

摘 要

机器学习是人工智能领域最重要的分支之一，主要研究计算机如何通过利用经验自动提高自身的性能，并已成为智能数据分析的主要方法。按照监督信息的不同，机器学习问题可以分为监督信息完全的监督学习、没有监督信息的无监督学习，以及介于两者之间的弱监督学习，其中弱监督学习包括监督信息滞后的强化学习、监督信息缺失的半监督学习、多示例学习等。AAAI Fellow美国华盛顿大学P. Domingos 教授指出“机器学习=表示+评估+优化”[41]，即不同的机器学习问题的解决方案都包括模型的表示、学习目标的评估，以及达到学习目标的优化方法三个部分。可见，机器学习任务最终承载在优化方法上，优化方法的能力也决定了能够使用的模型表示和评估函数。然而各类机器学习任务都面临着NP难的优化问题，因此机器学习需要强有力地优化方法。

Draft

Lecture Notes on Optimization

Zhouchen Lin (Computer Application)

Directed by

This lecture note will introduce the basics of convex analysis which is used in optimization and the basic optimization methods which are frequently used in machine learning. It is somewhat based on the famous textbook, Convex Analysis by Stephen Boyd and Lieven Vandenberghe, in the sense that if I need some materials this book will be the first-consulted one and I will take the needed materials if they are available in it. However, there are still many additions and deletions in order to better fit for machine learning applications. For example, the first-order methods are highlighted and the second-order methods and interior point methods are de-emphasized, and the non-convex and non-smooth optimization and stochastic gradient descent etc. are added. The examples and exercises will be gradually replaced with those emerging from machine learning (will require a lot of efforts, so will not be done in short time). The unification of notations is also in progress.

关键词: Convex Analysis, Optimization

Draft

第一章 Optimization Problems

1.1 History of Optimization

(Taken from Chapter 1 of [137])

Optimization is everywhere, from engineering design to financial markets, from our daily activity to planning our holidays, and computer sciences to industrial applications. We always intend to maximize or minimize something. An organization wants to maximize its profits, minimize costs, and maximize performance. Even when we plan our holidays, we want to maximize our enjoyment with least cost (or ideally free). In fact, we are constantly searching for the optimal solutions to every problem we meet, though we are not necessarily able to find such solutions.

It is no exaggeration to say that finding the solution to optimization problems, whether intentionally or subconsciously, is as old as human history itself. For example, the least effort principle can often explain many human behaviors. We know the shortest distance between any two different points on a plane is a straight line, though it often needs complex maths such as the calculus of variations to formally prove that a straight line segment between the two points is indeed the shortest.

In fact, many physical phenomena are governed by the so-called least action principle or its variants. For example, light travels and obeys Fermat's principle, that is to travel at the shortest time from one medium to another, thus resulting in Snell's law. The whole analytical mechanics is based on this least action principle.

1.2 Before 1900

The study of optimization problems is also as old as science itself. It is known that the ancient Greek mathematicians solved many optimization problems. For example, Euclid in around 300BC proved that a square encloses the greatest area among all possible rectangles with the same total length of four sides. Later, Heron in around 100BC suggested that the distance between two points along the path reflected by a mirror is the shortest when light travels and reflects from a mirror obeying some symmetry, that is the angle of incidence is equal to the angle of reflection. It is a well-known optimization problem, called Heron's problem, as it was first described in Heron's *Catoptrica* (or *On Mirrors*).

The celebrated German astronomer, Johannes Kepler, is mainly famous for the discovery of his three laws of planetary motion; however, in 1613, he solved an optimal solution to the so-called marriage problem or secretary problem when he started to look for his second wife. He described his method in his personal letter dated October 23, 1613 to Baron Strahlendorf, including the balance of virtues and drawbacks of each candidate, her dowry, hesitation, and advice of friends. Among the eleven candidates interviewed, Kepler chose the fifth, though his friend suggested him to choose the fourth candidate. This may imply that Kepler was trying to optimize some utility function of some sort. This problem was formally introduced by Martin Gardner in 1960 in his mathematical games column in the February 1960 issue of *Scientific American*. Since then, it has developed into a field of probability optimization such as optimal stopping problems.

W. van Royen Snell discovered in 1621 the law of refraction, which remained unpublished; later, Christiaan Huygens mentioned Snell's results in his *Dioptrica* in 1703. This law was independently rediscovered by Rene Descartes and published in his treatise *Discours de la Methode* in 1637. About 20 years later, when Descartes' students contacted Pierre de Fermat collecting his correspondence with Descartes, Fermat looked again in 1657 at his argument with the unsatisfactory description of light refraction by Descartes, and derived Snell and Descartes' results from a more fundamental principle – light always travels in the shortest time in any medium, and this principle for light is now referred to as Fermat's principle, which laid the foundation of modern optics.

In his *Principia Mathematica* published in 1687, Sir Isaac Newton solved the problem of the body shape of minimal resistance that he posed earlier in 1685 as a pioneering problem in optimization, now a problem of the calculus of variations. The main aim was to find the shape of a symmetrical revolution body so as to minimize the resistance to motion in a fluid. Subsequently, Newton derived the resistance law of the body. Interestingly, Galileo Galilei independently suggested a similar problem in 1638 in his *Discorsi*.

In June 1696, J. Bernoulli made some significant progress in calculus. In an article in *Acta Eruditorum*, he challenged all the mathematicians in the world to find the shape or curve connecting two points at different heights so that a body will fall along the curve in the shortest time due to gravity - the line of quickest descent, though Bernoulli already knew the solution. On January 29, 1697 the challenge was received by Newton when he came home at four in the afternoon and he did not sleep until he had solved it by about four the next morning and on the same day he sent out his solution. Though Newton managed to solve it in less than 12 hours as he became the Warden of the Royal Mint on March 19, 1696, some suggested that he, as such a genius, should have been able to

第一章 OPTIMIZATION PROBLEMS

solve it in half an hour. Some said this was the first hint or evidence that too much administrative work will slow down one's progress. The solution as we now know is a part of a cycloid. This steepest descent is now called Brachistochrone problem, which inspired Euler and Lagrange to formulate the general theory of calculus of variations.

In 1746, the principle of least action was proposed by P. L. de Maupertuis to unify various laws of physical motion and its application to explain all phenomena. In modern terminology, it is a variational principle of stationary action in terms of an integral equation of a functional in the framework of calculus of variations, which plays a central role in the Lagrangian and Hamiltonian classical mechanics. It is also an important principle in mathematics and physics.

In 1781, Gaspard Monge, a French civil engineer, investigated the transportation problem for optimal transportation and allocation of resources, if the initial and final spatial distribution are known. In 1942, Leonid Kantorovich showed that this combinatorial optimization problem is in fact a case of a linear programming problem.

Around 1801, Frederich Gauss claimed that he used the method of least-squares to predict the orbital location of the asteroid Ceres, though his version of the least squares with more rigorous mathematical foundation was published later in 1809. In 1805, Adrien Legendre was the first to describe the method of least squares in an appendix of his book *Nouvelle meethodes pour la determination des orbites des cometes*, and in 1806 he used the principle of least squares for curve fitting. Gauss later claimed that he had been using this method for more than 20 years, and laid the foundation for least-squares analysis in 1795. This led to some bitter disputes with Legendre. In 1808, Robert Adrain, unaware of Legendre's work, published the method of least squares studying the uncertainty and errors in making observations, not using the same terminology as those by Legendre.

In 1815, D. Ricardo proposed the law of diminishing returns for land cultivation, which can be applied in many activities. For example, the productivity of a piece of a land or a factory will only increase marginally with additional increase of inputs. This law is called law of increasing opportunity cost. It dictates that there is a fundamental relationship between opportunity and scarcity of resources, thus requiring that scarcely available resources be used efficiently.

In 1847 in a short note, L. A. Cauchy proposed a general method for solving systems of equations in an iterative way. This essentially leads to two iterative methods of minimization: now called the gradient method and steepest descent, for certain functions of more than one variable.

1.3 Twentieth Century

In 1906, Danish mathematician J. Jensen introduced the concept of convexity and derived an inequality, now referred to as Jensen's inequality, which plays an important role in convex optimization and other areas such as economics. Convex optimization is a special but very important class of mathematical optimization as any optimality found is also guaranteed to be the global optimality. A wider range of optimization problems can be reformulated in terms of convex optimization. Consequently, it has many applications including control systems, data fitting and modelling, optimal design, signal processing, mathematical finance, and others.

As early as 1766, Leonhard Euler studied the Knight tour problem, and T. P. Kirkman published a research article on the way to find a circuit which passes through each vertex once and only once for a give graph of polyhedra. In 1856, Sir William Rowan Hamilton popularized his Icosian Game. Then, in February 1930, Karl Menger posed the Messenger's problem at a mathematical colloquium in Vienna, as this problem is often encountered by postal messengers and travelers. His work was published later in 1932. The task is to find the shortest path connecting a finite number of points/cities whose pairwise distances are known. Though the problem is solvable in a finite number of trials and permutations, there is no efficient algorithm for finding such solutions. In general, the simple rule of going to the nearest points does not result in the shortest path. This problem is now referred to as Traveling Salesman Problem which is closely related to many different applications such as network routing, resource allocation, scheduling and operations research in general. In fact, as early as 1832, the 1832 traveling salesman manual described a tour along 45 German cities with a shortest route of 1248 km, though the exact mathematical roots of this problem are quite obscure, and might be well around for some time before the 1930s.

Interestingly, H. Hancock published in 1917 the first book on optimization “Theory of Minima and Maxima.”

In 1939, L. Kantorovich was the first to develop an algorithm for linear programming and use it in economics. He formulated the production problem of optimal planning and effective methods for finding solutions using linear programming. For this work, he shared the Nobel prize with T. Koopmans in 1975. The next important step of progress is that George Dantzig invented in 1947 the simplex method for solving large-scale linear programming problems. Dorfman in an article published in 1984 wrote that linear programming was discovered three times, independently, between 1939 and 1947, but each time in a somewhat different form. The first discovery was by the Russian mathemati-

第一章 OPTIMIZATION PROBLEMS

cian, L. Kantorovich, then by the Dutch economist, Koopmans, and the third in 1947 by the American mathematician George Dantzig. Dantzig's revolutionary simplex method is able to solve a wide range of optimal policy decision problems of great complexity. A classic example and one of the earliest of using linear programming as described in Dantzig's 1963 book was to find the solution to the special optimal diet problem involving 9 equations and 77 unknowns using hand-operated desk calculators. John von Neumann developed the theory of duality in the same year.

In 1951, Harold Kuhn and A. W. Tucker studied the nonlinear optimization problem and re-developed the optimality condition, as similar conditions were proposed by W. Karush in 1939 in his MSc dissertation. In fact, the optimality conditions are the generalization of Lagrange multipliers to nonlinear inequalities, and are now known as the Karush-Kuhn-Tucker conditions, or simply Kuhn-Tucker conditions, which are necessary conditions for a solution to be optimal in nonlinear programming.

Then, in 1957, Richard Bellman at Stanford University developed the dynamic programming and the optimality principle when studying multistage decision and planning processes while he spent some time at the RAND Corporation. He also coined the term Dynamic Programming. The idea of dynamic programming can date back to 1944 when John von Neumann and O. Morgenstern studied the sequential decision problems. John von Neumann also made important contribution to the development of operational research. As earlier as in 1840, Charles Babbage studied the cost of transportation and sorting mails; this could be the earliest research on the operational research. Significant progress was made during the Second World War, and ever since it expanded to find optimal or near optimal solutions in a wide range complex problems of interdisciplinary areas such as communication networks, project planning, scheduling, transport planning, and management.

[(Taken from [47])

In 1951, A.W. Tucker and his student H.W. Kuhn published the Kuhn-Tucker conditions. This is considered as an initial point of nonlinear programming. However, A. Takayama has an interesting comment on these condition: "Linear programming aroused interest in constraints in the form of inequalities and in the theory of linear inequalities and convex sets. The Kuhn-Tucker study appeared in the middle of this interest with a full recognition of such developments. However, the theory of nonlinear programming when constraints are all in the form of equalities has been known for a long time – in fact, since Euler and Lagrange. The inequality constraints were treated in a fairly satisfactory manner already in 1939 by Karush. Karush's work is apparently under the influence of a similar work in the calculus of variations by Valentine. Unfortunately, Karush's work

has been largely ignored.” Yet, this is another work that appeared before 1947 and it was ignored. In the 1960s, G. Zoutendijk, J.B. Rosen, P.Wolfe, M.J.D. Powell, and others published a number of algorithms for solving nonlinear optimization problems. These algorithms form the basis of contemporary nonlinear programming.

In 1954, L.R. Ford and D.R. Fulkerson initiated the study on network flows. This is considered as a starting point on combinatorial optimization although Fermat is the first one who studied a major combinatorial optimization problem. In fact, it was because of the influence of the results of Ford and Fulkerson, that interests on combinatorial optimization were growing, and so many problems, including Steiner trees, were proposed or re-discovered in history. In 1958, R.E. Gomory published the cutting plane method. This is considered as an initiation of integer programming, an important direction of combinatorial optimization.

In 1955, Dantzig published his paper “Linear programming under uncertainty” and E.M.L. Beale proposed an algorithm to solve similar problems. They started the study on stochastic programming. R.JB. Wets in the 1960s, and J.R. Birge and A. Prékopa in the 1980s made important contributions in this branch of optimization.]

After the 1960s, the literature on optimization exploded, and it would take a whole book to write even a brief history on optimization after the 1960s. As this book is mainly about the introduction to metaheuristic algorithms, we will then focus our attention on the development of heuristics and metaheuristics. In fact, quite a significant number of new algorithms in optimization are primarily metaheuristics.

(see <http://www.mitrikitti.fi/opthist.html> for a chronological list of optimization techniques.)

1.4 Heuristics and Metaheuristics

Heuristics is a solution strategy by trial-and-error to produce acceptable solutions to a complex problem in a reasonably practical time. The complexity of the problem of interest makes it impossible to search every possible solution or combination, the aim is to find good, feasible solutions in an acceptable timescale. There is no guarantee that the best solutions can be found, and we even do not know whether an algorithm will work and why if it does work. The idea is that an efficient but practical algorithm that will work most of the time and be able to produce good quality solutions. Among the found quality solutions, it is expected that some of them are nearly optimal, though there is no guarantee for such optimality.

Alan Turing was probably the first to use heuristic algorithms during the Second

第一章 OPTIMIZATION PROBLEMS

World War when he was breaking German Enigma ciphers at Bletchley Park where Turing, together with British mathematician Gordon Welchman, designed in 1940 a crypt-analytic electromechanical machine, the Bombe, to aid their code-breaking work. The bombe used a heuristic algorithm, as Turing called, to search, among about 10^{22} potential combinations, the possibly correct setting coded in an Enigma message. Turing called his search method heuristic search, as it could be expected it worked most of the time, but there was no guarantee to find the correct solution, but it was a tremendous success. In 1945, Turing was recruited to the National Physical Laboratory (NPL), UK where he set out his design for the Automatic Computing Engine (ACE). In an NPL report on “Intelligent Machinery” in 1948, he outlined his innovative ideas of machine intelligence and learning, neural networks and evolutionary algorithms or an early version of genetic algorithms.

The next significant step is the development of evolutionary algorithms in the 1960s and 1970s. First, John Holland and his collaborators at the University of Michigan developed the genetic algorithms in the 1960s and 1970s. As early as 1962, Holland studied the adaptive system and was the first to use crossover and recombination manipulations for modeling such systems. His seminal book summarizing the development of genetic algorithms was published in 1975. In the same year, Kenneth De Jong finished his important dissertation showing the potential and power of genetic algorithms for a wide range of objective functions, either noisy, multimodal or even discontinuous.

Genetic algorithms (GA) is a search method based on the abstraction of Darwin’s evolution and natural selection of biological systems and representing them in the mathematical operators: crossover or recombination, mutation, fitness, and selection of the fittest. Ever since, genetic algorithms become so successful in solving a wide range of optimization problems, several thousand research articles and hundreds of books have been written. Some statistics show that a vast majority of Fortune 500 companies are now using them routinely to solve tough combinatorial optimization problems such as planning, data-fitting, and scheduling.

During the same period, Ingo Rechenberg and Hans-Paul Schwefel both then at the Technical University of Berlin developed a search technique for solving optimization problem in aerospace engineering, called evolution strategy, in 1963. Later, Peter Bienert joined them and began to construct an automatic experimenter using simple rules of mutation and selection. There is no crossover in this technique; only mutation was used to produce an offspring and an improved solution was kept at each generation. This is essentially a simple trajectory-style hill-climbing algorithm with randomization. As early as 1960, Lawrence J. Fogel intended to use simulated evolution as a learning process

as a tool to study artificial intelligence. Then, in 1966, L. J. Fogel, with A. J. Owen and M. J. Walsh, developed the evolutionary programming technique by representing solutions as finite-state machines and randomly mutating one of these machines. The above innovative ideas and methods have evolved into a much wider discipline, called evolutionary algorithms and evolutionary computation.

The decades of the 1980s and 1990s were the most exciting time for metaheuristic algorithms. The next big step is the development of simulated annealing (SA) in 1983, an optimization technique, pioneered by S. Kirkpatrick, C. D. Gellat and M. P. Vecchi, inspired by the annealing process of metals. It is a trajectory-based search algorithm starting with an initial guess solution at a high temperature, and gradually cooling down the system. A move or new solution is accepted if it is better; otherwise, it is accepted with a probability, which makes it possible for the system to escape any local optima. It is then expected that if the system is cooled slowly enough, the global optimal solution can be reached.

The actual first usage of metaheuristic is probably due to Fred Glover's Tabu search in 1986, though his seminal book on Tabu search was published later in 1997.

In 1992, Marco Dorigo finished his PhD thesis on optimization and natural algorithms, in which he described his innovative work on ant colony optimization (ACO). This search technique was inspired by the swarm intelligence of social ants using pheromone as a chemical messenger. Then, in 1992, John R. Koza of Stanford University published a treatise on genetic programming which laid the foundation of a whole new area of machine learning, revolutionizing computer programming. As early as in 1988, Koza applied his first patent on genetic programming. The basic idea is to use the genetic principle to breed computer programs so as to produce the best programs for a given type of problem.

Slightly later in 1995, another significant step of progress is the development of the particle swarm optimization (PSO) by American social psychologist James Kennedy, and engineer Russell C. Eberhart. Loosely speaking, PSO is an optimization algorithm inspired by the swarm intelligence of fish and birds and even by human behavior. The multiple agents, called particles, swarm around the search space starting from some initial random guess. The swarm communicates the current best and shares the global best so as to focus on the quality solutions. Since its development, there have been about 20 different variants of particle swarm, and have been applied to almost all areas of tough optimization problems. There is some strong evidence that PSO is better than traditional search algorithms and even better than genetic algorithms for most type of problems, though this is far from conclusive.

In 1997, the publication of the ‘no free lunch theorems for optimization’ [133] by D. H. Wolpert and W. G. Macready sent out a shock wave to the optimization community. Researchers have always been trying to find better algorithms, or even universally robust algorithms, for optimization, especially for tough NP-hard optimization problems. However, these theorems state that if algorithm A performs better than algorithm B for some optimization functions, then B will outperform A for other functions. That is to say, if averaged over all possible function space, both algorithms A and B will perform on average equally well. Alternatively, there is no universally better algorithms exist. That is disappointing, right? Then, people realized that we do not need the average over all possible functions as for a given optimization problem. What we want is to find the best solutions; this has nothing to do with average over all the whole function space. In addition, we can accept the fact that there is no universal or magical tool, but we do know from our experience that some algorithms indeed outperform others for given types of optimization problems. So the research now focuses on finding the best and most efficient algorithm(s) for a given problem. The task is to design better algorithms for most types of problems, not for all the problems. Therefore, the search is still on.

At the turn of the twenty-first century, things became even more exciting. First, Zong Woo Geem et al. in 2001 developed the Harmony Search (HS) algorithm, which has been widely applied in solving various optimization problems such as water distribution, transport modelling and scheduling. In 2004, S. Nakrani and C. Tovey proposed the Honey Bee algorithm and its application for optimizing Internet hosting centers, which followed by the development of a novel bee algorithm by D. T. Pham et al. in 2005 and the Artificial Bee Colony (ABC) by D. Karaboga in 2005. In 2008, the author of this book developed the Firefly Algorithm (FA). Quite a few research articles on the Firefly Algorithm then followed, and this algorithm has attracted a wide range of interests.

As we can see, more and more metaheuristic algorithms are being developed. Such a diverse range of algorithms necessitates a system summary of various metaheuristic algorithms, and this book is such an attempt to introduce all the latest and major metaheuristics with applications.

1.4.1 No-free-lunch theorem and its proof – the case of static function

(Taken from [133])

Theorem 1. *For any two algorithms a_1 and a_2 on a finite search space, iterating m times,*

$$\sum_f P(\mathbf{d}_m^y | f, m, a_1) = \sum_f P(\mathbf{d}_m^y | f, m, a_2),$$

where $\mathbf{d}_m^y = \{\mathbf{d}_m^y(1), \dots, \mathbf{d}_m^y(m)\}$ is the cost of evaluation at points $\mathbf{d}_m^x = \{\mathbf{d}_m^x(1), \dots, \mathbf{d}_m^x(m)\}$ and P is a probability distribution.

Proof. The intuition behind the proof is straightforward: by summing over all we ensure that the past performance of an algorithm has no bearing on its future performance. Accordingly, under such a sum, all algorithms perform equally.

The proof is by induction. The induction is based on $m = 1$, and the inductive step is based on breaking f into two independent parts, one for $x \in \mathbf{d}_m^x$ and one for $x \notin \mathbf{d}_m^x$. These are evaluated separately, giving the desired result.

For $m = 1$, we write the first sample as where is set by $\mathbf{d}_1 = \{\mathbf{d}_1^x, f(\mathbf{d}_1^x)\}$, where \mathbf{d}_1^x is set by a . The only possible value for \mathbf{d}_1^y is $f(\mathbf{d}_1^x)$, so we have

$$\sum_f P(\mathbf{d}_1^y | f, m = 1, a) = \sum_f \delta(\mathbf{d}_1^y, f(\mathbf{d}_1^x)),$$

where δ is the Kronecker delta function.

Summing over all possible cost functions, $\delta(\mathbf{d}_1^y, f(\mathbf{d}_1^x))$ is one only for those functions which have cost \mathbf{d}_1^y at point \mathbf{d}_1^x . Therefore that sum equals $|\mathcal{Y}|^{|\mathcal{X}|-1}$, independent of \mathbf{d}_1^x :

$$\sum_f P(\mathbf{d}_1^y | f, m = 1, a) = |\mathcal{Y}|^{|\mathcal{X}|-1},$$

which is independent of a . This bases the induction.

The inductive step requires that if $\sum_f P(\mathbf{d}_m^y | f, m, a)$ is independent of a for all \mathbf{d}_m^y , then so also is $\sum_f P(\mathbf{d}_{m+1}^y | f, m + 1, a)$. Establishing this step completes the proof.

We begin by writing

$$\begin{aligned} P(\mathbf{d}_{m+1}^y | f, m + 1, a) &= P(\{\mathbf{d}_{m+1}^y(1), \dots, \mathbf{d}_{m+1}^y(m)\}, \mathbf{d}_{m+1}^y(m+1) | f, m + 1, a) \\ &= P(\mathbf{d}_m^y, \mathbf{d}_{m+1}^y(m+1) | f, m + 1, a) \\ &= P(\mathbf{d}_m^y | f, m + 1, a)P(\mathbf{d}_{m+1}^y(m+1) | f, m + 1, a), \end{aligned}$$

and thus

$$\sum_f P(\mathbf{d}_{m+1}^y | f, m + 1, a) = \sum_f P(\mathbf{d}_m^y | f, m + 1, a)P(\mathbf{d}_{m+1}^y(m+1) | f, m + 1, a).$$

The new y value, $\mathbf{d}_{m+1}^y(m+1)$, will depend on the new x value, f , and nothing else. So we expand over these possible x values, obtaining

$$\begin{aligned} \sum_f P(\mathbf{d}_{m+1}^y | f, m + 1, a) &= \sum_{f,x} P(\mathbf{d}_{m+1}^y(m+1) | f, x)P(x | \mathbf{d}_m^y, f, m + 1, a)P(\mathbf{d}_m^y | f, m + 1, a) \\ &= \sum_{f,x} \delta(\mathbf{d}_{m+1}^y(m+1), f(x))P(x | \mathbf{d}_m^y, f, m + 1, a)P(\mathbf{d}_m^y | f, m + 1, a). \end{aligned}$$

Next note that since $x = a(\mathbf{d}_m^x, \mathbf{d}_m^y)$, it does not depend directly on f . Consequently we expand in \mathbf{d}_m^x to remove the f dependence in $P(x|\mathbf{d}_m^y f, m + 1, a)$:

$$\begin{aligned} & \sum_f P(\mathbf{d}_{m+1}^y | f, m + 1, a) \\ &= \sum_{f, x, \mathbf{d}_m^x} \delta(\mathbf{d}_{m+1}^y(m+1), f(x)) P(x|\mathbf{d}_m, a) P(\mathbf{d}_m^x | \mathbf{d}_m^y, f, m + 1, a) P(\mathbf{d}_m^y | f, m + 1, a) \\ &= \sum_{f, \mathbf{d}_m^x} \delta(\mathbf{d}_{m+1}^y(m+1), f(a(\mathbf{d}_m))) P(\mathbf{d}_m | f, m, a), \end{aligned}$$

where we made use of the fact that and the facts that $P(x|\mathbf{d}_m, a) = \delta(x, a(\mathbf{d}_m))$ and $P(\mathbf{d}_m | f, m + 1, a) = P(\mathbf{d}_m | f, m, a)$.

The sum over cost functions f is done first. The cost function is defined both over those points restricted to \mathbf{d}_m^x and those points outside of \mathbf{d}_m^x . So $P(\mathbf{d}_m | f, m, a)$ will depend on the f values defined over points inside \mathbf{d}_m^x while $\delta(\mathbf{d}_{m+1}^y(m+1), f(a(\mathbf{d}_m)))$ depends only on the f values defined over points outside \mathbf{d}_m^x . (Recall that $a(\mathbf{d}_m^x) \notin \mathbf{d}_m^x$.) So we have

$$\sum_f P(\mathbf{d}_{m+1}^y | f, m + 1, a) = \sum_{\mathbf{d}_m^x} \sum_{f(x \in \mathbf{d}_m^x)} P(\mathbf{d}_m | f, m, a) \sum_{f(x \in \mathbf{d}_m^x)} \delta(\mathbf{d}_{m+1}^y(m+1), f(a(\mathbf{d}_m))).$$

The sum $\sum_{f(x \in \mathbf{d}_m^x)}$ contributes a constant, $|\mathcal{Y}|^{|\mathcal{X}-m-1|}$, equal to the number of functions defined over points not in \mathbf{d}_m^x passing through $(\mathbf{d}_{m+1}^x(m+1), f(a(\mathbf{d}_m)))$. So

$$\begin{aligned} \sum_f P(\mathbf{d}_{m+1}^y | f, m + 1, a) &= |\mathcal{Y}|^{|\mathcal{X}-m-1|} \sum_{\mathbf{d}_m^x, f(x \in \mathbf{d}_m^x)} P(\mathbf{d}_m | \mathbf{d}_m^x, f, m, a) \\ &= \frac{1}{|\mathcal{Y}|} \sum_{f, \mathbf{d}_m^x} P(\mathbf{d}_m | f, m, a), \\ &= \frac{1}{|\mathcal{Y}|} \sum_f P(\mathbf{d}_m^y | f, m, a). \end{aligned}$$

By hypothesis, the right-hand side of this equation is independent of a , so the left-hand side must also be. This completes the proof. \square

1.4.2 Exercises

Exercise 2. There is at least one photo of the people in the PPT of “History of optimization” which is incorrect (not accounting for the ages and poses). Find them, provide your correct photos and give justifications.

Exercise 3. The no-free-lunch (NFL) theorem is actually quite general. Give more instances that the NFL theorem applies.

1.5 Some convex optimization problems in machine learning

(Taken from Chapter 1.1 of [20])

Many fundamental convex optimization problems in machine learning take the following form:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \frac{1}{m} \sum_{i=1}^m f_i(\mathbf{x}) + \lambda \mathcal{R}(\mathbf{x}), \quad (1.1)$$

where the functions $f_1, \dots, f_m, \mathcal{R}$ are convex and $\lambda \geq 0$ is a fixed parameter. The interpretation is that $f_i(\mathbf{x})$ represents the cost of using \mathbf{x} on the i^{th} element of some data set, and $\mathcal{R}(\mathbf{x})$ is a regularization term which enforces some “simplicity” in \mathbf{x} . We discuss now major instances of (1.1). In all cases one has a data set of the form $(w_i, y_i) \in \mathbb{R}^n \times \mathcal{Y}, i = 1, \dots, m$, and the cost function f_i depends only on the pair (w_i, y_i) . We refer to [59, 115, 116] for more details on the origin of these important problems. The mere objective of this section is to expose the reader to a few concrete convex optimization problems which are routinely solved.

In classification one has $\mathcal{Y} = \mathbb{R}$. Taking $f_i(\mathbf{x}) = (\mathbf{x}^\top \mathbf{w}_i - y_i)^2$ and $\mathcal{R}(\mathbf{x}) = 0$ one obtains the vanilla least-squares problem which can be rewritten in vector notation as

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{W}\mathbf{x} - \mathbf{y}\|_2^2 \quad (1.2)$$

where $\mathbf{W} \in \mathbb{R}^{m \times n}$ is the matrix with \mathbf{w}_i^\top on the i^{th} row and $\mathbf{y} = (y_1, \dots, y_m)^\top$. With $\mathcal{R}(\mathbf{x}) = \|\mathbf{x}\|_2^2$ one obtains the ridge regression problem, while with $\mathcal{R}(\mathbf{x}) = \|\mathbf{x}\|_1$ this is the LASSO problem [124].

Our last two examples are of a slightly different flavor. In particular the design variable \mathbf{x} is now best viewed as a matrix, and thus we denote it by a capital letter \mathbf{X} . The sparse inverse covariance estimation problem can be written as follows, given some empirical covariance matrix \mathbf{Y} ,

$$\begin{aligned} & \min \text{tr } \mathbf{XY} - \log \det(\mathbf{X}) + \lambda \|\mathbf{X}\|_1 \\ & \text{s.t. } \mathbf{X} \in \mathbb{R}^{n \times n}, \mathbf{X}^\top = \mathbf{X}, \mathbf{X} \succeq \mathbf{0}. \end{aligned} \quad (1.3)$$

Intuitively the above problem is simply a regularized maximum likelihood estimator (under a Gaussian assumption).

Finally we introduce the convex version of the matrix completion problem. Here our data set consists of observations of some of the entries of an unknown matrix \mathbf{Y} , and we want to “complete” the unobserved entries of \mathbf{Y} in such a way that the resulting matrix is “simple” (in the sense that it has low rank). After some massaging (see [23]) the (convex)

matrix completion problem can be formulated as follows:

$$\begin{aligned} & \min \operatorname{tr} \mathbf{X} \\ \text{s.t. } & \mathbf{X} \in \mathbb{R}^{n \times n}, \mathbf{X}^\top = \mathbf{X}, \mathbf{X} \succeq \mathbf{0}, \mathbf{X}_{i,j} = \mathbf{Y}_{i,j} \text{ for } (i,j) \in \Omega \end{aligned} \quad (1.4)$$

where $\Omega \subset [n]^2$ and $(\mathbf{Y}_{i,j})_{(i,j) \in \Omega}$ are given.

(Taken from Chapter 2 of [127])

1.6 What is Optimization?

1.6.1 Classification of Optimization Problems

The classification of optimization problems is very important because it will guide us to devise strategies and techniques to solve the problems from different classes. Optimization problems can be classified according to several criteria related to the properties of the objective function and also of the solution set S . Thus, possible classifications take into account

1. the nature of the solution set S .
2. the description (definition) of the solution set S .
3. the properties of the objective function f .

The most important classification is the one based on the nature of the solution set S , which leads us to classify optimization problems into four classes: *continuous*, *discrete*, *combinatorial*, and *variational*.

A point \mathbf{x} of a topological space S is an accumulation point, if and only if for any open ball $B_{\mathbf{x}}$, with $\mathbf{x} \in B_{\mathbf{x}}$, there exists an element $\mathbf{y} \in S$ such that $\mathbf{y} \in B_{\mathbf{x}}$. A point $\mathbf{x} \in S$, which is not an accumulation point, is called an isolated point of S . Thus, \mathbf{x} is isolated if there exists an open ball $B_{\mathbf{x}}$, such that $\mathbf{x} \in B_{\mathbf{x}}$ and no other point of S belongs to $B_{\mathbf{x}}$.

A topological space is discrete if it contains no accumulation points. It is continuous when all its points are accumulation points.

Example 4 (discrete and continuous sets). *Any finite subset of an Euclidean space is obviously discrete. The set \mathbb{R}^n itself is continuous. The set $\mathbb{Z}^n = \{(i_1, \dots, i_n) | i_j \in \mathbb{Z}\}$ is discrete. The set $\{0\} \cup \{1/n | n \in \mathbb{Z}\}$ is not discrete because 0 is an accumulation point (no matter how small we take a ball with center at 0, it will contain some element $1/n$ for some large n); this set is not continuous either because with the exception of 0, all the points are isolated. However, the set $\{1/n | n \in \mathbb{Z}\}$ (without the 0) is discrete.*

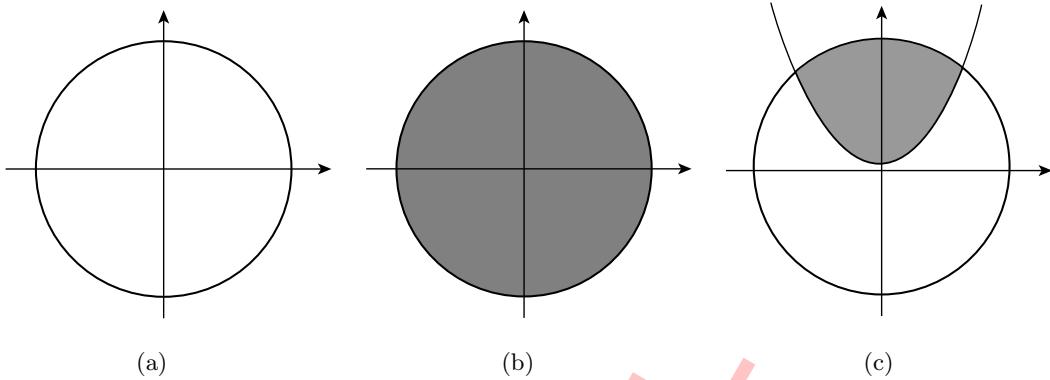


图 1.1: Three examples of a solution set.

1.6.2 Classification Based on The Nature of Solution

In this section, we discuss optimization methods according to the nature of the solution set.

1.6.2.1 Continuous Optimization Problems

An optimization problem is called *continuous* when the solution set S is a continuous subset of \mathbb{R}^n .

The most common cases occur when S is a differentiable m -dimensional surface of \mathbb{R}^n (with or without boundary), or $S \subseteq \mathbb{R}^n$ is a region of \mathbb{R}^n :

1. Minimize the function $f(x, y) = 3x + 4y + 1$ on the set $S = \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 = 1\}$. The solution set S is the unit circle S^1 , which is a curve of \mathbb{R}^2 , a 1D surface (see Figure 1.1(a)).
 2. Minimize the function $f(x, y) = 3x + 4y + 1$ on the set $S = \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 \leq 1\}$. The solution set is the disk of \mathbb{R}^2 , a 2D surface with boundary (see Figure 1.1(b)).
 3. Minimize the function $f(x, y) = 3x + 4y + 1$ on the set S , defined by the inequalities $x^2 + y^2 \leq 1$ and $y - x^2 \geq 0$. The solution set is a region defined by the intersection of two surfaces with boundary (see Figure 1.1(c)).

1.6.2.2 Discrete Optimization Problems

An optimization problem is called *discrete* when the solution set S is a discrete set (i.e., S has no accumulation points).

The most frequent case in the applications occurs for $S \subseteq \mathbb{Z}^n = \{(i_1, \dots, i_n) | i_n \in \mathbb{Z}\}$.

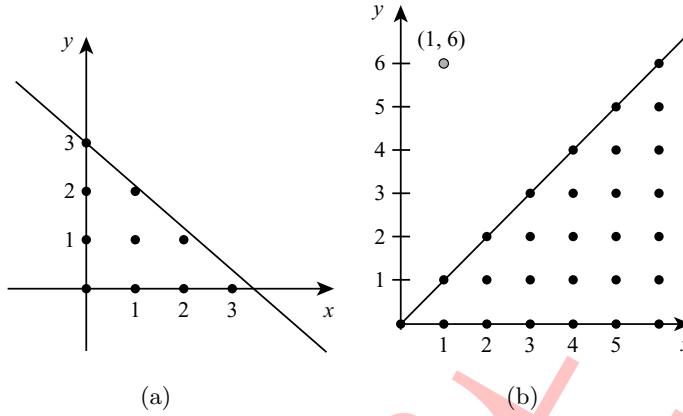


图 1.2: Discrete solution sets.

1. Minimize the function $f(x, y) = x + y$ on the set $S = \{(x, y) \in \mathbb{Z}^2 \mid 6x + 7y = 21, x \geq 0, y \geq 0\}$. The solution set S is a finite set, namely, the set of points with integer coordinates in a triangle whose sides are the axes and the line $6x + 7y = 21$ (see Figure 1.2(a)).
2. Minimize the function $f(x, y) = (x - 1)^2 + (y - 6)^2$ on the set $S = \{(x, y) \in \mathbb{Z}^2 \mid y \leq x, x \geq 0, y \geq 0\}$. The solution set S is infinite; the problem asks for finding among all points with integer coordinates in a cone, the one that is closest to the point $(1, 6)$ (see Figure 1.2(b)).

There is a close relationship between continuous and discrete optimization problems. In this respect, it is important to remark that some types of discrete problems $\min\{f(\mathbf{x}) \mid \mathbf{x} \in S\}$ with solution set S may be solved more easily by embedding S into a continuous domain S' , solving the new continuous optimization problem $\min\{f(x) \mid x \in S'\}$, and obtaining the solution to the original problem from the solution in the continuous domain. The example below illustrates this approach.

Example 5. Consider the discrete optimization problem $\min\{f(x) = -3x + 2x^2 \mid x \in \mathbb{Z}\}$. In order to solve this problem, we consider a similar problem on \mathbb{R} , that is, $\min\{f(x) = -3x + 2x^2 \mid x \in \mathbb{R}\}$.

The solution to the continuous problem is simple. In fact, since $f(x)$ is a quadratic function with positive second derivative, it has a unique minimum point given by $f'(x) = 0$, that is, $3 - 4x = 0$. Therefore, the solution is $m = 3/4$, and the minimum value is $9/8$. Now we obtain the solution to the discrete problem from this solution.

Note that the solution obtained for $S = \mathbb{R}$ is not a solution for the discrete case (because $m \notin \mathbb{Z}$). But it is possible to compute the solution for the discrete case from the

solution to the continuous problem. For this, we just need to choose the integer number that is closest to m . It is possible to prove that this provides indeed the solution. In fact, since f is a decreasing function in the interval $[-\infty, m]$ and increasing function in the interval $[m, \infty]$ and its graph is symmetrical with respect to the line $x = m$, the furthest x is from m , the larger is the value $f(x)$. From this, we conclude that the integer n closest to m is indeed the point where f attains its minimum on the set \mathbb{Z} .

We should remark that in general the solution to a discrete problem cannot be obtained from a corresponding continuous problem in such a simple way as shown in the example above. As an example, consider the polynomial function $f : \mathbb{R} \rightarrow \mathbb{R}$, $f(x) = x^4 - 14x$. It has a minimum at $x \approx 1.52$. The closest integer value to this minimum point is 2, where f assumes the value -12 . Nevertheless, the minimum of $f(x)$ when x is an integer number occurs for $x = 1$, where f attains the value -13 .

Thus, although solving the continuous version of a discrete problem is part of the several existing strategies in computing a solution to a discrete problem, in general, finding the discrete solution requires more specific techniques that go beyond rounding off solutions.

1.6.2.3 Combinatorial Optimization Problems

An optimization problem is said to be *combinatorial* when its solution set S is finite. Usually, the elements of S are not explicitly determined. Instead, they are indirectly specified through combinatorial relations. This allows S to be specified much more compactly than by simply enumerating its elements.

In contrast to continuous optimization, whose study has its roots in classical calculus, the interest in solution methods for combinatorial optimization problems is relatively recent and significantly associated with computer technology. Before computers, combinatorial optimization problems were less interesting since they admit an obvious method of solution that consists in examining all possible solutions in order to find the best one. The relative efficiency of the possible methods of solution was not a very relevant issue: for real problems, involving sets with a large number of elements, any solution method, efficient or not, was inherently unfeasible for requiring a number of operations too large to be done by hand.

With computers, the search for efficient solution methods became imperative: the practical feasibility of solving a large-scale problem by computational methods depends on the availability of an efficient solution method.

Example 6 (The Traveling Salesman Problem). *This need is well illustrated by the trav-*

eling salesman problem. *Given n towns, one wishes to find the minimum length route that starts in a given town, goes through each one of the other towns, and ends in the starting town.*

The combinatorial structure of the problem is quite simple. Each possible route corresponds to one of the $(n - 1)!$ circular permutations of the n towns. Thus, it suffices to enumerate these permutations, evaluate their lengths, and choose the optimal route.

This, however, becomes unpractical even for moderate values of n . For instance, for $n = 50$, there are approximately 10^{60} permutations to examine. Even if 1 billion of them were evaluated per second, examining all would require about 10^{51} seconds, or 10^{43} years! However, there are techniques that allow solving this problem, in practice, even for larger values of n .

1.6.2.4 Variational Optimization Problems

An optimization problem is called a *variational problem* when its solution set S is an infinite dimensional subset of a space of functions.

Among the most important examples, we could mention the *path* and *surface* problems. The problems consist in finding the best path (best surface) satisfying some conditions that define the solution set.

Typical examples of variational problems are the following.

Example 7 (Geodesic Problem). *Find the path of minimum length joining two points p_1 and p_2 of a given surface.*

Example 8 (Minimal Surface Problem). *Find the surface of minimum area for a given boundary curve.*

1.6.3 Other Classifications

Other forms of classifying optimization problems are based on characteristics that can be exploited in order to devise strategies for the solution.

1.6.3.1 Classification Based on Constraints

In many cases, the solution set S is specified by describing constraints that must be satisfied by its elements. A very common way to define constraints consists in using equalities and inequalities. The classification based on constraints takes into account the *nature* of the constraints and the *properties* of the functions that describe them.

With respect to the nature of the constraint functions, we obtain the following classification:

1. equality constraints: $h_i(\mathbf{x}) = 0, i = 1, \dots, m;$
2. inequality constraints: $g_j(\mathbf{x}) \leq 0.$

In the first case, the solution set S consists of the points that satisfy simultaneously the equations $h_i(\mathbf{x}) = 0$. We should remark that in general the equation $h_i(\mathbf{x}) = 0$ defines an m -dimensional surface of \mathbb{R}^n ; therefore, the simultaneous set of equations $h_i(\mathbf{x}) = 0, i = 1, \dots, m$ define the intersection of m surfaces.

In the second case, each inequality $g_j(\mathbf{x}) \leq 0$ in general represents a surface of \mathbb{R}^n with boundary (the boundary is given by the equality $g_j(\mathbf{x}) = 0$). Thus, the solution set is the intersection of a finite number of surfaces with boundary; in general, this intersection is a region of \mathbb{R}^n (see Figure 1.1(c)). A particular case of great importance occurs when the functions g_j are linear. Then, the region is a solid polyhedron of \mathbb{R}^n .

As we will see later, equality and inequality constraints lead us to different algorithmic strategies when looking for a solution.

The algorithms are also affected by the properties of the functions that define the constraints. Among the particular and relevant special cases that can be exploited are the ones where the constraint functions are linear, quadratic, convex (or concave), or sparse.

Proposition below shows that the nature of the restriction functions influences the geometry of the solution set, and this geometry is widely exploited in the strategies to compute a solution to the problem.

Proposition 9. *If the restriction functions are convex, the solution set S is also convex.*

The result of this proposition can be exploited both to determine optimality conditions and to develop algorithms.

1.6.3.2 Classification Based on The Objective Function

The properties of the objective function are also fundamental in order to devise strategies for the solution to optimization problems. The special cases that lead to particular solution strategies are the ones where the objective function is linear, quadratic, convex (or concave), sparse, or separable.

Linear Programs

A very important situation occurs when both the objective and the constraint functions are linear and, moreover, the constraints are defined either by equalities or by inequalities, that is, the constraints are given by linear equations and linear inequalities.

As we have remarked before, the solution set is a solid *polyhedron* of the Euclidean space. Optimization problems of this nature are called *linear programs*, and their study constitute a subarea of optimization. In fact, several practical problems can be posed as linear programs, and there exists a number of techniques that solve them, exploiting their characteristics: the linearity of the objective function and the piecewise linear structure of the solution set (polyhedra).

1.6.4 The Problem of Posing Problems

Most of the problems in general and the problems of computer graphics in particular can be posed using the concept of operator between two spaces.¹ These problems can be classified into two categories: *direct* and *inverse* problems.

Direct problem. Given are two spaces O_1 and O_2 of graphical objects, an operator $T : O_1 \rightarrow O_2$, and a graphical object $\mathbf{x} \in O_1$. Problem: compute the object $\mathbf{y} = T(\mathbf{x}) \in O_2$. That is, we are given the operator T and a point x in its domain, and we must compute the image $\mathbf{y} = T(\mathbf{x})$.

Because T is an operator, it is not multivalued; therefore, the direct problem always has a unique solution.

Inverse problems. There are two types of inverse problems.

Inverse problem of first kind. Given two spaces O_1 and O_2 of graphical objects, an operator $T : O_1 \rightarrow O_2$, and an element $\mathbf{y} \in O_2$, determine an object $\mathbf{x} \in O_1$ such that $T(\mathbf{x}) = \mathbf{y}$. That is, we are given an operator T and a point \mathbf{y} belonging to the arrival set of the operator, and we must compute a point \mathbf{x} in its domain whose image is \mathbf{y} .

Inverse problem of second kind. Given a finite sequence of elements $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n \in O_1$ and a finite sequence $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n \in O_2$, determine an operator $T : O_1 \rightarrow O_2$ such that $T(\mathbf{x}_i) = \mathbf{y}_i, i = 1, 2, \dots, n$. That is, we are given a finite sequence $\{\mathbf{x}_i\}$ of points in the domain and its image $\mathbf{y}_i = T(\mathbf{x}_i)$, we must compute the operator T .

If the operator T is invertible with inverse T^{-1} , the inverse problem of first kind admits a unique solution $\mathbf{x} = T^{-1}(\mathbf{y})$. However, invertibility is not necessary. In fact, a weaker condition occurs when T has a left inverse T^+ (i.e., $T^+T = \text{Identity}$), and the solution is given by $T^+(\mathbf{y})$. In this case, we cannot guarantee uniqueness of the solution because we do not have uniqueness of the left inverse.

The inverse problem of second type has two interesting interpretations:

¹In some books the use of the term operator implies that it is linear. Here we use the term to mean a continuous function.

1. We are given an operator $\bar{T} : \{\mathbf{x}_1, \dots, \mathbf{x}_n\} \rightarrow \{\mathbf{y}_1, \dots, \mathbf{y}_n\}$, and we must compute an extension T of \bar{T} to the entire space O_1 . Clearly, in general, this solution, if it exists, is not unique.
2. The solution operator T defines an interpolation from the samples $(\mathbf{x}_i, T(\mathbf{x}_i)), i = 1, \dots, n$.

From the above interpretations, it is easy to conclude that, in general, the solution to any type of inverse problems is not unique.

1.6.4.1 Well-Posed Problems

Hadamard² has established the concept of a *well-posed* problem as a problem in which the following conditions are satisfied:

1. There is a solution.
2. The solution is unique.
3. The solution depends continuously on the initial conditions.

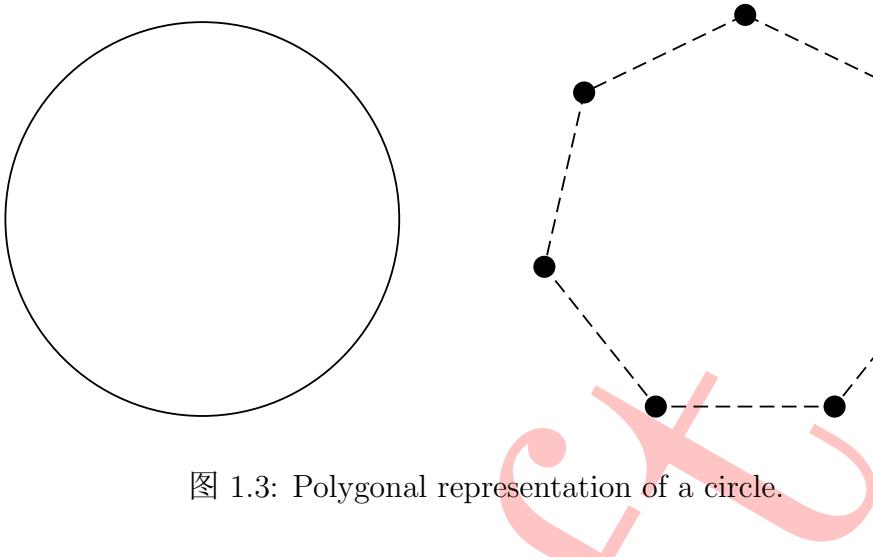
When at least one of the above conditions is not satisfied, we say that the problem is *ill posed*.

We have seen previously that, in general, inverse problems of the first and second kinds are ill posed in the sense of Hadamard because of the nonuniqueness of the solution.

The concept of an ill-posed problem needs to be further investigated very carefully. The continuity condition is important because it guarantees that small variations in the initial conditions will cause only small perturbations of the solution. Recall that, in practical problems, small variations in the initial conditions are quite common (and sometimes, inevitable) due to measurement imprecision or numerical errors. The nonuniqueness of solutions, however, should not be necessarily an obstacle. An example will make this point clear.

Example 10. Consider the operator $F : \mathbb{R}^2 : \mathbb{R}$ defined by $F(x, y) = x^2 + y^2 - 1$ and the inverse problem of the first kind $F(x, y) = 0$. That is, we must solve the quadratic equation $x^2 + y^2 - 1 = 0$. It is clear that it admits an infinite number of solutions and therefore it is ill posed in the sense of Hadamard. Geometrically, these solutions constitute the unit circle S^1 of the plane. The nonuniqueness of the solutions is very important in the representation of the circle. In fact, in order to obtain a sampled representation of the

²Jacques Hadamard (1865-1963), French mathematician.



circle, we need to determine a finite subset of these solutions. Figure 1.3 shows the use of seven solutions of the equation that allows us to obtain an approximate reconstruction of the circle by an heptagon.

The use of the term ill posed to designate the class of problems that do not satisfy one of the three Hadamard conditions may induce the reader to conclude that ill-posed problems cannot be solved, but this is far from the truth. Interpreting ill-posed problems as a category of intractable problems, the reader would be discouraged to study computer graphics, an area in which there is a large number of ill-posed problems, as we will see in this chapter. In fact, ill-posed problems constitute a fertile ground for the use of optimization methods. When a problem presents an infinite number of solutions and we wish to have unicity, optimization methods make it possible to reformulate the problem, so that a unique solution can be obtained. The general principle is to define an objective function such that an “optimal solution” can be chosen among the set of possible solutions. In this way, when we do not have a unique solution to a problem, we can have the best solution (which is unique).

It is interesting to note that, sometimes, well-posed problems may not present a solution due to the presence of noise or numerical errors. The example below shows one such case.

Example 11. Consider the solution of the linear system

$$x + 3y = a,$$

$$2x + 6y = 2a,$$

where $a \in \mathbb{R}$. In other words, we want to solve the inverse problem $\mathbf{T}\mathbf{x} = \mathbf{a}$, where \mathbf{T} is

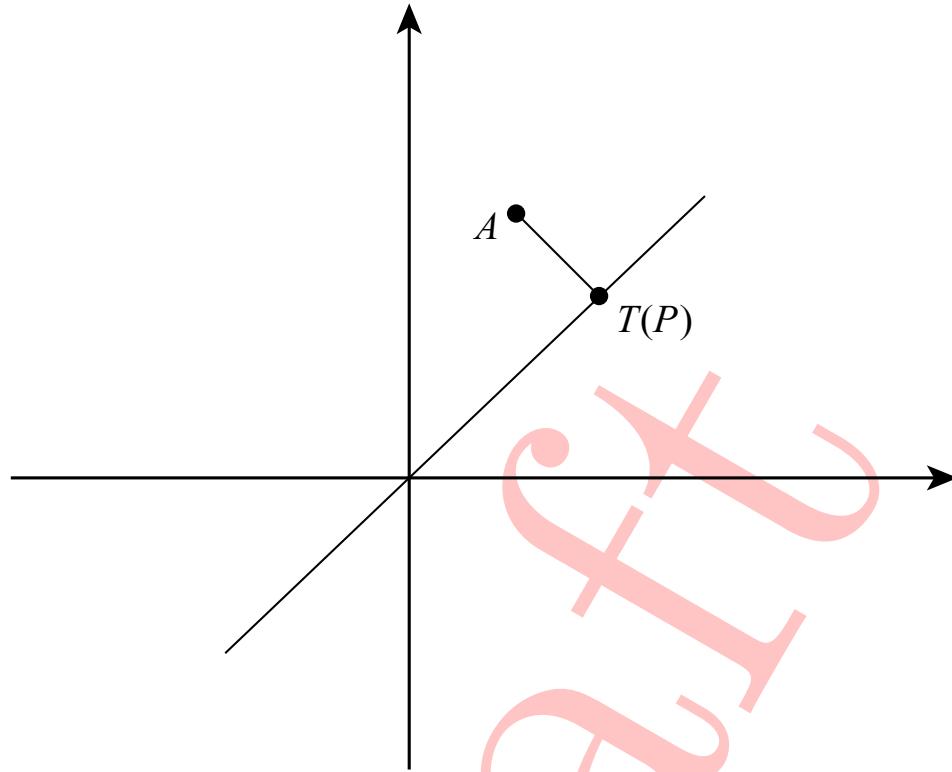


图 1.4: Well-posed formulation of a problem.

the linear transformation given by the matrix

$$\begin{pmatrix} 1 & 3 \\ 2 & 6 \end{pmatrix},$$

$\mathbf{x} = (x, y)$ and $\mathbf{a} = (a, 2a)$. It is easy to see that the image of \mathbb{R}^2 by \mathbf{T} is the line r given by the equation $2x - y = 0$. Because the point \mathbf{a} belongs to the line r , the problem has a solution (in fact, an infinite number of solutions).

However, a small perturbation in the value of a can move the point \mathbf{a} out of the line r , and the problem fails to have a solution.

To avoid this situation, a more appropriate way to pose the problem uses a formulation based on optimization: determine the element $\mathbf{p} \in \mathbb{R}^2$ such that the distance from $T(\mathbf{p})$ to the point \mathbf{a} is minimal. This form of the problem always has a solution. Moreover, this solution will be close to the point \mathbf{a} . Geometrically, the solution is the point \mathbf{p} , such that its image $T(\mathbf{p})$ is the closest point of the line r to the point \mathbf{a} (see Figure 1.4).

1.6.4.2 Problem Reduction

Another important concept related to problem characterization is the notion of *problem reduction*. It constitutes a powerful conceptual device that makes possible to analyze

and compare problems in the same class. Problem reduction also has practical relevance because it enables finding the right algorithm to resolve a problem.

Imagine you have a problem that you do not know how to solve. Very often, using a reduction technique, you can turn this problem into a prototypical one that can be solved. This not only gives you the methodology for implementing the solution but usually comes with guarantees associated with the knowledge of the problem nature.

A variation in the above strategy is when you can decompose a complex problem that you do not know how to solve into the aggregate solution of simpler subproblems that you can solve. This leads to some of the basic algorithm design methodologies in computer science, such as the *divide and conquer* technique.

Moreover, if your problem can be reduced to any instance of a known hard problem, then you can be sure that your problem is at least as hard. This issue will be discussed in more detail in the next section.

1.6.5 How to Solve It?

It is the purpose of this book to use optimization techniques to solve computer graphics problems.

Later, we present several examples of computer graphics problems that can be posed, in a natural way, as optimization problems. Nevertheless, we should remind that posing a problem as an optimization problem does not mean that it is automatically solved, at least in the sense of obtaining an exact solution.

In fact, optimization problems are, in general, very difficult to solve. Some of the difficulties are discussed below.

1.6.5.1 Global Versus Local Solutions

In continuous optimization problems, most of the optimality conditions are related to local optimality problems. The computation of global optimality solutions is, in general, very expensive from the computational point of view. Thus, in practice, we have to settle for the best local optimal solution found by the algorithm.

1.6.5.2 NP³-Complete Problems

There are combinatorial problems — for example, the so-called *NP-complete* problems — for which there are no efficient methods (i.e., running in polynomial time on the size of the instance to be solved) that provide exact solutions in all cases. The traveling salesman problem, mentioned before, is in that class. These problems are related in such

a way that if an algorithm with polynomially bounded running time is found for one of the problems, then polynomial time algorithms will be available for all. This makes it unlikely that such an algorithm will ever be found. When solving such problems, many times, we have to use heuristic solution processes that provide good (but not optimal) solutions.

(Special Optimization Problems: Semi-Definite Programming, Linear Programming, Integer Programming, Polynomial Programming, Fractional Programming, ...)

(The first chapter, Introduction to Optimization, of [109] is also valuable.)



第二章 Preliminaries of Mathematics

2.1 Notations

Notations	Meanings
Bold capital	A matrix.
Bold lowercase	A vector.
Calligraphic capital	A tensor, a subspace, an operator, or a set.
\mathbb{R}, \mathbb{R}^+	Set of real numbers, set of nonnegative real numbers.
$\mathbb{P}(x)$	Probability of event x .
$\mathbb{E}(x)$	Expectation of event x .
$\text{sgn}(x)$	Sign function, -1 if $x < 0$; 1 if $x > 0$; 0 if $x = 0$.
m, n	Size of data matrix \mathbf{D} .
$n_{(1)}, n_{(2)}$	$n_{(1)} = \max\{m, n\}$, $n_{(2)} = \min\{m, n\}$.
$\Theta(n)$	Grows in the same order of n .
$O(n)$	Grows no faster than the order of n .
\otimes	Tensor product.
\oplus	Direct sum of subspaces.
\mathbf{e}_i	Vector whose i -th entry is 1 and others are 0 's.
$\mathbf{I}, \mathbf{0}, \mathbf{1}$	The identity matrix, all-zero matrix or vector, and all-one vector.
\mathbf{D}	Noisy data matrix.
\mathbf{A} or \mathbf{A}_0	Clean data matrix.
\mathbf{E} or \mathbf{E}_0	Noise matrix.
\mathbf{A}^* or \mathbf{E}^*	Optimal solution to a certain optimization problem.
$\mathbf{D}_{:,j}$	The j -th column of matrix \mathbf{D} .
\mathbf{D}_{ij}	The entry at the i -th row and the j -th column of \mathbf{D} .
$\mathbf{D}_{\Omega,:}$	Subsampling the rows of \mathbf{D} whose indices are in Ω .
$\mathbf{D}_{:, \Omega}$	Subsampling the columns of \mathbf{D} whose indices are in Ω .
\mathbf{D}^T	Transpose of matrix \mathbf{D} .
\mathbf{D}^\dagger	Moore-Penrose pseudo-inverse of matrix \mathbf{D} .
$\text{diag}(\mathbf{D})$	Vector whose entries are diagonal entries of matrix \mathbf{D} .
$\text{Diag}(\mathbf{d})$	Diagonal matrix whose diagonal entries are entries of vector \mathbf{d} .
$\text{Range}(\mathbf{D})$	Range space of columns of \mathbf{D} .
$\sigma_i(\mathbf{D})$	The i -th largest singular value of matrix \mathbf{D} .
$\lambda_i(\mathbf{D})$	The i -th largest eigenvalue of matrix \mathbf{D} .

$ \mathbf{D} $	Matrix whose (i, j) -th entry is $ \mathbf{D}_{ij} $.
$\ \cdot\ $	Operator norm of an operator or a matrix.
$\ \cdot\ _2$ or $\ \cdot\ $	ℓ_2 norm of vectors, $\ \mathbf{v}\ _2 = \sqrt{\sum_i \mathbf{v}_i^2}$.
$\ \cdot\ _*$	Nuclear norm of matrices, the sum of singular values.
$\ \cdot\ _{Sp}$	Schatten- p pseudo-norm of matrices, the ℓ_p pseudo-norm of the vector of singular values.
$\ \cdot\ _0$	ℓ_0 pseudo-norm, number of nonzero entries.
$\ \cdot\ _{2,0}$	$\ell_{2,0}$ pseudo-norm of matrices, number of nonzero columns.
$\ \cdot\ _1$	ℓ_1 norm, $\ \mathbf{D}\ _1 = \sum_{i,j} \mathbf{D}_{ij} $.
$\ \cdot\ _p$	ℓ_p norm, $\ \mathbf{D}\ _p = \left(\sum_{i,j} \mathbf{D}_{ij} ^p \right)^{1/p}$.
$\ \cdot\ _{2,1}$	$\ell_{2,1}$ norm, $\ \mathbf{D}\ _{2,1} = \sum_j \ \mathbf{D}_{:j}\ _2$.
$\ \cdot\ _{2,p}$	$\ell_{2,p}$ norm, $\ \mathbf{D}\ _{2,p} = \left(\sum_j \ \mathbf{D}_{:j}\ _2^p \right)^{1/p}$.
$\ \cdot\ _{2,\infty}$	$\ell_{2,\infty}$ norm, $\ \mathbf{D}\ _{2,\infty} = \max_j \ \mathbf{D}_{:j}\ _2$.
$\ \cdot\ _F$	Frobenius norm of a matrix or a tensor, $\ \mathbf{D}\ _F = \sqrt{\sum_{i,j} \mathbf{D}_{ij}^2}$.
$\ \cdot\ _\infty$	ℓ_∞ norm, $\ \mathbf{D}\ _\infty = \max_{ij} \mathbf{D}_{ij} $.
∂f	Subgradient (resp. supergradient) of a convex (resp. concave) function f .
$\hat{\mathbf{U}}, \hat{\mathbf{V}}$	Left and right singular vectors of $\hat{\mathbf{A}}$.
$\mathcal{U}_0, \hat{\mathcal{U}}, \mathcal{U}^*$	Column space of $\mathbf{A}_0, \hat{\mathbf{A}}, \mathbf{A}^*$.
$\mathcal{V}_0, \hat{\mathcal{V}}, \mathcal{V}^*$	Row space of $\mathbf{A}_0, \hat{\mathbf{A}}, \mathbf{A}^*$.
$\hat{\mathcal{T}}$	Space $\hat{\mathcal{T}} = \left\{ \hat{\mathbf{U}}\mathbf{X}^T + \mathbf{Y}\hat{\mathbf{V}}^T, \forall \mathbf{X}, \mathbf{Y} \in \mathbb{R}^{n \times r} \right\}$, sum of the column space and the row space.
\mathcal{U}^\perp	Orthogonal complement of the space \mathcal{U} .
$\mathcal{I}_0, \hat{\mathcal{I}}, \mathcal{I}^*$	Indices of outliers (nonzero columns) of $\mathbf{E}_0, \hat{\mathbf{E}}, \mathbf{E}^*$.
$\mathcal{P}_{\hat{\mathcal{U}}}(\cdot), \mathcal{P}_{\hat{\mathcal{V}}}(\cdot)$	$\mathcal{P}_{\hat{\mathcal{U}}}(\mathbf{D}) = \hat{\mathbf{U}}\hat{\mathbf{U}}^T\mathbf{D}, \mathcal{P}_{\hat{\mathcal{V}}}(\mathbf{D}) = \mathbf{D}\hat{\mathbf{V}}\hat{\mathbf{V}}^T$.
$\mathcal{P}_{\hat{\mathcal{T}}^\perp}(\cdot)$	$\mathcal{P}_{\hat{\mathcal{T}}^\perp}(\mathbf{D}) = \mathcal{P}_{\hat{\mathcal{U}}^\perp}\mathcal{P}_{\hat{\mathcal{V}}^\perp}\mathbf{D}$.
$\mathcal{P}_\Omega(\cdot)$	$(\mathcal{P}_\Omega(\mathbf{D}))_{ij} = \begin{cases} \mathbf{D}_{ij}, & \text{if } (i, j) \in \Omega, \\ 0, & \text{if } (i, j) \notin \Omega. \end{cases}$
$\mathcal{P}_{\mathcal{I}}(\cdot)$	$\mathcal{P}_{\mathcal{I}}\mathbf{D} = \mathbf{D}_{:\mathcal{I}}$.
$\mathcal{S}_\varepsilon(\cdot)$	Soft thresholding operator, $\mathcal{S}_\varepsilon(x) = \text{sgn}(x) \max(x - \varepsilon, 0)$.
$\mathcal{A}^*(\cdot)$	Adjoint operator of $\mathcal{A}(\cdot)$.
$\mathcal{B}(\hat{\mathbf{E}})$	Operator normalizing nonzero columns of $\hat{\mathbf{E}}$, $\mathcal{B}(\hat{\mathbf{E}}) = \left\{ \mathbf{H} \middle \mathcal{P}_{\hat{\mathcal{T}}^\perp}(\mathbf{H}) = \mathbf{0}; \mathbf{H}_{:j} = \frac{\hat{\mathbf{E}}_{:j}}{\ \hat{\mathbf{E}}_{:j}\ _2}, j \in \hat{\mathcal{I}} \right\}.$
$\text{Prox}_\alpha f(\cdot)$	Proximal operator w.r.t. f and parameter α ,

$\text{Prox}_\alpha f(\mathbf{w}) = \operatorname{argmin}_{\mathbf{x}} \alpha f(\mathbf{x}) + \frac{1}{2} \ \mathbf{x} - \mathbf{w}\ _2^2.$	
$\text{Ber}(p)$	Bernoulli distribution with parameter p .
$\mathcal{N}(a, b^2)$	Gaussian distribution with mean a and variance b^2 .
$\text{Unif}(k)$	Uniform distribution on the set of subsets of cardinality k .

2.2 Sequences and Limits

(Taken from Chapter 5.1 of [30])

A *sequence of real numbers* is a function whose domain is the set of natural numbers $1, 2, \dots, k, \dots$ and whose range is contained in \mathbb{R} . Thus, a sequence of real numbers can be viewed as a set of numbers $\{x_1, x_2, \dots, x_k, \dots\}$, which is often also denoted as $\{x_k\}$ (or sometimes as $\{x_k\}_{k=1}^\infty$, to indicate explicitly the range of values that k can take).

A sequence $\{x_k\}$ is *increasing* if $x_1 < x_2 < \dots < x_k < \dots$; that is, $x_k < x_{k+1}$ for all k . If $x_k \leq x_{k+1}$, then we say that the sequence is *nondecreasing*. Similarly, we can define *decreasing* and *nonincreasing* sequences. Nonincreasing or nondecreasing sequences are called *monotone* sequences.

A number $x^* \in \mathbb{R}$ is called the *limit* of the sequence $\{x_k\}$ if for any positive $\varepsilon > 0$ there is a number K (which may depend on ε) such that for all $k > K$, $|x_k - x^*| < \varepsilon$ that is, x_k lies between $x^* - \varepsilon$ and $x^* + \varepsilon$ for all $k > K$. In this case we write

$$x^* = \lim_{k \rightarrow \infty} x_k$$

or

$$x_k \rightarrow x^*.$$

A sequence that has a limit is called a *convergent* sequence.

The notion of a sequence can be extended to sequences with elements in \mathbb{R}^n . Specifically, a sequence in \mathbb{R}^n is a function whose domain is the set of natural numbers $1, 2, \dots, k, \dots$ and whose range is contained in \mathbb{R}^n . We use the notation $\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots\}$ or $\{\mathbf{x}^{(k)}\}$ for sequences in \mathbb{R}^n . For limits of sequences in \mathbb{R}^n , we need to replace absolute values with vector norms. In other words, \mathbf{x}^* is the limit of $\{\mathbf{x}^{(k)}\}$ if for any positive $\varepsilon > 0$ there is a number K (which may depend on ε) such that for all $k > K$, $\|\mathbf{x}^{(k)} - \mathbf{x}^*\| < \varepsilon$. As before, if a sequence $\{\mathbf{x}^{(k)}\}$ is convergent, we write $\mathbf{x}^* = \lim_{k \rightarrow \infty} \mathbf{x}^{(k)}$ or $\mathbf{x}^{(k)} \rightarrow \mathbf{x}^*$.

Theorem 12. *A convergent sequence has only one limit.*

Proof. We prove this result by contradiction. Suppose that a sequence $\{\mathbf{x}^{(k)}\}$ has two different limits, say \mathbf{x}_1 and \mathbf{x}_2 . Then, we have $\|\mathbf{x}_1 - \mathbf{x}_2\| > 0$. Let

$$\varepsilon = \frac{1}{2} \|\mathbf{x}_1 - \mathbf{x}_2\|.$$

From the definition of a limit, there exist K_1 and K_2 such that for $k > K_1$ we have $\|\mathbf{x}^{(k)} - \mathbf{x}_1\| < \varepsilon$, and for $k > K_2$ we have $\|\mathbf{x}^{(k)} - \mathbf{x}_2\| < \varepsilon$. Let $K = \max\{K_1, K_2\}$. Then, if $k > K$, we have $\|\mathbf{x}^{(k)} - \mathbf{x}_1\| < \varepsilon$ and $\|\mathbf{x}^{(k)} - \mathbf{x}_2\| < \varepsilon$. Adding $\|\mathbf{x}^{(k)} - \mathbf{x}_1\| < \varepsilon$ and $\|\mathbf{x}^{(k)} - \mathbf{x}_2\| < \varepsilon$ yields

$$\|\mathbf{x}^{(k)} - \mathbf{x}_1\| + \|\mathbf{x}^{(k)} - \mathbf{x}_2\| < 2\varepsilon.$$

Applying the triangle inequality gives

$$\begin{aligned}\|-\mathbf{x}_1 + \mathbf{x}_2\| &= \|(\mathbf{x}^{(k)} - \mathbf{x}_1) - (\mathbf{x}^{(k)} - \mathbf{x}_2)\| \\ &\leq \|\mathbf{x}^{(k)} - \mathbf{x}_1\| + \|\mathbf{x}^{(k)} - \mathbf{x}_2\|.\end{aligned}\tag{2.1}$$

Therefore,

$$\|-\mathbf{x}_1 + \mathbf{x}_2\| = \|\mathbf{x}_1 - \mathbf{x}_2\| < 2\varepsilon.$$

However, this contradicts the assumption that $\|\mathbf{x}_1 - \mathbf{x}_2\| = 2\varepsilon$, which completes the proof. \square

A sequence $\{\mathbf{x}^{(k)}\}$ in \mathbb{R}^n is *bounded* if there exists a number $B \geq 0$ such that $\|\mathbf{x}^{(k)}\| \leq B$ for all $k = 1, 2, \dots$.

Theorem 13. *Every convergent sequence is bounded.*

Proof. Let $\{\mathbf{x}^{(k)}\}$ be a convergent sequence with limit \mathbf{x}^* . Choose $\varepsilon = 1$. Then, by definition of the limit, there exists a natural number K such that for all $k > K$,

$$\|\mathbf{x}^{(k)} - \mathbf{x}^*\| < 1.$$

Therefore,

$$\|\mathbf{x}^{(k)}\| < \|\mathbf{x}^*\| + 1, \quad \text{for all } k > K.$$

Letting

$$B = \max \{\|\mathbf{x}^{(1)}\|, \|\mathbf{x}^{(2)}\|, \dots, \|\mathbf{x}^{(K)}\|, \|\mathbf{x}^*\| + 1\},$$

we have

$$B > \|\mathbf{x}^{(k)}\| \quad \text{for all } k,$$

which means that the sequence $\{\mathbf{x}^*\}$ is bounded. \square

For a sequence $\{x_k\}$ in \mathbb{R} , a number B is called an *upper bound* if $x_k \leq B$ for all $k = 1, 2, \dots$. In this case, we say that $\{x_k\}$ is bounded above. Similarly, B is called a *lower bound* if $x_k \geq B$ for all $k = 1, 2, \dots$. In this case, we say that $\{x_k\}$ is bounded below. Clearly, a sequence is bounded if it is both bounded above and bounded below.

Any sequence $\{x_k\}$ in \mathbb{R} that has an upper bound has a *least upper bound* (also called the *supremum*), which is the smallest number B that is an upper bound of $\{x_k\}$. Similarly, any sequence $\{x_k\}$ in \mathbb{R} that has a lower bound has a *greatest lower bound* (also called the *infimum*). If B is the least upper bound of the sequence $\{x_k\}$, then $x_k \leq B$ for all k , and for any $\varepsilon > 0$, there exists a number K such that $x_K > B - \varepsilon$. An analogous statement applies to the greatest lower bound: If B is the greatest lower bound of $\{x_k\}$ then $x_k \geq B$ for all k , and for any $\varepsilon > 0$, there exists a number K such that $x_K < B + \varepsilon$.

Theorem 14. *Every monotone bounded sequence in \mathbb{R} is convergent.*

Proof. We prove the theorem for nondecreasing sequences. The proof for nonincreasing sequences is analogous.

Let $\{x_k\}$ be a bounded nondecreasing sequence in \mathbb{R} and x^* the least upper bound. Fix a number $\varepsilon > 0$. Then, there exists a number K such that $x_K > x^* - \varepsilon$. Because $\{x_k\}$ is nondecreasing, for any $k > K$,

$$x_k > x_K > x^* - \varepsilon.$$

Also, because x^* is an upper bound of $\{x_k\}$, we have

$$x_k \leq x^* < x^* + \varepsilon.$$

Therefore, for any $k > K$,

$$|x_k - x^*| < \varepsilon,$$

which means that $x_k \rightarrow x^*$. □

Suppose that we are given a sequence $\{\mathbf{x}^{(k)}\}$ and an increasing sequence of natural numbers $\{m_k\}$. The sequence

$$\{\mathbf{x}^{(m_k)}\} = \{\mathbf{x}^{(m_1)}, \mathbf{x}^{(m_2)}, \dots\}$$

is called a *subsequence* of the sequence $\{\mathbf{x}^{(k)}\}$. A subsequence of a given sequence can thus be obtained by neglecting some elements of the given sequence.

Theorem 15. *Consider a convergent sequence $\{\mathbf{x}^{(k)}\}$ with limit \mathbf{x}^* . Then, any subsequence of $\{\mathbf{x}^{(k)}\}$ also converges to \mathbf{x}^* .*

Proof. Let $\{\mathbf{x}^{(m_k)}\}$ be a subsequence of $\{\mathbf{x}^{(k)}\}$, where $\{m_k\}$ is an increasing sequence of natural numbers. Observe that $m_k > k$ for all $k = 1, 2, \dots$. To show this, first note that $m_1 \geq 1$ because m_1 is a natural number. Next, we proceed by induction by assuming that

$m_k \geq k$. Then, we have $m_{k+1} > m_k \geq k$, which implies that $m_{k+1} \geq k + 1$. Therefore, we have shown that $m_k \geq k$ for all $k = 1, 2, \dots$.

Let $\varepsilon > 0$ be given. Then, by definition of the limit, there exists K such that $\|\mathbf{x}^{(k)} - \mathbf{x}^*\| < \varepsilon$ for any $k > K$. Because $m_k \geq k$, we also have $\|\mathbf{x}^{(m_k)} - \mathbf{x}^*\| < \varepsilon$ for any $k > K$. This means that

$$\lim_{k \rightarrow \infty} \mathbf{x}^{(m_k)} = \mathbf{x}^*.$$

□

It turns out that any bounded sequence contains a convergent subsequence. This result is called the *Bolzano-Weierstrass theorem*.

Consider a function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and a point $\mathbf{x}_0 \in \mathbb{R}^n$. Suppose that there exists f^* such that for any convergent sequence $\{\mathbf{x}^{(k)}\}$ with limit \mathbf{x}_0 , we have

$$\lim_{k \rightarrow \infty} f(\mathbf{x}^{(k)}) = f^*.$$

Then, we use the notation

$$\lim_{\mathbf{x} \rightarrow \mathbf{x}_0} f(\mathbf{x})$$

to represent the limit f^* .

It turns out that f is continuous at \mathbf{x}_0 if and only if for any convergent sequence $\{\mathbf{x}^{(k)}\}$ with limit \mathbf{x}_0 , we have

$$\lim_{k \rightarrow \infty} f(\mathbf{x}^{(k)}) = f\left(\lim_{k \rightarrow \infty} \mathbf{x}^{(k)}\right) = f(\mathbf{x}_0).$$

Therefore, using the notation introduced above, the function f is continuous at \mathbf{x}_0 if and only if

$$\lim_{\mathbf{x} \rightarrow \mathbf{x}_0} f(\mathbf{x}) = f(\mathbf{x}_0).$$

We end this section with some results involving sequences and limits of matrices. These results are useful in the analysis of algorithms (e.g., Newton's algorithm in Chapter 5.3).

We say that a sequence $\{\mathbf{A}_k\}$ of $m \times n$ matrices converges to the $m \times n$ matrix \mathbf{A} if

$$\lim_{k \rightarrow \infty} \|\mathbf{A} - \mathbf{A}_k\| = 0.$$

Lemma 16. Let $\mathbf{A} \in \mathbb{R}^{n \times n}$. Then, $\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{O}$ if and only if the eigenvalues of \mathbf{A} satisfy $|\lambda_i(\mathbf{A})| < 1$, $i = 1, \dots, n$.

Proof. To prove this theorem, we use the Jordan form (see, e.g., [53]). Specifically, it is well known that any square matrix is similar to the Jordan form: There exists a nonsingular \mathbf{T} such that

$$\mathbf{T}\mathbf{A}\mathbf{T}^{-1} = \text{diag}[\mathbf{J}_{m_1}(\lambda_1), \dots, \mathbf{J}_{m_s}(\lambda_1), \mathbf{J}_{n_1}(\lambda_2), \dots, \mathbf{J}_{t_v}(\lambda_q)] \triangleq \mathbf{J},$$

where $\mathbf{J}_r(\lambda)$ is the $r \times r$ matrix:

$$\mathbf{J}_r(\lambda) = \begin{bmatrix} \lambda & 1 & & 0 \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ 0 & & & \lambda \end{bmatrix}.$$

The $\lambda_1, \dots, \lambda_q$ above are distinct eigenvalues of \mathbf{A} , the multiplicity of λ_1 is $m_1 + \dots + m_s$, and so on.

We may rewrite the above as $\mathbf{A} = \mathbf{T}^{-1}\mathbf{J}\mathbf{T}$. To complete the proof, observe that

$$(\mathbf{J}_r(\lambda))^k = \begin{bmatrix} \lambda^k & \binom{k}{k-1}\lambda^{k-1} & \cdots & \binom{k}{k-r+1}\lambda^{k-r+1} \\ 0 & \lambda^k & \cdots & \binom{k}{k-r+2}\lambda^{k-r+2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda^k \end{bmatrix},$$

where

$$\binom{k}{i} = \frac{k!}{i!(k-i)!}.$$

Furthermore,

$$\mathbf{A}^k = \mathbf{T}^{-1}\mathbf{J}^k\mathbf{T}.$$

Hence,

$$\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{T}^{-1} \left(\lim_{k \rightarrow \infty} \mathbf{J}^k \right) \mathbf{T} = \mathbf{O}$$

if and only if $|\lambda_i(\mathbf{A})| < 1$, $i = 1, \dots, n$.

□

Lemma 17. *The series of $n \times n$ matrices*

$$\mathbf{I}_n + \mathbf{A} + \mathbf{A}^2 + \cdots + \mathbf{A}^k + \cdots$$

converges if and only if $\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{O}$. In this case the sum of the series equals $(\mathbf{I}_n - \mathbf{A})^{-1}$.

Proof. The necessity of the condition is obvious.

To prove the sufficiency, suppose that $\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{O}$. By Lemma 16 we deduce that $|\lambda_i(\mathbf{A})| < 1$, $i = 1, \dots, n$. This implies that $\det(\mathbf{I}_n - \mathbf{A}) \neq 0$, and hence $(\mathbf{I}_n - \mathbf{A})^{-1}$ exists. Consider now the following relation:

$$(\mathbf{I}_n + \mathbf{A} + \mathbf{A}^2 + \cdots + \mathbf{A}^k)(\mathbf{I}_n - \mathbf{A}) = \mathbf{I}_n - \mathbf{A}^{k+1}.$$

Postmultiplying the equation above by $(\mathbf{I}_n - \mathbf{A})^{-1}$ yields

$$\mathbf{I}_n + \mathbf{A} + \mathbf{A}^2 + \cdots + \mathbf{A}^k = (\mathbf{I}_n - \mathbf{A})^{-1} - \mathbf{A}^{k+1}(\mathbf{I}_n - \mathbf{A})^{-1}.$$

Hence,

$$\lim_{k \rightarrow \infty} \sum_{j=0}^k \mathbf{A}^j = (\mathbf{I}_n - \mathbf{A})^{-1},$$

because $\lim_{k \rightarrow \infty} \mathbf{A}^{k+1} = \mathbf{O}$. Thus,

$$\sum_{j=0}^{\infty} \mathbf{A}^j = (\mathbf{I}_n - \mathbf{A})^{-1},$$

which completes the proof. \square

A matrix-valued function $\mathbf{A} : \mathbb{R}^r \rightarrow \mathbb{R}^{n \times n}$ is continuous at a point $\xi_0 \in \mathbb{R}^r$ if

$$\lim_{\|\xi - \xi_0\| \rightarrow 0} \|\mathbf{A}(\xi) - \mathbf{A}(\xi_0)\| = 0.$$

Lemma 18. Let $\mathbf{A} : \mathbb{R}^r \rightarrow \mathbb{R}^{n \times n}$ be an $n \times n$ matrix-valued function that is continuous at ξ_0 . If $\mathbf{A}(\xi_0)^{-1}$ exists, then $\mathbf{A}(\xi)^{-1}$ exists for ξ sufficiently close to ξ_0 and $\mathbf{A}(\cdot)^{-1}$ is continuous at ξ_0 .

Proof. We first prove the existence of $\mathbf{A}(\xi)^{-1}$ for all ξ sufficiently close to ξ_0 . We have

$$\mathbf{A}(\xi) = \mathbf{A}(\xi_0) - \mathbf{A}(\xi_0) + \mathbf{A}(\xi) = \mathbf{A}(\xi_0)(\mathbf{I}_n - \mathbf{K}(\xi)),$$

where

$$\mathbf{K}(\xi) = \mathbf{A}(\xi_0)^{-1}(\mathbf{A}(\xi_0) - \mathbf{A}(\xi)).$$

Thus,

$$\|\mathbf{K}(\xi)\| \leq \|\mathbf{A}(\xi_0)^{-1}\| \|\mathbf{A}(\xi_0) - \mathbf{A}(\xi)\|$$

and

$$\lim_{\|\xi - \xi_0\| \rightarrow 0} \|\mathbf{K}(\xi)\| = 0.$$

Because \mathbf{A} is continuous at ξ_0 , for all ξ close to enough ξ_0 , we have

$$\|\mathbf{A}(\xi_0) - \mathbf{A}(\xi)\| \leq \frac{\theta}{\|\mathbf{A}(\xi_0)^{-1}\|},$$

where $\theta \in (0, 1)$. Then,

$$\|\mathbf{K}(\xi)\| \leq \theta < 1$$

and

$$(\mathbf{I}_n - \mathbf{K}(\xi))^{-1}$$

exists. But then

$$\mathbf{A}(\boldsymbol{\xi})^{-1} = (\mathbf{A}(\boldsymbol{\xi}_0)(\mathbf{I}_n - \mathbf{K}(\boldsymbol{\xi})))^{-1} = (\mathbf{I}_n - \mathbf{K}(\boldsymbol{\xi}))^{-1}\mathbf{A}(\boldsymbol{\xi}_0)^{-1},$$

which means that $\mathbf{A}(\boldsymbol{\xi})^{-1}$ exists for $\boldsymbol{\xi}$ sufficiently close to $\boldsymbol{\xi}_0$.

To prove the continuity of $\mathbf{A}(\cdot)^{-1}$ note that

$$\begin{aligned}\|\mathbf{A}(\boldsymbol{\xi}_0)^{-1} - \mathbf{A}(\boldsymbol{\xi})^{-1}\| &= \|\mathbf{A}(\boldsymbol{\xi})^{-1} - \mathbf{A}(\boldsymbol{\xi}_0)^{-1}\| \\ &= \|((\mathbf{I}_n - \mathbf{K}(\boldsymbol{\xi}))^{-1} - \mathbf{I}_n)\mathbf{A}(\boldsymbol{\xi}_0)^{-1}\|.\end{aligned}$$

However, since $\|\mathbf{K}(\boldsymbol{\xi})\| < 1$, it follows from Lemma 17 that

$$(\mathbf{I}_n - \mathbf{K}(\boldsymbol{\xi}))^{-1} - \mathbf{I}_n = \mathbf{K}(\boldsymbol{\xi}) + \mathbf{K}^2(\boldsymbol{\xi}) + \cdots = \mathbf{K}(\boldsymbol{\xi})(\mathbf{I}_n + \mathbf{K}(\boldsymbol{\xi}) + \cdots).$$

Hence,

$$\begin{aligned}\|(\mathbf{I}_n - \mathbf{K}(\boldsymbol{\xi}))^{-1} - \mathbf{I}_n\| &\leq \|\mathbf{K}(\boldsymbol{\xi})\|(1 + \|\mathbf{K}(\boldsymbol{\xi})\| + \|\mathbf{K}(\boldsymbol{\xi})\|^2 + \cdots) \\ &= \frac{\|\mathbf{K}(\boldsymbol{\xi})\|}{1 - \|\mathbf{K}(\boldsymbol{\xi})\|},\end{aligned}$$

where $\|\mathbf{K}(\boldsymbol{\xi})\| < 1$. Therefore,

$$\|\mathbf{A}(\boldsymbol{\xi})^{-1} - \mathbf{A}(\boldsymbol{\xi}_0)^{-1}\| \leq \frac{\|\mathbf{K}(\boldsymbol{\xi})\|}{1 - \|\mathbf{K}(\boldsymbol{\xi})\|} \|\mathbf{A}(\boldsymbol{\xi}_0)^{-1}\|.$$

Because

$$\lim_{\|\boldsymbol{\xi} - \boldsymbol{\xi}_0\| \rightarrow 0} \|\mathbf{K}(\boldsymbol{\xi})\| = 0,$$

we obtain

$$\lim_{\|\boldsymbol{\xi} - \boldsymbol{\xi}_0\| \rightarrow 0} \|\mathbf{A}(\boldsymbol{\xi})^{-1} - \mathbf{A}(\boldsymbol{\xi}_0)^{-1}\| = 0,$$

which completes the proof. \square

2.3 Differentiability

Differential calculus is based on the idea of approximating an arbitrary function by an affine function. A function $\mathcal{A} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is affine if there exists a linear function $\mathcal{L} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and a vector $\mathbf{y} \in \mathbb{R}^m$ such that

$$\mathcal{A}(\mathbf{x}) = \mathcal{L}(\mathbf{x}) + \mathbf{y}$$

for every $\mathbf{x} \in \mathbb{R}^n$. Consider a function $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and a point $\mathbf{x}_0 \in \mathbb{R}^n$. We wish to find an affine function \mathcal{A} that approximates \mathbf{f} near the point \mathbf{x}_0 . First, it is natural to impose the condition

$$\mathcal{A}(\mathbf{x}_0) = \mathbf{f}(\mathbf{x}_0).$$

Because $\mathcal{A}(\mathbf{x}) = \mathcal{L}(\mathbf{x}) + \mathbf{y}$, we obtain $\mathbf{y} = \mathbf{f}(\mathbf{x}_0) - \mathcal{L}(\mathbf{x}_0)$. By the linearity of \mathcal{L} ,

$$\mathcal{L}(\mathbf{x}) + \mathbf{y} = \mathcal{L}(\mathbf{x}) - \mathcal{L}(\mathbf{x}_0) + \mathbf{f}(\mathbf{x}_0) = \mathcal{L}(\mathbf{x} - \mathbf{x}_0) + \mathbf{f}(\mathbf{x}_0).$$

Hence, we may write

$$\mathcal{A}(\mathbf{x}) = \mathcal{L}(\mathbf{x} - \mathbf{x}_0) + \mathbf{f}(\mathbf{x}_0).$$

Next, we require that $\mathcal{A}(\mathbf{x})$ approaches $\mathbf{f}(\mathbf{x})$ faster than \mathbf{x} approaches \mathbf{x}_0 ; that is,

$$\lim_{\mathbf{x} \rightarrow \mathbf{x}_0, \mathbf{x} \in \Omega} \frac{\|\mathbf{f}(\mathbf{x}) - \mathcal{A}(\mathbf{x})\|}{\|\mathbf{x} - \mathbf{x}_0\|} = 0.$$

The conditions above on \mathcal{A} ensure that \mathcal{A} approximates \mathbf{f} near \mathbf{x}_0 in the sense that the error in the approximation at a given point is “small” compared with the distance of the point from \mathbf{x}_0 .

In summary, a function $\mathbf{f} : \Omega \rightarrow \mathbb{R}^m$, $\Omega \subset \mathbb{R}^n$, is said to be differentiable at $\mathbf{x}_0 \in \Omega$ if there is an affine function that approximates \mathbf{f} near \mathbf{x}_0 ; that is, there exists a linear function $\mathcal{L} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ such that

$$\lim_{\mathbf{x} \rightarrow \mathbf{x}_0, \mathbf{x} \in \Omega} \frac{\|\mathbf{f}(\mathbf{x}) - (\mathcal{L}(\mathbf{x} - \mathbf{x}_0) + \mathbf{f}(\mathbf{x}_0))\|}{\|\mathbf{x} - \mathbf{x}_0\|} = 0.$$

The linear function \mathcal{L} above is determined uniquely by \mathbf{f} and \mathbf{x}_0 and is called the derivative of \mathbf{f} at \mathbf{x}_0 . The function \mathbf{f} is said to be differentiable on Ω if \mathbf{f} is differentiable at every point of its domain Ω .

In \mathbb{R} , an affine function has the form $ax + b$, with $a, b \in \mathbb{R}$. Hence, a real-valued function $f(x)$ of a real variable x that is differentiable at x_0 can be approximated near x_0 by a function

$$\mathcal{A}(x) = ax + b.$$

Because $f(x_0) = \mathcal{A}(x_0) = ax_0 + b$, we obtain

$$\mathcal{A}(x) = ax + b = a(x - x_0) + f(x_0).$$

The linear part of $\mathcal{A}(x)$, denoted earlier by $\mathcal{L}(x)$, is in this case just ax . The norm of a real number is its absolute value, so by the definition of differentiability we have

$$\lim_{x \rightarrow x_0} \frac{|f(x) - (a(x - x_0) + f(x_0))|}{|x - x_0|} = 0,$$

which is equivalent to

$$\lim_{x \rightarrow x_0} \frac{f(x) - f(x_0)}{x - x_0} = a,$$

The number a is commonly denoted $f'(x_0)$ and is called the derivative of f at x_0 . The affine function \mathcal{A} is therefore given by

$$\mathcal{A}(x) = f(x_0) + f'(x_0)(x - x_0).$$

This affine function is tangent to f at x_0 (see Figure 2.1).

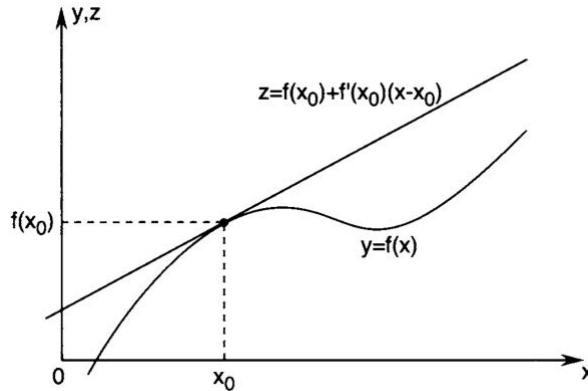


图 2.1: Illustration of the notion of the derivative.

2.4 The Derivative Matrix

Any linear transformation from \mathbb{R}^n to \mathbb{R}^m , and in particular the derivative \mathcal{L} of $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, can be represented by an $m \times n$ matrix. To find the matrix representation \mathbf{L} of the derivative \mathcal{L} of a differentiable function $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, we use the natural basis $\{\mathbf{e}_1, \dots, \mathbf{e}_n\}$ for \mathbb{R}^n . Consider the vectors

$$\mathbf{x}_j = \mathbf{x}_0 + t\mathbf{e}_j, j = 1, \dots, n.$$

By the definition of the derivative, we have

$$\lim_{t \rightarrow 0} \frac{\mathbf{f}(\mathbf{x}_j) - (\mathbf{t}\mathbf{L}\mathbf{e}_j + \mathbf{f}(\mathbf{x}_0))}{t} = \mathbf{0}$$

for $j = 1, \dots, n$. This means that

$$\lim_{t \rightarrow 0} \frac{\mathbf{f}(\mathbf{x}_j) - \mathbf{f}(\mathbf{x}_0)}{t} = \mathbf{L}\mathbf{e}_j$$

for $j = 1, \dots, n$. But $\mathbf{L}\mathbf{e}_j$ is the j th column of the matrix \mathbf{L} . On the other hand, the vector \mathbf{x}_j differs from \mathbf{x}_0 only in the j th coordinate, and in that coordinate the difference is just the number t . Therefore, the left side of the preceding equation is the partial derivative

$$\frac{\partial \mathbf{f}}{\partial x_j}(\mathbf{x}_0).$$

Because vector limits are computed by taking the limit of each coordinate function, it follows that if

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{bmatrix},$$

then

$$\frac{\partial \mathbf{f}}{\partial x_j}(\mathbf{x}_0) = \begin{bmatrix} \frac{\partial f_1}{\partial x_j}(\mathbf{x}_0) \\ \vdots \\ \frac{\partial f_m}{\partial x_j}(\mathbf{x}_0) \end{bmatrix},$$

and the matrix \mathbf{L} has the form

$$\begin{bmatrix} \frac{\partial \mathbf{f}}{\partial x_1}(\mathbf{x}_0) & \cdots & \frac{\partial \mathbf{f}}{\partial x_n}(\mathbf{x}_0) \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(\mathbf{x}_0) & \cdots & \frac{\partial f_1}{\partial x_n}(\mathbf{x}_0) \\ \vdots & & \vdots \\ \frac{\partial f_m}{\partial x_1}(\mathbf{x}_0) & \cdots & \frac{\partial f_m}{\partial x_n}(\mathbf{x}_0) \end{bmatrix}.$$

The matrix \mathbf{L} is called the Jacobian matrix, or derivative matrix, of \mathbf{f} at \mathbf{x}_0 , and is denoted $D\mathbf{f}(\mathbf{x}_0)$. For convenience, we often refer to $D\mathbf{f}(\mathbf{x}_0)$ simply as the derivative of \mathbf{f} at \mathbf{x}_0 . We summarize the foregoing discussion in the following theorem.

Theorem 19. *If a function $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is differentiable at \mathbf{x}_0 , then the derivative of \mathbf{f} at \mathbf{x}_0 is determined uniquely and is represented by the $m \times n$ derivative matrix $D\mathbf{f}(\mathbf{x}_0)$. The best affine approximation to \mathbf{f} near \mathbf{x}_0 is then given by*

$$\mathcal{A}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_0) + D\mathbf{f}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0),$$

in the sense that

$$\mathbf{f}(\mathbf{x}) = \mathcal{A}(\mathbf{x}) + \mathbf{r}(\mathbf{x})$$

and $\lim_{\mathbf{x} \rightarrow \mathbf{x}_0} \frac{\|\mathbf{r}(\mathbf{x})\|}{\|\mathbf{x} - \mathbf{x}_0\|} = 0$. The columns of the derivative matrix $D\mathbf{f}(\mathbf{x}_0)$ are vector partial derivatives. The vector

$$\frac{\partial \mathbf{f}}{\partial x_j}(\mathbf{x}_0)$$

is a tangent vector at \mathbf{x}_0 to the curve \mathbf{f} obtained by varying only the j th coordinate of \mathbf{x} .

If $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is differentiable, then the function ∇f defined by

$$\nabla f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(\mathbf{x}) \\ \vdots \\ \frac{\partial f}{\partial x_n}(\mathbf{x}) \end{bmatrix} = Df(\mathbf{x})^T$$

is called the gradient of f . The gradient is a function from \mathbb{R}^n to \mathbb{R}^n , and can be pictured as a vector field, by drawing the arrow representing $\nabla f(\mathbf{x})$ so that its tail starts at \mathbf{x} .

Given $f : \mathbb{R}^n \rightarrow \mathbb{R}$, if ∇f is differentiable, we say that f is twice differentiable, and we write the derivative of ∇f as

$$D^2 f = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_2 \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_1} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_1 \partial x_n} & \frac{\partial^2 f}{\partial x_2 \partial x_n} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}.$$

(The notation $\frac{\partial^2 f}{\partial x_i \partial x_j}$ represents taking the partial derivative of f with respect to x_j first, then with respect to x_i .) The matrix $D^2 f(\mathbf{x})$ is called the Hessian matrix of f at \mathbf{x} , and is often also denoted $\mathbf{F}(\mathbf{x})$.

A function $\mathbf{f} : \Omega \rightarrow \mathbb{R}^m$, $\Omega \subset \mathbb{R}^n$, is said to be continuously differentiable on Ω if it is differentiable (on Ω), and $D\mathbf{f} : \Omega \rightarrow \mathbb{R}^{m \times n}$ is continuous; that is, the components of \mathbf{f} have continuous partial derivatives. In this case, we write $\mathbf{f} \in \mathcal{C}^1$. If the components of \mathbf{f} have continuous partial derivatives of order p , then we write $\mathbf{f} \in \mathcal{C}^p$.

Note that the Hessian matrix of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ at \mathbf{x} is symmetric if f is twice continuously differentiable at \mathbf{x} . This is a well-known result from calculus called Clairaut's theorem or Schwarz's theorem. However, if the second partial derivatives of f are not continuous, then there is no guarantee that the Hessian is symmetric, as shown in the following well-known example.

Example 20. Consider the function

$$f(\mathbf{x}) = \begin{cases} x_1 x_2 (x_1^2 - x_2^2) / (x_1^2 + x_2^2), & \text{if } \mathbf{x} \neq \mathbf{0}, \\ 0, & \text{if } \mathbf{x} = \mathbf{0}. \end{cases}$$

Let us compute its Hessian at the point $\mathbf{0} = [0, 0]^T$. We have

$$\mathbf{F} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_2 \partial x_1} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix}.$$

We now proceed with computing the components of the Hessian and evaluating them at the point $[0, 0]^T$ one by one. We start with

$$\frac{\partial^2 f}{\partial x_1^2} = \frac{\partial}{\partial x_1} \left(\frac{\partial f}{\partial x_1} \right),$$

where

$$\frac{\partial f}{\partial x_1}(\mathbf{x}) = \begin{cases} x_2(x_1^4 - x_2^4 + 4x_1^2x_2^2)/(x_1^2 + x_2^2)^2, & \text{if } \mathbf{x} \neq \mathbf{0} \\ 0, & \text{if } \mathbf{x} = \mathbf{0}. \end{cases}$$

Note that

$$\frac{\partial f}{\partial x_1}([x_1, 0]^T) = 0.$$

Hence,

$$\frac{\partial^2 f}{\partial x_1^2}(\mathbf{0}) = 0.$$

Also,

$$\frac{\partial f}{\partial x_1}([0, x_2]^T) = -x_2.$$

Hence, the mixed partial is

$$\frac{\partial^2 f}{\partial x_2 \partial x_1}(\mathbf{0}) = -1.$$

We next compute

$$\frac{\partial^2 f}{\partial x_2^2} = \frac{\partial}{\partial x_2} \left(\frac{\partial f}{\partial x_2} \right),$$

where

$$\frac{\partial f}{\partial x_2}(x_1, x_2) = \begin{cases} x_1(x_1^4 - x_2^4 - 4x_1^2x_2^2)/(x_1^2 + x_2^2)^2, & \text{if } \mathbf{x} \neq \mathbf{0} \\ 0, & \text{if } \mathbf{x} = \mathbf{0}. \end{cases}$$

Note that

$$\frac{\partial f}{\partial x_2}([0, x_2]^T) = 0.$$

Hence,

$$\frac{\partial^2 f}{\partial x_2^2}(\mathbf{0}) = 0.$$

Also,

$$\frac{\partial f}{\partial x_2}([x_1, 0]^T) = x_1.$$

Hence, the mixed partial is

$$\frac{\partial^2 f}{\partial x_1 \partial x_2}(\mathbf{0}) = 1.$$

Therefore, the Hessian evaluated at the point $\mathbf{0}$ is

$$\mathbf{F}(\mathbf{0}) = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},$$

which is not symmetric.

2.5 Differentiation Rules

We now introduce the chain rule for differentiating the composition $g(\mathbf{f}(t))$, of a function $\mathbf{f} : \mathbb{R} \rightarrow \mathbb{R}^n$ and a function $g : \mathbb{R}^n \rightarrow \mathbb{R}$.

Theorem 21. *Let $g : \mathcal{D} \rightarrow \mathbb{R}$ be differentiable on an open set $\mathcal{D} \subset \mathbb{R}^n$, and let $\mathbf{f} : (a, b) \rightarrow \mathcal{D}$ be differentiable on (a, b) . Then, the composite function $h : (a, b) \rightarrow \mathbb{R}$ given by $h(t) = g(\mathbf{f}(t))$ is differentiable on (a, b) , and*

$$h'(t) = Dg(\mathbf{f}(t))D\mathbf{f}(t) = \nabla g(\mathbf{f}(t))^T \begin{bmatrix} f'_1(t) \\ \vdots \\ f'_n(t) \end{bmatrix}.$$

Proof. By definition,

$$h'(t) = \lim_{s \rightarrow t} \frac{h(s) - h(t)}{s - t} = \lim_{s \rightarrow t} \frac{g(\mathbf{f}(s)) - g(\mathbf{f}(t))}{s - t}$$

if the limit exists. By Theorem 19 we write

$$g(\mathbf{f}(s)) - g(\mathbf{f}(t)) = Dg(\mathbf{f}(t))(\mathbf{f}(s) - \mathbf{f}(t)) + r(s),$$

where $\lim_{s \rightarrow t} \frac{r(s)}{s - t} = 0$. Therefore,

$$\frac{h(s) - h(t)}{s - t} = Dg(\mathbf{f}(t)) \frac{\mathbf{f}(s) - \mathbf{f}(t)}{s - t} + \frac{r(s)}{s - t}.$$

Letting $s \rightarrow t$ yields

$$h'(t) = \lim_{s \rightarrow t} Dg(\mathbf{f}(t)) \frac{\mathbf{f}(s) - \mathbf{f}(t)}{s - t} + \frac{r(s)}{s - t} = Dg(\mathbf{f}(t))D\mathbf{f}(t).$$

□

Next, we present the product rule. Let $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ be two differentiable functions. Define the function $h : \mathbb{R}^n \rightarrow \mathbb{R}$ by $h(\mathbf{x}) = \mathbf{f}(\mathbf{x})^T \mathbf{g}(\mathbf{x})$. Then, h is also differentiable and

$$Dh(\mathbf{x}) = \mathbf{f}(\mathbf{x})^T D\mathbf{g}(\mathbf{x}) + \mathbf{g}(\mathbf{x})^T D\mathbf{f}(\mathbf{x}).$$

We end this section with a list of some useful formulas from multivariable calculus. In each case, we compute the derivative with respect to \mathbf{x} . Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be a given matrix and $\mathbf{y} \in \mathbb{R}^m$ a given vector. Then,

$$\begin{aligned} D(\mathbf{y}^T \mathbf{A} \mathbf{x}) &= \mathbf{y}^T \mathbf{A} \\ D(\mathbf{x}^T \mathbf{A} \mathbf{x}) &= \mathbf{x}^T (\mathbf{A} + \mathbf{A}^T), \quad \text{if } m = n. \end{aligned}$$

It follows from the first formula above that if $\mathbf{y} \in \mathbb{R}^n$, then

$$D(\mathbf{y}^T \mathbf{x}) = \mathbf{y}^T.$$

It follows from the second formula above that if \mathbf{Q} is a symmetric matrix, then

$$D(\mathbf{x}^T \mathbf{Q} \mathbf{x}) = 2\mathbf{x}^T \mathbf{Q}.$$

In particular,

$$D(\mathbf{x}^T \mathbf{x}) = 2\mathbf{x}^T.$$

2.6 Level Sets and Gradients

The level set of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ at level c is the set of points

$$S = \{\mathbf{x} : f(\mathbf{x}) = c\}.$$

For $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, we are usually interested in S when it is a curve. For $f : \mathbb{R}^3 \rightarrow \mathbb{R}$, the sets S most often considered are surfaces.

Example 22. Consider the following real-valued function on \mathbb{R}^2 :

$$f(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2, \mathbf{x} = [x_1, x_2]^T. \quad (2.2)$$

The function above is called Rosenbrock's function. A plot of the function f is shown in Figure 2.2. The level sets of f at levels 0.7, 7, 70, 200, and 700 are depicted in Figure 2.3. These level sets have a particular shape resembling bananas. For this reason, Rosenbrock's function is also called the banana function.

To say that a point \mathbf{x}_0 is on the level set S at level c means that $f(\mathbf{x}_0) = c$. Now suppose that there is a curve γ lying in S and parameterized by a continuously differentiable function $\mathbf{g} : \mathbb{R} \rightarrow \mathbb{R}^n$. Suppose also that $\mathbf{g}(t_0) = \mathbf{x}_0$ and $D\mathbf{g}(t_0) = \mathbf{v} \neq \mathbf{0}$, so that \mathbf{v} is a tangent vector to γ at \mathbf{x}_0 (see Figure 2.4). Applying the chain rule to the function $h(t) = f(\mathbf{g}(t))$ at t_0 gives

$$h'(t_0) = Df(\mathbf{g}(t_0))D\mathbf{g}(t_0) = Df(\mathbf{x}_0)\mathbf{v}.$$

But since γ lies on S , we have

$$h(t) = f(\mathbf{g}(t)) = c;$$

that is, h is constant. Thus, $h'(t_0) = 0$ and

$$Df(\mathbf{x}_0)\mathbf{v} = \nabla f(\mathbf{x}_0)^T \mathbf{v} = 0.$$

Hence, we have proved, assuming f continuously differentiable, the following theorem (see Figure 2.4).

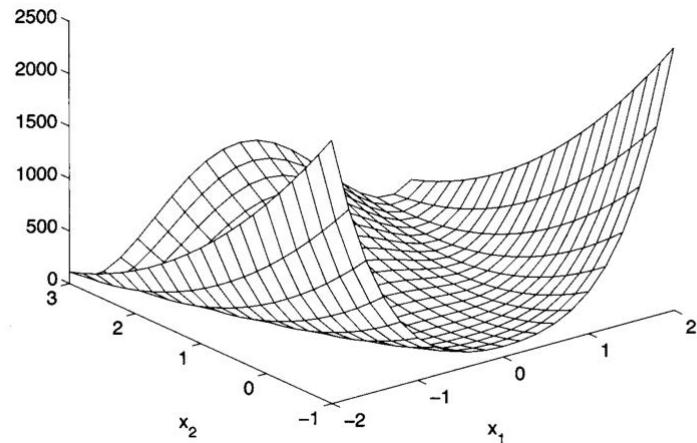


图 2.2: Graph of Rosenbrock's function.

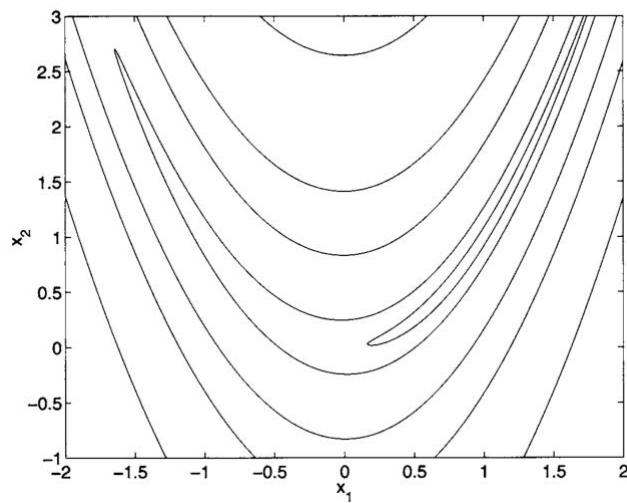


图 2.3: Level sets of Rosenbrock's (banana) function.

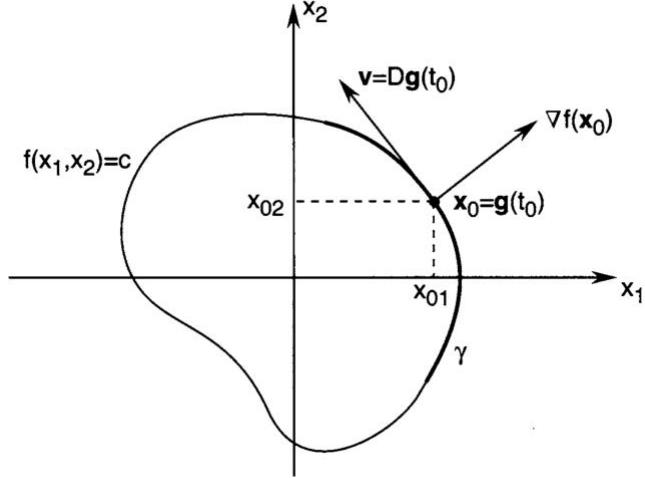


图 2.4: Orthogonality of the gradient to the level set.

Theorem 23. *The vector $\nabla f(\mathbf{x}_0)$ is orthogonal to the tangent vector to an arbitrary smooth curve passing through \mathbf{x}_0 on the level set determined by $f(\mathbf{x}) = f(\mathbf{x}_0)$.*

It is natural to say that $\nabla f(\mathbf{x}_0)$ is orthogonal or normal to the level set S corresponding to \mathbf{x}_0 , and it is also natural to take as the tangent plane (or line) to S at \mathbf{x}_0 the set of all points \mathbf{x} satisfying

$$\nabla f(\mathbf{x}_0)^T (\mathbf{x} - \mathbf{x}_0) = 0 \quad \text{if } \nabla f(\mathbf{x}_0) \neq \mathbf{0}.$$

As we shall see later, $\nabla f(\mathbf{x}_0)$ is the direction of maximum rate of increase of f at \mathbf{x}_0 . Because $\nabla f(\mathbf{x}_0)$ is orthogonal to the level set through \mathbf{x}_0 determined by $f(\mathbf{x}) = f(\mathbf{x}_0)$, we deduce the following fact: The direction of maximum rate of increase of a real-valued differentiable function at a point is orthogonal to the level set of the function through that point.

Figure 2.5 illustrates the discussion above for the case $f : \mathbb{R}^2 \rightarrow \mathbb{R}$. The curve on the shaded surface in Figure 2.5 running from bottom to top has the property that its projection onto the (x_1, x_2) -plane is always orthogonal to the level curves and is called a path of steepest ascent because it always heads in the direction of maximum rate of increase for f .

The graph of $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is the set $\{[\mathbf{x}^T, f(\mathbf{x})]^T : \mathbf{x} \in \mathbb{R}^n\} \subset \mathbb{R}^{n+1}$. The notion of the gradient of a function has an alternative useful interpretation in terms of the tangent hyperplane to its graph. To proceed, let $\mathbf{x}_0 \in \mathbb{R}^n$ and $z_0 = f(\mathbf{x}_0)$. The point $[\mathbf{x}_0^T, z_0]^T \in \mathbb{R}^{n+1}$ is a point on the graph of f . If f is differentiable at ξ , then the graph

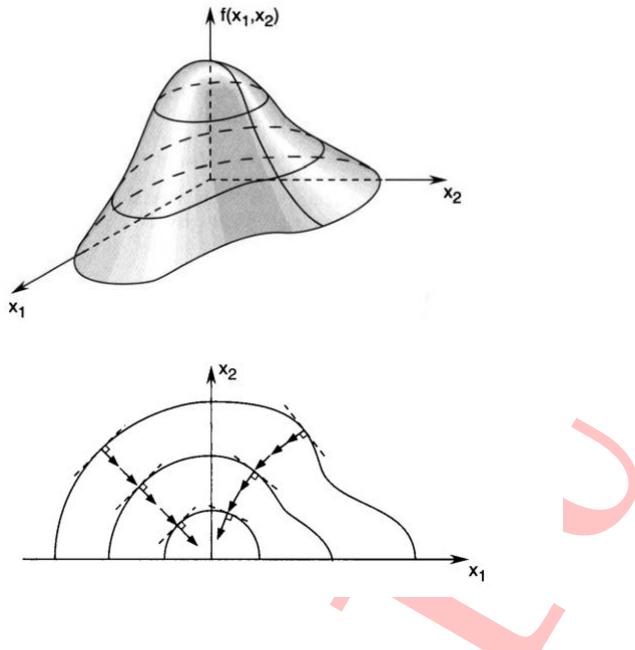


图 2.5: Illustration of a path of steepest ascent.

admits a nonvertical tangent hyperplane at $\xi = [\mathbf{x}_0^T, z_0]^T$. The hyperplane through ξ is the set of all points $[x_1, \dots, x_n, z]^T \in \mathbb{R}^{n+1}$ satisfying the equation

$$u_1(x_1 - x_{01}) + \dots + u_n(x_n - x_{0n}) + v(z - z_0) = 0,$$

where the vector $[u_1, \dots, u_n, v]^T \in \mathbb{R}^{n+1}$ is normal to the hyperplane. Assuming that this hyperplane is nonvertical (that is, $v \neq 0$), let

$$d_i = -\frac{u_i}{v}.$$

Thus, we can rewrite the hyperplane equation above as

$$z = d_1(x_1 - x_{01}) + \dots + d_n(x_n - x_{0n}) + z_0.$$

We can think of the right side of the above equation as a function $z : \mathbb{R}^n \rightarrow \mathbb{R}$. Observe that for the hyperplane to be tangent to the graph of f , the functions f and z must have the same partial derivatives at the point \mathbf{x}_0 . Hence, if f is differentiable at \mathbf{x}_0 , its tangent hyperplane can be written in terms of its gradient, as given by the equation

$$z - z_0 = Df(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) = (\mathbf{x} - \mathbf{x}_0)^T \nabla f(\mathbf{x}_0).$$

2.7 Taylor Series

The basis for many numerical methods and models for optimization is Taylor's formula, which is given by Taylor's theorem.

Theorem 24 (Taylor's Theorem). Assume that a function $f : \mathbb{R} \rightarrow \mathbb{R}$ is m times continuously differentiable (i.e., $f \in \mathcal{C}^m$) on an interval $[a, b]$. Denote $h = b - a$. Then,

$$f(b) = f(a) + \frac{h}{1!} f^{(1)}(a) + \frac{h^2}{2!} f^{(2)}(a) + \cdots + \frac{h^{m-1}}{(m-1)!} f^{(m-1)}(a) + R_m,$$

(called Taylor's formula) where $f^{(i)}$ is the i th derivative of f , and

$$R_m = \frac{h^m (1-\theta)^{m-1}}{(m-1)!} f^{(m)}(a + \theta h) = \frac{h^m}{m!} f^{(m)}(a + \theta' h),$$

with $\theta, \theta' \in (0, 1)$.

Proof. We have

$$R_m = f(b) - f(a) - \frac{h}{1!} f^{(1)}(a) - \frac{h^2}{2!} f^{(2)}(a) - \cdots - \frac{h^{m-1}}{(m-1)!} f^{(m-1)}(a).$$

Denote by $g_m(x)$ an auxiliary function obtained from R_m by replacing a by x . Hence,

$$\begin{aligned} g_m(x) &= f(b) - f(x) - \frac{b-x}{1!} f^{(1)}(x) - \frac{(b-x)^2}{2!} f^{(2)}(x) \\ &\quad - \cdots - \frac{(b-x)^{m-1}}{(m-1)!} f^{(m-1)}(x). \end{aligned}$$

Differentiating $g_m(x)$ yields

$$\begin{aligned} g_m^{(1)}(x) &= -f^{(1)}(x) + \left[f^{(1)}(x) - \frac{b-x}{1!} f^{(2)}(x) \right] \\ &\quad + \left[2 \frac{b-x}{2!} f^{(2)}(x) - \frac{(b-x)^2}{2!} f^{(3)}(x) \right] + \cdots \\ &\quad + \left[(m-1) \frac{(b-x)^{m-2}}{(m-1)!} f^{(m-1)}(x) - \frac{(b-x)^{m-1}}{(m-1)!} f^{(m)}(x) \right] \\ &= -\frac{(b-x)^{m-1}}{(m-1)!} f^{(m)}(x). \end{aligned}$$

Observe that $g_m(b) = 0$ and $g_m(a) = R_m$. Applying the mean-value theorem yields

$$\frac{g_m(b) - g_m(a)}{b-a} = g_m^{(1)}(a + \theta h),$$

where $\theta \in (0, 1)$. The equation above is equivalent to

$$-\frac{R_m}{h} = -\frac{(b-a-\theta h)^{m-1}}{(m-1)!} f^{(m)}(a + \theta h) = -\frac{h^{m-1} (1-\theta)^{m-1}}{(m-1)!} f^{(m)}(a + \theta h).$$

Hence,

$$R_m = \frac{h^m (1-\theta)^{m-1}}{(m-1)!} f^{(m)}(a + \theta h).$$

To derive the formula

$$R_m = \frac{h^m}{m!} f^{(m)}(a + \theta' h),$$

see, e.g., [72]. □

An important property of Taylor's theorem arises from the form of the remainder R_m . To discuss this property further, we introduce the order symbols, O and o .

Let g be a real-valued function defined in some neighborhood of $\mathbf{0} \in \mathbb{R}^n$, with $g(\mathbf{x}) \neq 0$ if $\mathbf{x} \neq \mathbf{0}$. Let $\mathbf{f} : \Omega \rightarrow \mathbb{R}^m$ be defined in a domain $\Omega \subset \mathbb{R}^n$ that includes $\mathbf{0}$. Then, we write

1. $\mathbf{f}(\mathbf{x}) = O(g(\mathbf{x}))$ to mean that the quotient $\|\mathbf{f}(\mathbf{x})\|/|g(\mathbf{x})|$ is bounded near $\mathbf{0}$; that is, there exist numbers $K > 0$ and $\delta > 0$ such that if $\|\mathbf{x}\| < \delta$, $\mathbf{x} \in \Omega$, then $\|\mathbf{f}(\mathbf{x})\|/|g(\mathbf{x})| \leq K$.
2. $\mathbf{f}(\mathbf{x}) = o(g(\mathbf{x}))$ to mean that

$$\lim_{\mathbf{x} \rightarrow \mathbf{0}, \mathbf{x} \in \Omega} \frac{\|\mathbf{f}(\mathbf{x})\|}{|g(\mathbf{x})|} = 0.$$

The symbol $O(g(\mathbf{x}))$ [read “big-oh of $g(\mathbf{x})$ ”] is used to represent a function that is bounded by a scaled version of g in a neighborhood of $\mathbf{0}$. Examples of such a function are:

- $x = O(x)$.
- $\begin{bmatrix} x^3 \\ 2x^2 + 3x^4 \end{bmatrix} = O(x^2)$.
- $\cos x = O(1)$.
- $\sin x = O(x)$.

On the other hand, $o(g(\mathbf{x}))$ [read “little-oh of $g(\mathbf{x})$ ”] represents a function that goes to zero “faster” than $g(\mathbf{x})$ in the sense that $\lim_{\mathbf{x} \rightarrow \mathbf{0}} \|o(g(\mathbf{x}))\|/|g(\mathbf{x})| = 0$. Examples of such functions are:

- $x^2 = o(x)$.
- $\begin{bmatrix} x^3 \\ 2x^2 + 3x^4 \end{bmatrix} = o(x)$.
- $x^3 = o(x^2)$.
- $x = o(1)$.

Note that if $\mathbf{f}(\mathbf{x}) = o(g(\mathbf{x}))$, then $\mathbf{f}(\mathbf{x}) = O(g(\mathbf{x}))$ (but the converse is not necessarily true). Also, if $\mathbf{f}(\mathbf{x}) = O(\|\mathbf{x}\|^p)$, then $\mathbf{f}(\mathbf{x}) = o(\|\mathbf{x}\|^{p-\varepsilon})$ for any $\varepsilon > 0$.

Suppose that $f \in \mathcal{C}^m$. Recall that the remainder term in Taylor's theorem has the form

$$R_m = \frac{h^m}{m!} f^{(m)}(a + \theta h),$$

where $\theta \in (0, 1)$. Substituting this into Taylor's formula, we get

$$f(b) = f(a) + \frac{h}{1!} f^{(1)}(a) + \frac{h^2}{2!} f^{(2)}(a) + \cdots + \frac{h^{m-1}}{(m-1)!} f^{(m-1)}(a) + \frac{h^m}{m!} f^{(m)}(a + \theta h).$$

By the continuity of $f^{(m)}$, we have $f^{(m)}(a + \theta h) \rightarrow f^{(m)}(a)$ as $h \rightarrow 0$; that is, $f^{(m)}(a + \theta h) = f^{(m)}(a) + o(1)$. Therefore,

$$\frac{h^m}{m!} f^{(m)}(a + \theta h) = \frac{h^m}{m!} f^{(m)}(a) + o(h^m),$$

since $h^m o(1) = o(h^m)$. We may then write Taylor's formula as

$$f(b) = f(a) + \frac{h}{1!} f^{(1)}(a) + \frac{h^2}{2!} f^{(2)}(a) + \cdots + \frac{h^m}{m!} f^{(m)}(a) + o(h^m).$$

If, in addition, we assume that $f \in \mathcal{C}^{m+1}$, we may replace the term $o(h^m)$ above by $O(h^{m+1})$. To see this, we first write Taylor's formula with R_{m+1} :

$$f(b) = f(a) + \frac{h}{1!} f^{(1)}(a) + \frac{h^2}{2!} f^{(2)}(a) + \cdots + \frac{h^m}{m!} f^{(m)}(a) + R_{m+1},$$

where

$$R_{m+1} = \frac{h^{m+1}}{(m+1)!} f^{(m+1)}(a + \theta' h),$$

with $\theta' \in (0, 1)$. Because $f^{(m+1)}$ is bounded on $[a, b]$,

$$R_{m+1} = O(h^{m+1}).$$

Therefore, if $f \in \mathcal{C}^{m+1}$, we may write Taylor's formula as

$$f(b) = f(a) + \frac{h}{1!} f^{(1)}(a) + \frac{h^2}{2!} f^{(2)}(a) + \cdots + \frac{h^m}{m!} f^{(m)}(a) + O(h^{m+1}).$$

We now turn to the Taylor series expansion of a real-valued function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ about the point $\mathbf{x}_0 \in \mathbb{R}^n$. Suppose that $f \in \mathcal{C}^2$. Let \mathbf{x} and \mathbf{x}_0 be points in \mathbb{R}^n , and let $\mathbf{z}(\alpha) = \mathbf{x}_0 + \alpha(\mathbf{x} - \mathbf{x}_0)/\|\mathbf{x} - \mathbf{x}_0\|$. Define $\phi : \mathbb{R} \rightarrow \mathbb{R}$ by

$$\phi(\alpha) = f(\mathbf{z}(\alpha)) = f(\mathbf{x}_0 + \alpha(\mathbf{x} - \mathbf{x}_0)/\|\mathbf{x} - \mathbf{x}_0\|).$$

Using the chain rule, we obtain

$$\begin{aligned}\phi'(\alpha) &= \frac{d\phi}{d\alpha}(\alpha) \\ &= Df(\mathbf{z}(\alpha))D\mathbf{z}(\alpha) = Df(\mathbf{z}(\alpha))\frac{(\mathbf{x} - \mathbf{x}_0)}{\|\mathbf{x} - \mathbf{x}_0\|} \\ &= (\mathbf{x} - \mathbf{x}_0)^T Df(\mathbf{z}(\alpha))^T / \|\mathbf{x} - \mathbf{x}_0\|\end{aligned}$$

and

$$\begin{aligned}\phi''(\alpha) &= \frac{d^2\phi}{d\alpha^2}(\alpha) \\ &= \frac{d}{d\alpha} \left(\frac{d\phi}{d\alpha} \right)(\alpha) \\ &= \frac{(\mathbf{x} - \mathbf{x}_0)^T}{\|\mathbf{x} - \mathbf{x}_0\|} \frac{d}{d\alpha} Df(\mathbf{z}(\alpha))^T \\ &= \frac{(\mathbf{x} - \mathbf{x}_0)^T}{\|\mathbf{x} - \mathbf{x}_0\|} D(Df)(\mathbf{z}(\alpha))^T \frac{d\mathbf{z}}{d\alpha}(\alpha) \\ &= \frac{1}{\|\mathbf{x} - \mathbf{x}_0\|^2} (\mathbf{x} - \mathbf{x}_0)^T D^2 f(\mathbf{z}(\alpha))^T (\mathbf{x} - \mathbf{x}_0) \\ &= \frac{1}{\|\mathbf{x} - \mathbf{x}_0\|^2} (\mathbf{x} - \mathbf{x}_0)^T D^2 f(\mathbf{z}(\alpha))(\mathbf{x} - \mathbf{x}_0),\end{aligned}$$

where we recall that

$$D^2 f = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_2 \partial x_1} & \dots & \frac{\partial^2 f}{\partial x_n \partial x_1} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} & \dots & \frac{\partial^2 f}{\partial x_n \partial x_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_1 \partial x_n} & \frac{\partial^2 f}{\partial x_2 \partial x_n} & \dots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}.$$

Observe that

$$\begin{aligned}f(\mathbf{x}) &= \phi(\|\mathbf{x} - \mathbf{x}_0\|) \\ &= \phi(0) + \frac{\|\mathbf{x} - \mathbf{x}_0\|}{1!} \phi'(0) + \frac{\|\mathbf{x} - \mathbf{x}_0\|^2}{2!} \phi''(0) + o(\|\mathbf{x} - \mathbf{x}_0\|^2).\end{aligned}$$

Hence,

$$\begin{aligned}f(\mathbf{x}) &= f(\mathbf{x}_0) + \frac{1}{1!} Df(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) \\ &\quad + \frac{1}{2!} (\mathbf{x} - \mathbf{x}_0)^T D^2 f(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) + o(\|\mathbf{x} - \mathbf{x}_0\|^2).\end{aligned}$$

If we assume that $f \in \mathcal{C}^3$, we may use the formula for the remainder term R_3 to conclude that

$$\begin{aligned} f(\mathbf{x}) &= f(\mathbf{x}_0) + \frac{1}{1!} Df(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) \\ &\quad + \frac{1}{2!} (\mathbf{x} - \mathbf{x}_0)^T D^2 f(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) + O(\|\mathbf{x} - \mathbf{x}_0\|^3). \end{aligned}$$

We end with a statement of the mean value theorem, which is closely related to Taylor's theorem.

Theorem 25. *If a function $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is differentiable on an open set $\Omega \subset \mathbb{R}^n$, then for any pair of points $\mathbf{x}, \mathbf{y} \in \Omega$, there exists a matrix \mathbf{M} such that*

$$\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y}) = \mathbf{M}(\mathbf{x} - \mathbf{y}).$$

The mean value theorem follows from Taylor's theorem (for the case where $m = 1$) applied to each component of \mathbf{f} . It is easy to see that \mathbf{M} is a matrix whose rows are the rows of $D\mathbf{f}$ evaluated at points that lie on the line segment joining \mathbf{x} and \mathbf{y} (these points may differ from row to row).

For further reading in calculus, consult [72].

2.7.1 Exercises

Exercise 26. Show that a sufficient condition for $\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{O}$ is $\|\mathbf{A}\| < 1$.

Exercise 27. Show that for any matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$,

$$\|\mathbf{A}\| \geq \max_{1 \leq i \leq n} |\lambda_i(\mathbf{A})|.$$

Exercise 28. Consider the function

$$f(\mathbf{x}) = (\mathbf{a}^T \mathbf{x})(\mathbf{b}^T \mathbf{x}),$$

where \mathbf{a} , \mathbf{b} , and \mathbf{x} are n -dimensional vectors.

- a. Find $\nabla f(\mathbf{x})$.
- b. Find the Hessian $\mathbf{F}(\mathbf{x})$.

Exercise 29. Define the functions $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ and $g : \mathbb{R} \rightarrow \mathbb{R}^2$ by $f(\mathbf{x}) = \frac{x_1^2}{6} + \frac{x_2^2}{4}$, $g(t) = [3t + 5, 2t - 6]^T$. Let $F : \mathbb{R} \rightarrow \mathbb{R}$ be given by $F(t) = f(g(t))$. Evaluate $\frac{dF}{dt}(t)$ using the chain rule.

Exercise 30. Consider $f(\mathbf{x}) = \frac{x_1 x_2}{2}$, $\mathbf{g}(s, t) = [4s + 3t, 2s + t]^T$. Evaluate $\frac{\partial}{\partial s} f(\mathbf{g}(s, t))$ and $\frac{\partial}{\partial t} f(\mathbf{g}(s, t))$ using the chain rule.

Exercise 31. Let $\mathbf{x}(t) = [e^t + t^3, t^2, t + 1]^T$, $t \in \mathbb{R}$, and $f(\mathbf{x}) = x_1^3 x_2 x_3^2 + x_1 x_2 + x_3$, $\mathbf{x} = [x_1, x_2, x_3]^T \in \mathbb{R}^3$. Find $\frac{d}{dt} f(\mathbf{x}(t))$ in terms of t .

Exercise 32. Suppose that $f(\mathbf{x}) = o(g(\mathbf{x}))$. Show that for any given $\varepsilon > 0$, there exists $\delta > 0$ such that if $\|\mathbf{x}\| < \delta$, then $\|f(\mathbf{x})\| < \varepsilon |g(\mathbf{x})|$.

Exercise 33. Use Exercise 32 to show that if functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}$ satisfy $f(\mathbf{x}) = -g(\mathbf{x}) + o(g(\mathbf{x}))$ and $g(\mathbf{x}) > 0$ for all $\mathbf{x} \neq \mathbf{0}$, then for all $\mathbf{x} \neq \mathbf{0}$ sufficiently small, we have $f(\mathbf{x}) < 0$.

Exercise 34. Let

$$\begin{aligned} f_1(x_1, x_2) &= x_1^2 - x_2^2, \\ f_2(x_1, x_2) &= 2x_1 x_2. \end{aligned}$$

Sketch the level sets associated with $f_1(x_1, x_2) = 12$ and $f_2(x_1, x_2) = 16$ on the same diagram. Indicate on the diagram the values of $\mathbf{x} = [x_1, x_2]^T$ for which $\mathbf{f}(\mathbf{x}) = [f_1(x_1, x_2), f_2(x_1, x_2)]^T = [12, 16]^T$.

Exercise 35. Write down the Taylor series expansion of the following functions about the given points \mathbf{x}_0 . Neglect terms of order three or higher.

- a. $f(\mathbf{x}) = x_1 e^{-x_2} + x_2 + 1$, $\mathbf{x}_0 = [1, 0]^T$.
- b. $f(\mathbf{x}) = x_1^4 + 2x_1^2 x_2^2 + x_2^4$, $\mathbf{x}_0 = [1, 1]^T$.
- c. $f(\mathbf{x}) = e^{x_1-x_2} + e^{x_1+x_2} + x_1 + x_2 + 1$, $\mathbf{x}_0 = [1, 0]^T$.

2.8 Rates of Convergence

(Taken from Chapter 2.5 of [57])

Many of the algorithms discussed in this book do not find a solution in a finite number of steps. Instead these algorithms compute a sequence of approximate solutions that we hope get closer and closer to a solution. When discussing such an algorithm, the following two questions are often asked:

1. Does it converge?
2. How fast does it converge?

It is the second question that is the topic of this section.

If an algorithm converges in a finite number of steps, the cost of that algorithm is often measured by counting the number of steps required, or by counting the number of arithmetic operations required. For example, if Gaussian elimination is applied to a system of n linear equations, then it will require about n^3 operations. This cost is referred to as the *computational complexity* of the algorithm.

For many optimization methods, the number of operations or steps required to find an exact solution will be infinite, so some other measure of efficiency must be used. The rate of convergence is one such measure. It describes how quickly the estimates of the solution approach the exact solution.

Let us assume that we have a sequence of points x_k converging to a solution x_* . We define the sequence of errors to be

$$e_k = x_k - x_*.$$

Note that

$$\lim_{k \rightarrow \infty} e_k = 0.$$

We say that the sequence $\{x_k\}$ converges to x_* with rate r and rate constant C if

$$\lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|^r} = C$$

and $C < \infty$. To understand this idea better, let us look at some examples.

Initially let us assume that we have ideal convergence behavior

$$\|e_{k+1}\| = C\|e_k\|^r \quad \text{for all } k,$$

so that we can avoid having to deal with limits. When $r = 1$ this is referred to as *linear* convergence:

$$\|e_{k+1}\| = C\|e_k\|.$$

If $0 < C < 1$, then the norm of the error is reduced by a constant factor at every iteration. If $C > 1$, then the sequence diverges. (What can happen when $C = 1$?) If we choose $C = 0.1 = 10^{-1}$ and $\|e\| = 1$, then the norms of the errors are

$$1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7},$$

and seven-digit accuracy is obtained in seven iterations, a good result. On the other hand, if $C = 0.99$, then the norms of the errors take on the values

$$1, 0.99, 0.9801, 0.9703, 0.9606, 0.9510, 0.9415, 0.9321, \dots,$$

and it would take about 1600 iterations to reduce the error to 10^{-7} , a less impressive result.

If $r = 1$ and $C = 0$, the convergence is called *superlinear*. Superlinear convergence includes all cases where $r > 1$ since if

$$\lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|^r} = C < \infty,$$

then

$$\lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|} = \lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|^r} \|e_k\|^{r-1} = C \times \lim_{k \rightarrow \infty} \|e_k\|^{r-1} = 0.$$

When $r = 2$, the convergence is called *quadratic*. As an example, let $r = 2, C = 1$, and $\|e_0\| = 10^{-1}$. Then the sequence of error norms is

$$10^{-1}, 10^{-2}, 10^{-4}, 10^{-8},$$

and so three iterations are sufficient to achieve seven-digit accuracy. In this form of quadratic convergence the error is squared at each iteration. Another way of saying this is that the number of correct digits in x_k doubles at every iteration. Of course, if the constant $C \neq 1$, then this is not an accurate statement, but it gives an intuitive sense of the attractions of a quadratic convergence rate.

For optimization algorithms there is one other important case, and that is when $1 < r < 2$. This is another special case of superlinear convergence. This case is important because (a) it is qualitatively similar to quadratic convergence for the precision of common computer calculations, and (b) it can be achieved by algorithms that only compute first derivatives, whereas to achieve quadratic convergence it is often necessary to compute second derivatives as well. To get a sense of what this form of superlinear convergence looks like, let $r = 1.5, C = 1$, and $\|e_0\| = 10^{-1}$. Then the sequence of error norms is

$$1 \times 10^{-1}, 3 \times 10^{-2}, 6 \times 10^{-3}, 4 \times 10^{-4}, 9 \times 10^{-6}, 3 \times 10^{-8},$$

and five iterations are required to achieve single-precision accuracy.

Example 36 (Rate of Convergence of a Sequence). Consider the sequence

$$2, 1.1, 1.01, 1.001, 1.0001, 1.00001, \dots$$

with general term $x_k = 1 + 10^{-k}$. This sequence converges to $x_* = 1$ and $e_k = x_k - x_* = 10^{-k}$. Hence

$$\lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|} = \lim_{k \rightarrow \infty} \frac{10^{-(k+1)}}{10^{-k}} = \frac{1}{10},$$

so that the sequence converges linearly with rate constant $\frac{1}{10}$.

Now consider the sequence

$$4, 2.5, 2.05, 2.00060975, \dots$$

defined by the formula

$$x_{k+1} = \frac{1}{2} \left(x_k + \frac{4}{x_k} \right) = \frac{x_k}{2} + \frac{2}{x_k}$$

with $x_0 = 4$. It can be shown that $x_k \rightarrow 2$. Also

$$\begin{aligned} e_{k+1} &= x_{k+1} - x_* \\ &= \frac{x_k}{2} + \frac{2}{x_k} - 2 \\ &= \frac{1}{2x_k}(x_k^2 + 4 - 4x_k) \\ &= \frac{1}{2x_k}(x_k - 2)^2 = \frac{1}{2x_k}e_k^2. \end{aligned}$$

From this it follows that

$$\lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|^2} = \frac{1}{2|x_*|} = \frac{1}{4}.$$

Hence this sequence converges quadratically with rate constant $\frac{1}{4}$.

(The above definition of convergence rates are in the sense of Q -linear or Q -quadratic, etc., where ‘ Q ’ stands for “Quotient”. See <http://www.math.unl.edu/~s-bockel1/833-notes/node7.html> for definition of R -linear, where ‘ R ’ stands for “Root”, and an example for why this definition is necessary.)

In practical situations ideal convergence behavior is not always observed. The rate of convergence is only observed in the limit, so at the initial iterations there is no guarantee that the norm of the error will be reduced at all, let alone at any predictable rate. In fact, it is not uncommon for an algorithm to expend almost all of its effort far from the solution, with this asymptotic convergence rate only becoming apparent at the last few iterations. In addition, the algorithm will be terminated after a finite number of iterations when the error in the solution is below some tolerance, and so the limiting behavior described here may be only imperfectly observed.

There is ambiguity in the definition of the rate of convergence. For instance, any sequence that converges quadratically also converges linearly, but with rate constant equal to zero. It is common when discussing algorithms to refer to the *fastest* rate at which the

algorithm *typically* converges. For example, in Newton's Method we show that a certain sequence $\{x_k\}$ satisfies

$$x_{k+1} - x_* \approx \left(\frac{f''(x_*)}{2f'(x_*)} \right) (x_k - x_*)^2,$$

where $x_* = \lim_{k \rightarrow +\infty} x_k$ and f is a function used to define the sequence. Based on this formula, the sequence $\{x_k\}$ is said to converge quadratically. However, if $f'(x_*) \neq 0$ but $f''(x_*) = 0$, then the sequence can converge faster than quadratically. “Typically” these things do not happen.

In many situations people use a sort of shorthand and only refer to the convergence rate without mention of the rate constant. For quadratic rates of convergence this is not too misleading, since the ideal behavior and the observed behavior are similar unless the rate constant is exceptionally large or small. However, in the linear case the rate constant plays an important role. It is not uncommon to see rate constants that are close to one, and more unusual to see rate constants near zero. As a result, linear convergence rates are often considered to be inferior. However, if the rate constant is small, then there is little practical difference between linear and higher rates of convergence at the level of precision common on many computers. In summary, even though it is generally true that higher rates of convergence often represent improvements in performance, this is not guaranteed, and an algorithm with a linear rate of convergence can sometimes be effective in a practical setting.

2.8.1 Exercises

Exercise 37. For each of the following sequences, prove that the sequence converges, find its limit, and determine the rate of convergence and the rate constant.

1. The sequence

$$\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \dots$$

with general term $x_k = 2^{-k}$, for $k = 1, 2, \dots$.

2. The sequence

$$1.05, 1.0005, 1.000005, \dots$$

with general term $x_k = 1 + 5 \times 10^{-2k}$, for $k = 1, 2, \dots$.

3. The sequence with general term $x_k = 2^{-2^k}$.

4. The sequence with general term $x_k = 3^{-k^2}$.

5. The sequence with general term $x_k = 1 - 2^{-2^k}$ for k odd, and $x_k = 1 + 2^{-2^k}$ for k even.

Exercise 38. Consider the sequence defined by $x_0 = a > 0$ and

$$x_{k+1} = \frac{1}{2} \left(x_k + \frac{a}{x_k} \right).$$

Prove that this sequence converges to $x_* = \sqrt{a}$ and that the convergence rate is quadratic, and determine the rate constant.

Exercise 39. Consider a convergent sequence $\{x_k\}$ and define a second sequence $\{y_k\}$ with $y_k = cx_k$ where c is some nonzero constant. What is the relationship between the convergence rates and rate constants of the two sequences?

Exercise 40. Let $\{x_k\}$ and $\{c_k\}$ be convergent sequences, and assume that

$$\lim_{k \rightarrow \infty} c_k = c \neq 0.$$

Consider the sequence $\{y_k\}$ with $y_k = c_k x_k$. Is this sequence guaranteed to converge? If so, can its convergence rate and rate constant be determined from the rates and rate constants for the sequences $\{x_k\}$ and $\{c_k\}$?

(Taken from Chapter 8 of [30])

Exercise 41. Let $\{\mathbf{x}^{(k)}\}$ be a sequence that converges to \mathbf{x}^* . Show that if there exists $c > 0$ such that

$$\|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| > c \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^p$$

for sufficiently large k , then the order of convergence (if it exists) is at most p .

Exercise 42. Let $\{\mathbf{x}^{(k)}\}$ be a sequence that converges to \mathbf{x}^* . Show that there does not exist $p < 1$ such that

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|}{\|\mathbf{x}^{(k)} - \mathbf{x}^*\|^p} > 0.$$

Exercise 43. Consider the sequence $\{x^{(k)}\}$ given by $x^{(k)} = 2^{-2^k}$.

a. Write down the value of the limit of $\{x^{(k)}\}$.

b. Find the order of convergence of $\{x^{(k)}\}$.

Exercise 44. Consider the two sequences $\{x^{(k)}\}$ and $\{y^{(k)}\}$ defined iteratively as follows:

$$x^{(k+1)} = ax^{(k)}, \quad y^{(k+1)} = (y^{(k)})^b,$$

where $a \in \mathbb{R}$, $b \in \mathbb{R}$, $0 < a < 1$, $b > 1$, $x^{(0)} \neq 0$, $y^{(0)} \neq 0$, and $|y^{(0)}| < 1$.

- a. Derive a formula for $x^{(k)}$ in terms of $x^{(0)}$ and a . Use this to deduce that $x^{(k)} \rightarrow 0$.
- b. Derive a formula for $y^{(k)}$ in terms of $y^{(0)}$ and b . Use this to deduce that $y^{(k)} \rightarrow 0$.
- c. Find the orders of convergence of $\{x^{(k)}\}$ and $\{y^{(k)}\}$.
- d. Calculate the smallest number of iterations k such that $|x^{(k)}| < c|x^{(0)}|$, where $0 < c < 1$.
- e. Calculate the smallest number of iterations k such that $|y^{(k)}| < c|y^{(0)}|$, where $0 < c < 1$.
- f. Compare the answer of part e with that of part d, focusing on the case where c is very small.

(Proposed by Zhouchen Lin.)

Exercise 45. Given $r \geq 1$ and $C \geq 0$, where r may not be an integer, construct a convergent sequence $\{\mathbf{x}_k\}$ such that its convergent rate is r and its rate constant is C .

Exercise 46. Let $\{\theta_k\}$ be the magic sequence used in Nesterov's acceleration tricks, defined as: $\theta_0 = 1$, $\theta_{k+1} = \frac{\sqrt{\theta_k^4 + 4\theta_k^2} - \theta_k^2}{2}$. Then what is the limit of $\{\theta_k\}$ and what is its convergence rate? Further define $\beta_k = \frac{\theta_k(1 - \theta_{k-1})}{\theta_{k-1}}$, what is the limit of $\{\beta_k\}$ and what is its convergence rate?

2.9 Global and Local Convergence

We say that an iterative algorithm is globally convergent if for any arbitrary starting point the algorithm is guaranteed to generate a sequence of points converging to a point that satisfies the first-order necessary condition (FONC) for a minimizer. When the algorithm is not globally convergent, it may still generate a sequence that converges to a point satisfying the FONC, provided that the initial point is sufficiently close to the point. In this case we say that the algorithm is locally convergent. How close to a solution point we need to start for the algorithm to converge depends on the local convergence properties of the algorithm.

2.10 Error analysis in numerical modeling

(Taken from [https://en.wikipedia.org/wiki/Error_analysis_\(mathematics\)](https://en.wikipedia.org/wiki/Error_analysis_(mathematics)))

In numerical simulation or modeling of real systems, error analysis is concerned with the changes in the output of the model as the parameters to the model vary about a mean.

For instance, in a system modeled as a function of two variables $z = f(x, y)$. Error analysis deals with the propagation of the numerical errors in x and y (around mean values \bar{x} and \bar{y}) to error in z (around a mean \bar{z}).

In numerical analysis, error analysis comprises both *forward error analysis* and *backward error analysis*.

2.10.1 Forward error analysis

Forward error analysis involves the analysis of a function $z' = f'(a_0, a_1, \dots, a_n)$ which is an approximation (usually a finite polynomial) to a function $z = f(a_0, a_1, \dots, a_n)$ to determine the bounds on the error in the approximation; i.e., to find ε such that $0 \leq |z - z'| \varepsilon$.

2.10.2 Backward error analysis

Backward error analysis involves the analysis of the approximation function $z' = f'(a_0, a_1, \dots, a_n)$, to determine the bounds on the parameters $a_i = \bar{a}_i \pm \varepsilon_i$ such that the result $z' = z$.

Backward error analysis, the theory of which was developed and popularized by James H. Wilkinson, can be used to establish that an algorithm implementing a numerical function is numerically stable. The basic approach is to show that although the calculated result, due to roundoff errors, will not be exactly correct, it is the exact solution to a nearby problem with slightly perturbed input data. If the perturbation required is small, on the order of the uncertainty in the input data, then the results are in some sense as accurate as the data “deserves”. The algorithm is then defined as backward stable. Stability is a measure of the sensitivity to rounding errors of a given numerical procedure; by contrast, the condition number of a function for a given problem indicates the inherent sensitivity of the function to small perturbations in its input and is independent of the implementation used to solve the problem.

2.11 Mathematical Background

(Taken from Appendix A of [18])

2.11.1 Norms

2.11.1.1 Inner product, Euclidean norms, and angle

The *standard inner product* on \mathbb{R}^n , the set of real n -vectors, is given by

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^\top \mathbf{y} = \sum_{i=1}^n x_i y_i,$$

for $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. In this book we use the notation $\mathbf{x}^\top \mathbf{y}$, instead of $\langle \mathbf{x}, \mathbf{y} \rangle$. The *Euclidean norm*, or l_2 -norm, of a vector $\mathbf{x} \in \mathbb{R}^n$ is defined as

$$\|\mathbf{x}\|_2 = (\mathbf{x}^\top \mathbf{x})^{1/2} = (\mathbf{x}_1^2 + \cdots + \mathbf{x}_n^2)^{1/2} \quad (2.3)$$

The *Cauchy-Schwartz inequality* states that $|\mathbf{x}^\top \mathbf{y}| \leq \|\mathbf{x}\|_2 \|\mathbf{y}\|_2$ for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. The (unsigned) *angle* between nonzero vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ is defined as

$$\angle(\mathbf{x}, \mathbf{y}) = \cos^{-1} \left(\frac{\mathbf{x}^\top \mathbf{y}}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2} \right),$$

where we take $\cos^{-1}(u) \in [0, \pi]$. We say \mathbf{x} and \mathbf{y} are *orthogonal* if $\mathbf{x}^\top \mathbf{y} = 0$.

The standard inner product on $\mathbb{R}^{m \times n}$, the set of $m \times n$ real matrices, is given by

$$\langle \mathbf{X}, \mathbf{Y} \rangle = \text{tr}(\mathbf{X}^\top \mathbf{Y}) = \sum_{i=1}^m \sum_{j=1}^n X_{ij} Y_{ij}$$

for $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{m \times n}$. (Here tr denotes *trace* of a matrix, *i.e.*, the sum of its diagonal elements.) We use the notation $\text{tr}(\mathbf{X}^\top \mathbf{Y})$ instead of $\langle \mathbf{X}, \mathbf{Y} \rangle$. Note that the inner product of two matrices is the inner product of the associated vectors, in \mathbb{R}^{mn} , obtained by listing the coefficients of the matrices in some order, such as row major.

The *Frobenius norm* of a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ is given by

$$\|\mathbf{X}\|_F = (\text{tr}(\mathbf{X}^\top \mathbf{X}))^{1/2} = \left(\sum_{i=1}^m \sum_{j=1}^n \mathbf{X}_{ij}^2 \right)^{1/2}. \quad (2.4)$$

The Frobenius norm is the Euclidean norm of the vector obtained by listing the coefficients of the matrix. (The l_2 -norm of a matrix is a different norm; see Section 2.11.1.5.)

The standard inner product on \mathbf{S}^n , the set of symmetric $n \times n$ matrices, is given by

$$\langle \mathbf{X}, \mathbf{Y} \rangle = \text{tr}(\mathbf{XY}) = \sum_{i=1}^n \sum_{j=1}^n X_{ij} Y_{ij} = \sum_{i=1}^n \mathbf{X}_{ii} \mathbf{Y}_{ii} + 2 \sum_{i < j} \mathbf{X}_{ij} \mathbf{Y}_{ij}.$$

2.11.1.2 Norms, distance, and unit ball

A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ with $\text{dom } f = \mathbb{R}^n$ is called a *norm* if

- f is nonnegative: $f(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in \mathbb{R}^n$;
- f is definite: $f(\mathbf{x}) = 0$ only if $\mathbf{x} = \mathbf{0}$;
- f is homogeneous: $f(t\mathbf{x}) = |t|f(\mathbf{x})$, for all $\mathbf{x} \in \mathbb{R}^n$ and $t \in \mathbb{R}$;
- f satisfies the triangle inequality: $f(\mathbf{x} + \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y})$, for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$.

We use the notation $f(\mathbf{x}) = \|\mathbf{x}\|$, which is meant to suggest that a norm is a generalization of the absolute value on \mathbb{R} . When we specify a particular norm, we use the notation $\|\mathbf{x}\|_{\text{symb}}$, where the subscript is a mnemonic to indicate which norm is meant.

A norm is a measure of the *length* of a vector \mathbf{x} ; we can measure the distance between two vectors \mathbf{x} and \mathbf{y} as the length of their difference, *i.e.*,

$$\text{dist}(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|.$$

We refer to $\text{dist}(\mathbf{x}, \mathbf{y})$ as the distance between \mathbf{x} and \mathbf{y} , in the norm $\|\cdot\|$.

The set of all vectors with norm less than or equal to one,

$$\mathcal{B} = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\| \leq 1\},$$

is called the *unit ball* of the norm $\|\cdot\|$. The unit ball satisfies the following properties:

- \mathcal{B} is symmetric about the origin, *i.e.*, $\mathbf{x} \in \mathcal{B}$ if and only if $-\mathbf{x} \in \mathcal{B}$;
- \mathcal{B} is convex;
- \mathcal{B} is closed, bounded, and has nonempty interior.

Conversely, if $C \subseteq \mathbb{R}^n$ is any set satisfying these three conditions, then it is the unit ball of a norm, which is given by

$$\|\mathbf{x}\| = (\sup \{t \geq 0 \mid t\mathbf{x} \in C\})^{-1}.$$

2.11.1.3 Examples

The simplest example of a norm is the absolute value on \mathbb{R} . Another simple example is the Euclidean or l_2 -norm on \mathbb{R}^n , defined above in (2.3). Two other frequently used norms on \mathbb{R}^n are the *sum-absolute-value*, or l_1 -norm, given by

$$\|\mathbf{x}\|_1 = |\mathbf{x}_1| + \cdots + |\mathbf{x}_n|,$$

and the *Chebyshev* or l_∞ -norm, given by

$$\|\mathbf{x}\|_\infty = \max\{|\mathbf{x}_1|, \dots, |\mathbf{x}_n|\}.$$

These three norms are part of a family parametrized by a constant traditionally denoted p , with $p \geq 1$: the l_p -norm is defined by

$$\|\mathbf{x}\|_p = (|\mathbf{x}_1|^p + \cdots + |\mathbf{x}_n|^p)^{1/p}.$$

This yields the l_1 -norm when $p = 1$ and the Euclidean norm when $p = 2$. It is easy to show that for any $\mathbf{x} \in \mathbb{R}^n$,

$$\lim_{p \rightarrow \infty} \|\mathbf{x}\|_p = \max\{|\mathbf{x}_1|, \dots, |\mathbf{x}_n|\},$$

so the l_∞ -norm also fits in this family, as a limit.

Another important family of norms are the *quadratic norms*. For $\mathbf{P} \in \mathbf{S}_{++}^n$, we define the \mathbf{P} -quadratic norm as

$$\|\mathbf{x}\|_P = (\mathbf{x}^\top \mathbf{P} \mathbf{x})^{1/2} = \|\mathbf{P}^{1/2} \mathbf{x}\|_2.$$

The unit ball of a quadratic norm is an ellipsoid (and conversely, if the unit ball of a norm is an ellipsoid, the norm is a quadratic norm).

Some common norms on $\mathbb{R}^{m \times n}$ are the Frobenius norm, defined above in (2.4), the sum-absolute-value norm,

$$\|\mathbf{X}\|_{sav} = \sum_{i=1}^m \sum_{j=1}^n |\mathbf{X}_{ij}|,$$

and the maximum-absolute-value norm,

$$\|\mathbf{X}\|_{mav} = \max\{|\mathbf{X}_{ij}|\}, \quad i = 1, \dots, m, j = 1, \dots, n.$$

We will encounter several other important norms of matrices in Section 2.11.1.5.

2.11.1.4 Equivalence of norms

Suppose that $\|\cdot\|_a$ and $\|\cdot\|_b$ are norms on \mathbb{R}^n . A basic result of analysis is that there exist positive constants α and β such that, for all $\mathbf{x} \in \mathbb{R}^n$,

$$\alpha \|\mathbf{x}\|_a \leq \|\mathbf{x}\|_b \leq \beta \|\mathbf{x}\|_a.$$

This means that the norms are *equivalent*, i.e., they define the same set of open subsets, the same set of convergent sequences, and so on (see Section 2.11.2). (We conclude that any norms on any finite-dimensional vector space are equivalent, but on infinite-dimensional vector spaces, the result need not hold.) Using convex analysis, we can give a more specific result: If $\|\cdot\|$ is any norm on \mathbb{R}^n , then there exists a quadratic norm $\|\cdot\|_P$ for which

$$\|\mathbf{x}\|_P \leq \|\mathbf{x}\| \leq \sqrt{n} \|\mathbf{x}\|_P \quad (2.5)$$

holds for all \mathbf{x} . In other words, any norm on \mathbb{R}^n can be uniformly approximated, within a factor of \sqrt{n} , by a quadratic norm.

2.11.1.5 Operator norms

Suppose $\|\cdot\|_a$ and $\|\cdot\|_b$ are norms on \mathbb{R}^m and \mathbb{R}^n , respectively. We define the *operator norm* of $\mathbf{X} \in \mathbb{R}^{m \times n}$, induced by the norms $\|\cdot\|_a$ and $\|\cdot\|_b$, as

$$\|\mathbf{X}\|_{a,b} = \sup \{ \|\mathbf{X}\mathbf{u}\|_a \mid \|\mathbf{u}\|_b \leq 1 \}.$$

(It can be shown that this defines a norm on $\mathbb{R}^{m \times n}$.)

When $\|\cdot\|_a$ and $\|\cdot\|_b$ are both Euclidean norms, the operator norm of \mathbf{X} is its *maximum singular value*, and is denoted $\|\mathbf{X}\|_2$:

$$\|\mathbf{X}\|_2 = \sigma_{\max}(\mathbf{X}) = (\lambda_{\max}(\mathbf{X}^\top \mathbf{X}))^{1/2}.$$

(This agrees with the Euclidean norm on \mathbb{R}^m , when $\mathbf{X} \in \mathbb{R}^{m \times 1}$, so there is no clash of notation.) This norm is also called the *spectral norm* or *l_2 -norm* of \mathbf{X} .

As another example, the norm induced by the l_∞ -norm on \mathbb{R}^m and \mathbb{R}^n , denoted $\|\mathbf{X}\|_\infty$,¹ is the *max-row-sum norm*,

$$\|\mathbf{X}\|_\infty = \sup \{ \|\mathbf{X}\mathbf{u}\|_\infty \mid \|\mathbf{u}\|_\infty \leq 1 \} = \max_{i=1,\dots,m} \sum_{j=1}^n |\mathbf{X}_{ij}|. \quad (2.6)$$

¹In the context of sparse recovery, $\|\mathbf{X}\|_\infty$ and $\|\mathbf{X}\|_1$ that follows immediately may be defined as $\|\mathbf{X}\|_\infty = \max_{i,j} |\mathbf{X}_{ij}|$ and $\|\mathbf{X}\|_1 = \sum_{i,j} |\mathbf{X}_{ij}|$, respectively, with abused use of notations.

The norm induced by the l_1 -norm on \mathbb{R}^m and \mathbb{R}^n , denoted $\|\mathbf{X}\|_1$, is the *max-column-sum norm*,

$$\|\mathbf{X}\|_1 = \max_{j=1,\dots,n} \sum_{i=1}^m |\mathbf{X}_{ij}|. \quad (2.7)$$

2.11.1.6 Matrix Norm

Definition 47. A matrix norm $\|\cdot\|$ is a vector norm that further satisfies

$$\|\mathbf{AB}\| \leq \|\mathbf{A}\| \|\mathbf{B}\|.$$

Definition 48. Suppose $\|\cdot\|$ is a vector norm. The matrix norm $\|\cdot\|$ induced by the vector norm is defined as

$$\|\mathbf{A}\| = \sup_{\|\mathbf{x}\|=1} \|\mathbf{Ax}\|.$$

Common vector norm induced matrix norms include:

1. $\|\mathbf{A}\|_1 = \max_j \sum_i |A_{ij}|$, induced by vector 1-norm;
2. $\|\mathbf{A}\|_\infty = \max_i \sum_j |A_{ij}|$, induced by vector ∞ -norm;
3. $\|\mathbf{A}\|_2 = \sigma_1(\mathbf{A})$, where $\sigma_1(\mathbf{A})$ denotes the largest singular value of \mathbf{A} . It is induced by vector 2-norm.

Remark 49. Not all matrix norms are induced norms. According to Theorem 5.6.32 of *Matrix Analysis* by Horn & Johnson (2nd Ed.), a matrix norm $\|\cdot\|$ is induced iff it is a minimal matrix norm, i.e., the only matrix norm $N(\cdot)$ that satisfies $N(\mathbf{X}) \leq \|\mathbf{X}\|, \forall \mathbf{X}$, is $N(\cdot) = \|\cdot\|$. By this theorem, the Frobenius norm cannot be induced.

Theorem 50. Any norm for matrices can be scaled appropriately so that it becomes a matrix norm.

Define

$$a = \sup_{\|\mathbf{A}\|=1, \|\mathbf{B}\|=1} \|\mathbf{AB}\|.$$

Then $\|\mathbf{X}\|_m = a\|\mathbf{X}\|$ is a matrix norm:

$$\begin{aligned} \|\mathbf{AB}\|_m &= a\|(\mathbf{A}/\|\mathbf{A}\|)(\mathbf{B}/\|\mathbf{B}\|)\| \|\mathbf{A}\| \|\mathbf{B}\| \\ &= a^{-1}\|(\mathbf{A}/\|\mathbf{A}\|)(\mathbf{B}/\|\mathbf{B}\|)\| \|\mathbf{A}\|_m \|\mathbf{B}\|_m \\ &\leq \|\mathbf{A}\|_m \|\mathbf{B}\|_m. \end{aligned}$$

2.11.1.7 Nuclear Norm

Definition 51. The nuclear norm $\|\mathbf{A}\|_*$ of a matrix \mathbf{A} is defined as the sum of all the singular values of \mathbf{A} .

It is usually used for approximating the rank of a matrix, because:

Theorem 52. The nuclear norm is the convex envelope of the rank function on the unit ball $B_1 = \{\mathbf{A} \mid \|\mathbf{A}\|_2 \leq 1\}$ of matrix 2-norm.

Definition 53. Convex envelope is the largest convex function upper bounded by the given function

Think 54. Is nuclear norm $\|\cdot\|_*$ a matrix norm?

Proposition 55. The matrix norms have the following relationship:

$$1. \quad \|\mathbf{A}\|_2 \leq \|\mathbf{A}\|_F \leq \|\mathbf{A}\|_* . \quad (2.8)$$

$$2. \quad \|\mathbf{A}\|_2 \leq \sqrt{\|\mathbf{A}\|_1 \|\mathbf{A}\|_\infty} . \quad (2.9)$$

Proof. $\sigma_1^2 = \|\mathbf{A}\|_2^2$ is the largest eigenvalue of $\mathbf{A}^T \mathbf{A}$. Let its associated eigenvector be \mathbf{x} , normalized such that $\|\mathbf{x}\|_1 = 1$. Then

$$\sigma_1^2 = \|\sigma_1^2 \mathbf{x}\|_1 = \|\mathbf{A}^T \mathbf{A} \mathbf{x}\|_1 \leq \|\mathbf{A}^T\|_1 \|\mathbf{A}\|_1 \|\mathbf{x}\|_1 = \|\mathbf{A}\|_\infty \|\mathbf{A}\|_1 .$$

□

2.11.1.8 (p, q) -Norm

The (p, q) -norm of matrices are also widely used in sparse representation.

Definition 56.

$$\|\mathbf{A}\|_{p,q} = \left(\sum_{i=1}^n \|\mathbf{A}_i\|_p^q \right)^{\frac{1}{q}} ,$$

where $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_n)$ and $p, q \geq 1$.

Note that here we use an intuitive notation for the above norm. It is different from the induced norm of linear mapping: $\mathbb{R}^p \rightarrow \mathbb{R}^q$.

When $q = 1$, minimizing $\|\mathbf{A}\|_{p,q}$ will encourage the columns of \mathbf{A} to be all zeros, which is called the group sparsity².

Think 57. Is (p, q) -Norm $\|\cdot\|_{p,q}$ a matrix norm?

²In sparse representation, we often allow p or q to be smaller than 1. In this case the (p, q) -norm is not a real norm, but only a pseudo-norm.

2.11.1.9 Dual norms

Let $\|\cdot\|$ be a norm on \mathbb{R}^n . The associated *dual norm*, denoted $\|\cdot\|_*$, is defined as

$$\|\mathbf{z}\|_* = \sup \{\mathbf{z}^\top \mathbf{x} \mid \|\mathbf{x}\| \leq 1\}. \quad (2.10)$$

(This can be shown to be a norm.) The dual norm can be interpreted as the operator norm of \mathbf{z}^\top , interpreted as a $1 \times n$ matrix, with the norm $\|\cdot\|$ on \mathbb{R}^n , and the absolute value on \mathbb{R} :

$$\|\mathbf{z}\|_* = \sup \{|\mathbf{z}^\top \mathbf{x}| \mid \|\mathbf{x}\| \leq 1\}.$$

From the definition of dual norm we have the inequality

$$\mathbf{z}^\top \mathbf{x} \leq \|\mathbf{x}\| \|\mathbf{z}\|_*,$$

which holds for all \mathbf{x} and \mathbf{z} . This inequality is tight, in the following sense: for any \mathbf{x} there is a \mathbf{z} for which the inequality holds with equality. (Similarly, for any \mathbf{z} there is an \mathbf{x} that gives equality.) The dual of the dual norm is the original norm: we have $\|\mathbf{x}\|_{**} = \|\mathbf{x}\|$ for all \mathbf{x} . (This need not hold in infinite-dimensional vector spaces.)

The dual of the Euclidean norm is the Euclidean norm, since

$$\sup \{\mathbf{z}^\top \mathbf{x} \mid \|\mathbf{x}\|_2 \leq 1\} = \|\mathbf{z}\|_2.$$

(This follows from the Cauchy-Schwarz inequality; for nonzero \mathbf{z} , the value of \mathbf{x} that maximizes $\mathbf{z}^\top \mathbf{x}$ over $\|\mathbf{x}\|_2 \leq 1$ is $\mathbf{z}/\|\mathbf{z}\|_2$.)

The dual of the l_1 -norm is the l_∞ -norm:

$$\sup \{\mathbf{z}^\top \mathbf{x} \mid \|\mathbf{x}\|_\infty \leq 1\} = \sum_{i=1}^n |z_i| = \|\mathbf{z}\|_1.$$

and the dual of the l_∞ -norm is the l_1 -norm. More generally, the dual of the l_p -norm is the l_q -norm, where q satisfies $1/p + 1/q = 1$, i.e., $q = p/(p-1)$.

As another example, consider the l_2 -or spectral norm on $\mathbb{R}^{m \times n}$. The associated dual norm is³

$$\|\mathbf{Z}\|_{2*} = \sup \{\text{tr}(\mathbf{Z}^\top \mathbf{X}) \mid \|\mathbf{X}\|_2 \leq 1\}, \quad (2.11)$$

which turns out to be the sum of the singular values,

$$\|\mathbf{Z}\|_{2*} = \sigma_1(\mathbf{Z}) + \dots + \sigma_r(\mathbf{Z}) = \text{tr}(\mathbf{Z}^\top \mathbf{Z})^{1/2},$$

where $r = \text{rank } \mathbf{Z}$. This norm is sometimes called the *nuclear norm*.

³In the context of low-rank recovery, $\|\mathbf{Z}\|_{2*}$ is usually written as $\|\mathbf{Z}\|_*$ for brevity.

2.11.1.10 Exercises

Exercise 58. Let $\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \end{pmatrix}$, compute its 1-norm, 2-norm, Frobenious norm, ∞ -norm, nuclear norm, and $(2, 1)$ -norm.

Exercise 59. Prove the following identities:

1.

$$\langle \mathbf{a}, \mathbf{b} \rangle = \frac{1}{2} (\|\mathbf{a} + \mathbf{b}\|^2 - \|\mathbf{a}\|^2 - \|\mathbf{b}\|^2). \quad (2.12)$$

2.

$$\langle \mathbf{a} + \mathbf{c}, \mathbf{b} \rangle = \frac{1}{4} (\|\mathbf{a} + 2\mathbf{b} + \mathbf{c}\|^2 - \|\mathbf{a} + \mathbf{c}\|^2 - 4\|\mathbf{b}\|^2). \quad (2.13)$$

3.

$$\langle \mathbf{a} - \mathbf{b}, \mathbf{c} - \mathbf{d} \rangle = \frac{1}{2} (\|\mathbf{b} - \mathbf{c}\|^2 - \|\mathbf{b} - \mathbf{d}\|^2 - \|\mathbf{a} - \mathbf{c}\|^2 + \|\mathbf{a} - \mathbf{d}\|^2). \quad (2.14)$$

4.

$$\langle \mathbf{a} - \mathbf{b}, \mathbf{a} \rangle = \frac{1}{2} (\|\mathbf{a}\|^2 - \|\mathbf{b}\|^2 + \|\mathbf{a} - \mathbf{b}\|^2). \quad (2.15)$$

5.

$$\langle \mathbf{a} - \mathbf{b}, \mathbf{c} - \mathbf{b} \rangle = \frac{1}{2} (\|\mathbf{a} - \mathbf{b}\|^2 + \|\mathbf{b} - \mathbf{c}\|^2 - \|\mathbf{a} - \mathbf{c}\|^2). \quad (2.16)$$

They are useful in proving the convergence of iterations as the squared norms are known to be nonnegative. Can you find more such identities that relate inner products with squared norms?

Exercise 60. Let $\|\cdot\|$ be a norm on \mathcal{V} that is derived from an inner product, i.e., $\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}$.

1. Show that it satisfies the parallelogram identity

2. Extend the above identity to any number m of vectors $\mathbf{x}_1, \dots, \mathbf{x}_m \in \mathcal{V}$, $m > 2$.

Exercise 61. An important discrete family of norms on \mathbb{R}^n bridges the gap between the l_1 -norm and the l_∞ -norm. For each $k = 1, \dots, n$ the k -norm of a vector \mathbf{x} is obtained by nonincreasingly ordering the absolute values of the entries of \mathbf{x} and adding the k largest values, that is,

$$\|\mathbf{x}\|_{[k]} = |x_{i_1}| + \dots + |x_{i_k}|, \text{ in which } |x_{i_1}| \geq \dots \geq |x_{i_n}|. \quad (2.17)$$

Verify that $\|\mathbf{x}\|_{[k]}$ is a norm on \mathbb{R}^n for each $k = 1, 2, \dots, n$ and that

$$\|\mathbf{x}\|_\infty = \|\mathbf{x}\|_{[1]} \leq \|\mathbf{x}\|_{[2]} \leq \dots \leq \|\mathbf{x}\|_{[n]} = \|\mathbf{x}\|_1, \quad \forall \mathbf{x}.$$

Exercise 62. Prove (2.5), (2.6), and (2.7).

Exercise 63. Define weighted nuclear norm:

$$\|\mathbf{X}\|_{\mathbf{w}*} = \sum_{i=1}^p w_i \sigma_i(\mathbf{X}), \quad \text{where } \mathbf{X} \in \mathbb{R}^{m \times n}, p = \min(m, n), \mathbf{w} \geq \mathbf{0}, \mathbf{w} \neq \mathbf{0}. \quad (2.18)$$

Prove that $\|\cdot\|_{\mathbf{w}*}$ is a true norm for matrices if and only if $w_1 \geq w_2 \geq \dots \geq w_p$. For comparison, what properties should $\mathbf{w} \geq \mathbf{0}$ hold for the weighted l_1 -norm $\|\mathbf{x}\|_{\mathbf{w},1} = \sum_{i=1}^n w_i |x_i|$ to be a vector norm? Why there is such difference between the weighted nuclear norm and weighted l_1 -norm?

Exercise 64. Prove that (2.10) defines a norm.

Exercise 65. Prove that for a norm $\|\cdot\|$ on a finite dimensional space, $\|\mathbf{x}\|_{**} = \|\mathbf{x}\|$ for all \mathbf{x} . If possible, give an example showing that this statement is not true if the linear space is infinitely dimensional.

Exercise 66. Prove that the dual of the l_p -norm is the l_q -norm, where $p \geq 1$ and q satisfies $1/p + 1/q = 1$.

Exercise 67. Prove (2.11).

Exercise 68. Find the dual norm of the (p, q) -norm.

Exercise 69. If none of the columns of \mathbf{D} is zero, prove that

(a) $\|\mathbf{D} \operatorname{Diag}(\mathbf{x})\|_*$ is a norm of \mathbf{x} . This norm is call the trace lasso.

(b) if the columns of \mathbf{D} are all normalized so that their ℓ_2 norms are all 1, then $\|\mathbf{x}\|_2 \leq \|\mathbf{D} \operatorname{Diag}(\mathbf{x})\|_* \leq \|\mathbf{x}\|_1$ and the equalities are both achievable.

Exercise 70. If none of the columns of \mathbf{D} is zero, find the dual norm of the trace lasso $\|\mathbf{D} \operatorname{Diag}(\mathbf{x})\|_*$.

2.11.2 Analysis

2.11.2.1 Open and closed sets

An element $\mathbf{x} \in C \subseteq \mathbb{R}^n$ is called an *interior* point of C if there exists an $\epsilon > 0$ for which

$$\{\mathbf{y} \mid \|\mathbf{y} - \mathbf{x}\|_2 \leq \epsilon\} \subseteq C,$$

i.e., there exists a ball centered at \mathbf{x} that lies entirely in C . The set of all points interior to C is called the *interior* of C and is denoted $\text{int } C$. (Since all norms on \mathbb{R}^n are equivalent to the Euclidean norm, all norms generate the same set of interior points.) A set C is *open* if $\text{int } C = C$, i.e., every point in C is an interior point. A set $C \subseteq \mathbb{R}^n$ is *closed* if its complement $\mathbb{R}^n \setminus C = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{x} \notin C\}$ is open.

The *closure* of a set C is defined as

$$\text{cl } C = \mathbb{R}^n \setminus \text{int } (\mathbb{R}^n \setminus C)$$

i.e., the complement of the interior of the complement of C . A point \mathbf{x} is in the closure of C if for every $\epsilon > 0$, there is a $\mathbf{y} \in C$ with $\|\mathbf{x} - \mathbf{y}\|_2 \leq \epsilon$.

We can also describe closed sets and the closure in terms of convergent sequences and limit points. A set C is closed if and only if it contains the limit point of every convergent sequence in it. In other words, if $\mathbf{x}_1, \mathbf{x}_2, \dots$ converges to \mathbf{x} , and $\mathbf{x}_i \in C$, then $\mathbf{x} \in C$. The closure of C is the set of all limit points of convergent sequences in C .

The *boundary* of the set C is defined as

$$\text{bd } C = \text{cl } C \setminus \text{int } C.$$

A *boundary point* x (i.e., a point $\mathbf{x} \in \text{bd } C$) satisfies the following property: For all $\epsilon > 0$, there exists $\mathbf{y} \in C$ and $\mathbf{z} \notin C$ with

$$\|\mathbf{y} - \mathbf{x}\|_2 \leq \epsilon, \quad \|\mathbf{z} - \mathbf{x}\|_2 \leq \epsilon,$$

i.e., there exist arbitrarily close points in C , and also arbitrarily close points not in C . We can characterize closed and open sets in terms of the boundary operation: C is *closed* if it contains its boundary, i.e., $\text{bd } C \subseteq C$. It is *open* if it contains no boundary points, i.e., $C \cap \text{bd } C = \emptyset$.

2.11.2.2 Supremum and infimum

Suppose $C \subseteq \mathbb{R}$. A number a is an *upper bound* on C if for each $x \in C$, $x \leq a$. The set of upper bounds on a set C is either empty (in which case we say C is unbounded above), all of \mathbb{R} (only when $C = \emptyset$), or a closed infinite interval $[b, \infty)$. The number b is called the *least upper bound* or *supremum* of the set C , and is denoted $\sup C$. We take $\sup \emptyset = -\infty$, and $\sup C = \infty$ if C is unbounded above. When $\sup C \in C$, we say the supremum of C is attained or achieved.

When the set C is finite, $\sup C$ is the maximum of its elements. Some authors use the notation $\max C$ to denote supremum, when it is attained, but we follow standard mathematical convention, using $\max C$ only when the set C is finite.

We define lower bound, and infimum, in a similar way. A number a is a lower bound on $C \subseteq \mathbb{R}$ if for each $x \in C$, $a \leq x$. The *infimum* (or *greatest lower bound*) of a set $C \subseteq \mathbb{R}$ is defined as $\inf C = -\sup(-C)$. When C is finite, the infimum is the minimum of its elements. We take $\inf \emptyset = \infty$, and $\inf C = -\infty$ if C is unbounded below, *i.e.*, has no lower bound.

2.11.2.3 Exercises

Exercise 71. Prove that:

1. the closure of a set \mathcal{A} is the smallest closed set that contains \mathcal{A} .
2. \mathcal{A} is closed iff $\mathcal{A} = \text{cl } \mathcal{A}$.
3. if $\mathcal{A} \subseteq \mathcal{B}$, then $\text{cl } \mathcal{A} \subseteq \text{cl } \mathcal{B}$.
4. $\text{cl}(\mathcal{A} \cup \mathcal{B}) = \text{cl } \mathcal{A} \cup \text{cl } \mathcal{B}$; $\text{cl}(\mathcal{A} \cap \mathcal{B}) \subseteq \text{cl } \mathcal{A} \cap \text{cl } \mathcal{B}$. Give an example showing that $\text{cl}(\mathcal{A} \cap \mathcal{B}) \neq \text{cl } \mathcal{A} \cap \text{cl } \mathcal{B}$.
5. $\text{int } \mathcal{A}$ is the largest open set contained in \mathcal{A} .
6. \mathcal{A} is open iff $\mathcal{A} = \text{int } \mathcal{A}$.
7. if $\mathcal{A} \subseteq \mathcal{B}$ then $\text{int } \mathcal{A} \subseteq \text{int } \mathcal{B}$.
8. $\mathbb{R}^n \setminus \text{int } \mathcal{A} = \text{cl}(\mathbb{R}^n - \mathcal{A})$; $\mathbb{R}^n - \text{cl } \mathcal{A} = \text{int}(\text{cl}(\mathbb{R}^n - \mathcal{A}))$.
9. $\text{int}(\mathcal{A} \cap \mathcal{B}) = \text{int } \mathcal{A} \cap \text{int } \mathcal{B}$; $\text{int}(\mathcal{A} \cup \mathcal{B}) \supseteq \text{int } \mathcal{A} \cup \text{int } \mathcal{B}$. Give an example showing that $\text{int}(\mathcal{A} \cup \mathcal{B}) \neq \text{int } \mathcal{A} \cup \text{int } \mathcal{B}$.
10. $\text{bd } \mathcal{A}$ is a closed set.
11. $\text{bd } \mathcal{A} = \text{cl } \mathcal{A} \cap \text{cl}(\mathbb{R}^n - \mathcal{A})$.
12. $\text{bd } \mathcal{A} = \text{bd}(\mathbb{R}^n - \mathcal{A})$.
13. $\text{cl}(\mathcal{A} \times \mathcal{B}) = \text{cl } \mathcal{A} \times \text{cl } \mathcal{B}$.
14. $\text{int}(\text{cl}(\mathcal{A} \times \mathcal{B})) = \text{int } \mathcal{A} \times \text{cl } \mathcal{B}$. Can you find an example showing that $\text{int}(\mathcal{A} \times \mathcal{B}) \neq \text{int } \mathcal{A} \times \text{cl } \mathcal{B}$?
15. $\text{bd}(\mathcal{A} \times \mathcal{B}) = (\text{bd } \mathcal{A} \times \text{cl } \mathcal{B}) \cup (\text{cl } \mathcal{A} \times \text{bd } \mathcal{B})$.

Exercise 72. Suppose that (\mathcal{X}, d) is a metric space and $\mathcal{A} \subseteq \mathcal{X}$. Prove that:

1. $\text{cl } \mathbf{A} = \{\mathbf{x} \in \mathcal{X} | d(\mathbf{x}, \mathcal{A}) = 0\}$.
2. \mathcal{A} is a closed set iff $\forall \mathbf{x} \in \mathcal{X}$, if $d(\mathbf{x}, \mathcal{A}) = 0$ then $\mathbf{x} \in \mathcal{A}$.

2.11.3 Functions

2.11.3.1 Function notation

Our notation for functions is mostly standard, with one exception. When we write

$$f : A \rightarrow B$$

we mean that f is a function on the set $\mathbf{dom} f \subseteq A$ into the set B ; in particular we can have $\mathbf{dom} f$ a proper subset of the set A . Thus the notation $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ means that f maps (some) n -vectors into m -vectors; it does not mean that $f(x)$ is defined for every $\mathbf{x} \in \mathbb{R}^n$. This convention is similar to function declarations in computer languages. Specifying the data types of the input and output arguments of a function gives the *syntax* of that function; it does not guarantee that any input argument with the specified data type is valid.

As an example consider the function $f : \mathbf{S}^n \rightarrow \mathbb{R}$, given by

$$f(\mathbf{X}) = \log \det \mathbf{X}, \quad (2.19)$$

with $\mathbf{dom} f = \mathbf{S}_{++}^n$. The notation $f : \mathbf{S}^n \rightarrow \mathbb{R}$ specifies the *syntax* of f : it takes as argument a symmetric $n \times n$ matrix, and returns a real number. The notation $\mathbf{dom} f = \mathbf{S}_{++}^n$ specifies which symmetric $n \times n$ matrices are valid input arguments for f (*i.e.*, only positive definite ones). The formula (2.19) specifies what $f(\mathbf{X})$ is, for $\mathbf{X} \in \mathbf{dom} f$.

2.11.3.2 Continuity

A function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is *continuous* at $\mathbf{x} \in \mathbf{dom} f$ if for all $\epsilon > 0$ there exists a δ such that

$$\mathbf{y} \in \mathbf{dom} f, \quad \|\mathbf{y} - \mathbf{x}\|_2 \leq \delta \Rightarrow \|f(\mathbf{y}) - f(\mathbf{x})\|_2 \leq \epsilon.$$

Continuity can be described in terms of limits: whenever the sequence $\mathbf{x}_1, \mathbf{x}_2, \dots$ in $\mathbf{dom} f$ converges to a point $\mathbf{x} \in \mathbf{dom} f$, the sequence $f(\mathbf{x}_1), f(\mathbf{x}_2), \dots$ converges to $f(\mathbf{x})$, *i.e.*,

$$\lim_{i \rightarrow \infty} f(\mathbf{x}_i) = f\left(\lim_{i \rightarrow \infty} \mathbf{x}_i\right).$$

A function f is continuous if it is continuous at every point in its domain.

2.11.3.3 Closed functions

A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is said to be *closed* if, for each $\alpha \in \mathbb{R}$, the sublevel set

$$\{\mathbf{x} \in \mathbf{dom} f | f(\mathbf{x}) \leq \alpha\}$$

is closed. This is equivalent to the condition that the epigraph of f ,

$$\text{epi } f = \{(\mathbf{x}, t) \in \mathbb{R}^{n+1} \mid \mathbf{x} \in \text{dom } f, f(\mathbf{x}) \leq t\},$$

is closed. (This definition is general, but is usually only applied to convex functions.)

If $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is continuous, and $\text{dom } f$ is closed, then f is closed. If $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is continuous, with $\text{dom } f$ open, then f is closed if and only if f converges to ∞ along every sequence converging to a boundary point of $\text{dom } f$. In other words, if $\lim_{i \rightarrow \infty} \mathbf{x}_i = \mathbf{x} \in \text{bd dom } f$, with $\mathbf{x}_i \in \text{dom } f$, we have $\lim_{i \rightarrow \infty} f(\mathbf{x}_i) = \infty$.

Example 73. Examples on \mathbb{R} .

- The function $f : \mathbb{R} \rightarrow \mathbb{R}$, with $f(x) = x \log x$, $\text{dom } f = \mathbb{R}_{++}$, is *not* closed.
- The function $f : \mathbb{R} \rightarrow \mathbb{R}$, with

$$f(x) = \begin{cases} x \log x, & x > 0 \\ 0, & x = 0, \end{cases} \quad \text{dom } f = \mathbb{R}_+,$$

is closed.

- The function $f(x) = -\log x$, $\text{dom } f = \mathbb{R}_{++}$, is closed.

2.11.4 Derivatives

2.11.4.1 Derivative and gradient

Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, and $\mathbf{x} \in \text{int dom } f$. The *derivative* (or *Jacobian*) of f at \mathbf{x} is the matrix $Df(\mathbf{x}) \in \mathbb{R}^{m \times n}$, given by

$$Df(\mathbf{x})_{ij} = \frac{\partial f_i(\mathbf{x})}{\partial \mathbf{x}_j}, \quad i = 1, \dots, m, \quad j = 1, \dots, n, \quad (2.20)$$

provided the partial derivatives exist. If the partial derivatives exist, we say f is differentiable at \mathbf{x} . The function f is differentiable if $\text{dom } f$ is open, and it is differentiable at every point in its domain.

The affine function of \mathbf{z} given by

$$f(\mathbf{x}) + Df(\mathbf{x})(\mathbf{z} - \mathbf{x})$$

is called the *first-order approximation* of f at (or near) \mathbf{x} . Evidently this function agrees with f at $\mathbf{z} = \mathbf{x}$; when \mathbf{z} is close to \mathbf{x} , this affine function is *very close* to f :

$$\lim_{\mathbf{z} \in \text{dom } f, \mathbf{z} \neq \mathbf{x}, \mathbf{z} \rightarrow \mathbf{x}} \frac{\|f(\mathbf{z}) - f(\mathbf{x}) - Df(\mathbf{x})(\mathbf{z} - \mathbf{x})\|_2}{\|\mathbf{z} - \mathbf{x}\|_2} = 0. \quad (2.21)$$

The derivative matrix $Df(\mathbf{x})$ is the *only* matrix in $\mathbb{R}^{m \times n}$ that satisfies the condition (2.21). This gives an alternative method for finding the derivative, by deriving a first-order approximation of the function f at \mathbf{x} .

(added by Zhouchen Lin)

When f is a real valued matrix function, i.e., $f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$, its derivative is defined as:

$$\frac{\partial f}{\partial \mathbf{X}} = \left(\frac{\partial f}{\partial X_{ij}} \right) \in \mathbb{R}^{m \times n}. \quad (2.22)$$

The above definition is actually not inconsistent with the definition in (2.20). By treating $m \times n$ matrices as $(mn) \times 1$ vectors, $\frac{\partial f}{\partial \mathbf{X}}$ is a $1 \times (mn)$ vector. Since we actually don't compute matrices by reshaping them as vectors, $\frac{\partial f}{\partial \mathbf{X}}$ has to be reshaped back as matrices.

The most natural way is to put $\frac{\partial f}{\partial X_{ij}}$ at (i, j) . This will be convenient when updating the matrix variable using gradient related algorithms, e.g., we may simply write

$$\mathbf{X}_{k+1} = \mathbf{X}_k - \alpha_k \frac{\partial f}{\partial \mathbf{X}}.$$

2.11.4.1.1 Gradient When f is real-valued (i.e., $f : \mathbb{R}^n \rightarrow \mathbb{R}$) the derivative $Df(\mathbf{x})$ is a $1 \times n$ matrix, i.e., it is a *row* vector. Its transpose is called the *gradient* of the function:

$$\nabla f(\mathbf{x}) = Df(\mathbf{x})^\top,$$

which is a (column) vector, i.e., in \mathbb{R}^n . Its components are the partial derivatives of f :

$$\nabla f(\mathbf{x})_i = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}_i}, \quad i = 1, \dots, n.$$

The first-order approximation of f at a point $\mathbf{x} \in \text{int dom } f$ can be expressed as (the affine function of \mathbf{z})

$$f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{z} - \mathbf{x}).$$

(added by Zhouchen Lin)

Why we bother with defining gradient while we have already defined derivative? This is because we always treat vectors without the transpose sign as *column* vectors. If f is a real valued vector function, its derivative is a *row* vector. So when writing gradient related algorithms, such as the gradient descent, we will always have to transpose the derivative, e.g.,

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \left(\frac{\partial f}{\partial \mathbf{x}} \right)^T,$$

which is rather cumbersome. By defining $\nabla f = \left(\frac{\partial f}{\partial \mathbf{x}} \right)^T$, the related expression can be greatly simplified.

However, when f is a real valued matrix function, i.e., $f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$, we don't transpose the variable matrix. Even if the variable matrix actually reduces to a row vector, conceptually we still view it as a degenerated matrix and still bear the boldface capital notation. Then since $\frac{\partial f}{\partial \mathbf{X}}$ is of the same dimension as \mathbf{X} , the gradient is actually equal to derivative.

In a nutshell, gradient is only special when f is a real-valued vector function.

2.11.4.1.2 Examples As a simple example consider the quadratic function $f : \mathbb{R}^n \rightarrow \mathbb{R}$,

$$f(\mathbf{x}) = (1/2)\mathbf{x}^\top \mathbf{P}\mathbf{x} + \mathbf{q}^\top \mathbf{x} + r,$$

where $\mathbf{P} \in \mathbf{S}^n$, $\mathbf{q} \in \mathbb{R}^n$, and $r \in \mathbb{R}$. Its derivative at \mathbf{x} is the row vector $Df(\mathbf{x}) = \mathbf{x}^\top \mathbf{P} + \mathbf{q}^\top$, and its gradient is

$$\nabla f(\mathbf{x}) = \mathbf{P}\mathbf{x} + \mathbf{q}.$$

As a more interesting example, we consider the function $f : \mathbf{S}^n \rightarrow \mathbb{R}$, given by

$$f(\mathbf{X}) = \log \det \mathbf{X}, \quad \text{dom } f = \mathbf{S}_{++}^n.$$

One (tedious) way to find the gradient of f is to introduce a basis for \mathbf{S}^n , find the gradient of the associated function, and finally translate the result back to \mathbf{S}^n . Instead, we will directly find the first-order approximation of f at $\mathbf{X} \in \mathbf{S}_{++}^n$. Let $\mathbf{Z} \in \mathbf{S}_{++}^n$ be close to \mathbf{X} , and let $\Delta\mathbf{X} = \mathbf{Z} - \mathbf{X}$ (which is assumed to be small). We have

$$\begin{aligned} \log \det \mathbf{Z} &= \log \det (\mathbf{X} + \Delta\mathbf{X}) \\ &= \log \det (\mathbf{X}^{1/2} (\mathbf{I} + \mathbf{X}^{-1/2} \Delta\mathbf{X} \mathbf{X}^{-1/2}) \mathbf{X}^{1/2}) \\ &= \log \det \mathbf{X} + \log \det (\mathbf{I} + \mathbf{X}^{-1/2} \Delta\mathbf{X} \mathbf{X}^{-1/2}) \\ &= \log \det \mathbf{X} + \sum_{i=1}^n \log (1 + \lambda_i), \end{aligned}$$

where λ_i is the i th eigenvalue of $\mathbf{X}^{-1/2} \Delta\mathbf{X} \mathbf{X}^{-1/2}$. Now we use the fact that $\Delta\mathbf{X}$ is small, which implies λ_i are small, so to first order we have $\log (1 + \lambda_i) \approx \lambda_i$. Using this first-order

approximation in the expression above, we get

$$\begin{aligned}\log \det \mathbf{Z} &\approx \log \det \mathbf{X} + \sum_{i=1}^n \lambda_i \\&= \log \det \mathbf{X} + \operatorname{tr}(\mathbf{X}^{-1/2} \Delta \mathbf{X} \mathbf{X}^{-1/2}) \\&= \log \det \mathbf{X} + \operatorname{tr}(\mathbf{X}^{-1} \Delta \mathbf{X}) \\&= \log \det \mathbf{X} + \operatorname{tr}(\mathbf{X}^{-1}(\mathbf{Z} - \mathbf{X})),\end{aligned}$$

where we have used the fact that the sum of the eigenvalues is the trace, and the property $\operatorname{tr}(\mathbf{AB}) = \operatorname{tr}(\mathbf{BA})$.

Thus, the first-order approximation of f at \mathbf{X} is the affine function of \mathbf{Z} given by

$$f(\mathbf{Z}) \approx f(\mathbf{X}) + \operatorname{tr}(\mathbf{X}^{-1}(\mathbf{Z} - \mathbf{X})).$$

Noting that the second term on the righthand side is the standard inner product of \mathbf{X}^{-1} and $\mathbf{Z} - \mathbf{X}$, we can identify \mathbf{X}^{-1} as the gradient of f at \mathbf{X} . Thus, we can write the simple formula

$$\nabla f(\mathbf{X}) = \mathbf{X}^{-1}.$$

This result should not be surprising, since the derivative of $\log x$, on \mathbb{R}_{++} , is $1/x$.

2.11.4.2 Chain rule

Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is differentiable at $\mathbf{x} \in \operatorname{int} \operatorname{dom} f$ and $g : \mathbb{R}^m \rightarrow \mathbb{R}^p$ is differentiable at $f(\mathbf{x}) \in \operatorname{int} \operatorname{dom} g$. Define the composition $h : \mathbb{R}^n \rightarrow \mathbb{R}^p$ by $h(\mathbf{z}) = g(f(\mathbf{z}))$. Then h is differentiable at \mathbf{x} , with derivative

$$Dh(\mathbf{x}) = Dg(f(\mathbf{x}))Df(\mathbf{x}). \quad (2.23)$$

As an example, suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g : \mathbb{R} \rightarrow \mathbb{R}$, and $h(\mathbf{x}) = g(f(\mathbf{x}))$. Taking the transpose of $Dh(\mathbf{x}) = Dg(f(\mathbf{x}))Df(\mathbf{x})$ yields

$$\nabla h(\mathbf{x}) = g'(f(\mathbf{x}))\nabla f(\mathbf{x}). \quad (2.24)$$

2.11.4.2.1 Composition with affine function Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is differentiable, $\mathbf{A} \in \mathbb{R}^{n \times p}$, and $\mathbf{b} \in \mathbb{R}^n$. Define $g : \mathbb{R}^p \rightarrow \mathbb{R}^m$ as $g(\mathbf{x}) = f(\mathbf{Ax} + \mathbf{b})$, with $\operatorname{dom} g = \{\mathbf{x} | \mathbf{Ax} + \mathbf{b} \in \operatorname{dom} f\}$. The derivative of g is, by the chain rule (2.23), $Dg(\mathbf{x}) = Df(\mathbf{Ax} + \mathbf{b})\mathbf{A}$.

When f is real-valued (*i.e.*, $m = 1$), we obtain the formula for the gradient of a composition of a function with an affine function,

$$\nabla g(\mathbf{x}) = \mathbf{A}^\top \nabla f(\mathbf{Ax} + \mathbf{b}).$$

For example, suppose that $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $\mathbf{x}, \mathbf{v} \in \mathbb{R}^n$, and we define the function $\tilde{f} : \mathbb{R} \rightarrow \mathbb{R}$ by $\tilde{f}(t) = f(\mathbf{x} + t\mathbf{v})$. (Roughly speaking, \tilde{f} is f , restricted to the line $\{\mathbf{x} + t\mathbf{v} | t \in \mathbb{R}\}$.) Then we have

$$D\tilde{f}(t) = \tilde{f}'(t) = \nabla f(\mathbf{x} + t\mathbf{v})^\top \mathbf{v}.$$

(The scalar $\tilde{f}'(0)$ is the *directional derivative* of f , at \mathbf{x} , in the direction \mathbf{v} .)

Example 74. Consider the function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, with $\text{dom } f = \mathbb{R}^n$ and

$$f(\mathbf{x}) = \log \sum_{i=1}^m \exp (\mathbf{a}_i^\top \mathbf{x} + b_i),$$

where $\mathbf{a}_1, \dots, \mathbf{a}_m \in \mathbb{R}^n$, and $b_1, \dots, b_m \in \mathbb{R}$. We can find a simple expression for its gradient by noting that it is the composition of the affine function $\mathbf{Ax} + \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{m \times n}$ with rows $\mathbf{a}_1^\top, \dots, \mathbf{a}_m^\top$, and the function $g : \mathbb{R}^m \rightarrow \mathbb{R}$ given by $g(\mathbf{y}) = \log (\sum_{i=1}^m \exp y_i)$. Simple differentiation (or the formula (2.24)) shows that

$$\nabla g(\mathbf{y}) = \frac{1}{\sum_{i=1}^m \exp y_i} \begin{bmatrix} \exp y_1 \\ \vdots \\ \exp y_m \end{bmatrix}, \quad (2.25)$$

so by the composition formula we have

$$\nabla f(\mathbf{x}) = \frac{1}{\mathbf{1}^\top \mathbf{z}} \mathbf{A}^\top \mathbf{z}$$

where $z_i = \exp (\mathbf{a}_i^\top \mathbf{x} + b_i)$, $i = 1, \dots, m$.

Example 75. We derive an expression for $\nabla f(\mathbf{x})$, where

$$f(\mathbf{x}) = \log \det (\mathbf{F}_0 + x_1 \mathbf{F}_1 + \cdots + x_n \mathbf{F}_n),$$

where $\mathbf{F}_0, \dots, \mathbf{F}_n \in \mathbb{S}_{++}^p$, and

$$\text{dom } f = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{F}_0 + x_1 \mathbf{F}_1 + \cdots + x_n \mathbf{F}_n \succ 0\}.$$

The function f is the composition of the affine mapping from $\mathbf{x} \in \mathbb{R}^n$ to $\mathbf{F}_0 + x_1 \mathbf{F}_1 + \cdots + x_n \mathbf{F}_n \in \mathbb{S}_{++}^p$, with the function $\log \det \mathbf{X}$. We use the chain rule to evaluate

$$\frac{\partial f(\mathbf{x})}{\partial x_i} = \text{tr} (\mathbf{F}_i \nabla \log \det (\mathbf{F})) = \text{tr} (\mathbf{F}^{-1} \mathbf{F}_i),$$

where $\mathbf{F} = \mathbf{F}_0 + x_1 \mathbf{F}_1 + \cdots + x_n \mathbf{F}_n$. Thus we have

$$\nabla f(\mathbf{x}) = \begin{bmatrix} \text{tr} (\mathbf{F}^{-1} \mathbf{F}_1) \\ \vdots \\ \text{tr} (\mathbf{F}^{-1} \mathbf{F}_n) \end{bmatrix}.$$

2.11.4.3 Second derivative

In this section we review the second derivative of a real-valued function $f : \mathbb{R}^n \rightarrow \mathbb{R}$. The second derivative or *Hessian matrix* of f at $\mathbf{x} \in \text{int dom } f$, denoted $\nabla^2 f(\mathbf{x})$, is given by

$$\nabla^2 f(\mathbf{x})_{ij} = \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j}, \quad i = 1, \dots, n, \quad j = 1, \dots, n,$$

provided f is twice differentiable at \mathbf{x} , where the partial derivatives are evaluated at \mathbf{x} . The *second-order approximation* of f , at or near \mathbf{x} , is the quadratic function of \mathbf{z} defined by

$$\hat{f}(\mathbf{z}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{z} - \mathbf{x}) + (1/2)(\mathbf{z} - \mathbf{x})^\top \nabla^2 f(\mathbf{x})(\mathbf{z} - \mathbf{x}).$$

This second-order approximation satisfies

$$\lim_{\mathbf{z} \in \text{dom } f, \mathbf{z} \neq \mathbf{x}, \mathbf{z} \rightarrow \mathbf{x}} \frac{|f(\mathbf{z}) - \hat{f}(\mathbf{z})|}{\|\mathbf{z} - \mathbf{x}\|_2^2} = 0.$$

Not surprisingly, the second derivative can be interpreted as the derivative of the first derivative. If f is differentiable, the *gradient mapping* is the function $\nabla f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, with $\text{dom } \nabla f = \text{dom } f$, with value $\nabla f(\mathbf{x})$ at \mathbf{x} . The derivative mapping is

$$D\nabla f(\mathbf{x}) = \nabla^2 f(\mathbf{x}).$$

2.11.4.3.1 Examples

As a simple example consider the quadratic function $f : \mathbb{R}^n \rightarrow \mathbb{R}$,

$$f(\mathbf{x}) = (1/2)\mathbf{x}^\top \mathbf{P}\mathbf{x} + \mathbf{q}^\top \mathbf{x} + r,$$

where $\mathbf{P} \in \mathbf{S}^n$, $\mathbf{q} \in \mathbb{R}^n$, and $r \in \mathbb{R}$. Its gradient is $\nabla f(\mathbf{x}) = \mathbf{P}\mathbf{x} + \mathbf{q}$, so its Hessian is given by $\nabla^2 f(\mathbf{x}) = \mathbf{P}$. The second-order approximation of a quadratic function is itself.

As a more complicated example, we consider again the function $f : \mathbf{S}^n \rightarrow \mathbb{R}$, given by $f(\mathbf{X}) = \log \det \mathbf{X}$, with $\text{dom } f = \mathbf{S}_{++}^n$. To find the second-order approximation (and therefore, the Hessian), we will derive a first-order approximation of the gradient, $\nabla f(\mathbf{X}) = \mathbf{X}^{-1}$. For $\mathbf{Z} \in \mathbf{S}_{++}^n$ near $\mathbf{X} \in \mathbf{S}_{++}^n$, and $\Delta\mathbf{X} = \mathbf{Z} - \mathbf{X}$, we have

$$\begin{aligned} \mathbf{Z}^{-1} &= (\mathbf{X} + \Delta\mathbf{X})^{-1} \\ &= (\mathbf{X}^{1/2}(\mathbf{I} + \mathbf{X}^{-1/2}\Delta\mathbf{X}\mathbf{X}^{-1/2})\mathbf{X}^{1/2})^{-1} \\ &= \mathbf{X}^{-1/2}(\mathbf{I} + \mathbf{X}^{-1/2}\Delta\mathbf{X}\mathbf{X}^{-1/2})^{-1}\mathbf{X}^{-1/2} \\ &\approx \mathbf{X}^{-1/2}(\mathbf{I} - \mathbf{X}^{-1/2}\Delta\mathbf{X}\mathbf{X}^{-1/2})\mathbf{X}^{-1/2} \\ &= \mathbf{X}^{-1} - \mathbf{X}^{-1}\Delta\mathbf{X}\mathbf{X}^{-1}, \end{aligned}$$

using the first-order approximation $(\mathbf{I} + \mathbf{A})^{-1} \approx \mathbf{I} - \mathbf{A}$, valid for \mathbf{A} being small.

This approximation is enough for us to identify the Hessian of f at \mathbf{X} . The Hessian is a quadratic form on \mathbf{S}^n . Such a quadratic form is cumbersome to describe in the general case, since it requires four indices. But from the first-order approximation of the gradient above, the quadratic form can be expressed as

$$-\operatorname{tr}(\mathbf{X}^{-1}\mathbf{U}\mathbf{X}^{-1}\mathbf{V}),$$

where $\mathbf{U}, \mathbf{V} \in \mathbf{S}^n$ are the arguments of the quadratic form. (This generalizes the expression for the scalar case: $(\log x)^n = -1/x^2$.)

Now we have the second-order approximation of f near \mathbf{X} :

$$\begin{aligned} f(\mathbf{Z}) &= f(\mathbf{X} + \Delta\mathbf{X}) \\ &\approx f(\mathbf{X}) + \operatorname{tr}(\mathbf{X}^{-1}\Delta\mathbf{X}) - (1/2)\operatorname{tr}(\mathbf{X}^{-1}\Delta\mathbf{X}\mathbf{X}\mathbf{X}^{-1}\Delta\mathbf{X}) \\ &\approx f(\mathbf{X}) + \operatorname{tr}(\mathbf{X}^{-1}(\mathbf{Z} - \mathbf{X})) - (1/2)\operatorname{tr}(\mathbf{X}^{-1}(\mathbf{Z} - \mathbf{X})\mathbf{X}^{-1}(\mathbf{Z} - \mathbf{X})). \end{aligned}$$

2.11.4.4 Chain rule for second derivative

A general chain rule for the second derivative is cumbersome in most cases, so we will state it only for some special cases that we will need.

2.11.4.4.1 Composition with scalar function Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g : \mathbb{R} \rightarrow \mathbb{R}$, and $h(\mathbf{x}) = g(f(\mathbf{x}))$. Simply working out the partial derivatives yields

$$\nabla^2 h(\mathbf{x}) = g'(f(\mathbf{x}))\nabla^2 f(\mathbf{x}) + g''(f(\mathbf{x}))\nabla f(\mathbf{x})\nabla f(\mathbf{x})^\top. \quad (2.26)$$

2.11.4.4.2 Composition with affine function Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $\mathbf{A} \in \mathbb{R}^{n \times m}$, and $\mathbf{b} \in \mathbb{R}^n$. Define $g : \mathbb{R}^m \rightarrow \mathbb{R}$ by $g(\mathbf{x}) = f(\mathbf{Ax} + \mathbf{b})$. Then we have

$$\nabla^2 g(\mathbf{x}) = \mathbf{A}^\top \nabla^2 f(\mathbf{Ax} + \mathbf{b}) \mathbf{A}.$$

As an example, consider the restriction of a real-valued function f to a line, *i.e.*, the function $\tilde{f}(t) = f(\mathbf{x} + t\mathbf{v})$, where \mathbf{x} and \mathbf{v} are fixed. Then we have

$$\nabla^2 \tilde{f}(t) = \tilde{f}''(t) = \mathbf{v}^\top \nabla^2 f(\mathbf{x} + t\mathbf{v}) \mathbf{v}.$$

Example 76. We consider the function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ from Example 74,

$$f(\mathbf{x}) = \log \sum_{i=1}^m \exp(\mathbf{a}_i^\top \mathbf{x} + b_i),$$

where $\mathbf{a}_1, \dots, \mathbf{a}_m \in \mathbb{R}^n$, and $b_1, \dots, b_m \in \mathbb{R}$. By noting that $f(\mathbf{x}) = g(\mathbf{Ax} + \mathbf{b})$, where $g(\mathbf{y}) = \log(\sum_{i=1}^m \exp y_i)$, we can obtain a simple formula for the Hessian of f . Taking partial derivatives, or using the formula (2.26), noting that g is the composition of \log with $\sum_{i=1}^m \exp y_i$, yields

$$\nabla^2 g(\mathbf{y}) = \text{diag}(\nabla g(\mathbf{y})) - \nabla g(\mathbf{y})\nabla g(\mathbf{y})^\top,$$

where $\nabla g(\mathbf{y})$ is given in (2.25). By the composition formula we have

$$\nabla^2 f(\mathbf{x}) = \mathbf{A}^\top \left(\frac{1}{\mathbf{1}^\top \mathbf{z}} \text{diag}(\mathbf{z}) - \frac{1}{(\mathbf{1}^\top \mathbf{z})^2} \mathbf{z}\mathbf{z}^\top \right) \mathbf{A},$$

where $z_i = \exp(\mathbf{a}_i^\top \mathbf{x} + b_i)$, $i = 1, \dots, m$.

(Taken from Chapters 5.5.2 - 5.5.3 of [56])

2.11.5 Backpropagation

Backpropagation is likely to be the most popular word in machine learning. Yet, it is quite often the source of a surprising misunderstanding. More than a learning algorithm, it is an efficient gradient computation algorithm, which, as it will be seen later on in this section, is in fact optimal! Learning algorithms typically require computing the gradient of the loss for any example \mathbf{v} , that is, ∇e , where $e(\mathbf{w}, \mathbf{v}, \mathbf{y}) = V(\mathbf{v}, \mathbf{y}, f(\mathbf{w}, \mathbf{v}))$. In order to grasp the idea, it is important to realize that the derivatives of a function can either be computed numerically or symbolically. For instance, if we want to compute $\sigma'(a)$, where $\sigma(a) = 1/(1 + e^{-a})$, the symbolic derivation immediately leads us to notice that

$$\sigma'(a) = \sigma(a)(1 - \sigma(a)). \quad (2.27)$$

Alternatively, one can use numerical schemes that are typically based on clever approximations; for instance, we have

$$\sigma^{(1)}(a) = \frac{\sigma(a+h) - \sigma(a-h)}{2h} - \frac{h^2}{6} \sigma^{(3)}(\tilde{a}), \quad (2.28)$$

where $\tilde{a} \in (a-h, a+h)$ and $\sigma^{(k)}$ is the k -th order derivative of σ . Of course, for “small” h we have $\sigma^{(1)}(a) \approx (\sigma(a+h) - \sigma(a-h))/2h$, which gives rise to a good numerical scheme to compute $\sigma^{(1)}(a)$.

Suppose use a numerical computation of the gradient, where any of its components $\partial e / \partial w_{ij}$ is computed using the idea sketched in Eq. (2.28), where w_{ij} is the weight connecting node i and j . Hence

$$\frac{\partial e}{\partial w_{ij}} \leftarrow \frac{e(w_{ij} + h, \mathbf{v}, \mathbf{y}) - e(w_{ij} - h, \mathbf{v}, \mathbf{y})}{2h}. \quad (2.29)$$

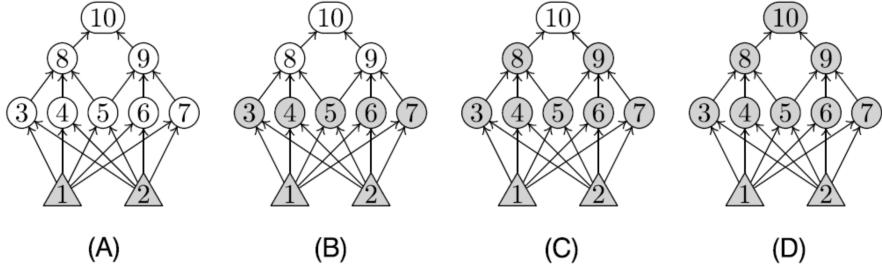


图 2.6: Data flow computation: The input is applied at the first layer (A). It is then propagated forward to the second (B), third (C), and fourth (D) layer, which contains the output.

Let $\mathcal{N} = (\mathcal{V}, \mathcal{A})$ be a feedforward network and \mathcal{I} be the set of indices of the input nodes. According to the above equation, the computation of $\partial e / \partial w_{ij}$ requires three floating-points operations. However, since we are interested in an asymptotical analysis, we can promptly see that we are reduced to determining the complexity of computing $e(\mathbf{w}, \mathbf{v}, \mathbf{y})$ that, in turn, is reduced to establishing the complexity of the computation of $f(\mathbf{w}, \mathbf{v})$. Data flow computation. Fig. 2.6 shows how such a computation takes place in a feedforward network with two hidden layers. First (see (A)) the input is applied to the inputs (grey level, units 1, 2). Then it is propagated forward to the second (B), third (C), and fourth layer (D) (output). Any of the three forward computations, which construct the outputs of the two hidden layers (B, C) and of the output (D), require (asymptotically) as many floating-point operations as the number of connections with the previous layer. For example, on layer (B), we need to compute $x_i = \sigma(w_{i1}x_1 + w_{i2}x_2 + b_i)$ for every $i = 3, \dots, 7$. When considering a neural network as simple as this, modeling the cost of x_i is not a trivial issue. In particular, it is important to know what kind of threshold function $\sigma(\cdot)$ we are considering and, in addition, how we compute it on the given platform. As an extreme case, we can dramatically optimize the computation of σ if we use its tabular-based approximation. Clearly, with large numbers of inputs and neurons one can regard the cost of σ as $O(1)$. Hence, the computation of outputs of any layers requires a number of floating-point operations that corresponds with the number of weights (including the bias) that are connected with the previous layer. Overall, the complexity grows proportionally to the number of weights. Of course, this holds true also in case of a generic DAG, since the data flow computation requires finding $x_i = \sigma(\sum_j w_{ij}x_j)$, where $j \in \text{pa}(i)$ is any parent of i in the DAG.

Algorithm F (Forward propagation). Given a neural network $\mathcal{N} = (\mathcal{G}, \mathbf{w})$ based on the DAG \mathcal{G} and on the weights \mathbf{w} , a vector \mathbf{m} , that will be used as a weight modifier, and a

vector of inputs \mathbf{v} , for all $i \in \mathcal{V} \setminus \mathcal{I}$ the algorithm computes the state of vertex i and stores its value into the vector x_i . We assume that we have already defined $\text{TOPSORT}(\mathcal{S}, \mathbf{s})$ that takes a set \mathcal{S} equipped with an ordering \prec and copies the elements of this set into the topologically sorted array \mathbf{s} , so that for each i and j with $i < j$ we have $s_i \prec s_j$. In what follows this algorithm will be invoked by $\text{FORWARD}(\mathcal{G}, \mathbf{w}, \mathbf{m}, \mathbf{v}, \mathbf{x})$.

- F1. [Initialize.] For all $i \in \mathcal{I}$ set $x_i \leftarrow v_i$ and initialize an integer variable $k \leftarrow 1$.
- F2. [Topsort.] Invoke TOPSORT on $\mathcal{V} \setminus \mathcal{I}$, so that now the vector \mathbf{s} contains the topological sorting of the nodes of the net. Set the variable l to the dimension of the vector \mathbf{s} .
- F3. [Finished yet?] If $k \leq l$ go on to step F4, otherwise the algorithm stops.
- F4. [Compute the state \mathbf{x} .] If $\mathbf{m} = (1, 1, \dots, 1)^T$ set $x_{s_k} \leftarrow \sigma(\sum_{j \in \text{pa}(s_k)} w_{s_k j} x_j)$ otherwise set $x_{s_k} \leftarrow m_{s_k} \sum_{j \in \text{pa}(s_k)} w_{s_k j} x_j$. (The nonuniform weights \mathbf{m} are for backpropagating errors using the same FORWARD subroutine. See (2.34).) Increase k by one and go back to step F3.

This is sketched in Algorithm F, which relies on the topological sort of the neurons identified by the vertices $\mathcal{V} \setminus \mathcal{I}$ in $\mathcal{N} = (\mathcal{V}, \mathcal{A})$. For example, in Fig. 2.6, $\mathcal{V} \setminus \mathcal{I} = \{3, 4, 5, 6, 7, 8, 9, 10\}$ and, amongst the possible topological sorts, we can clearly choose $\mathbf{s} = (3, 4, 5, 6, 7, 8, 9, 10)^T$. While this is a trivial issue in multilayer nets, in a general digraph, there are clearly a lot of different ways of sorting the vertices, but one of them can be found in linear time – this is in fact the cost of TOPSORT . The subsequent loop for the forward step takes $O(|\mathcal{A}|)$, since we need to accumulate the value of the activations for all the arcs. This dominates in the algorithm, so that the computation of $f(\mathbf{w}, x_k)$ and, consequently, of $\partial e / \partial w_{ij}$ is clearly optimal, that is, $\Theta(|\mathcal{A}|)$, since it is also $\Omega(|\mathcal{A}|)$. Let $m = |\mathcal{A}|$ be the number of arcs, which corresponds to the number of weights. Hence the numerical computation of all the $|\mathcal{A}|$ components of the gradient requires $O(m^2)$. Feedforward neural networks are sometimes applied in problems where m is on the order of millions! The numerical computation of the gradient in those cases would require teraflops. This is a remarkable computational burden when considering that this is only for the computation of the gradient associated with a single pattern! As we will see, Backpropagation is a clever algorithm to dramatically cut this bound to $O(m)$.

In order to come up with a solution to compute the gradient smarter than based on Eq. (2.29), one should realize that the same forward step is repeated for all the weights m times, and we do not capitalize from previous computations. Let us attack the problem

by analytically expressing the gradient with symbolic manipulations. We start noticing that

$$\frac{\partial e}{\partial \mathbf{w}} = \frac{\partial V}{\partial f} \frac{\partial f}{\partial \mathbf{w}} = \sum_{o \in \mathcal{O}} \frac{\partial V}{\partial f_o} \frac{\partial f_o}{\partial \mathbf{w}}. \quad (2.30)$$

Whenever we are given a symbolic expression for $V(y, f(\mathbf{w}, \mathbf{v}))$, the first term in Eq. (2.30) can also be given a corresponding symbolic expression. For example, in case $V(y, f) = \frac{1}{2}(y - f)^2$, we have $\nabla_f V = -(y - f)$ and, therefore, its computation requires a forward step to determine $f(\mathbf{w}, \mathbf{v})$. The symbolic expression of $\partial f / \partial \mathbf{w}$ can be gained if we exploit the DAG structure of feedforward nets. Consider the derivative of $f_o(\mathbf{w}, \mathbf{v})$ with respect to the (i, j) th weight w_{ij} , and call this quantity g_{ij}^o ; by using the chain rule, we get

$$g_{ij}^o = \frac{\partial x_o}{\partial w_{ij}} = \frac{\partial x_o}{\partial a_i} \frac{\partial a_i}{\partial w_{ij}} = \frac{\partial x_o}{\partial a_i} \frac{\partial}{\partial w_{ij}} \sum_{h \in \text{pa}(i)} w_{ih} x_h = \delta_i^o x_j, \quad (2.31)$$

where $a_i = \sum_{h \in \text{pa}(i)} w_{ih} x_h$ and we have defined $\delta_i^o := \partial x_o / \partial a_i$. This definition, which is motivated by the computation of g_{ij}^o , can be generalized when considering the transfer of the activation a_i onto the unit j . That is, we can replace δ_i^o with δ_i^j by assuming that the role of $o \in \mathcal{O}$ is moved to $j \in \mathcal{H}$. Clearly, $\delta_i^j = 0$ whenever $i \succ j$. We can immediately determine the gradient with respect to the bias, since⁴ $\partial x_o / \partial b_i = \delta_i^o$. The term δ_i^o is referred to as the delta error.

Let $m \in \mathcal{O}$ be the index of an output neuron. Then, by definition, the delta error is different from zero only when $m = o$, and in that case we have

$$\delta_o^o = \sigma'(a_o). \quad (2.32)$$

For asymmetric sigmoidal functions, from Eq. (2.27), we get $\delta_o^o = x_o(1 - x_o)$. In case of symmetric sigmoidal functions $\sigma(a) = \tanh(a)$, similarly, we have

$$\delta_o^o = \frac{1}{2}(1 + x_o)(1 - x_o),$$

and related symbolic expressions can be found for other LTU units that directly involve the value of x_o . Basically, once the forward step has been completed and x_o is known, we can compute δ_o^o directly. If $i \in \mathcal{H}$ is the index of any hidden unit then δ_i^o cannot be directly expressed like for the case of output units. However, by using the chain rule we have

$$\delta_i^o = \frac{\partial x_o}{\partial a_i} = \sum_{h \in \text{ch}(i)} \frac{\partial x_o}{\partial a_h} \frac{\partial a_h}{\partial x_i} \frac{\partial x_i}{\partial a_i} = \sigma'(a_i) \sum_{h \in \text{ch}(i)} w_{hi} \delta_h^o. \quad (2.33)$$

⁴For the sake of simplicity, in the following discussion we will incorporate the bias as an ordinary weight, by assuming the \mathbf{x} has been enriched, as usual, by $\hat{\mathbf{x}} = (\mathbf{x}^T, 1)^T$.

Eqs. (2.32) and (2.33) allow us to determine δ_i^o by propagating backward the values δ_o^o throughout the hidden units $i \in \mathcal{H}$. This is shown in Fig. 2.7, where we can see the recursive propagation based on the children of the output.⁵

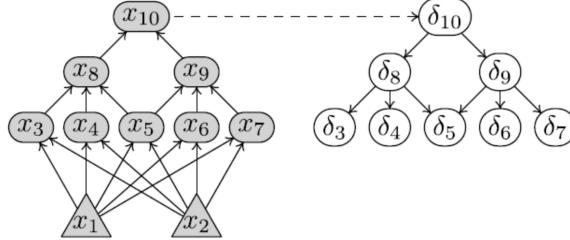


图 2.7: The backward step propagates recursively the delta error beginning from the output through its children. For example, $\delta_5 = \sigma'(a_5)(w_{85}\delta_8 + w_{95}\delta_9)$. Since there is only one output, we don't bother to write down the index o .

Now suppose that instead of the derivative of each output x_o we want to calculate the derivative of the loss V with respect to the generic weight w_{ij} . We immediately realize that we can follow the steps outlined above since we can exploit the chain rule

$$\frac{\partial V}{\partial w_{ij}} = \frac{\partial V}{\partial a_i} \frac{\partial a_i}{\partial w_{ij}} = \delta_i x_j,$$

where this time δ_i is simply $\partial V / \partial a_i$. As before, after the forward phase, we can immediately evaluate the δ_i for $i \in \mathcal{O}$ once we know the symbolic expression of V ; for example, in case of quadratic loss $V(y, f) = \frac{1}{2}(y - f)^2$, then $\delta_o = (\sigma(a_o) - y_o)\sigma'(a_o)$. Of course, we can recursively evaluate all the other δ_i using an analogue of Eq. (2.33),

$$\delta_i = \sum_{h \in \text{ch}(i)} \frac{\partial V}{\partial a_h} \frac{\partial a_h}{\partial x_i} \frac{\partial x_i}{\partial a_i} = \sigma'(a_i) \sum_{h \in \text{ch}(i)} w_{hi} \delta_h. \quad (2.34)$$

We will now show how these ideas can be used to write an algorithm that computes the derivatives either of the output or of the loss function for a general DAG with respect to the weights.

Algorithm B (Backward propagation). Given a network $\mathcal{N} = (\mathcal{G}, \mathbf{w})$ based on the DAG \mathcal{G} , all the states x_i of the vertices of \mathcal{G} , a parameter q , and the symbolic expression of a loss function $V(y, f)$, depending on whether q is positive or not, it returns the derivatives g_{ij}^q , if $q > 0$ and $q \in \mathcal{O}$; otherwise for any $q \leq 0$, it returns the derivatives of the loss $\partial V / \partial w_{ij}$. In what follows the algorithm is invoked as $\text{BACKWARD}(\mathcal{G}, \mathbf{w}, \mathbf{x}, q, V)$, where V is the name of the loss.

⁵Since δ_i^o is needed in Eq. (2.31) for the gradient computation, we can immediately see that there is no propagation throughout the inputs.

- B1. [Loss or output?] If $q \leq 0$ go to step B2, otherwise jump to step B3.
- B2. [Initialize the loss.] For all $o \in \mathcal{O}$ set $v_o \leftarrow \partial V / \partial a_o$ and go to step B4.
- B3. [Initialize x_q .] For each $o \in \mathcal{O}$ if $o \neq q$ set $v_o \leftarrow 0$, else if $o = q$ make the assignment $v_o \leftarrow \sigma'(\sigma^{-1}(x_o))$.
- B4. [Compute backwards.] For each $k \in \mathcal{V} \setminus \mathcal{I}$ set $m_k \leftarrow \sigma'(\sigma^{-1}(x_k))$, then invoke FORWARD($(\mathcal{G} \setminus \mathcal{I})^T, \mathbf{w}^T, \mathbf{m}, \mathbf{v}, \boldsymbol{\delta}$), where $(\mathcal{G} \setminus \mathcal{I})^T$ and \mathbf{w}^T means inverting the direction of the graph \mathcal{G} and weight.
- B5. [Output the gradient.] For each $i \in \mathcal{V} \setminus \mathcal{I}$ and then for each $j \in \text{pa}(i)$ set $g_{ij} \leftarrow \delta_i x_j$ and output g_{ij} . Terminate the algorithm.

In the latter algorithm we are assuming – specifically in step B2 – that we are able to handle symbolic differentiation; this topic will be discussed in more details in Section 2.11.6. It's interesting to notice that the assignments $v_o \leftarrow \sigma'(\sigma^{-1}(x_o))$ in step B3 and $m_k \leftarrow \sigma'(\sigma^{-1}(x_k))$ are almost immediate once we have chosen a specific σ function; for example if $\sigma = \tanh$ we have that $\sigma'(\sigma^{-1}(x_k)) = 1/2(1 + x_k)(1 - x_k)$. A last comment on this algorithm is in order: In step B4 FORWARD is invoked on the graph $(\mathcal{G} \setminus \mathcal{I})'$; we have used this notation to indicate the graph obtained by \mathcal{G} by pruning the input nodes (together with the arcs attached to those nodes) and reversing the direction of the arrows in the remaining graph. This way of performing the calculations is what characterize the algorithm as the “backward propagation.”

Now that we have defined Algorithms F and B, we are ready to introduce the back-propagation algorithm.

Algorithm FB (Backpropagation). Given a network $\mathcal{N} = (\mathcal{G}, \mathbf{w})$ based on the DAG \mathcal{G} , a vector of inputs \mathbf{v} , and a loss function V , the algorithm returns the gradient of the loss with respect to \mathbf{w} .

- FB1. [Forward] Invoke FORWARD($\mathcal{G}, \mathbf{w}, (1, 1, \dots, 1)^T, \mathbf{v}, \mathbf{x}$).
- FB2. [Backward] Invoke BACKWARD($\mathcal{G}, \mathbf{w}, \mathbf{x}, -1, V$). Terminate the algorithm.

The algorithm requires that the forward step has already been carried out, whose effect is that of determining all the x_k . Once the output values for all the neurons are given, we start computing the δ s by means of the backward step. At this point the gradients are obtained using Eq. (2.31). Notice also that in Algorithms F we have introduced the modifier vector \mathbf{m} so that we could use the same algorithm to compute

the backward step; this is necessary since in order to compute the i -th delta error we need to multiply the i -th activation by $\sigma'(a_i)$. From the analysis of the algorithm we can easily draw the conclusion that it has complexity $O(m^2)$, just like for the forward step of Algorithm F. The cost arises from the computation of the delta error as well as from the computation of the gradients. As clearly shown also in Fig. 2.7, the backpropagation of the error is restricted to the hidden units, but the gradient computation involves all the weights, which indicates that it is dominant and involves all the weights of the neural network.

In case of layered structures, which is common in many applications, the forward/backward steps get a very simple structure that is sketched in Fig. 2.6 and Fig. 2.7, respectively. We can also express the forward/backward equations by referring to the indexes of the layers using the tensor formalism. We have

$$\hat{X}_q = \sigma(\hat{X}_{q-1}\hat{W}_q), \quad q = 0, \dots, Q. \quad (2.35)$$

This expression clearly shows the composition of the map that takes place on a layered architecture with Q layers, which returns

$$\hat{X}_Q = \sigma(\cdots \sigma(\sigma(\hat{X}_0\hat{W}_1)\hat{W}_2) \cdots \hat{W}_Q).$$

For $Q = 3$ we have $\hat{X}_3 = \sigma(\sigma(\sigma(\hat{X}_0\hat{W}_1)\hat{W}_2)\hat{W}_3)$, which indicates a nice symmetry: The input \hat{X}_0 is right-multiplied by the weight matrices and left-processed by the σ . Likewise the backward step returns both the delta error and the gradient according to

$$\begin{aligned} \Delta_{q-1} &= \sigma' \odot (\Delta_q W_l), \\ G_q &= \hat{X}'_{q-1} \Delta_q, \end{aligned} \quad (2.36)$$

where $\sigma' \in \mathbb{R}^{L,q-1}$ is the matrix with coordinates $\sigma'(a_{i,k})$, \odot is the Hadamard product, and $\Delta_q := (\delta_1, \dots, \delta_{n(q)}) \in \mathbb{R}^{q,n(q)}$.

Now we use similar arguments rooted in the graphical structure of feedforward nets to express the Hessian matrix, which turns out to be useful to investigate the nature of the critical points of the error function. Let $v_k := V(x_k, y_k, f(x_k))$, then a generic

coordinate of the Hessian matrix can be expressed as

$$\begin{aligned}
 h_{ij,lm} &= \frac{\partial^2 v_k}{\partial w_{ij} \partial w_{lm}} = \frac{\partial}{\partial w_{ij}} \sum_{o \in \mathcal{O}} \frac{\partial v_k}{\partial x_{ko}} \frac{\partial x_{ko}}{\partial w_{lm}} \\
 &= \sum_{o \in \mathcal{O}} \frac{\partial^2 v_k}{\partial w_{ij} \partial x_{ko}} \frac{\partial x_{ko}}{\partial w_{lm}} + \sum_{o \in \mathcal{O}} \frac{\partial v_k}{\partial x_{ko}} \frac{\partial^2 x_{ko}}{\partial w_{ij} \partial w_{lm}} \\
 &= \sum_{o \in \mathcal{O}} \sum_{q \in \mathcal{O}} \frac{\partial^2 v_k}{\partial x_{ko} \partial x_{kq}} \frac{\partial x_{kq}}{\partial w_{ij}} \frac{\partial x_{ko}}{\partial w_{lm}} + \sum_{o \in \mathcal{O}} \frac{\partial v_k}{\partial x_{ko}} \frac{\partial^2 x_{ko}}{\partial w_{ij} \partial w_{lm}} \\
 &= \sum_{o \in \mathcal{O}} \sum_{q \in \mathcal{O}} \frac{\partial^2 v_k}{\partial x_{ko} \partial x_{kq}} \delta_{ki}^q \delta_{kl}^o x_{kj} x_{km} + \sum_{o \in \mathcal{O}} \frac{\partial v_k}{\partial x_{ko}} \bar{h}_{ij,lm},
 \end{aligned} \tag{2.37}$$

where

$$\bar{h}_{ij,lm} = \frac{\partial^2 x_{ko}}{\partial w_{ij} \partial w_{lm}}. \tag{2.38}$$

When using backpropagation rule again, we get

$$\begin{aligned}
 \bar{h}_{ij,lm} &= \frac{\partial}{\partial w_{ij}} (\delta_k^o x_{km}) = x_{km} \frac{\partial \delta_k^o}{\partial w_{ij}} + \delta_{kl}^o \frac{\partial x_{km}}{\partial w_{ij}} \\
 &= x_{km} \frac{\partial}{\partial w_{ij}} \frac{\partial x_{ko}}{\partial a_{kl}} + [i \prec m] \delta_{kl}^o \delta_{ki}^m x_{kj} \\
 &= x_{km} \frac{\partial^2 x_{ko}}{\partial a_{kl} \partial a_{ki}} \frac{\partial a_{ki}}{\partial w_{ij}} + [i \prec m] \delta_{kl}^o \delta_{ki}^m x_{kj} \\
 &= x_{km} x_{kj} \frac{\partial^2 x_{ko}}{\partial a_{kl} \partial a_{ki}} + [i \prec m] \delta_{kl}^o \delta_{ki}^m x_{kj} \\
 &= x_{km} x_{kj} \delta_{kli}^{o2} + [i \prec m] \delta_{kl}^o \delta_{ki}^m x_{kj}
 \end{aligned} \tag{2.39}$$

where

$$\delta_{kli}^{o2} = \frac{\partial^2 x_{ko}}{\partial a_{kl} \partial a_{ki}} \tag{2.40}$$

is referred to as the square delta error. Now we show that, just like the delta error, it can be optimally computed in feedforward nets. We start noticing that

$$\delta_{kli}^{o2} \neq 0 \text{ if and only if } i \leftrightarrow l \in \mathcal{A},$$

meaning that either $i \rightarrow l$ or $l \rightarrow i$ is an arch of the DAG \mathcal{G} , or said in another way i and l must be connected vertexes, with a directed path from one to the other. For example, neurons on the same layer do not satisfy this property and, consequently, $\delta_{kli}^{o2} = 0$. Like for the delta error, we distinguish the case of $i, l \in \mathcal{O}$ and $i, l \in \mathcal{H}$. If $i, l \in \mathcal{O}$, we get $\delta_{kli}^{o2} = 0$ if $i \neq l$. If $i = l = o$, we have

$$\delta_{koo}^{o2} = \frac{\partial}{\partial a_{ko}} \frac{\partial x_{ko}}{\partial a_{ko}} = \frac{\partial}{\partial a_{ko}} \sigma'(a_{ko}) = \sigma''(a_{ko}). \tag{2.41}$$

For the generic term δ_{kli}^{o2} we have

$$\begin{aligned}
 \delta_{kli}^{o2} &= \frac{\partial}{\partial a_{kl}} \delta_{ki}^o = \frac{\partial}{\partial a_{kl}} \left(\sigma'(a_{ki}) \sum_{j \in \text{ch}(i)} w_{ji} \delta_{kj}^o \right) \\
 &= \frac{\partial}{\partial a_{kl}} \frac{d \sigma(a_{ki})}{d a_{ki}} \sum_{j \in \text{ch}(i)} w_{ji} \delta_{kj}^o + \sigma'(a_{ki}) \sum_{j \in \text{ch}(i)} w_{ji} \frac{\partial \delta_{kj}^o}{\partial a_{kl}} \\
 &= \frac{d \sigma(a_{ki})}{d a_{ki}} \frac{\partial x_{ki}}{\partial a_{kl}} \sum_{j \in \text{ch}(i)} w_{ji} \delta_{kj}^o + \sigma'(a_{ki}) \sum_{j \in \text{ch}(i)} w_{ji} \delta_{kj}^{o2} \\
 &= \frac{d \delta_{kl}^i}{d a_{ki}} \sum_{j \in \text{ch}(i)} w_{ji} \delta_{kj}^o + \sigma'(a_{ki}) \sum_{j \in \text{ch}(i)} w_{ji} \delta_{kj}^{o2}.
 \end{aligned} \tag{2.42}$$

Now we are ready to define an algorithm for the computation of the Hessian. The generic term $h_{ij,lm}$ is computed by Eq. (2.37) that relies on the chain of variables $\bar{h}_{ij,lm}$ and δ_{kli}^{o2} according to the following expression tree in Fig. 2.8.⁶

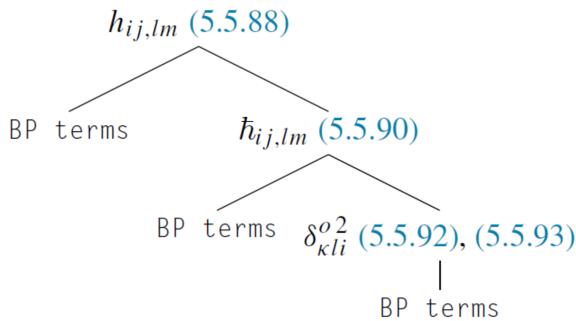


图 2.8: Expression tree used in computing $h_{ij,lm}$.

This expression tree indicates how to compute $h_{ij,lm}$ beginning from the leaves of the tree, where the vertexes also indicate the equations to be used. We can easily see that this computational scheme, referred to as Hessian BP, allows us to compute the Hessian with complexity $\Theta(m^2)$, where m is the number of weights. Exercise 84 proposes the detailed formulation of Hessian BP.

2.11.6 Symbolic and Automatic Differentiation

Learning algorithms rely on the computation mostly of the gradient and of the Hessian of appropriate objective functions. Differentiation can be done manually, but there are nice tools to perform symbolic differentiation. As discussed in the previous section, we can use numerical differentiation, but we can do better by properly exploiting the

⁶The value $h_{ij,lm}$ is computed by using the postorder visit of the tree.

structure of the function to be optimized. Backpropagation neither performs numerical nor symbolic differentiation. Unlike numerical analysis, it returns a precise expression for the gradient and, unlike symbolic differentiation, it computes the gradient on a given point with optimal complexity, but it doesn't return the symbolic expression. Since, nowadays learning schemes go well beyond supervised learning with feedforward nets, one may be interested in understanding the generality of the discussed backpropagation computational scheme. In order to shed light on the essence of this computational scheme, let us consider the following example. Suppose we want to compute the gradient of the function

$$y_0 = f(x_1, x_2) = (1 + x_2) \ln x_1 + \cos x_2. \quad (2.43)$$

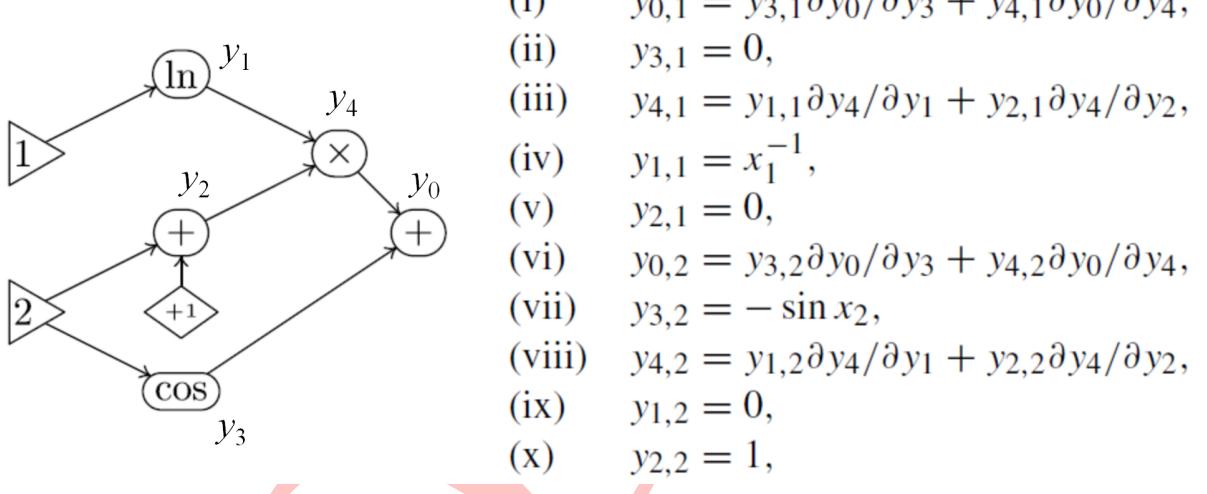


图 2.9: Expression DAG for (2.43).

We can provide the expression DAG in Fig. 2.9, which suggests a recursive structure for the computation of ∇f . The end of the recursion is reached for the vertexes of the graph whose parents are the variables x_1 and x_2 . We can easily see that the computation of ∇f can be carried out by a forward step. We distinguish two different differentiations: $y_{i,j} = \partial y_i / \partial x_j$, with $j = 1, 2$, and $\partial y_i / \partial y_k$. The objective of the computation is to determine $y_{i,j}$ for $i = 0$. We can determine $y_{1,j}$, $y_{2,j}$, and $y_{3,j}$ directly, whereas we need a forward propagation for computing $y_{4,j}$ and $y_{0,j}$. The difference with $\partial y_i / \partial y_k$ is that they can be symbolically determined immediately from the expression DAG. We have

$$\partial y_0 / \partial y_3 = 1, \quad \partial y_0 / \partial y_4 = 1, \quad \partial y_4 / \partial y_1 = y_2, \quad \partial y_4 / \partial y_2 = y_1.$$

Now, if we refer to the DAG expression and to the side equations, then the computation

of $\nabla f = (y_{0,1}, y_{0,2})^T$ takes place according to a data flow scheme which follows the sorting:

$$y_{0,1} \rightsquigarrow \{(iv), (v), (ii)\}, (iii), (i), \quad y_{0,2} \rightsquigarrow \{(vii), (ix), (x)\}, (viii), (vi),$$

where the numbers in curly braces indicate that there are no sorting constraints, so they can be associated with parallel computations. Exercise 85 proposes the computation of the network sensibility by automatic differentiation using the described forward step. Notice that the computation of the network sensibility is closely related to the computation of the gradient with respect to the weights connected to the inputs. In case of feedforward networks, because of the bilinear structure in which weights and inputs are involved, the equations are very similar (see Exercise 86). It is easy to realize that the described automatic differentiation has a general structure, which is dictated by the DAG expression.⁷

The computation of ∇f can be carried out also by using a generalization of backpropagation. In particular, we have

$$\begin{aligned} y_{0,1} &= \frac{\partial y_1}{\partial x_1} \frac{\partial y_0}{\partial y_1} + \frac{\partial y_2}{\partial x_1} \frac{\partial y_0}{\partial y_2} + \frac{\partial y_3}{\partial x_1} \frac{\partial y_0}{\partial y_3} = \frac{1}{x_1} \frac{\partial y_0}{\partial y_1}, \\ y_{0,2} &= \frac{\partial y_1}{\partial x_2} \frac{\partial y_0}{\partial y_1} + \frac{\partial y_2}{\partial x_2} \frac{\partial y_0}{\partial y_2} + \frac{\partial y_3}{\partial x_2} \frac{\partial y_0}{\partial y_3} = \frac{\partial y_0}{\partial y_2} - \frac{\partial y_0}{\partial y_3}. \end{aligned} \quad (2.44)$$

The auxiliary variables $\partial y_0 / \partial y_1$, $\partial y_0 / \partial y_2$, and $\partial y_0 / \partial y_3$ correspond with backpropagation delta error and can be determined by a backward step.⁸

Like for the forward step computation, this scheme is general and can be applied as soon as the DAG expression is given. It is not difficult to realize that the forward step technique is more efficient than the backward step for functions $f : \mathbb{R}^d \rightarrow \mathbb{R}^m$, where $m \gg d$, whereas the backward step technique is more efficient in case $d \gg m$ (see Exercise 87).

2.11.6.1 Exercises

Exercise 77. Consider a feedforward neural network whose connections are modeled by weights depending on the applied input. The generic weight of the connection can be modeled by a function $w_{i,j} : \mathbb{R} \rightarrow \mathbb{R}$ so that the activation of neuron i is dictated by

$$x_i = \sum_{j=1}^d x_j w_{i,j}(x_j) + b_i, \quad (2.45)$$

where $j \prec i$ and $w_{i,j}(x_j)$ is the corresponding weight of the connection. Here the output function of the neuron is linear, but all the connections are functionally dependent on

⁷Sometimes, it is called a computational graph.

⁸In the automatic differential literature, this is also referred to as reverse accumulation.

the input. This neuron is used to compose feedforward architectures according to graphic connections based on DAGs. Formulate supervised learning in the framework of regularization proposed in Section 4.4 of [56]. Prove that

$$w_{i,j}(x_j) = \sum_{k=1}^l \lambda_{i,j,k} g(x_j - x_{j,k}), \quad (2.46)$$

so that Eq. (2.45) reads

$$x_i = \sum_{j=1}^d \sum_{k=1}^l x_j g(x_j - x_{j,k}) \lambda_{i,j,k} + b_i. \quad (2.47)$$

This can be given a simple interpretation: Any input x_j , which is fed through connection (i, j) to neuron i , is filtered out by the training set so as to return the equivalent input $x_j^k := x_j g(x_j - x_{j,k})$. In so doing we end up with a classic ridge linear neuron model. Propose a corresponding learning algorithm with regularization based on the new weight $\lambda_{j,k}$.

Exercise 78. Given the neural network as defined in Exercise 77, prove that we can construct an equivalent network where the nonlinearity is moved to the vertexes, that is,

$$x_i = \gamma \left(\sum_{j=1}^d w_{i,j} x_j \right).$$

Prove that also the inverse construction is possible, and reformulate the learning algorithm of Exercise 77 for this new feedforward network.

Exercise 79. Using the Lagrange remainder in Taylor expansion, give a proof of Eq. 2.28. Then consider the asymmetric approximation and prove that it is significantly worse than the symmetric one.

Exercise 80. Given the neural network of Fig. 2.6, count the number of different topological sorts.

Exercise 81. Let us consider the feedforward neural network in Fig. 2.10, where $w = 4$, $w_u = 2$, $b = -2$, and the units are based on the sigmoidal function $\sigma(a) = \frac{1}{1+e^{-a}}$. This network is the time-unfolding of $x_{t+1} = \sigma(wx_t + b + u_t)$, where $u_0 = 1$ and $u_t = 0$, for $t > 0$. What happens when $t \rightarrow \infty$? Determine $\lim_{t \rightarrow \infty} x_t$ and $\lim_{t \rightarrow \infty} \nabla e(\mathbf{w}, b)$.

Exercise 82. Suppose you are given a multilayered network with one output such that all its weights (including the bias terms) are null. Compute the gradient for the cases of hyperbolic tangent, logistic sigmoid, and rectifier units.

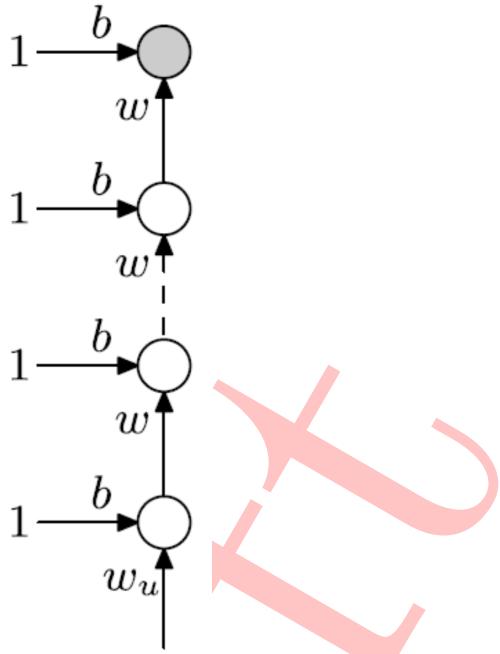


图 2.10: A network.

Exercise 83. Let us consider a feedforward network whose output neurons are computed by softmax. What are the backprop equations in this case?

Exercise 84. Based on the Hessian BP computational scheme defined by Fig. 2.8 construct a correspondent algorithm for the Hessian computation.

Exercise 85. Use the automatic differentiation by forward step to determine the gradient of the output with respect to the input (network sensibility).

Exercise 86. Discuss the differences between the network sensibility and the gradient equations with respect to the weights connected to the inputs.

Exercise 87. Prove that the forward step technique is more efficient than the backward step for functions $f : \mathbb{R}^d \rightarrow \mathbb{R}^m$, where $m \gg d$, whereas the backward step technique is more efficient in case $d \gg m$.

Exercise 88. Let us consider a single neuron with logistic sigmoid as a nonlinear transfer function, and the XOR training set

$$\mathcal{L} = \{(\mathbf{a}^T, 0)^T, (\mathbf{b}^T, 1)^T, (\mathbf{c}^T, 0)^T, (\mathbf{d}^T, 1)^T\},$$

where $\mathbf{a} = (0, 0)^T$, $\mathbf{b} = (1, 0)^T$, $\mathbf{c} = (1, 1)^T$, and $\mathbf{d} = (0, 1)^T$. Determine the stationary points and discuss their nature.

Exercise 89. Suppose we are given the following simple training set:

$$\mathcal{L} = \{((0, 0)^T, y_0), ((1, 0)^T, y_1)\}$$

and a single neuron with logistic sigmoid. Discuss the solution of the loading problem. What about generalization to new examples? What is the relationship between the choice of the nonasymptotic targets and the generalization? Determine which values of y_0 and y_1 lead to the same solution that one would determine by maximizing the maximum margin.

2.11.7 Linear algebra

2.11.7.1 Range and nullspace

Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ (*i.e.*, \mathbf{A} is a real matrix with m rows and n columns). The *range* of \mathbf{A} , denoted $\mathcal{R}(\mathbf{A})$, is the set of all vectors in \mathbb{R}^m that can be written as linear combinations of the columns of \mathbf{A} , *i.e.*,

$$\mathcal{R}(\mathbf{A}) = \{\mathbf{Ax} \mid \mathbf{x} \in \mathbb{R}^n\}.$$

The range $\mathcal{R}(\mathbf{A})$ is a subspace of \mathbb{R}^m , *i.e.*, it is itself a vector space. Its dimension is the *rank* of \mathbf{A} , denoted $\text{rank } \mathbf{A}$. The rank of \mathbf{A} can never be greater than the minimum of m and n . We say \mathbf{A} has *full rank* if $\text{rank } \mathbf{A} = \min\{m, n\}$.

The *nullspace* (or *kernel*) of \mathbf{A} , denoted $\mathcal{N}(\mathbf{A})$, is the set of all vectors \mathbf{x} mapped into zero by \mathbf{A} :

$$\mathcal{N}(\mathbf{A}) = \{\mathbf{x} \mid \mathbf{Ax} = \mathbf{0}\}.$$

The nullspace is a subspace of \mathbb{R}^n .

2.11.7.1.1 Orthogonal decomposition induced by \mathbf{A} If \mathcal{V} is a subspace of \mathbb{R}^n , its *orthogonal complement*, denoted \mathcal{V}^\perp , is defined as

$$\mathcal{V}^\perp = \{\mathbf{x} \mid \mathbf{z}^\top \mathbf{x} = 0 \text{ for all } \mathbf{z} \in \mathcal{V}\}.$$

(As one would expect of a complement, we have $\mathcal{V}^{\perp\perp} = \mathcal{V}$.)

A basic result of linear algebra is that, for any $\mathbf{A} \in \mathbb{R}^{m \times n}$, we have

$$\mathcal{N}(\mathbf{A}) = \mathcal{R}(\mathbf{A}^\top)^\perp$$

(Applying the result to \mathbf{A}^\top we also have $\mathcal{R}(\mathbf{A}) = \mathcal{N}(\mathbf{A}^\top)^\perp$.) This result is often stated as

$$\mathcal{N}(\mathbf{A}) \overset{\perp}{\oplus} \mathcal{R}(\mathbf{A}^\top) = \mathbb{R}^n. \quad (2.48)$$

Here the symbol $\overset{\perp}{\oplus}$ refers to *orthogonal direct sum*, *i.e.*, the sum of two subspaces that are orthogonal. The decomposition (2.48) of \mathbb{R}^n is called the *orthogonal decomposition induced by \mathbf{A}* .

2.11.7.2 Symmetric eigenvalue decomposition

Suppose $\mathbf{A} \in \mathbb{S}^n$, i.e., \mathbf{A} is a real symmetric $n \times n$ matrix. Then \mathbf{A} can be factored as

$$\mathbf{A} = \mathbf{Q}\Lambda\mathbf{Q}^\top, \quad (2.49)$$

where $\mathbf{Q} \in \mathbb{R}^{n \times n}$ is *orthogonal*, i.e., satisfies $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$, and $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$. The (real) numbers λ_i are the *eigenvalues* of \mathbf{A} , and are the roots of the *characteristic polynomial* $\det(s\mathbf{I} - \mathbf{A})$. The columns of \mathbf{Q} form an orthonormal set of *eigenvectors* of \mathbf{A} . The factorization (2.49) is called the *spectral decomposition* or (symmetric) *eigenvalue decomposition* of \mathbf{A} .

We order the eigenvalues as $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. We use the notation $\lambda_i(\mathbf{A})$ to refer to the i th largest eigenvalue of $\mathbf{A} \in \mathbb{S}$. We usually write the largest or maximum eigenvalue as $\lambda_1(\mathbf{A}) = \lambda_{\max}(\mathbf{A})$, and the least or minimum eigenvalue as $\lambda_n(\mathbf{A}) = \lambda_{\min}(\mathbf{A})$.

The determinant and trace can be expressed in terms of the eigenvalues,

$$\det \mathbf{A} = \prod_{i=1}^n \lambda_i, \quad \text{tr } \mathbf{A} = \sum_{i=1}^n \lambda_i,$$

as can the spectral and Frobenius norms,

$$\|\mathbf{A}\|_2 = \max_{i=1, \dots, n} |\lambda_i| = \max\{\lambda_1, -\lambda_n\}, \quad \|\mathbf{A}\|_F = \left(\sum_{i=1}^n \lambda_i^2 \right)^{1/2}.$$

2.11.7.2.1 Definiteness and matrix inequalities The largest and smallest eigenvalues satisfy

$$\lambda_{\max}(\mathbf{A}) = \sup_{\mathbf{x} \neq 0} \frac{\mathbf{x}^\top \mathbf{A} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}}, \quad \lambda_{\min}(\mathbf{A}) = \inf_{\mathbf{x} \neq 0} \frac{\mathbf{x}^\top \mathbf{A} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}}.$$

In particular, for any \mathbf{x} , we have

$$\lambda_{\min}(\mathbf{A}) \mathbf{x}^\top \mathbf{x} \leq \mathbf{x}^\top \mathbf{A} \mathbf{x} \leq \lambda_{\max}(\mathbf{A}) \mathbf{x}^\top \mathbf{x}$$

with both inequalities tight for (different) choices of \mathbf{x} .

A matrix $\mathbf{A} \in \mathbb{S}^n$ is called *positive definite* if for all $\mathbf{x} \neq \mathbf{0}$, $\mathbf{x}^\top \mathbf{A} \mathbf{x} > 0$. We denote this as $\mathbf{A} \succ \mathbf{0}$. By the inequality above, we see that $\mathbf{A} \succ \mathbf{0}$ if and only all its eigenvalues are positive, i.e., $\lambda_{\min}(\mathbf{A}) > 0$. If $-\mathbf{A}$ is positive definite, we say \mathbf{A} is *negative definite*, which we write as $\mathbf{A} \prec \mathbf{0}$. We use \mathbb{S}_{++}^n to denote the set of positive definite matrices in \mathbb{S}^n .

If \mathbf{A} satisfies $\mathbf{x}^\top \mathbf{A} \mathbf{x} \geq 0$ for all \mathbf{x} , we say that \mathbf{A} is *positive semidefinite* or *non-negative definite*. If $-\mathbf{A}$ is nonnegative definite, i.e., if $\mathbf{x}^\top \mathbf{A} \mathbf{x} \leq 0$ for all \mathbf{x} , we say

that \mathbf{A} is *negative semidefinite* or *nonpositive definite*. We use \mathbf{S}_+^n to denote the set of nonnegative definite matrices in \mathbf{S}^n .

For $\mathbf{A}, \mathbf{B} \in \mathbf{S}^n$, we use $\mathbf{A} \prec \mathbf{B}$ to mean $\mathbf{B} - \mathbf{A} \succ \mathbf{0}$, and so on. These inequalities are called *matrix inequalities*, or generalized inequalities associated with the positive semidefinite cone.

2.11.7.2.2 Symmetric squareroot Let $\mathbf{A} \in \mathbf{S}_+^n$, with eigenvalue decomposition $\mathbf{A} = \mathbf{Q} \operatorname{diag}(\lambda_1, \dots, \lambda_n) \mathbf{Q}^\top$. We define the (symmetric) squareroot of \mathbf{A} as

$$\mathbf{A}^{1/2} = \mathbf{Q} \operatorname{diag}(\lambda_1^{1/2}, \dots, \lambda_n^{1/2}) \mathbf{Q}^\top.$$

The squareroot $\mathbf{A}^{1/2}$ is the unique symmetric positive semidefinite solution of the equation $\mathbf{X}^2 = \mathbf{A}$.

2.11.7.3 Generalized eigenvalue decomposition

The *generalized eigenvalues* of a pair of symmetric matrices $(\mathbf{A}, \mathbf{B}) \in \mathbf{S}^n \times \mathbf{S}^n$ are defined as the roots of the polynomial $\det(\lambda \mathbf{B} - \mathbf{A})$.

We are usually interested in matrix pairs with $\mathbf{B} \in \mathbf{S}_{++}^n$. In this case the generalized eigenvalues are also the eigenvalues of $\mathbf{B}^{1/2} \mathbf{A} \mathbf{B}^{-1/2}$ (which are real). As with the standard eigenvalue decomposition, we order the generalized eigenvalues in nonincreasing order, as $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$, and denote the maximum generalized eigenvalue by $\lambda_{\max}(\mathbf{A}, \mathbf{B})$.

When $\mathbf{B} \in \mathbf{S}_{++}^n$, the pair of matrices can be factored as

$$\mathbf{A} = \mathbf{V} \Lambda \mathbf{V}^\top, \quad \mathbf{B} = \mathbf{V} \mathbf{V}^\top \tag{2.50}$$

where $\mathbf{V} \in \mathbb{R}^{n \times n}$ is nonsingular, and $\Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_n)$, where λ_i are the generalized eigenvalues of the pair (\mathbf{A}, \mathbf{B}) . The decomposition (2.50) is called the *generalized eigenvalue decomposition*.

The generalized eigenvalue decomposition is related to the standard eigenvalue decomposition of the matrix $\mathbf{B}^{-1/2} \mathbf{A} \mathbf{B}^{-1/2}$. If $\mathbf{Q} \Lambda \mathbf{Q}^\top$ is the eigenvalue decomposition of $\mathbf{B}^{-1/2} \mathbf{A} \mathbf{B}^{-1/2}$, then (2.50) holds with $\mathbf{V} = \mathbf{B}^{1/2} \mathbf{Q}$.

2.11.7.4 Singular value decomposition

Suppose $\mathbf{A} \in \mathbb{R}^{m \times n}$ with $\operatorname{rank} \mathbf{A} = r$. Then \mathbf{A} can be factored as

$$\mathbf{A} = \mathbf{U} \Sigma \mathbf{V}^\top, \tag{2.51}$$

where $\mathbf{U} \in \mathbb{R}^{m \times r}$ satisfies $\mathbf{U}^\top \mathbf{U} = \mathbf{I}$, $\mathbf{V} \in \mathbb{R}^{n \times r}$ satisfies $\mathbf{V}^\top \mathbf{V} = \mathbf{I}$, and $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r)$, with

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0.$$

The factorization (2.51) is called the *singular value decomposition* (SVD) of \mathbf{A} . The columns of \mathbf{U} are called *left singular vectors* of \mathbf{A} , the columns of \mathbf{V} are *right singular vectors*, and the numbers σ_i are the *singular values*. The singular value decomposition can be written

$$\mathbf{A} = \sum_{i=1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^\top,$$

where $\mathbf{u}_i \in \mathbb{R}^m$ are the left singular vectors, and $\mathbf{v}_i \in \mathbb{R}^n$ are the right singular vectors.

The singular value decomposition of a matrix \mathbf{A} is closely related to the eigenvalue decomposition of the (symmetric, nonnegative definite) matrix $\mathbf{A}^\top \mathbf{A}$. Using (2.51) we can write

$$\mathbf{A}^\top \mathbf{A} = \mathbf{V} \Sigma^2 \mathbf{V}^\top = [\mathbf{V} \quad \tilde{\mathbf{V}}] \begin{bmatrix} \Sigma^2 & 0 \\ 0 & 0 \end{bmatrix} [\mathbf{V} \quad \tilde{\mathbf{V}}]^\top,$$

where $\tilde{\mathbf{V}}$ is any matrix for which $[\mathbf{V} \quad \tilde{\mathbf{V}}]$ is orthogonal. The righthand expression is the eigenvalue decomposition of $\mathbf{A}^\top \mathbf{A}$, so we conclude that its nonzero eigenvalues are the singular values of \mathbf{A} squared, and the associated eigenvectors of $\mathbf{A}^\top \mathbf{A}$ are the right singular vectors of \mathbf{A} . A similar analysis of $\mathbf{A} \mathbf{A}^\top$ shows that its nonzero eigenvalues are also the squares of the singular values of \mathbf{A} , and the associated eigenvectors are the left singular vectors of \mathbf{A} .

The first or largest singular value is also written as $\sigma_{\max}(\mathbf{A})$. It can be expressed as

$$\sigma_{\max}(\mathbf{A}) = \sup_{\mathbf{x}, \mathbf{y} \neq 0} \frac{\mathbf{x}^\top \mathbf{A} \mathbf{y}}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2} = \sup_{\mathbf{y} \neq 0} \frac{\|\mathbf{A} \mathbf{y}\|_2}{\|\mathbf{y}\|_2}.$$

The righthand expression shows that the maximum singular value is the ℓ_2 operator norm of \mathbf{A} . The *minimum singular value* of $\mathbf{A} \in \mathbb{R}^{m \times n}$ is given by

$$\sigma_{\min}(\mathbf{A}) = \begin{cases} \sigma_r(\mathbf{A}), & r = \min\{m, n\} \\ 0, & r < \min\{m, n\}, \end{cases}$$

which is positive if and only if \mathbf{A} is full rank.

The singular values of a symmetric matrix are the absolute values of its nonzero eigenvalues, sorted into descending order. The singular values of a symmetric positive semidefinite matrix are the same as its nonzero eigenvalues.

The *condition number* of a nonsingular $\mathbf{A} \in \mathbb{R}^{n \times n}$, denoted $\text{cond}(\mathbf{A})$ or $\kappa(\mathbf{A})$, is defined as

$$\text{cond}(\mathbf{A}) = \|\mathbf{A}\|_2 \|\mathbf{A}^{-1}\|_2 = \sigma_{\max}(\mathbf{A}) / \sigma_{\min}(\mathbf{A}).$$

2.11.7.4.1 Pseudo-inverse Let $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^\top$ be the singular value decomposition of $\mathbf{A} \in \mathbb{R}^{m \times n}$, with rank $\mathbf{A} = r$. We define the *pseudo-inverse* or *Moore-Penrose inverse* of \mathbf{A} as

$$\mathbf{A}^\dagger = \mathbf{V}\Sigma^{-1}\mathbf{U}^\top \in \mathbb{R}^{n \times m}.$$

Alternative expressions are

$$\mathbf{A}^\dagger = \lim_{\epsilon \rightarrow 0} (\mathbf{A}^\top \mathbf{A} + \epsilon \mathbf{I})^{-1} \mathbf{A}^\top = \lim_{\epsilon \rightarrow 0} \mathbf{A}^\top (\mathbf{A}\mathbf{A}^\top + \epsilon \mathbf{I})^{-1},$$

where the limits are taken with $\epsilon > 0$, which ensures that the inverses in the expressions exist. If rank $\mathbf{A} = n$, then $\mathbf{A}^\dagger = (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top$. If rank $\mathbf{A} = m$, then $\mathbf{A}^\dagger = \mathbf{A}^\top (\mathbf{A}\mathbf{A}^\top)^{-1}$. If \mathbf{A} is square and nonsingular, then $\mathbf{A}^\dagger = \mathbf{A}^{-1}$.

The pseudo-inverse comes up in problems involving least-squares, minimum norm, quadratic minimization, and (Euclidean) projection. For example, $\mathbf{A}^\dagger \mathbf{b}$ is a solution of the least-squares problem

$$\min \|\mathbf{Ax} - \mathbf{b}\|_2^2$$

in general. When the solution is not unique, $\mathbf{A}^\dagger \mathbf{b}$ gives the solution with minimum (Euclidean) norm. As another example, the matrix $\mathbf{A}\mathbf{A}^\dagger = \mathbf{U}\mathbf{U}^\top$ gives (Euclidean) projection on $\mathcal{R}(\mathbf{A})$. The matrix $\mathbf{A}^\dagger \mathbf{A} = \mathbf{V}\mathbf{V}^\top$ gives (Euclidean) projection on $\mathcal{R}(\mathbf{A}^\top)$.

The optimal value p^* of the (general, nonconvex) quadratic optimization problem

$$\min (1/2)\mathbf{x}^\top \mathbf{Px} + \mathbf{q}^\top \mathbf{x} + r$$

where $\mathbf{P} \in \mathbf{S}^n$, can be expressed as

$$p^* = \begin{cases} -(1/2)\mathbf{q}^\top \mathbf{P}^\dagger \mathbf{q} + r, & \mathbf{P} \succeq \mathbf{0}, \mathbf{q} \in \mathcal{R}(\mathbf{P}) \\ -\infty, & \text{otherwise.} \end{cases}$$

(This generalizes the expression $p^* = -(1/2)\mathbf{q}^\top \mathbf{P}^{-1} \mathbf{q} + r$, valid for $\mathbf{P} \succ \mathbf{0}$.)

2.11.7.5 Schur complement

Consider a matrix $\mathbf{X} \in \mathbf{S}^n$ partitioned as

$$\mathbf{X} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{C} \end{bmatrix},$$

where $\mathbf{A} \in \mathbf{S}^k$. If $\det \mathbf{A} \neq 0$, the matrix

$$\mathbf{S} = \mathbf{C} - \mathbf{B}^\top \mathbf{A}^{-1} \mathbf{B}$$

is called the *Schur complement* of \mathbf{A} in \mathbf{X} . Schur complements arise in several contexts, and appear in many important formulas and theorems. For example, we have

$$\det \mathbf{X} = \det \mathbf{A} \det \mathbf{S}.$$

2.11.7.5.1 Inverse of block matrix The Schur complement comes up in solving linear equations, by eliminating one block of variables. We start with

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}$$

and assume that $\det \mathbf{A} \neq 0$. If we eliminate \mathbf{x} from the top block equation and substitute it into the bottom block equation, we obtain $\mathbf{v} = \mathbf{B}^\top \mathbf{A}^{-1} \mathbf{u} + \mathbf{S} \mathbf{y}$, so

$$\mathbf{y} = \mathbf{S}^{-1}(\mathbf{v} - \mathbf{B}^\top \mathbf{A}^{-1} \mathbf{u}).$$

Substituting this into the first equation yields

$$\mathbf{x} = (\mathbf{A}^{-1} + \mathbf{A}^{-1} \mathbf{B} \mathbf{S}^{-1} \mathbf{B}^\top \mathbf{A}^{-1}) \mathbf{u} - \mathbf{A}^{-1} \mathbf{B} \mathbf{S}^{-1} \mathbf{v}.$$

We can express these two equations as a formula for the inverse of a block matrix:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{C} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} + \mathbf{A}^{-1} \mathbf{B} \mathbf{S}^{-1} \mathbf{B}^\top \mathbf{A}^{-1} & -\mathbf{A}^{-1} \mathbf{B} \mathbf{S}^{-1} \\ -\mathbf{S}^{-1} \mathbf{B}^\top \mathbf{A}^{-1} & \mathbf{S}^{-1} \end{bmatrix}.$$

In particular, we see that the Schur complement is the inverse of the $(2, 2)$ block entry of the inverse of \mathbf{X} .

2.11.7.5.2 Minimization and definiteness The Schur complement arises when you minimize a quadratic form over some of the variables. Suppose $\mathbf{A} \succ \mathbf{0}$, and consider the minimization problem

$$\min \quad \mathbf{u}^\top \mathbf{A} \mathbf{u} + 2\mathbf{v}^\top \mathbf{B}^\top \mathbf{u} + \mathbf{v}^\top \mathbf{C} \mathbf{v} \tag{2.52}$$

with variable \mathbf{u} . The solution is $\mathbf{u} = -\mathbf{A}^{-1} \mathbf{B} \mathbf{v}$, and the optimal value is

$$\inf_{\mathbf{u}} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}^\top \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \mathbf{v}^\top \mathbf{S} \mathbf{v}. \tag{2.53}$$

From this we can derive the following characterizations of positive definiteness or semidefiniteness of the block matrix \mathbf{X} :

- $\mathbf{X} \succ \mathbf{0}$ if and only if $\mathbf{A} \succ \mathbf{0}$ and $\mathbf{S} \succ \mathbf{0}$.
- If $\mathbf{A} \succ \mathbf{0}$, then $\mathbf{X} \succeq \mathbf{0}$ if and only if $\mathbf{S} \succeq \mathbf{0}$.

2.11.7.5.3 Schur complement with singular \mathbf{A} Some Schur complement results have generalizations to the case when \mathbf{A} is singular, although the details are more complicated. As an example, if $\mathbf{A} \succeq \mathbf{0}$ and $\mathbf{B}\mathbf{v} \in \mathcal{R}(\mathbf{A})$, then the quadratic minimization problem (2.52) (with variable \mathbf{u}) is solvable, and has optimal value

$$\mathbf{v}^\top (\mathbf{C} - \mathbf{B}^\top \mathbf{A}^\dagger \mathbf{B}) \mathbf{v},$$

where \mathbf{A}^\dagger is the pseudo-inverse of \mathbf{A} . The problem is unbounded if $\mathbf{B}\mathbf{v} \notin \mathcal{R}(\mathbf{A})$ or if $\mathbf{A} \not\succeq \mathbf{0}$.

The range condition $\mathbf{B}\mathbf{v} \in \mathcal{R}(\mathbf{A})$ can also be expressed as $(\mathbf{I} - \mathbf{A}\mathbf{A}^\dagger)\mathbf{B}\mathbf{v} = \mathbf{0}$, so we have the following characterization of positive semidefiniteness of the block matrix \mathbf{X} :

$$\mathbf{X} \succeq \mathbf{0} \iff \mathbf{A} \succeq \mathbf{0}, (\mathbf{I} - \mathbf{A}\mathbf{A}^\dagger)\mathbf{B} = \mathbf{0}, \mathbf{C} - \mathbf{B}^\top \mathbf{A}^\dagger \mathbf{B} \succ \mathbf{0}.$$

Here the matrix $\mathbf{C} - \mathbf{B}^\top \mathbf{A}^\dagger \mathbf{B}$ serves as a generalization of the Schur complement, when \mathbf{A} is singular.

2.11.7.5.4 Adjoint Operator

Definition 90. Given a linear operator $\mathcal{A} : \mathbb{R}^m \rightarrow \mathbb{R}^n$, its adjoint operator is defined as the linear operator \mathcal{A}^* that satisfies:

$$\langle \mathcal{A}^*(\mathbf{x}), \mathbf{y} \rangle = \langle \mathbf{x}, \mathcal{A}(\mathbf{y}) \rangle, \quad \forall \mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^m. \quad (2.54)$$

Examples:

1. If $\mathcal{A}(\mathbf{x}) = \mathbf{Ax}$, then $\mathcal{A}^*(\mathbf{x}) = \mathbf{A}^T \mathbf{x}$.
2. If $\mathcal{A}(\mathbf{X}) = \mathbf{x}$ is the linear operator that extracts entries from \mathbf{X} , then $\mathcal{A}^*(\mathbf{x})$ is the linear operator that puts entries \mathbf{x} back into the corresponding positions in \mathbf{X} and fills the remaining entries with zeros. Especially, suppose that $\text{diag}(\mathbf{X})$ extracts the diagonal entries of \mathbf{X} , then $\text{diag}^*(\mathbf{x})$ is the diagonal matrix with \mathbf{x} on its diagonal.

2.11.7.5.5 von Neumann Trace Theorem

Theorem 91. Suppose $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m \times n}$. Then

$$|\langle \mathbf{A}, \mathbf{B} \rangle| \leq \sum_{i=1}^{\min(m,n)} \sigma_i(\mathbf{A}) \sigma_i(\mathbf{B}). \quad (2.55)$$

In particular, when both \mathbf{A} and \mathbf{B} are $n \times n$ p.s.d. matrices, the inequality becomes

$$\sum_{i=1}^n \sigma_i(\mathbf{A}) \sigma_{n-i+1}(\mathbf{B}) \leq \text{tr}(\mathbf{AB}) \leq \sum_{i=1}^n \sigma_i(\mathbf{A}) \sigma_i(\mathbf{B}). \quad (2.56)$$

For more practical information about matrix analysis, please refer to [95, 106].

2.11.7.6 Exercises

Exercise 92. Find the determinant and inverse (if it exists) of

$$\mathbf{B} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{a}^T & 1 \end{bmatrix}.$$

Exercise 93. Let \mathbf{A} be an $m \times n$ matrix of rank r . Let \mathbf{S} be a matrix such that $\mathbf{AS} = \mathbf{0}$. Show that $\text{rank}(\mathbf{S}) \leq n - r$.

Exercise 94. If \mathbf{A} is a positive semidefinite $n \times n$ matrix with $\text{rank}(\mathbf{A}) = r$, then there exists an $n \times r$ matrix \mathbf{G} such that $\mathbf{A} = \mathbf{GG}^T$ and $\mathbf{G}^T \mathbf{G} = \mathbf{\Lambda}$, where $\mathbf{\Lambda}$ is an $r \times r$ diagonal matrix containing the positive eigenvalues of \mathbf{A} .

Exercise 95. Let \mathbf{A} be positive definite and \mathbf{B} symmetric of the same order. Then there exist a non-singular matrix \mathbf{P} and a diagonal matrix $\mathbf{\Lambda}$ such that $\mathbf{A} = \mathbf{PP}^T$ and $\mathbf{B} = \mathbf{P}\mathbf{\Lambda}\mathbf{P}^T$.

Exercise 96. If \mathbf{A} is positive definite show that the matrix $\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{b}^T & \mathbf{b}^T \mathbf{A}^{-1} \mathbf{b} \end{bmatrix}$ is positive semidefinite and singular, and find the eigenvector associated with the zero eigenvalue.

Exercise 97. Prove that the eigenvalues λ_i of $(\mathbf{A} + \mathbf{B})^{-1} \mathbf{A}$, where \mathbf{A} is positive semidefinite and \mathbf{B} is positive definite, satisfy $0 \leq \lambda_i < 1$.

Exercise 98. Prove the Sherman-Morrison-Woodbury formula:

$$(\mathbf{A} + \mathbf{UCV}^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{U} (\mathbf{C}^{-1} + \mathbf{V}^T \mathbf{A}^{-1} \mathbf{U})^{-1} \mathbf{V}^T \mathbf{A}^{-1}, \quad (2.57)$$

where \mathbf{A} and \mathbf{C} are invertible and the sizes of \mathbf{U} , \mathbf{C} , and \mathbf{V} are compatible. In particular,

$$(\mathbf{A} + \mathbf{uv}^T)^{-1} = \mathbf{A}^{-1} - \frac{(\mathbf{A}^{-1} \mathbf{u})(\mathbf{v}^T \mathbf{A}^{-1})}{1 + \mathbf{v}^T \mathbf{A}^{-1} \mathbf{u}}. \quad (2.58)$$

It is useful for online update.

Exercise 99. Show that $(\mathbf{I} + \mathbf{AA}^T)^{-1} \mathbf{A} = \mathbf{A} (\mathbf{I} + \mathbf{A}^T \mathbf{A})^{-1}$ and $\mathbf{A} (\mathbf{A}^T \mathbf{A})^{1/2} = (\mathbf{AA}^T)^{1/2} \mathbf{A}$.

Exercise 100. Let $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a real-valued function defined by the equation $\phi(x, y) = xy^2(x^2 + y)$. Find its gradient and Hessian.

Exercise 101. Compute $\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}$, where $f(\mathbf{X}) = \|\mathbf{X}^T \mathbf{AX}\|_F^2$ and \mathbf{A} is a symmetric matrix.

Exercise 102. Compute $\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}$, where $f(\mathbf{X}) = \|\text{diag}(\mathbf{X}^T \mathbf{AX})\|_F^2$ and \mathbf{A} is a symmetric matrix.

Exercise 103. Let $\mathbf{U} = (\mathbf{u}_1, \dots, \mathbf{u}_n)$ and $\Phi : \mathbb{R} \rightarrow \mathbb{R}$ be a twice differentiable function.

$\Phi(\mathbf{x})$ means $(\Phi(x_1), \dots, \Phi(x_n))^T$. Compute $\frac{\partial}{\partial \mathbf{x}} \left(\sum_{i=1}^n \Phi(\mathbf{u}_i^T \mathbf{x}) \right)$ and $\frac{\partial^2}{\partial \mathbf{x} \partial \mathbf{x}^T} \left(\sum_{i=1}^n \Phi(\mathbf{u}_i^T \mathbf{x}) \right)$.

Exercise 104. Prove that

$$1. \frac{\partial \det \mathbf{X}}{\partial \mathbf{X}} = \det \mathbf{X} (\mathbf{X}^{-1})^T.$$

$$2. \frac{\partial \ln \det \mathbf{X}}{\partial \mathbf{X}} = (\mathbf{X}^T)^{-1}.$$

$$3. \text{ If } \mathbf{X} \text{ is square and invertible, then } \frac{\partial \det(\mathbf{X}^T \mathbf{A} \mathbf{X})}{\partial \mathbf{X}} = 2 \det(\mathbf{X}^T \mathbf{A} \mathbf{X}) (\mathbf{X}^T)^{-1}.$$

$$4. \frac{\partial \mathbf{a}^T \mathbf{X}^{-1} \mathbf{b}}{\partial \mathbf{X}} = -\mathbf{X}^{-T} \mathbf{a} \mathbf{b}^T \mathbf{X}^{-T}.$$

$$5. \frac{\partial \text{tr}(\mathbf{A} \mathbf{X}^{-1} \mathbf{B})}{\partial \mathbf{X}} = -(\mathbf{X}^{-1} \mathbf{B} \mathbf{A} \mathbf{X}^{-1})^T.$$

$$6. \frac{\partial \text{tr}((\mathbf{A} + \mathbf{X})^{-1})}{\partial \mathbf{X}} = -((\mathbf{A} + \mathbf{X})^{-2})^T.$$

Exercise 105. Suppose $\mathbf{X} \in \mathbb{R}^{3 \times 3}$, $\mathcal{A}(\mathbf{X}) = X_{11} + X_{12} - X_{31} + 2X_{33}$, find \mathcal{A}^* .

Exercise 106. Use von Neumann inequality to prove that the solution to

$$\min_{\mathbf{Y}} \text{tr}(\mathbf{Y} \mathbf{K} \mathbf{Y}^T), \quad \text{s.t.} \quad \mathbf{Y} \mathbf{Y}^T = \mathbf{I},$$

is the tailing eigenvectors of \mathbf{K} . The solution to

$$\max_{\mathbf{Y}} \text{tr}(\mathbf{Y} \mathbf{K} \mathbf{Y}^T), \quad \text{s.t.} \quad \mathbf{Y} \mathbf{Y}^T = \mathbf{I},$$

is the leading eigenvectors of \mathbf{K} . Try whether the above can be deduced by using Lagrange multiplier.

Draft

第三章 Convex Sets

(Taken from Chapter 2 of [18])

3.1 Affine and convex sets

3.1.1 Lines and line segments

Suppose $\mathbf{x}_1 \neq \mathbf{x}_2$ are two points in \mathbb{R}^n . Points of the form

$$\mathbf{y} = \theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2, \quad (3.1)$$

where $\theta \in \mathbb{R}$, form the line passing through \mathbf{x}_1 and \mathbf{x}_2 . The parameter value $\theta = 0$ corresponds to $\mathbf{y} = \mathbf{x}_2$, and the parameter value $\theta = 1$ corresponds to $\mathbf{y} = \mathbf{x}_1$. Values of the parameter θ between 0 and 1 correspond to the (closed) line segment between \mathbf{x}_1 and \mathbf{x}_2 .

Expressing \mathbf{y} in the form

$$\mathbf{y} = \mathbf{x}_2 + \theta(\mathbf{x}_1 - \mathbf{x}_2)$$

gives another interpretation: \mathbf{y} is the sum of the base point \mathbf{x}_2 (corresponding to $\theta = 0$) and the direction $\mathbf{x}_1 - \mathbf{x}_2$ (which points from \mathbf{x}_2 to \mathbf{x}_1) scaled by the parameter θ . Thus, θ gives the fraction of the way from \mathbf{x}_2 to \mathbf{x}_1 where \mathbf{y} lies. As θ increases from 0 to 1, the point \mathbf{y} moves from \mathbf{x}_2 to \mathbf{x}_1 ; for $\theta > 1$, the point \mathbf{y} lies on the line beyond \mathbf{x}_1 . This is illustrated in Figure 3.1.

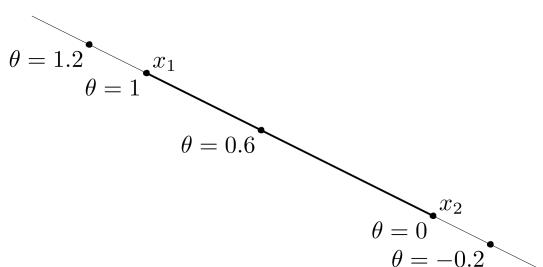


图 3.1: The line passing through \mathbf{x}_1 and \mathbf{x}_2 is described parametrically by $\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2$, where θ varies over \mathbb{R} . The line segment between \mathbf{x}_1 and \mathbf{x}_2 , which corresponds to θ between 0 and 1, is shown darker.

3.1.2 Affine sets

A set $C \subseteq \mathbb{R}^n$ is affine if the line through any two distinct points in C lies in C , i.e., if for any $\mathbf{x}_1, \mathbf{x}_2 \in C$ and $\theta \in \mathbb{R}$, we have $\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2 \in C$. In other words, C contains the linear combination of any two points in C , provided the coefficients in the linear combination sum to one.

This idea can be generalized to more than two points. We refer to a point of the form $\theta_1\mathbf{x}_1 + \dots + \theta_k\mathbf{x}_k$, where $\theta_1 + \dots + \theta_k = 1$, as an *affine combination* of the points $\mathbf{x}_1, \dots, \mathbf{x}_k$. All the affine combinations form the *affine subspace* C . It is the *affine hull* of the point set $\{\mathbf{x}_1, \dots, \mathbf{x}_k\}$. Using induction from the definition of affine subspace, it can be shown that an affine subspace contains every affine combination of its points: If C is an affine subspace, $\mathbf{x}_1, \dots, \mathbf{x}_k \in C$, and $\theta_1 + \dots + \theta_k = 1$, then the point $\theta_1\mathbf{x}_1 + \dots + \theta_k\mathbf{x}_k$ also belongs to C .

If C is an affine subspace and $\mathbf{x}_0 \in C$, then the set

$$V = C - \mathbf{x}_0 = \{\mathbf{x} - \mathbf{x}_0 \mid \mathbf{x} \in C\}$$

is a linear subspace (or simply called subspace), i.e., closed under sums and scalar multiplication. To see this, suppose $\mathbf{v}_1, \mathbf{v}_2 \in V$ and $\alpha, \beta \in \mathbb{R}$. Then we have $\mathbf{v}_1 + \mathbf{x}_0 \in C$ and $\mathbf{v}_2 + \mathbf{x}_0 \in C$, and so

$$\alpha\mathbf{v}_1 + \beta\mathbf{v}_2 + \mathbf{x}_0 = \alpha(\mathbf{v}_1 + \mathbf{x}_0) + \beta(\mathbf{v}_2 + \mathbf{x}_0) + (1 - \alpha - \beta)\mathbf{x}_0 \in C.$$

since C is affine, and $\alpha + \beta + (1 - \alpha - \beta) = 1$. We conclude that $\alpha\mathbf{v}_1 + \beta\mathbf{v}_2 \in V$, since $\alpha\mathbf{v}_1 + \beta\mathbf{v}_2 + \mathbf{x}_0 \in C$.

Thus, the affine subspace C can be expressed as

$$C = V + \mathbf{x}_0 = \{\mathbf{v} + \mathbf{x}_0 \mid \mathbf{v} \in V\},$$

i.e., as a subspace plus an offset. The subspace V associated with the affine set C does not depend on the choice of \mathbf{x}_0 , so \mathbf{x}_0 can be chosen as any point in C . We define the dimension of an affine set C as the dimension of the subspace $V = C - \mathbf{x}_0$, where \mathbf{x}_0 is any element of C .

Example 107. *Solution set of linear equations.* The solution set of a system of linear equations, $C = \{\mathbf{x} \mid \mathbf{Ax} = \mathbf{b}\}$, where $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$, is an affine set. To show this, suppose $\mathbf{x}_1, \mathbf{x}_2 \in C$, i.e., $\mathbf{Ax}_1 = \mathbf{b}, \mathbf{Ax}_2 = \mathbf{b}$. Then for any θ , we have

$$\mathbf{A}(\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2) = \theta\mathbf{Ax}_1 + (1 - \theta)\mathbf{Ax}_2 = \theta\mathbf{b} + (1 - \theta)\mathbf{b} = \mathbf{b},$$

which shows that the affine combination $\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2$ is also in C . The subspace associated with the affine set C is the nullspace of \mathbf{A} .

We also have a converse: every affine set can be expressed as the solution set of a system of linear equations.

The set of all affine combinations of points in some set $C \subseteq \mathbb{R}^n$ is called the affine hull of C , and denoted $\mathbf{aff} C$:

$$\mathbf{aff} C = \{\theta_1\mathbf{x}_1 + \dots + \theta_k\mathbf{x}_k \mid \mathbf{x}_1, \dots, \mathbf{x}_k \in C, \theta_1 + \dots + \theta_k = 1\}.$$

The affine hull is the smallest affine set that contains C , in the following sense: if S is any affine set with $C \subseteq S$, then $\mathbf{aff} C \subseteq S$.

3.1.3 Affine dimension and relative interior

We define the *affine dimension* of a set C as the dimension of its affine hull. Affine dimension is useful in the context of convex analysis and optimization, but is not always consistent with other definitions of dimension. As an example consider the unit circle in \mathbb{R}^2 , i.e., $\{\mathbf{x} \in \mathbb{R}^2 \mid \mathbf{x}_1^2 + \mathbf{x}_2^2 = 1\}$. Its affine hull is all of \mathbb{R}^2 , so its affine dimension is two. By most definitions of dimension, however, the unit circle in \mathbb{R}^2 has dimension one.

If the affine dimension of a set $C \subseteq \mathbb{R}^n$ is less than n , then the set lies in the affine set $\mathbf{aff} C \neq \mathbb{R}^n$. We define the *relative interior* of the set C , denoted $\mathbf{relint} C$, as its interior relative to $\mathbf{aff} C$:

$$\mathbf{relint} C = \{\mathbf{x} \in C \mid B(\mathbf{x}, r) \cap \mathbf{aff} C \subseteq C \text{ for some } r > 0\},$$

where $B(\mathbf{x}, r) = \{\mathbf{y} \mid \|\mathbf{y} - \mathbf{x}\| \leq r\}$, the ball of radius r and center \mathbf{x} in the norm $\|\cdot\|$. (Here $\|\cdot\|$ is any norm, all norms define the same relative interior.) We can then define the *relative boundary* of a set C as $\mathbf{cl} C \setminus \mathbf{relint} C$, where $\mathbf{cl} C$ is the closure of C .

Example 108. Consider a square in the $(\mathbf{x}_1, \mathbf{x}_2)$ -plane in \mathbb{R}^3 , defined as

$$C = \{\mathbf{x} \in \mathbb{R}^3 \mid -1 \leq \mathbf{x}_1 \leq 1, -1 \leq \mathbf{x}_2 \leq 1, \mathbf{x}_3 = 0\}.$$

Its affine hull is the $(\mathbf{x}_1, \mathbf{x}_2)$ -plane, i.e., $\mathbf{aff} C = \{\mathbf{x} \in \mathbb{R}^3 \mid \mathbf{x}_3 = 0\}$. The interior of C is empty, but the relative interior is

$$\mathbf{relint} C = \{\mathbf{x} \in \mathbb{R}^3 \mid -1 < \mathbf{x}_1 < 1, -1 < \mathbf{x}_2 < 1, \mathbf{x}_3 = 0\}.$$

Its boundary (in \mathbb{R}^3) is itself, its relative boundary is the wire-frame outline,

$$\{\mathbf{x} \in \mathbb{R}^3 \mid \max\{|\mathbf{x}_1|, |\mathbf{x}_2|\} = 1, \mathbf{x}_3 = 0\}.$$

3.1.4 Convex sets

A set C is *convex* if the line segment between any two points in C lies in C , i.e., if for any $\mathbf{x}_1, \mathbf{x}_2 \in C$ and any θ with $0 \leq \theta \leq 1$, we have

$$\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2 \in C.$$

Roughly speaking, a set is convex if every point in the set can be seen by every other point, along an unobstructed straight path between them, where unobstructed means lying in the set. Every affine set is also convex, since it contains the entire line between any two distinct points in it, and therefore also the line segment between the points. Figure 3.2 illustrates some simple convex and nonconvex sets in \mathbb{R}^2 .

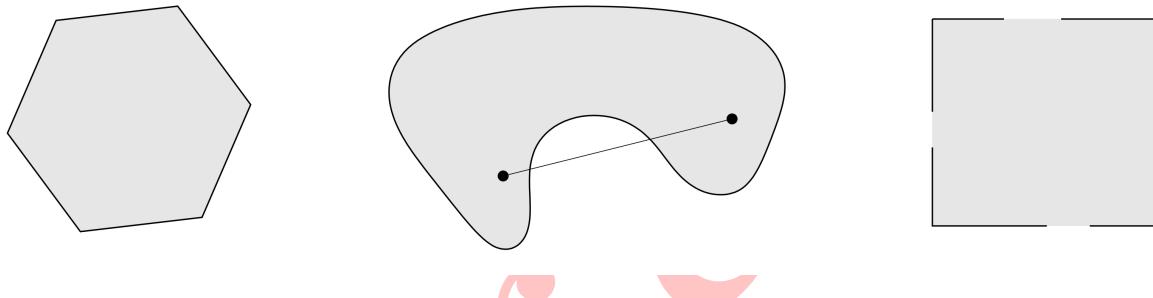


图 3.2: Some simple convex and nonconvex sets. *Left*. The hexagon, which includes its boundary (shown darker), is convex. *Middle*. The kidney shaped set is not convex, since the line segment between the two points in the set shown as dots is not contained in the set. *Right*. The square contains some boundary points but not others, and is not convex.

We call a point of the form $\theta_1\mathbf{x}_1 + \dots + \theta_k\mathbf{x}_k$, where $\theta_1 + \dots + \theta_k = 1$ and $\theta_i \geq 0, i = 1, \dots, k$, a *convex combination* of the points $\mathbf{x}_1, \dots, \mathbf{x}_k$. As with affine sets, it can be shown that a set is convex if and only if it contains every convex combination of its points. A convex combination of points can be thought of as a mixture or weighted average of the points, with θ_i the fraction of \mathbf{x}_i in the mixture.

The *convex hull* of a set C , denoted **conv** C , is the set of all convex combinations of points in C :

$$\mathbf{conv} C = \{\theta_1\mathbf{x}_1 + \dots + \theta_k\mathbf{x}_k \mid \mathbf{x}_i \in C, \theta_i \geq 0, i = 1, \dots, k, \theta_1 + \dots + \theta_k = 1\}.$$

As the name suggests, the convex hull **conv** C is always convex. It is the smallest convex set that contains C : If B is any convex set that contains C , then **conv** $C \subseteq B$. Figure 3.3 illustrates the definition of convex hull.

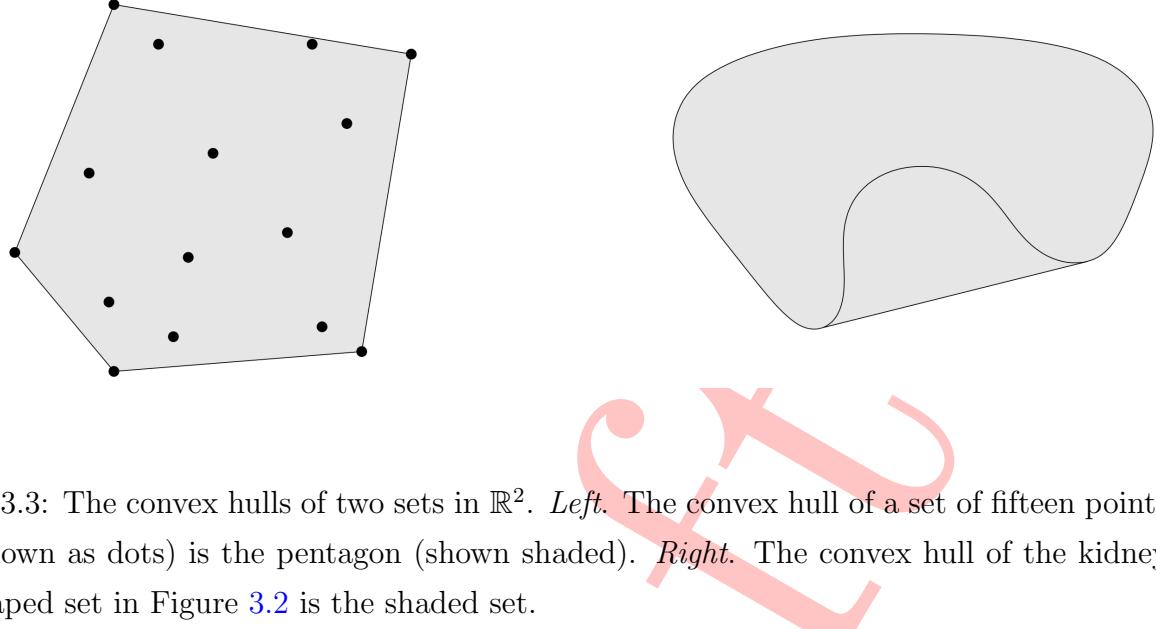


图 3.3: The convex hulls of two sets in \mathbb{R}^2 . *Left*. The convex hull of a set of fifteen points (shown as dots) is the pentagon (shown shaded). *Right*. The convex hull of the kidney shaped set in Figure 3.2 is the shaded set.

The idea of a convex combination can be generalized to include infinite sums, integrals, and, in the most general form, probability distributions. Suppose $\theta_1, \theta_2, \dots$ satisfy

$$\theta_i \geq 0, \quad i = 1, 2, \dots, \quad \sum_{i=1}^{\infty} \theta_i = 1,$$

and $\mathbf{x}_1, \mathbf{x}_2, \dots \in C$, where $C \subseteq \mathbb{R}^n$ is convex. Then

$$\sum_{i=1}^{\infty} \theta_i \mathbf{x}_i \in C,$$

if the series converges. More generally, suppose $p : \mathbb{R}^n \rightarrow \mathbb{R}$ satisfies $p(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in C$ and $\int_C p(\mathbf{x}) d\mathbf{x} = 1$, where $C \subseteq \mathbb{R}^n$ is convex. Then

$$\int_C p(\mathbf{x}) \mathbf{x} d\mathbf{x} \in C,$$

if the integral exists.

In the most general form, suppose $C \subseteq \mathbb{R}^n$ is convex and \mathbf{x} is a random vector with $\mathbf{x} \in C$ with probability one. Then $\mathbb{E}\mathbf{x} \in C$. Indeed, this form includes all the others as special cases. For example, suppose the random variable \mathbf{x} only takes on the two values \mathbf{x}_1 and \mathbf{x}_2 , with $\text{prob}(\mathbf{x} = \mathbf{x}_1) = \theta$ and $\text{prob}(\mathbf{x} = \mathbf{x}_2) = 1 - \theta$, where $0 \leq \theta \leq 1$. Then $\mathbb{E}\mathbf{x} = \theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2$, and we are back to a simple convex combination of two points.

3.1.5 Cones

A set C is called a *cone*, or nonnegative homogeneous, if for every $\mathbf{x} \in C$ and $\theta \geq 0$ we have $\theta\mathbf{x} \in C$. A set C is a convex cone if it is convex and a cone, which means that

for any $\mathbf{x}_1, \mathbf{x}_2 \in C$ and $\theta_1, \theta_2 \geq 0$, we have

$$\theta_1 \mathbf{x}_1 + \theta_2 \mathbf{x}_2 \in C.$$

Points of this form can be described geometrically as forming the two-dimensional pie slice with apex 0 and edges passing through \mathbf{x}_1 and \mathbf{x}_2 (See Figure 3.4).

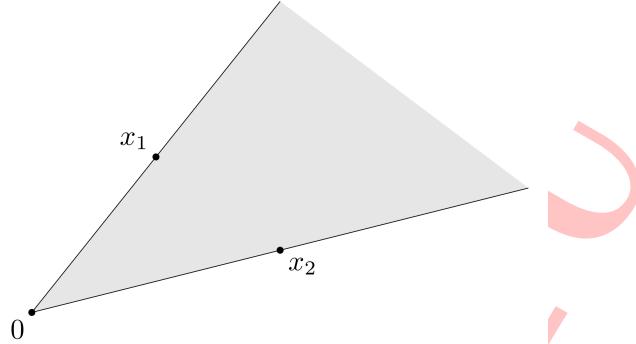


图 3.4: The pie slice shows all points of the form $\theta_1 \mathbf{x}_1 + \theta_2 \mathbf{x}_2$, where $\theta_1, \theta_2 \geq 0$. The apex of the slice (which corresponds to $\theta_1 = \theta_2 = 0$) is at 0; its edges (which correspond to $\theta_1 = 0$ or $\theta_2 = 0$) pass through the points \mathbf{x}_1 and \mathbf{x}_2 .

A point of the form $\theta_1 \mathbf{x}_1 + \dots + \theta_k \mathbf{x}_k$ with $\theta_1, \dots, \theta_k \geq 0$ is called a conic combination (or a nonnegative linear combination) of $\mathbf{x}_1, \dots, \mathbf{x}_k$. If \mathbf{x}_i are in a convex cone C , then every conic combination of \mathbf{x}_i is in C . Conversely, a set C is a convex cone if and only if it contains all conic combinations of its elements. Like convex (or affine) combinations, the idea of conic combination can be generalized to infinite sums and integrals.

The *conic hull* of a set C is the set of all conic combinations of points in C , i.e.,

$$\{\theta_1 \mathbf{x}_1 + \dots + \theta_k \mathbf{x}_k \mid \mathbf{x}_i \in C, \theta_i \geq 0, i = 1, \dots, k\},$$

which is also the smallest convex cone that contains C (see Figure 3.5).

3.2 Some important examples

In this section we describe some important examples of convex sets which we will encounter throughout the rest of the book. We start with some simple examples.

- The empty set \emptyset , any single point (i.e., singleton) $\{\mathbf{x}_0\}$, and the whole space \mathbb{R}^n are affine (hence, convex) subsets of \mathbb{R}^n .
- Any line is affine. If it passes through zero, it is a subspace, hence also a convex cone.

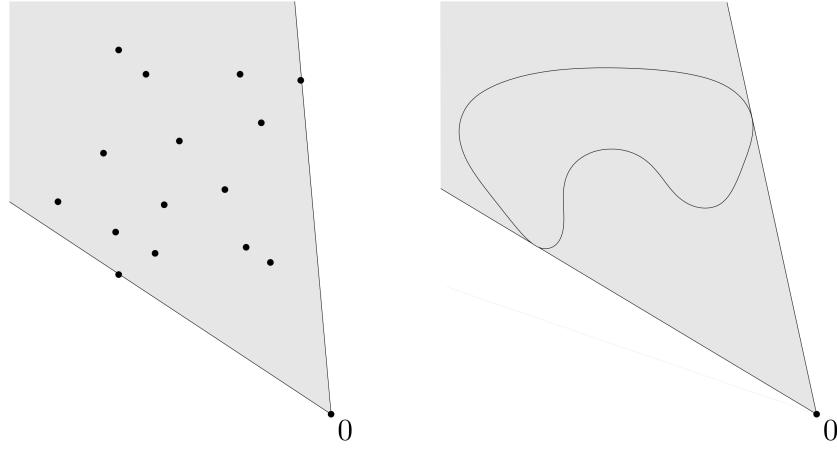


图 3.5: The conic hulls (shown shaded) of the two sets of Figure 3.3.

- A line segment is convex, but not affine (unless it reduces to a point).
- A ray, which has the form $\{\mathbf{x}_0 + \theta\mathbf{v} | \theta \geq 0\}$, where $\mathbf{v} \neq 0$, is convex, but not affine. It is a convex cone if its base \mathbf{x}_0 is 0.
- Any subspace is affine, and a convex cone (hence convex).

3.2.1 Hyperplanes and halfspaces

A hyperplane is a set of the form

$$\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = b\},$$

where $\mathbf{a} \in \mathbb{R}^n$, $\mathbf{a} \neq 0$, and $b \in \mathbb{R}$. Analytically it is the solution set of a nontrivial linear equation among the components of \mathbf{x} (and hence an affine set). Geometrically, the hyperplane $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = b\}$ can be interpreted as the set of points with a constant inner product to a given vector \mathbf{a} , or as a hyperplane with normal vector \mathbf{a} , the constant $b \in \mathbb{R}$ determines the offset of the hyperplane from the origin. This geometric interpretation can be understood by expressing the hyperplane in the form

$$\{\mathbf{x} | \mathbf{a}^T (\mathbf{x} - \mathbf{x}_0) = 0\},$$

where \mathbf{x}_0 is any point in the hyperplane (i.e., any point that satisfies $\mathbf{a}^T \mathbf{x}_0 = b$). This representation can in turn be expressed as

$$\{\mathbf{x} | \mathbf{a}^T (\mathbf{x} - \mathbf{x}_0) = 0\} = \mathbf{x}_0 + \mathbf{a}^\perp,$$

where \mathbf{a}^\perp denotes the orthogonal complement of \mathbf{a} , i.e., the set of all vectors orthogonal to it:

$$\mathbf{a}^\perp = \{\mathbf{v} | \mathbf{a}^T \mathbf{v} = 0\}.$$

This shows that the hyperplane consists of an offset \mathbf{x}_0 , plus all vectors orthogonal to the (normal) vector \mathbf{a} . These geometric interpretations are illustrated in Figure 3.6.

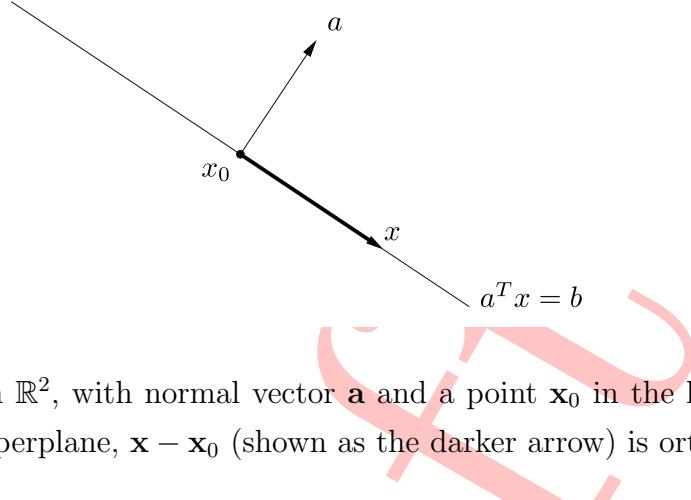


图 3.6: Hyperplane in \mathbb{R}^2 , with normal vector \mathbf{a} and a point \mathbf{x}_0 in the hyperplane. For any point \mathbf{x} in the hyperplane, $\mathbf{x} - \mathbf{x}_0$ (shown as the darker arrow) is orthogonal to \mathbf{a} .

A hyperplane divides \mathbb{R}^n into two halfspaces. A (closed) halfspace is a set of the form

$$\{\mathbf{x} \mid \mathbf{a}^T \mathbf{x} \leq b\}, \quad (3.2)$$

where $\mathbf{a} \neq 0$, i.e., the solution set of one (nontrivial) linear inequality. Halfspaces are convex, but not affine. This is illustrated in Figure 3.7.

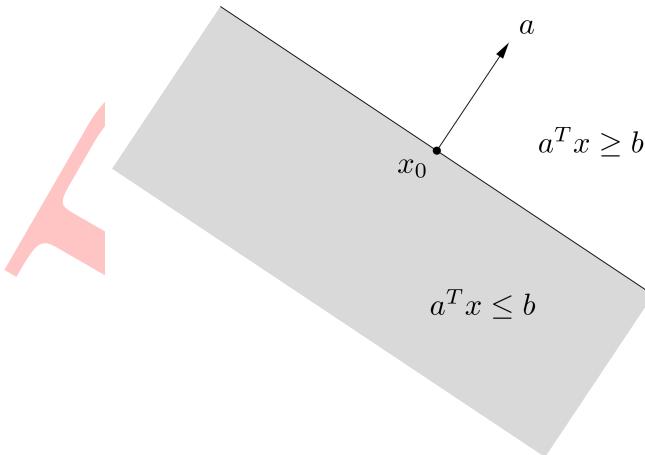


图 3.7: A hyperplane defined by $\mathbf{a}^T \mathbf{x} = b$ in \mathbb{R}^2 determines two halfspaces. The halfspace determined by $\mathbf{a}^T \mathbf{x} \geq b$ (not shaded) is the halfspace extending in the direction \mathbf{a} . The halfspace determined by $\mathbf{a}^T \mathbf{x} \leq b$ (which is shown shaded) extends in the direction \mathbf{a} . The vector \mathbf{a} is the outward normal of this halfspace.

The halfspace (3.2) can also be expressed as

$$\{\mathbf{x} \mid \mathbf{a}^T (\mathbf{x} - \mathbf{x}_0) \leq 0\}, \quad (3.3)$$

where \mathbf{x}_0 is any point on the associated hyperplane, i.e., satisfies $\mathbf{a}^T \mathbf{x}_0 = b$. The representation (3.3) suggests a simple geometric interpretation: the halfspace consists of \mathbf{x}_0 plus any vector that makes an obtuse (or right) angle with the (outward normal) vector \mathbf{a} . This is illustrated in Figure 3.8.

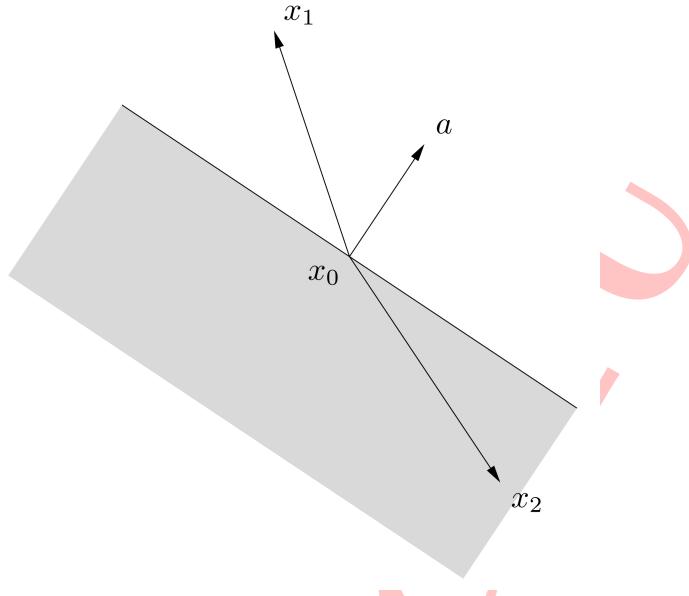


图 3.8: The shaded set is the halfspace determined by $\mathbf{a}^T(\mathbf{x} - \mathbf{x}_0) \leq 0$. The vector $\mathbf{x}_1 - \mathbf{x}_0$ makes an acute angle with \mathbf{a} , so \mathbf{x}_1 is not in the halfspace. The vector $\mathbf{x}_2 - \mathbf{x}_0$ makes an obtuse angle with \mathbf{a} , and so is in the halfspace.

The boundary of the halfspace (3.2) is the hyperplane $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = b\}$. The set $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} < b\}$, which is the interior of the halfspace $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} \leq b\}$, is called an open halfspace.

3.2.2 Euclidean balls and ellipsoids

A (Euclidean) ball (or just ball) in \mathbb{R}^n has the form

$$B(\mathbf{x}_c, r) = \{\mathbf{x} | \|\mathbf{x} - \mathbf{x}_c\|_2 \leq r\} = \{\mathbf{x} | (\mathbf{x} - \mathbf{x}_c)^T(\mathbf{x} - \mathbf{x}_c) \leq r^2\},$$

where $r > 0$ and $\|\cdot\|_2$ denotes the Euclidean norm, i.e., $\|\mathbf{u}\|_2 = (\mathbf{u}^T \mathbf{u})^{1/2}$. The vector \mathbf{x}_c is the center of the ball and the scalar r is its radius; $B(\mathbf{x}_c, r)$ consists of all points within a distance r of the center \mathbf{x}_c . Another common representation for the Euclidean ball is

$$B(\mathbf{x}_c, r) = \{\mathbf{x}_c + r\mathbf{u} | \|\mathbf{u}\|_2 \leq 1\}.$$

A Euclidean ball is a convex set: if $\|\mathbf{x}_1 - \mathbf{x}_c\|_2 \leq r, \|\mathbf{x}_2 - \mathbf{x}_c\|_2 \leq r$, and $0 \leq \theta \leq 1$, then

$$\begin{aligned} & \|\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2 - \mathbf{x}_c\|_2 \\ &= \|\theta(\mathbf{x}_1 - \mathbf{x}_c) + (1 - \theta)(\mathbf{x}_2 - \mathbf{x}_c)\|_2 \\ &\leq \theta\|\mathbf{x}_1 - \mathbf{x}_c\|_2 + (1 - \theta)\|\mathbf{x}_2 - \mathbf{x}_c\|_2 \\ &\leq r. \end{aligned}$$

(Here we use the homogeneity property and triangle inequality for $\|\cdot\|_2$; see Section 2.11.1.2.)

A related family of convex sets is the ellipsoids, which have the form

$$\varepsilon = \{\mathbf{x} \mid (\mathbf{x} - \mathbf{x}_c)^T \mathbf{P}^{-1}(\mathbf{x} - \mathbf{x}_c) \leq 1\}, \quad (3.4)$$

where $\mathbf{P} = \mathbf{P}^T \succ \mathbf{0}$, i.e., \mathbf{P} is symmetric and positive definite. The vector $\mathbf{x}_c \in \mathbb{R}^n$ is the center of the ellipsoid. The matrix \mathbf{P} determines how far the ellipsoid extends in every direction from \mathbf{x}_c , the lengths of the semi-axes of ε are given by $\sqrt{\lambda_i}$, where λ_i are the eigenvalues of \mathbf{P} . A ball is an ellipsoid with $\mathbf{P} = r^2 \mathbf{I}$. Figure 3.9 shows an ellipsoid in \mathbb{R}^2 .

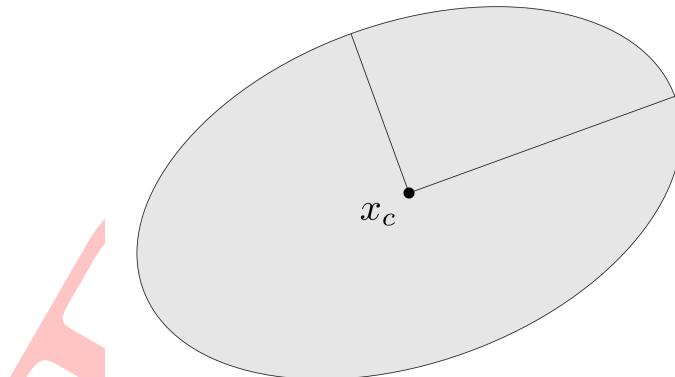


图 3.9: An ellipsoid in \mathbb{R}^2 , shown shaded. The center x_c is shown as a dot, and the two semi-axes are shown as line segments.

Another common representation of an ellipsoid is

$$\varepsilon = \{\mathbf{x}_c + \mathbf{A}\mathbf{u} \mid \|\mathbf{u}\|_2 \leq 1\}, \quad (3.5)$$

where \mathbf{A} is square and nonsingular. In this representation we can assume without loss of generality that \mathbf{A} is symmetric and positive definite. By taking $\mathbf{A} = \mathbf{P}^{1/2}$, this representation gives the ellipsoid defined in (3.4). When the matrix \mathbf{A} in (3.5) is symmetric positive semidefinite but singular, the set in (3.5) is called a degenerate ellipsoid, its affine dimension is equal to the rank of \mathbf{A} . Degenerate ellipsoids are also convex.

3.2.3 Norm balls and norm cones

Suppose $\|\cdot\|$ is any norm on \mathbb{R}^n (see Section 2.11.1.2). From the general properties of norms it can be shown that a norm ball of radius r and center \mathbf{x}_c , given by $\{\mathbf{x} \mid \|\mathbf{x} - \mathbf{x}_c\| \leq r\}$, is convex. The norm cone associated with the norm $\|\cdot\|$ is the set

$$C = \{(\mathbf{x}, t) \mid \|\mathbf{x}\| \leq t\} \subseteq \mathbb{R}^{n+1}.$$

It is (as the name suggests) a convex cone.

Example 109. The second-order cone is the norm cone for the Euclidean norm, i.e.,

$$\begin{aligned} C &= \{(\mathbf{x}, t) \in \mathbb{R}^{n+1} \mid \|\mathbf{x}\|_2 \leq t\} \\ &= \left\{ \begin{bmatrix} \mathbf{x} \\ t \end{bmatrix} \mid \begin{bmatrix} \mathbf{x} \\ t \end{bmatrix}^T \begin{bmatrix} \mathbf{I} & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ t \end{bmatrix} \leq 0, t \geq 0 \right\}. \end{aligned}$$

The second-order cone is also known by several other names. It is called the quadratic cone, since it is defined by a quadratic inequality. It is also called the Lorentz cone or ice-cream cone. Figure 3.10 shows the second-order cone in \mathbb{R}^3 .

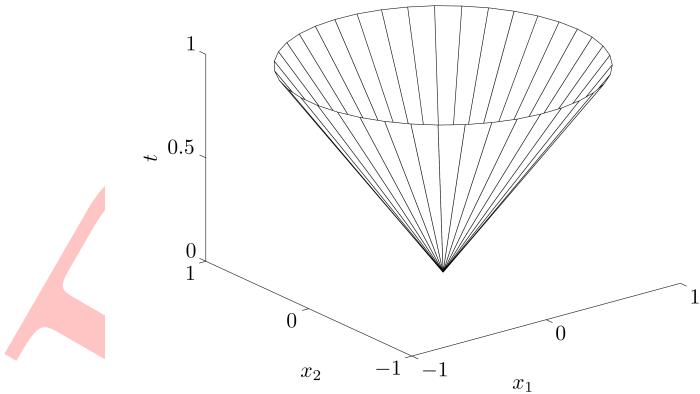


图 3.10: Boundary of second-order cone in \mathbb{R}^3 , $\{(x_1, x_2, t) \mid (x_1^2 + x_2^2)^{1/2} \leq t\}$.

3.2.4 Polyhedra

A polyhedron is defined as the solution set of a finite number of linear equalities and inequalities:

$$\mathcal{P} = \{\mathbf{x} \mid \mathbf{a}_j^T \mathbf{x} \leq b_j, j = 1, \dots, m, \mathbf{c}_j^T \mathbf{x} = d_j, j = 1, \dots, p\}. \quad (3.6)$$

A polyhedron is thus the intersection of a finite number of halfspaces and hyperplanes. Affine sets (e.g., subspaces, hyperplanes, lines), rays, line segments, and halfspaces are all

polyhedra. It is easily shown that polyhedra are convex sets. A bounded polyhedron is sometimes called a polytope, but some authors use the opposite convention (i.e., polytope for any set of the form (3.6), and polyhedron when it is bounded). Figure 3.11 shows an example of a polyhedron defined as the intersection of five halfspaces.

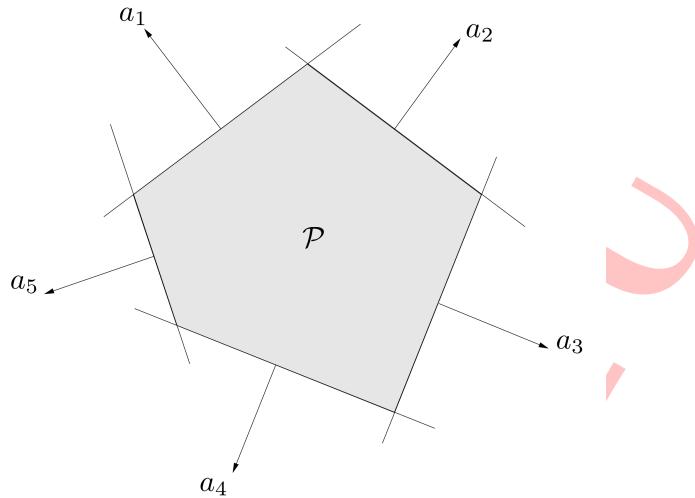


图 3.11: The polyhedron P (shown shaded) is the intersection of five halfspaces, with outward normal vectors $\mathbf{a}_1, \dots, \mathbf{a}_5$.

It will be convenient to use the compact notation

$$P = \{\mathbf{x} \mid \mathbf{Ax} \preceq \mathbf{b}, \mathbf{Cx} = \mathbf{d}\} \quad (3.7)$$

for (3.6), where

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1^T \\ \vdots \\ \mathbf{a}_m^T \end{bmatrix}, \mathbf{C} = \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_p^T \end{bmatrix},$$

and the symbol \preceq denotes vector inequality or componentwise inequality in \mathbb{R}^m : $\mathbf{u} \preceq \mathbf{v}$ means $\mathbf{u}_i \leq \mathbf{v}_i$ for $i = 1, \dots, m$.

Example 110. The nonnegative orthant is the set of points with nonnegative components, i.e.,

$$\mathbb{R}_+^n = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{x}_i \geq 0, i = 1, \dots, n\} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{x} \succeq \mathbf{0}\}.$$

(Here \mathbb{R}_+ denotes the set of nonnegative numbers: $\mathbb{R}_+ = \{x \in \mathbb{R} \mid x \geq 0\}$.) The nonnegative orthant is a polyhedron and a cone (and therefore called a polyhedral cone).

Simplexes

Simplexes are another important family of polyhedra. Suppose the $k + 1$ points $\mathbf{v}_0, \dots, \mathbf{v}_k \in \mathbb{R}^n$ are affinely independent, which means $\mathbf{v}_1 - \mathbf{v}_0, \dots, \mathbf{v}_k - \mathbf{v}_0$ are linearly independent. The simplex determined by them is given by

$$C = \text{conv}\{\mathbf{v}_0, \dots, \mathbf{v}_k\} = \{\theta_0\mathbf{v}_0 + \dots + \theta_k\mathbf{v}_k | \boldsymbol{\theta} \succeq \mathbf{0}, \mathbf{1}^T \boldsymbol{\theta} = 1\}, \quad (3.8)$$

where $\mathbf{1}$ denotes the vector with all entries one. The affine dimension of this simplex is k , so it is sometimes referred to as a k -dimensional simplex in \mathbb{R}^n .

Example 111 (Some common simplexes). A 1-dimensional simplex is a line segment, a 2-dimensional simplex is a triangle (including its interior), and a 3-dimensional simplex is a tetrahedron.

The unit simplex is the n -dimensional simplex determined by the zero vector and the unit vectors, i.e., $\mathbf{0}, \mathbf{e}_1, \dots, \mathbf{e}_n \in \mathbb{R}^n$. It can be expressed as the set of vectors that satisfy

$$\mathbf{x} \succeq \mathbf{0}, \mathbf{1}^T \mathbf{x} \leq 1.$$

The probability simplex is the $(n - 1)$ -dimensional simplex determined by the unit vectors $\mathbf{e}_1, \dots, \mathbf{e}_n \in \mathbb{R}^n$. It is the set of vectors that satisfy

$$\mathbf{x} \succeq \mathbf{0}, \mathbf{1}^T \mathbf{x} = 1.$$

Vectors in the probability simplex correspond to probability distributions on a set with n elements, with \mathbf{x}_i interpreted as the probability of the i th element.

To describe the simplex (3.8) as a polyhedron, i.e., in the form (3.7), we proceed as follows. By definition, $\mathbf{x} \in C$ if and only if $\mathbf{x} = \theta_0\mathbf{v}_0 + \theta_1\mathbf{v}_1 + \dots + \theta_k\mathbf{v}_k$ for some $\boldsymbol{\theta} \succeq \mathbf{0}$ with $\mathbf{1}^T \boldsymbol{\theta} = 1$. Equivalently, if we define $\mathbf{y} = (\theta_1, \dots, \theta_k)^T$ and

$$\mathbf{B} = [\mathbf{v}_1 - \mathbf{v}_0 \ \cdots \ \mathbf{v}_k - \mathbf{v}_0] \in \mathbb{R}^{n \times k}, \quad (3.9)$$

we can say that $\mathbf{x} \in C$ if and only if

$$\mathbf{x} = \mathbf{v}_0 + \mathbf{B}\mathbf{y}$$

for some $\mathbf{y} \succeq \mathbf{0}$ with $\mathbf{1}^T \mathbf{y} \leq 1$. Now we note that affine independence of the points $\mathbf{v}_0, \dots, \mathbf{v}_k$ implies that the matrix \mathbf{B} has rank k . Therefore there exists a nonsingular matrix $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2) \in \mathbb{R}^{n \times n}$ such that

$$\mathbf{AB} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \end{bmatrix} \mathbf{B} = \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix}.$$

Multiplying (3.9) on the left with \mathbf{A} , we obtain

$$\mathbf{A}_1 \mathbf{x} = \mathbf{A}_1 \mathbf{v}_0 + \mathbf{y}, \quad \mathbf{A}_2 \mathbf{x} = \mathbf{A}_2 \mathbf{v}_0.$$

From this we see that $\mathbf{x} \in C$ if and only if $\mathbf{A}_2 \mathbf{x} = \mathbf{A}_2 \mathbf{v}_0$, and the vector $\mathbf{y} = \mathbf{A}_1 \mathbf{x} - \mathbf{A}_1 \mathbf{v}_0$ satisfies $y \succeq \mathbf{0}$ and $\mathbf{1}^T \mathbf{y} \leq 1$. In other words we have $\mathbf{x} \in C$ if and only if

$$\mathbf{A}_2 \mathbf{x} = \mathbf{A}_2 \mathbf{v}_0, \quad \mathbf{A}_1 \mathbf{x} \succeq \mathbf{A}_1 \mathbf{v}_0, \quad \mathbf{1}^T \mathbf{A}_1 \mathbf{x} \leq 1 + \mathbf{1}^T \mathbf{A}_1 \mathbf{v}_0,$$

which is a set of linear equalities and inequalities in \mathbf{x} , and so describes a polyhedron. Convex hull description of polyhedra The convex hull of the finite set $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ is

$$\mathbf{conv}\{\mathbf{v}_1, \dots, \mathbf{v}_k\} = \{\theta_1 \mathbf{v}_1 + \dots + \theta_k \mathbf{v}_k \mid \boldsymbol{\theta} \succeq \mathbf{0}, \mathbf{1}^T \boldsymbol{\theta} = 1\}.$$

This set is a polyhedron, and bounded, but (except in special cases, e.g., a simplex) it is not simple to express it in the form (3.6), i.e., by a set of linear equalities and inequalities.

A generalization of this convex hull description is

$$\{\theta_1 \mathbf{v}_1 + \dots + \theta_k \mathbf{v}_k \mid \theta_1 + \dots + \theta_m = 1, \theta_i \geq 0, i = 1, \dots, k\}, \quad (3.10)$$

where $m \leq k$. Here we consider nonnegative linear combinations of \mathbf{v}_i , but only the first m coefficients are required to sum to one. Alternatively, we can interpret (3.10) as the convex hull of the points $\mathbf{v}_1, \dots, \mathbf{v}_m$, plus the conic hull of the points $\mathbf{v}_{m+1}, \dots, \mathbf{v}_k$. The set (3.10) defines a polyhedron, and conversely, every polyhedron can be represented in this form (although we will not show this).

The question of how a polyhedron is represented is subtle, and has very important practical consequences. As a simple example consider the unit ball in the l_∞ -norm in \mathbb{R}^n ,

$$C = \{\mathbf{x} \mid |\mathbf{x}_i| \leq 1, i = 1, \dots, n\}.$$

The set C can be described in the form (3.6) with $2n$ linear inequalities $\pm \mathbf{e}_i^T \mathbf{x} \leq 1$, where \mathbf{e}_i is the i th unit vector. To describe it in the convex hull form (3.10) requires at least 2^n points:

$$C = \mathbf{conv}\{\mathbf{v}_1, \dots, \mathbf{v}_{2^n}\},$$

where $\mathbf{v}_1, \dots, \mathbf{v}_{2^n}$ are the 2^n vectors all of whose components are 1 or -1 . Thus the size of the two descriptions differs greatly, for large n .

3.2.5 The positive semidefinite cone

We use the notation \mathbb{S}^n to denote the set of symmetric $n \times n$ matrices,

$$\mathbb{S}^n = \{\mathbf{X} \in \mathbb{R}^{n \times n} \mid \mathbf{X} = \mathbf{X}^T\},$$

which is a vector space with dimension $n(n + 1) = 2$. We use the notation \mathbb{S}_+^n to denote the set of symmetric positive semidefinite matrices:

$$\mathbb{S}_+^n = \{\mathbf{X} \in S^n | \mathbf{X} \succeq \mathbf{0}\},$$

and the notation \mathbb{S}_{++}^n to denote the set of symmetric positive definite matrices:

$$\mathbb{S}_{++}^n = \{\mathbf{X} \in S^n | \mathbf{X} \succ \mathbf{0}\}.$$

(This notation is meant to be analogous to \mathbb{R}_+ , which denotes the nonnegative reals, and \mathbb{R}_{++} , which denotes the positive reals.)

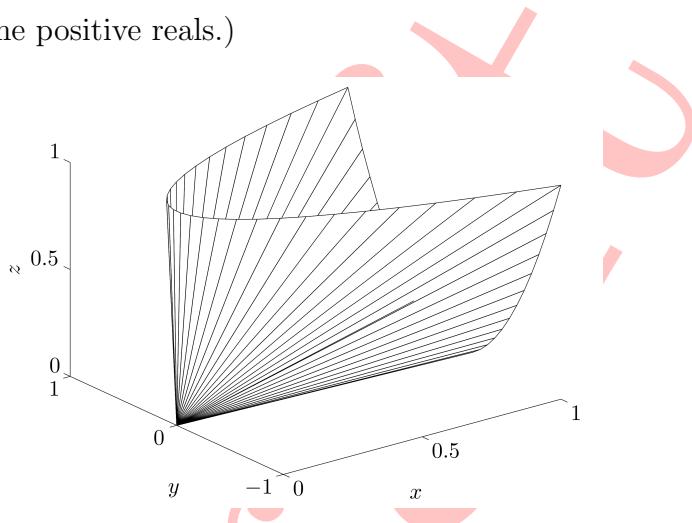


图 3.12: Boundary of positive semidefinite cone in \mathbb{S}^2 .

The set \mathbb{S}_+^n is a convex cone: if $\theta_1, \theta_2 \geq 0$ and $\mathbf{A}, \mathbf{B} \in \mathbb{S}_+^n$, then $\theta_1\mathbf{A} + \theta_2\mathbf{B} \in \mathbb{S}_+^n$. This can be seen directly from the definition of positive semidefiniteness: for any $\mathbf{x} \in \mathbb{R}^n$, we have

$$\mathbf{x}^T(\theta_1\mathbf{A} + \theta_2\mathbf{B})\mathbf{x} = \theta_1\mathbf{x}^T\mathbf{A}\mathbf{x} + \theta_2\mathbf{x}^T\mathbf{B}\mathbf{x} \geq 0,$$

if $\mathbf{A} \succeq \mathbf{0}$, $\mathbf{B} \succeq \mathbf{0}$ and $\theta_1, \theta_2 \geq 0$.

Example 112 (Positive semidefinite cone in \mathbb{S}^2). *We have*

$$\mathbf{X} = \begin{bmatrix} x & y \\ y & z \end{bmatrix} \in \mathbb{S}_+^2 \iff x \geq 0, z \geq 0, xz \geq y^2.$$

The boundary of this cone is shown in Figure 3.12, plotted in \mathbb{R}^3 as (x, y, z) .

3.3 Operations that preserve convexity

In this section we describe some operations that preserve convexity of sets, or allow us to construct convex sets from others. These operations, together with the simple examples described in, form a calculus of convex sets that is useful for determining or establishing convexity of sets.

3.3.1 Intersection

Convexity is preserved under intersection: if S_1 and S_2 are convex, then $S_1 \cap S_2$ is convex. This property extends to the intersection of an infinite number of sets: if S_α is convex for every $\alpha \in \mathcal{A}$, then $\cap_{\alpha \in \mathcal{A}} S_\alpha$ is convex. (Subspaces, affine sets, and convex cones are also closed under arbitrary intersections.) As a simple example, a polyhedron is the intersection of halfspaces and hyperplanes (which are convex), and therefore is convex.

Example 113. The positive semidefinite cone \mathbb{S}_+^n can be expressed as

$$\bigcap_{\mathbf{z} \neq 0} \{\mathbf{X} \in \mathbb{S}^n \mid \mathbf{z}^T \mathbf{X} \mathbf{z} \geq 0\}.$$

For each $\mathbf{z} \neq 0$, $\mathbf{z}^T \mathbf{X} \mathbf{z}$ is a (not identically zero) linear function of \mathbf{X} , so the sets

$$\{\mathbf{X} \in \mathbb{S}^n \mid \mathbf{z}^T \mathbf{X} \mathbf{z} \geq 0\}$$

are, in fact, halfspaces in \mathbb{S}^n . Thus the positive semidefinite cone is the intersection of an infinite number of halfspaces, and so is convex.

Example 114. We consider the set

$$S = \{\mathbf{x} \in \mathbb{R}^m \mid |p(t)| \leq 1 \text{ for } |t| \leq \pi/3\}, \quad (3.11)$$

where $p(t) = \sum_{k=1}^m \mathbf{x}_k \cos kt$. The set S can be expressed as the intersection of an infinite number of slabs: $S = \cap_{|t| \leq \pi/3} S_t$, where

$$S_t = \{\mathbf{x} \mid -1 \leq (\cos t, \dots, \cos mt)^T \mathbf{x} \leq 1\},$$

and so is convex. The definition and the set are illustrated in Figures 3.13 and 3.14, for $m = 2$.

In the examples above we establish convexity of a set by expressing it as a (possibly infinite) intersection of halfspaces. We will see in Section 3.5.1 that a converse holds: every closed convex set S is a (usually infinite) intersection of halfspaces. In fact, a closed convex set S is the intersection of all halfspaces that contain it:

$$S = \bigcap \{\mathcal{H} \mid \mathcal{H} \text{ halfspace, } S \subseteq \mathcal{H}\}.$$

3.3.2 Affine functions

Recall that a function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is affine if it is a sum of a linear function and a constant, i.e., if it has the form $f(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$. Suppose $S \subseteq \mathbb{R}^n$ is convex and $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is an affine function. Then the image of S under f ,

$$f(S) = \{f(\mathbf{x}) \mid \mathbf{x} \in S\},$$

is convex. Similarly, if $f : \mathbb{R}^k \rightarrow \mathbb{R}^n$ is an affine function, the inverse image of S under f ,

$$f^{-1}(S) = \{\mathbf{x} | f(\mathbf{x}) \in S\},$$

is convex.

Two simple examples are scaling and translation. If $S \subseteq \mathbb{R}^n$ is convex, $\alpha \in \mathbb{R}$, and $\mathbf{a} \in \mathbb{R}^n$, then the sets αS and $S + \mathbf{a}$ are convex, where

$$\alpha S = \{\alpha \mathbf{x} | \mathbf{x} \in S\}, \quad S + \mathbf{a} = \{\mathbf{x} + \mathbf{a} | \mathbf{x} \in S\}.$$

The projection of a convex set onto some of its coordinates is convex: if $S \subseteq \mathbb{R}^m \times \mathbb{R}^n$ is convex, then

$$T = \{\mathbf{x}_1 \in \mathbb{R}^m | (\mathbf{x}_1, \mathbf{x}_2) \in S \text{ for some } \mathbf{x}_2 \in \mathbb{R}^n\}$$

is convex.

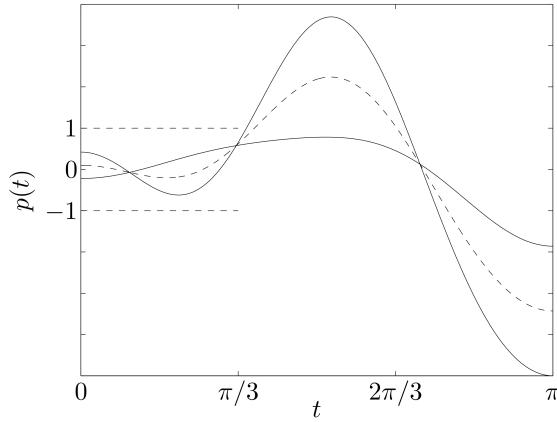


图 3.13: Three trigonometric polynomials associated with points in the set S defined in (3.11), for $m = 2$. The trigonometric polynomial plotted with dashed line type is the average of the other two.

The sum of two sets is defined as

$$S_1 + S_2 = \{\mathbf{x} + \mathbf{y} | \mathbf{x} \in S_1, \mathbf{y} \in S_2\}.$$

If S_1 and S_2 are convex, then $S_1 + S_2$ is convex. To see this, if S_1 and S_2 are convex, then so is the direct or Cartesian product

$$S_1 \times S_2 = \{(\mathbf{x}_1, \mathbf{x}_2) | \mathbf{x}_1 \in S_1, \mathbf{x}_2 \in S_2\}.$$

The image of this set under the linear function $f(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1 + \mathbf{x}_2$ is the sum $S_1 + S_2$. We can also consider the partial sum of $S_1, S_2 \in \mathbb{R}^n \times \mathbb{R}^m$, defined as

$$S = \{(\mathbf{x}, \mathbf{y}_1 + \mathbf{y}_2) | (\mathbf{x}, \mathbf{y}_1) \in S_1, (\mathbf{x}, \mathbf{y}_2) \in S_2\},$$

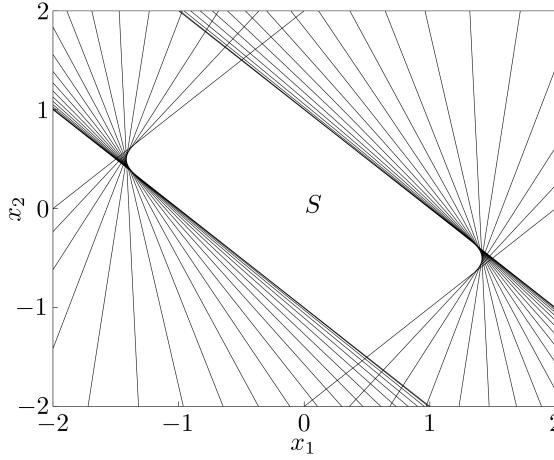


图 3.14: The set S defined in (3.11), for $m = 2$, is shown as the white area in the middle of the plot. The set is the intersection of an infinite number of slabs (20 of which are shown), hence convex.

where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y}_i \in \mathbb{R}^m$. For $m = 0$, the partial sum gives the intersection of S_1 and S_2 ; for $n = 0$, it is set addition. Partial sums of convex sets are convex (see Exercise 152).

Example 115 (Polyhedron). *The polyhedron $\{\mathbf{x} | \mathbf{Ax} \leq \mathbf{b}, \mathbf{Cx} = \mathbf{d}\}$ can be expressed as the inverse image of the Cartesian product of the nonnegative orthant and the origin under the affine function $f(\mathbf{x}) = (\mathbf{b} - \mathbf{Ax}, \mathbf{d} - \mathbf{Cx})$:*

$$\{\mathbf{x} | \mathbf{Ax} \leq \mathbf{b}, \mathbf{Cx} = \mathbf{d}\} = \{\mathbf{x} | f(\mathbf{x}) \in \mathbb{R}_+^m \times \{0\}\}.$$

Example 116 (Solution set of linear matrix inequality). *The condition*

$$A(\mathbf{x}) = \mathbf{x}_1 \mathbf{A}_1 + \dots + \mathbf{x}_n \mathbf{A}_n \preceq \mathbf{B}, \quad (3.12)$$

where $\mathbf{B}, \mathbf{A}_i \in \mathbb{S}^m$, is called a linear matrix inequality (LMI) in \mathbf{x} (Note the similarity to an ordinary linear inequality,

$$\mathbf{a}^T \mathbf{x} = \mathbf{x}_1 a_1 + \dots + \mathbf{x}_n a_n \leq b,$$

with $b, a_i \in \mathbb{R}$.)

The solution set of a linear matrix inequality, $\{\mathbf{x} | A(\mathbf{x}) \preceq \mathbf{B}\}$, is convex. Indeed, it is the inverse image of the positive semidefinite cone under the affine function $f : \mathbb{R}^n \rightarrow \mathbb{S}^m$ given by $f(\mathbf{x}) = \mathbf{B} - A(\mathbf{x})$.

Example 117 (Hyperbolic cone). *The set*

$$\{\mathbf{x} | \mathbf{x}^T \mathbf{P} \mathbf{x} \leq (\mathbf{c}^T \mathbf{x})^2, \mathbf{c}^T \mathbf{x} \geq 0\}$$

where $\mathbf{P} \in \mathbb{S}_+^n$ and $\mathbf{c} \in \mathbb{R}^n$, is convex, since it is the inverse image of the second-order cone,

$$\{(\mathbf{z}, t) \mid \mathbf{z}^T \mathbf{z} \leq t^2, t \geq 0\},$$

under the affine function $f(\mathbf{x}) = (\mathbf{P}^{1/2}\mathbf{x}, \mathbf{c}^T \mathbf{x})$.

Example 118 (Ellipsoid). *The ellipsoid*

$$\epsilon = \{\mathbf{x} \mid (\mathbf{x} - \mathbf{x}_c)^T \mathbf{P}^{-1}(\mathbf{x} - \mathbf{x}_c) \leq 1\},$$

where $\mathbf{P} \in \mathbb{S}_{++}^n$, is the image of the unit Euclidean ball $\{\mathbf{u} \mid \|\mathbf{u}\|_2 \leq 1\}$ under the affine mapping $f(\mathbf{u}) = \mathbf{P}^{1/2}\mathbf{u} + \mathbf{x}_c$. (It is also the inverse image of the unit ball under the affine mapping $g(\mathbf{x}) = \mathbf{P}^{-1/2}(\mathbf{x} - \mathbf{x}_c)$.)

3.3.3 Linear-fractional and perspective functions

In this section we explore a class of functions, called linear-fractional, that is more general than affine but still preserves convexity.

3.3.3.1 The perspective function

We define the *perspective function* $P : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$, with domain $\text{dom } P = \mathbb{R}^n \times \mathbb{R}_{++}$, as $P(\mathbf{z}, t) = \mathbf{z}/t$. (Here \mathbb{R}_{++} denotes the set of positive numbers: $\mathbb{R}_{++} = \{x \in \mathbb{R} \mid x > 0\}$.) The perspective function scales or normalizes vectors so the last component is one, and then drops the last component.

Remark 119. We can interpret the perspective function as the action of a pin-hole camera. A pin-hole camera (in \mathbb{R}^3) consists of an opaque horizontal plane $\mathbf{x}_3 = 0$, with a single pin-hole at the origin, through which light can pass, and a horizontal image plane $\mathbf{x}_3 = -1$. An object at \mathbf{x} , above the camera (i.e., with $\mathbf{x}_3 > 0$), forms an image at the point $-(\mathbf{x}_1/\mathbf{x}_3, \mathbf{x}_2/\mathbf{x}_3, 1)$ on the image plane. Dropping the last component of the image point (since it is always -1), the image of a point at \mathbf{x} appears at $\mathbf{y} = -(\mathbf{x}_1/\mathbf{x}_3, \mathbf{x}_2/\mathbf{x}_3) = -P(\mathbf{x})$ on the image plane. This is illustrated in Figure 3.15.

If $C \subseteq \text{dom } P$ is convex, then its image

$$P(C) = \{P(\mathbf{x}) \mid \mathbf{x} \in C\}$$

is convex. This result is certainly intuitive: a convex object, viewed through a pin-hole camera, yields a convex image. To establish this fact we show that line segments are mapped to line segments under the perspective function. (This too makes sense: a line

segment, viewed through a pin-hole camera, yields a line segment image.) Suppose that $\mathbf{x} = (\tilde{\mathbf{x}}, x_{n+1}), \mathbf{y} = (\tilde{\mathbf{y}}, y_{n+1}) \in \mathbb{R}^{n+1}$ with $x_{n+1} > 0, y_{n+1} > 0$. Then for $0 \leq \theta \leq 1$,

$$P(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) = \frac{\theta\tilde{\mathbf{x}} + (1 - \theta)\tilde{\mathbf{y}}}{\theta x_{n+1} + (1 - \theta)y_{n+1}} = \mu P(\mathbf{x}) + (1 - \mu)P(\mathbf{y}),$$

where

$$\mu = \frac{\theta x_{n+1}}{\theta x_{n+1} + (1 - \theta)y_{n+1}} \in [0, 1].$$

This correspondence between θ and μ is monotonic: as θ varies between 0 and 1 (which sweeps out the line segment $[\mathbf{x}, \mathbf{y}]$), μ varies between 0 and 1 (which sweeps out the line segment $[P(\mathbf{x}), P(\mathbf{y})]$). This shows that $P([\mathbf{x}, \mathbf{y}]) = [P(\mathbf{x}), P(\mathbf{y})]$.

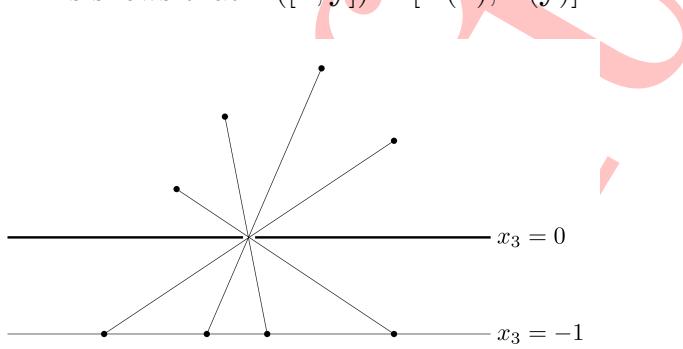


图 3.15: Pin-hole camera interpretation of perspective function. The dark horizontal line represents the plane $\mathbf{x}_3 = 0$ in \mathbb{R}^3 , which is opaque, except for a pin-hole at the origin. Objects or light sources above the plane appear on the image plane $\mathbf{x}_3 = -1$, which is shown as the lighter horizontal line. The mapping of the position of a source to the position of its image is related to the perspective function.

Now suppose C is convex with $C \subseteq \text{dom } P$ (i.e., $\mathbf{x}_{n+1} > 0$ for all $\mathbf{x} \in C$), and $\mathbf{x}, \mathbf{y} \in C$. To establish convexity of $P(C)$ we need to show that the line segment $[P(\mathbf{x}), P(\mathbf{y})]$ is in $P(C)$. But this line segment is the image of the line segment $[\mathbf{x}, \mathbf{y}]$ under P , and so lies in $P(C)$.

The inverse image of a convex set under the perspective function is also convex: if $C \subseteq \mathbb{R}^n$ is convex, then

$$P^{-1}(C) = \{(\mathbf{x}, t) \in \mathbb{R}^{n+1} | \mathbf{x}/t \in C, t > 0\}$$

is convex. To show this, suppose $(\mathbf{x}, t) \in P^{-1}(C), (\mathbf{y}, s) \in P^{-1}(C)$, and $0 \leq \theta \leq 1$. We need to show that

$$\theta(\mathbf{x}, t) + (1 - \theta)(\mathbf{y}, s) \in P^{-1}(C),$$

i.e., that

$$\frac{\theta\mathbf{x} + (1 - \theta)\mathbf{y}}{\theta t + (1 - \theta)s} \in C,$$

$(\theta t + (1 - \theta)s > 0)$ is obvious). This follows from

$$\frac{\theta \mathbf{x} + (1 - \theta)\mathbf{y}}{\theta t + (1 - \theta)s} = \mu(\mathbf{x}/t) + (1 - \mu)(\mathbf{y}/s),$$

where

$$\mu = \frac{\theta t}{\theta t + (1 - \theta)s} \in [0, 1].$$

3.3.3.2 Linear-fractional functions

A linear-fractional function is formed by composing the perspective function with an affine function. Suppose $g : \mathbb{R}^n \rightarrow \mathbb{R}^{m+1}$ is affine, i.e.,

$$g(\mathbf{x}) = \begin{bmatrix} \mathbf{A} \\ \mathbf{c}^T \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{b} \\ d \end{bmatrix}, \quad (3.13)$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{c} \in \mathbb{R}^n$, and $d \in \mathbb{R}$. The function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ given by $f = P \circ g$, i.e.,

$$f(\mathbf{x}) = (\mathbf{A}\mathbf{x} + \mathbf{b}) / (\mathbf{c}^T \mathbf{x} + d), \quad \text{dom } f = \{\mathbf{x} | \mathbf{c}^T \mathbf{x} + d > 0\}, \quad (3.14)$$

is called a linear-fractional (or projective) function. If $\mathbf{c} = \mathbf{0}$ and $d > 0$, the domain of f is \mathbb{R}^n , and f is an affine function. So we can think of affine and linear functions as special cases of linear-fractional functions.

Remark 120 (Projective interpretation). *It is often convenient to represent a linear-fractional function as a matrix*

$$\mathbf{Q} = \begin{bmatrix} \mathbf{A}^T & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} \in \mathbb{R}^{(m+1) \times (n+1)} \quad (3.15)$$

that acts on (multiplies) points of form $(\mathbf{x}, 1)$, which yields $(\mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{c}^T \mathbf{x} + d)$. This result is then scaled or normalized so that its last component is one, which yields $(f(\mathbf{x}), 1)$. This representation can be interpreted geometrically by associating \mathbb{R}^n with a set of rays in \mathbb{R}^{n+1} as follows. With each point \mathbf{z} in \mathbb{R}^n we associate the (open) ray $P(\mathbf{z}) = \{t(\mathbf{z}, 1) | t > 0\}$ in \mathbb{R}^{n+1} . The last component of this ray takes on positive values. Conversely any ray in \mathbb{R}^{n+1} , with base at the origin and last component which takes on positive values, can be written as $P(\mathbf{v}) = \{t(\mathbf{v}, 1) | t \geq 0\}$ for some $\mathbf{v} \in \mathbb{R}^n$. This (projective) correspondence P between \mathbb{R}^n and the halfspace of rays with positive last component is one-to-one and onto. The linear-fractional function (3.14) can be expressed as

$$f(\mathbf{x}) = \mathbf{P}^{-1}(\mathbf{Q}\mathbf{P}(\mathbf{x})).$$

Thus, we start with $\mathbf{x} \in \text{dom } f$, i.e., $\mathbf{c}^T \mathbf{x} + d > 0$. We then form the ray $\mathbf{P}(\mathbf{x})$ in \mathbb{R}^{n+1} . The linear transformation with matrix \mathbf{Q} acts on this ray to produce another ray $\mathbf{QP}(\mathbf{x})$. Since $\mathbf{x} \in \text{dom } f$, the last component of this ray assumes positive values. Finally we take the inverse projective transformation to recover $f(\mathbf{x})$.

Like the perspective function, linear-fractional functions preserve convexity. If C is convex and lies in the domain of f (i.e., $\mathbf{c}^T \mathbf{x} + d > 0$ for $\mathbf{x} \in C$), then its image $f(C)$ is convex. This follows immediately from results above: the image of C under the affine mapping (3.13) is convex, and the image of the resulting set under the perspective function P , which yields $f(C)$, is convex. Similarly, if $C \subseteq \mathbb{R}^m$ is convex, then the inverse image $f^{-1}(C)$ is convex.

Example 121 (Conditional probabilities). Suppose u and v are random variables that take on values in $\{1, \dots, n\}$ and $\{1, \dots, m\}$, respectively, and let p_{ij} denote $\text{prob}(u = i, v = j)$. Then the conditional probability $f_{ij} = \text{prob}(u = i | v = j)$ is given by

$$f_{ij} = \frac{p_{ij}}{\sum_{k=1}^n p_{kj}}.$$

Thus \mathbf{f} is obtained by a linear-fractional mapping from \mathbf{p} . It follows that if C is a convex set of joint probabilities for (u, v) , then the associated set of conditional probabilities of u given v is also convex.

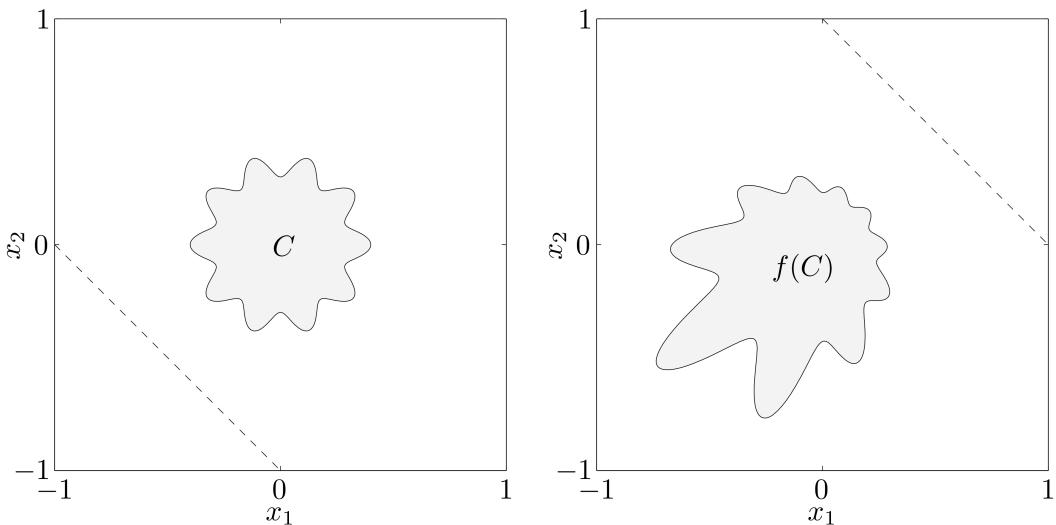


图 3.16: Left. A set $C \subseteq \mathbb{R}^2$. The dashed line shows the boundary of the domain of the linear-fractional function $f(\mathbf{x}) = \mathbf{x}/(\mathbf{x}_1 + \mathbf{x}_2 + 1)$ with $\text{dom } f = \{(x_1, x_2) | x_1 + x_2 + 1 > 0\}$. Right. Image of C under f . The dashed line shows the boundary of the domain of f^{-1} .

Figure 3.16 shows a set $C \subseteq \mathbb{R}^2$, and its image under the linear-fractional function

$$f(\mathbf{x}) = \frac{1}{\mathbf{x}_1 + \mathbf{x}_2 + 1} \mathbf{x}, \quad \text{dom } f = \{(x_1, x_2) | x_1 + x_2 + 1 > 0\}.$$

3.4 Generalized inequalities

3.4.1 Proper cones and generalized inequalities

A cone $K \subseteq \mathbb{R}^n$ is called a *proper cone* if it satisfies the following:

- K is convex.
- K is closed.
- K is solid, which means it has nonempty interior.
- K is pointed, which means that it contains no line (or equivalently, $\mathbf{x} \in K, -\mathbf{x} \in K \Rightarrow \mathbf{x} = 0$).

A proper cone K can be used to define a generalized inequality, which is a partial ordering on \mathbb{R}^n that has many of the properties of the standard ordering on \mathbb{R} . We associate the proper cone K with the partial ordering on \mathbb{R}^n defined by

$$\mathbf{x} \preceq_K \mathbf{y} \iff \mathbf{y} - \mathbf{x} \in K.$$

We also write $\mathbf{x} \succeq_K \mathbf{y}$ for $\mathbf{y} \preceq_K \mathbf{x}$. Similarly, we define an associated strict partial ordering by

$$\mathbf{x} \prec_K \mathbf{y} \iff \mathbf{y} - \mathbf{x} \in \text{int } K,$$

and write $\mathbf{x} \succ_K \mathbf{y}$ for $\mathbf{y} \prec_K \mathbf{x}$. (To distinguish the generalized inequality \preceq_K from the strict generalized inequality \prec_K , we sometimes refer to \preceq_K as the nonstrict generalized inequality.)

When $K = \mathbb{R}_+$, the partial ordering \preceq_K is the usual ordering \leq on \mathbb{R} , and the strict partial ordering \prec_K is the same as the usual strict ordering $<$ on \mathbb{R} . So generalized inequalities include as a special case ordinary (nonstrict and strict) inequality in \mathbb{R} .

Example 122 (Nonnegative orthant and componentwise inequality). *The nonnegative orthant $K = \mathbb{R}_+^n$ is a proper cone. The associated generalized inequality \preceq_K corresponds to componentwise inequality between vectors: $\mathbf{x} \preceq_K \mathbf{y}$ means that $\mathbf{x}_i \leq \mathbf{y}_i, i = 1, \dots, n$. The associated strict inequality corresponds to componentwise strict inequality: $\mathbf{x} \prec_K \mathbf{y}$ means that $\mathbf{x}_i < \mathbf{y}_i, i = 1, \dots, n$.*

The nonstrict and strict partial orderings associated with the nonnegative orthant arise so frequently that we drop the subscript \mathbb{R}_+^n ; it is understood when the symbol \preceq or \prec appears between vectors.

Example 123 (Positive semidefinite cone and matrix inequality). The positive semidefinite cone S_+^n is a proper cone in S^n . The associated generalized inequality \preceq_K is the usual matrix inequality: $\mathbf{X} \preceq_K \mathbf{Y}$ means $\mathbf{Y} - \mathbf{X}$ is positive semidefinite. The interior of S_+^n (in S^n) consists of the positive definite matrices, so the strict generalized inequality also agrees with the usual strict inequality between symmetric matrices: $\mathbf{X} \prec_K \mathbf{Y}$ means $\mathbf{Y} - \mathbf{X}$ is positive definite.

Here, too, the partial ordering arises so frequently that we drop the subscript: for symmetric matrices we write simply $\mathbf{X} \preceq \mathbf{Y}$ or $\mathbf{X} \prec \mathbf{Y}$. It is understood that the generalized inequalities are with respect to the positive semidefinite cone.

Example 124 (Cone of polynomials nonnegative on $[0, 1]$). Let K be defined as

$$K = \{\mathbf{c} \in \mathbb{R}^n \mid c_1 + c_2 t + \dots + c_n t^{n-1} \geq 0 \text{ for } t \in [0, 1]\}, \quad (3.16)$$

i.e., K is the cone of (coefficients of) polynomials of degree $n - 1$ that are nonnegative on the interval $[0, 1]$. It can be shown that K is a proper cone, its interior is the set of coefficients of polynomials that are positive on the interval $[0, 1]$.

Two vectors $\mathbf{c}, \mathbf{d} \in \mathbb{R}^n$ satisfy $\mathbf{c} \prec_K \mathbf{d}$ if and only if

$$c_1 + c_2 t + \dots + c_n t^{n-1} \leq d_1 + d_2 t + \dots + d_n t^{n-1}$$

for all $t \in [0, 1]$.

3.4.1.1 Properties of generalized inequalities

A generalized inequality \preceq_K satisfies many properties, such as

- \preceq_K is preserved under addition: if $\mathbf{x} \preceq_K \mathbf{y}$ and $\mathbf{u} \preceq_K \mathbf{v}$, then $\mathbf{x} + \mathbf{u} \preceq_K \mathbf{y} + \mathbf{v}$.
- \preceq_K is transitive: if $\mathbf{x} \preceq_K \mathbf{y}$ and $\mathbf{y} \preceq_K \mathbf{z}$ then $\mathbf{x} \preceq_K \mathbf{z}$.
- \preceq_K is preserved under nonnegative scaling: if $\mathbf{x} \preceq_K \mathbf{y}$ and $\alpha \geq 0$ then $\alpha\mathbf{x} \preceq_K \alpha\mathbf{y}$.
- \preceq_K is reflexive: $\mathbf{x} \preceq_K \mathbf{x}$.
- \preceq_K is antisymmetric: if $\mathbf{x} \preceq_K \mathbf{y}$ and $\mathbf{y} \preceq_K \mathbf{x}$, then $\mathbf{x} = \mathbf{y}$.
- \preceq_K is preserved under limits: if $\mathbf{x}_i \preceq_K \mathbf{y}_i$ for $i = 1, 2, \dots$, $\mathbf{x}_i \rightarrow \mathbf{x}$ and $\mathbf{y}_i \rightarrow \mathbf{y}$ as $i \rightarrow \infty$, then $\mathbf{x} \preceq_K \mathbf{y}$.

The corresponding strict generalized inequality \prec_K satisfies, for example,

- if $\mathbf{x} \prec_K \mathbf{y}$ then $\mathbf{x} \preceq_K \mathbf{y}$.
- if $\mathbf{x} \prec_K \mathbf{y}$ and $\mathbf{u} \preceq_K \mathbf{v}$ then $\mathbf{x} + \mathbf{u} \prec_K \mathbf{y} + \mathbf{v}$.
- if $\mathbf{x} \prec_K \mathbf{y}$ and $\alpha > 0$ then $\alpha\mathbf{x} \prec_K \alpha\mathbf{y}$.
- $\mathbf{x} \not\prec_K \mathbf{x}$.
- if $\mathbf{x} \prec_K \mathbf{y}$, then for \mathbf{u} and \mathbf{v} small enough, $\mathbf{x} + \mathbf{u} \prec_K \mathbf{y} + \mathbf{v}$.

These properties are inherited from the definitions of \preceq_K and \prec_K , and the properties of proper cones, see Exercise 167.

3.4.2 Minimum and minimal elements

The notation of generalized inequality (i.e., \preceq_K , \prec_K) is meant to suggest the analogy to ordinary inequality on \mathbb{R} (i.e., \leq , $<$). While many properties of ordinary inequality do hold for generalized inequalities, some important ones do not. The most obvious difference is that \leq on \mathbb{R} is a linear ordering: any two points are comparable, meaning either $\mathbf{x} \leq \mathbf{y}$ or $\mathbf{y} \leq \mathbf{x}$. This property does not hold for other generalized inequalities. One implication is that concepts like minimum and maximum are more complicated in the context of generalized inequalities. We briefly discuss this in this section.

We say that $\mathbf{x} \in S$ is the *minimum element* of S (with respect to the generalized inequality \preceq_K) if for every $\mathbf{y} \in S$ we have $\mathbf{x} \prec_K \mathbf{y}$. We define the *maximum element* of a set S , with respect to a generalized inequality, in a similar way. If a set has a minimum (maximum) element, then it is unique. A related concept is minimal element. We say that $\mathbf{x} \in S$ is a *minimal element* of S (with respect to the generalized inequality \preceq_K) if $\mathbf{y} \in S$, $\mathbf{y} \preceq_K \mathbf{x}$ only if $\mathbf{y} = \mathbf{x}$. We define *maximal element* in a similar way. A set can have many different minimal (maximal) elements.

We can describe minimum and minimal elements using simple set notation. A point $\mathbf{x} \in S$ is the *minimum element* of S if and only if

$$S \subseteq \mathbf{x} + K.$$

Here $\mathbf{x} + K$ denotes all the points that are comparable to \mathbf{x} and greater than or equal to \mathbf{x} (according to \preceq_K). A point $\mathbf{x} \in S$ is a *minimal element* if and only if

$$(\mathbf{x} - K) \cap S = \{\mathbf{x}\}.$$

Here $\mathbf{x} - K$ denotes all the points that are comparable to \mathbf{x} and less than or equal to \mathbf{x} (according to \preceq_K), the only point in common with S is \mathbf{x} .

For $K = \mathbb{R}_+$, which induces the usual ordering on \mathbb{R} , the concepts of minimal and minimum are the same, and agree with the usual definition of the minimum element of a set.

Example 125. Consider the cone \mathbb{R}_+^2 , which induces componentwise inequality in \mathbb{R}^2 . Here we can give some simple geometric descriptions of minimal and minimum elements. The inequality $\mathbf{x} \preceq \mathbf{y}$ means \mathbf{y} is above and to the right of \mathbf{x} . To say that $\mathbf{x} \in S$ is the minimum element of a set S means that all other points of S lie above and to the right. To say that \mathbf{x} is a minimal element of a set S means that no other point of S lies to the left and below \mathbf{x} . This is illustrated in Figure 3.17.

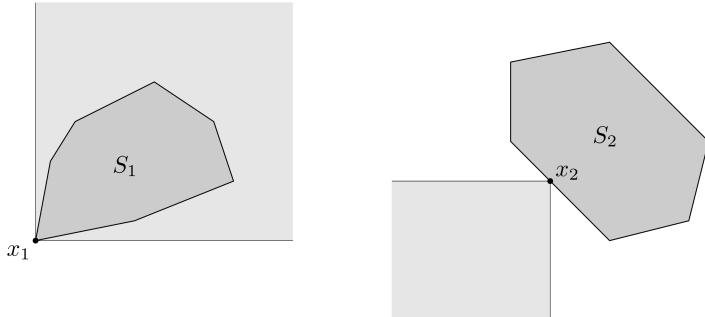


图 3.17: Left. The set S_1 has a minimum element \mathbf{x}_1 with respect to componentwise inequality in \mathbb{R}^2 . The set $\mathbf{x}_1 + K$ is shaded lightly, \mathbf{x}_1 is the minimum element of S_1 since $S_1 \subseteq \mathbf{x}_1 + K$. Right. The point \mathbf{x}_2 is a minimal point of S_2 . The set $\mathbf{x}_2 - K$ is shown lightly shaded. The point \mathbf{x}_2 is minimal because $\mathbf{x}_2 - K$ and S_2 intersect only at \mathbf{x}_2 .

Example 126 (Minimum and minimal elements of a set of symmetric matrices). We associate with each $\mathbf{A} \in \mathbb{S}_{++}^n$ an ellipsoid centered at the origin, given by

$$\varepsilon_{\mathbf{A}} = \{\mathbf{x} \mid \mathbf{x}^T \mathbf{A}^{-1} \mathbf{x} \leq 1\}.$$

We have $\mathbf{A} \preceq \mathbf{B}$ if and only if $\varepsilon_{\mathbf{A}} \subseteq \varepsilon_{\mathbf{B}}$. Let $\mathbf{v}_1, \dots, \mathbf{v}_k \in \mathbb{R}^n$ be given and define

$$S = \{\mathbf{P} \in \mathbb{S}_{++}^n \mid \mathbf{v}_i^T \mathbf{P}^{-1} \mathbf{v}_i \leq 1, i = 1, \dots, k\},$$

which corresponds to the set of ellipsoids that contain the points $\mathbf{v}_1, \dots, \mathbf{v}_k$. The set S does not have a minimum element: for any ellipsoid that contains the points $\mathbf{v}_1, \dots, \mathbf{v}_k$ we can find another one that contains the points, and is not comparable to it. An ellipsoid is minimal if it contains the points, but no smaller ellipsoid does. Figure 3.18 shows an example in \mathbb{R}^2 with $k = 2$.

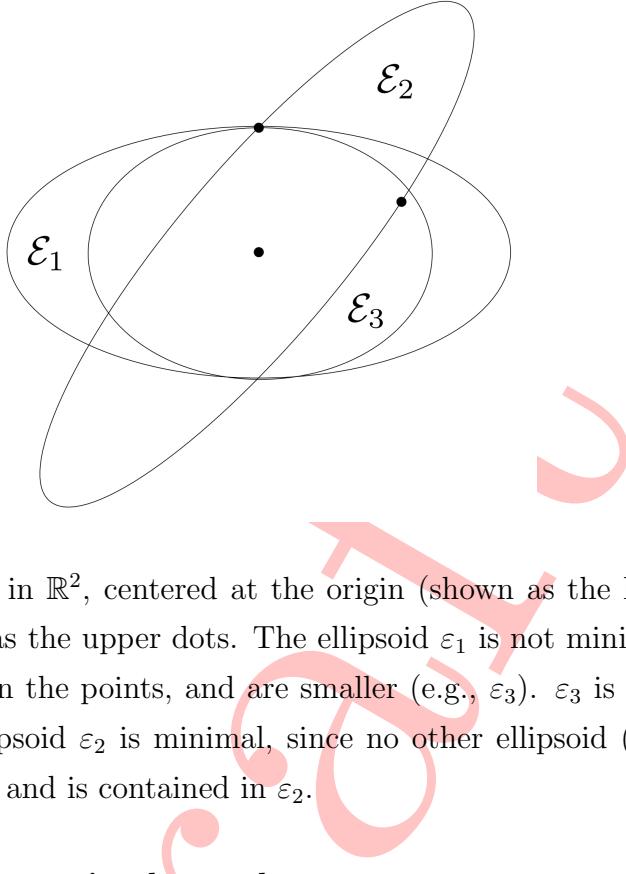


图 3.18: Three ellipsoids in \mathbb{R}^2 , centered at the origin (shown as the lower dot), that contain the points shown as the upper dots. The ellipsoid ε_1 is not minimal, since there exist ellipsoids that contain the points, and are smaller (e.g., ε_3). ε_3 is not minimal for the same reason. The ellipsoid ε_2 is minimal, since no other ellipsoid (centered at the origin) contains the points and is contained in ε_2 .

3.5 Separating and supporting hyperplanes

3.5.1 Separating hyperplane theorem

In this section we describe an idea that will be important later: the use of hyperplanes or affine functions to separate convex sets that do not intersect. The basic result is the separating hyperplane theorem: Suppose C and D are two convex sets that do not intersect, i.e., $C \cap D = \emptyset$. Then there exist $\mathbf{a} \neq \mathbf{0}$ and b such that $\mathbf{a}^T \mathbf{x} \leq b$ for all $\mathbf{x} \in C$ and $\mathbf{a}^T \mathbf{x} \geq b$ for all $\mathbf{x} \in D$. In other words, the affine function $\mathbf{a}^T \mathbf{x} - b$ is nonpositive on C and nonnegative on D . The hyperplane $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = b\}$ is called a separating hyperplane for the sets C and D , or is said to separate the sets C and D . This is illustrated in Figure 3.19.

3.5.1.1 Proof of separating hyperplane theorem

Here we consider a special case, and leave the extension of the proof to the general case as an exercise (Exercise 158). We assume that the (Euclidean) distance between C and D , defined as

$$\text{dist}(C, D) = \inf\{\|\mathbf{u} - \mathbf{v}\|_2 | \mathbf{u} \in C, \mathbf{v} \in D\},$$

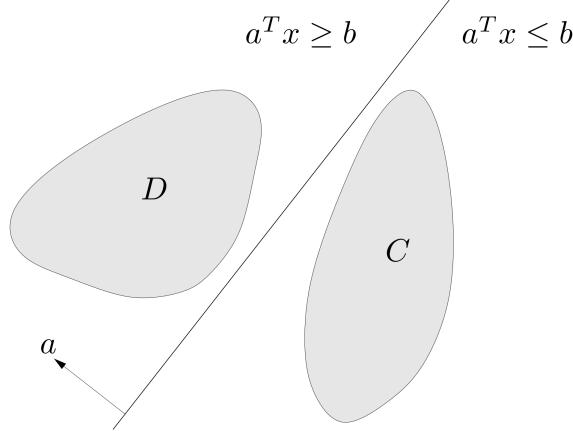


图 3.19: The hyperplane $\{\mathbf{x} \mid \mathbf{a}^T \mathbf{x} = \mathbf{b}\}$ separates the disjoint convex sets C and D . The affine function $\mathbf{a}^T \mathbf{x} - b$ is nonpositive on C and nonnegative on D .

is positive, and that there exist points $\mathbf{c} \in C$ and $\mathbf{d} \in D$ that achieve the minimum distance, i.e., $\|\mathbf{c} - \mathbf{d}\|_2 = \text{dist}(C, D)$. (These conditions are satisfied, for example, when C and D are closed and one set is bounded.)

The hyperplane passing through the mid-point of \mathbf{c} and \mathbf{d} and having $\mathbf{d} - \mathbf{c}$ as its normal is

$$(\mathbf{d} - \mathbf{c})^T (\mathbf{x} - (1/2)(\mathbf{d} + \mathbf{c})) = 0,$$

which corresponds to defining

$$\mathbf{a} = \mathbf{d} - \mathbf{c}, b = \frac{\|\mathbf{d}\|_2^2 - \|\mathbf{c}\|_2^2}{2}.$$

We will show that the affine function

$$f(\mathbf{x}) = (\mathbf{d} - \mathbf{c})^T (\mathbf{x} - (1/2)(\mathbf{d} + \mathbf{c}))$$

is nonpositive on C and nonnegative on D , i.e., that the hyperplane $\{\mathbf{x} \mid \mathbf{a}^T \mathbf{x} = b\}$ separates C and D . This hyperplane is perpendicular to the line segment between \mathbf{c} and \mathbf{d} , and passes through its midpoint, as shown in Figure 3.20.

We first show that f is nonnegative on D . The proof that f is nonpositive on C is similar (or follows by swapping C and D and considering $-f$). Suppose there were a point $\mathbf{u} \in D$ for which

$$f(\mathbf{u}) = (\mathbf{d} - \mathbf{c})^T (\mathbf{u} - (1/2)(\mathbf{d} + \mathbf{c})) < 0. \quad (3.17)$$

We can express $f(\mathbf{u})$ as

$$\begin{aligned} f(\mathbf{u}) &= (\mathbf{d} - \mathbf{c})^T (\mathbf{u} - \mathbf{d} + (1/2)(\mathbf{d} - \mathbf{c})) \\ &= (\mathbf{d} - \mathbf{c})^T (\mathbf{u} - \mathbf{d}) + (1/2)\|\mathbf{d} - \mathbf{c}\|_2^2. \end{aligned}$$

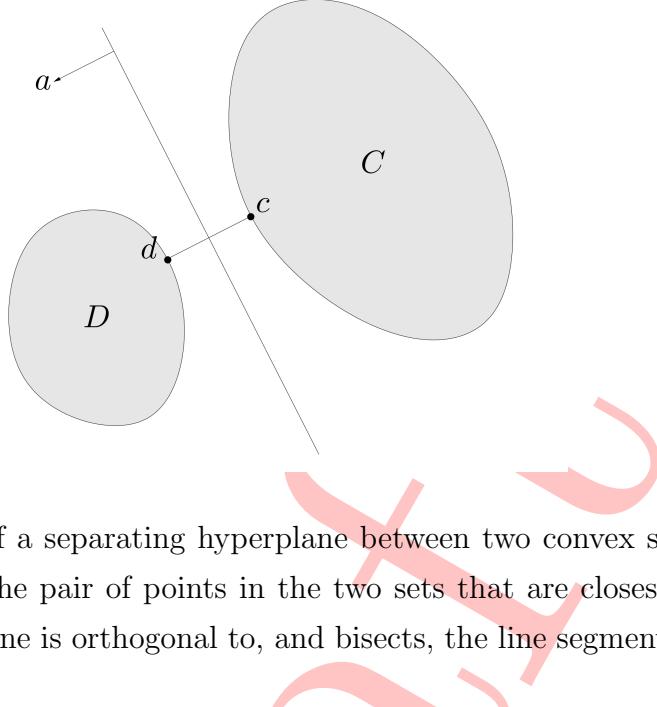


图 3.20: Construction of a separating hyperplane between two convex sets. The points $\mathbf{c} \in C$ and $\mathbf{d} \in D$ are the pair of points in the two sets that are closest to each other. The separating hyperplane is orthogonal to, and bisects, the line segment between \mathbf{c} and \mathbf{d} .

We see that (3.17) implies $(\mathbf{d} - \mathbf{c})^T(\mathbf{u} - \mathbf{d}) < 0$. Now we observe that

$$\frac{d}{dt} \|\mathbf{d} + t(\mathbf{u} - \mathbf{d}) - \mathbf{c}\|_2^2 \Big|_{t=0} = 2(\mathbf{d} - \mathbf{c})^T(\mathbf{u} - \mathbf{d}) < 0,$$

so for some small $t > 0$, with $t \leq 1$, we have

$$\|\mathbf{d} + t(\mathbf{u} - \mathbf{d}) - \mathbf{c}\|_2 < \|\mathbf{d} - \mathbf{c}\|_2,$$

i.e., the point $\mathbf{d} + t(\mathbf{u} - \mathbf{d})$ is closer to \mathbf{c} than \mathbf{d} is. Since D is convex and contains \mathbf{d} and \mathbf{u} , we have $\mathbf{d} + t(\mathbf{u} - \mathbf{d}) \in D$. But this is impossible, since \mathbf{d} is assumed to be the point in D that is closest to C .

Example 127 (Separation of an affine and a convex set). Suppose C is convex and D is affine, i.e., $D = \{\mathbf{Fu} + \mathbf{g} \mid \mathbf{u} \in \mathbb{R}^m\}$, where $\mathbf{F} \in \mathbb{R}^{n \times m}$. Suppose C and D are disjoint, so by the separating hyperplane theorem there are $\mathbf{a} \neq 0$ and b such that $\mathbf{a}^T \mathbf{x} \leq b$ for all $\mathbf{x} \in C$ and $\mathbf{a}^T \mathbf{x} \geq b$ for all $\mathbf{x} \in D$.

Now $\mathbf{a}^T \mathbf{x} \geq b$ for all $\mathbf{x} \in D$ means $\mathbf{a}^T \mathbf{Fu} \geq b - \mathbf{a}^T \mathbf{g}$ for all $\mathbf{u} \in \mathbb{R}^m$. But a linear function is bounded below on \mathbb{R}^m only when it is zero, so we conclude $\mathbf{a}^T \mathbf{F} = \mathbf{0}$ (and hence, $b \leq \mathbf{a}^T \mathbf{g}$).

Thus we conclude that there exists $\mathbf{a} \neq 0$ such that $\mathbf{F}^T \mathbf{a} = \mathbf{0}$ and $\mathbf{a}^T \mathbf{x} \leq \mathbf{a}^T \mathbf{g}$ for all $\mathbf{x} \in C$.

3.5.1.2 Strict separation

The separating hyperplane we constructed above satisfies the stronger condition that $\mathbf{a}^T \mathbf{x} < b$ for all $\mathbf{x} \in C$ and $\mathbf{a}^T \mathbf{x} > b$ for all $\mathbf{x} \in D$. This is called *strict separation* of the sets C and D . Simple examples show that in general, disjoint convex sets need not be strictly separable by a hyperplane (even when the sets are closed, see Exercise 159). In many special cases, however, strict separation can be established.

Example 128 (Strict separation of a point and a closed convex set). *Let C be a closed convex set and $\mathbf{x}_0 \notin C$. Then there exists a hyperplane that strictly separates \mathbf{x}_0 from C .*

To see this, note that the two sets C and $B(\mathbf{x}_0, \varepsilon)$ do not intersect for some $\varepsilon > 0$. By the separating hyperplane theorem, there exist $\mathbf{a} \neq \mathbf{0}$ and b such that $\mathbf{a}^T \mathbf{x} \leq b$ for $\mathbf{x} \in C$ and $\mathbf{a}^T \mathbf{x} \geq b$ for $\mathbf{x} \in B(\mathbf{x}_0, \varepsilon)$.

Using $B(\mathbf{x}_0, \varepsilon) = \{\mathbf{x}_0 + \mathbf{u} \mid \|\mathbf{u}\|_2 \leq \varepsilon\}$, the second condition can be expressed as

$$\mathbf{a}^T (\mathbf{x}_0 + \mathbf{u}) \geq b \text{ for all } \|\mathbf{u}\|_2 \leq \varepsilon.$$

The \mathbf{u} that minimizes the lefthand side is $\mathbf{u} = -\varepsilon \mathbf{a} / \|\mathbf{a}\|_2$; using this value we have

$$\mathbf{a}^T \mathbf{x}_0 - \varepsilon \|\mathbf{a}\|_2 \geq b.$$

Therefore the affine function

$$f(\mathbf{x}) = \mathbf{a}^T \mathbf{x} - b - \varepsilon \|\mathbf{a}\|_2 / 2$$

is negative on C and positive at \mathbf{x}_0 .

As an immediate consequence we can establish a fact that we already mentioned above: a closed convex set is the intersection of all halfspaces that contain it. Indeed, let C be closed and convex, and let S be the intersection of all halfspaces containing C . Obviously $\mathbf{x} \in C \implies \mathbf{x} \in S$. To show the converse, suppose there exists $\mathbf{x} \in S, \mathbf{x} \notin C$. By the strict separation result there exists a hyperplane that strictly separates \mathbf{x} from C , i.e., there is a halfspace containing C but not \mathbf{x} . In other words, $\mathbf{x} \notin S$.

3.5.1.3 Converse separating hyperplane theorems

The converse of the separating hyperplane theorem (i.e., existence of a separating hyperplane implies that C and D do not intersect) is not true, unless one imposes additional constraints on C or D , even beyond convexity. As a simple counterexample, consider $C = D = \{0\} \subseteq \mathbb{R}$. Here the hyperplane $\mathbf{x} = 0$ separates C and D .

By adding conditions on C and D various converse separation theorems can be derived. As a very simple example, suppose C and D are convex sets, with C open,

and there exists an affine function f that is nonpositive on C and nonnegative on D . Then C and D are disjoint. (To see this we first note that f must be negative on C , for if f were zero at a point of C then f would take on positive values near the point, which is a contradiction. But then C and D must be disjoint since f is negative on C and nonnegative on D .) Putting this converse together with the separating hyperplane theorem, we have the following result: any two convex sets C and D , at least one of which is open, are disjoint if and only if there exists a separating hyperplane.

Example 129 (Theorem of alternatives for strict linear inequalities). *We derive the necessary and sufficient conditions for solvability of a system of strict linear inequalities*

$$\mathbf{Ax} \prec \mathbf{b}. \quad (3.18)$$

These inequalities are infeasible if and only if the (convex) sets

$$C = \{\mathbf{b} - \mathbf{Ax} \mid \mathbf{x} \in \mathbb{R}^n\}, \quad D = \mathbb{R}_{++}^m = \{\mathbf{y} \in \mathbb{R}^m \mid \mathbf{y} \succ \mathbf{0}\}$$

do not intersect. The set D is open, C is an affine set. Hence by the result above, C and D are disjoint if and only if there exists a separating hyperplane, i.e., a nonzero $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\mu \in \mathbb{R}$ such that $\boldsymbol{\lambda}^T \mathbf{y} \leq \mu$ on C and $\boldsymbol{\lambda}^T \mathbf{y} \geq \mu$ on D .

Each of these conditions can be simplified. The first means $\boldsymbol{\lambda}^T(\mathbf{b} - \mathbf{Ax}) \leq \mu$ for all \mathbf{x} . This implies (as in Example 127) that $\mathbf{A}^T \boldsymbol{\lambda} = \mathbf{0}$ and $\boldsymbol{\lambda}^T \mathbf{b} \leq \mu$. The second inequality means $\boldsymbol{\lambda}^T \mathbf{y} \geq \mu$ for all $\mathbf{y} \succ \mathbf{0}$. This implies $\mu \leq 0$ and $\boldsymbol{\lambda} \succeq \mathbf{0}$, $\boldsymbol{\lambda} \neq \mathbf{0}$.

Putting it all together, we find that the set of strict inequalities (3.18) is infeasible if and only if there exists $\boldsymbol{\lambda} \in \mathbb{R}^m$ such that

$$\boldsymbol{\lambda} \neq \mathbf{0}, \boldsymbol{\lambda} \succeq \mathbf{0}, \mathbf{A}^T \boldsymbol{\lambda} = \mathbf{0}, \boldsymbol{\lambda}^T \mathbf{b} \leq 0. \quad (3.19)$$

This is also a system of linear inequalities and linear equations in the variable $\boldsymbol{\lambda} \in \mathbb{R}^m$. We say that (3.18) and (3.19) form a pair of alternatives: for any data \mathbf{A} and \mathbf{b} , exactly one of them is solvable.

The above example can be better summarized by the following:

Theorem 130 (Theorem of the Alternative (Farkas (1902))). *For $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$ the following are strong alternatives:*

1. $\exists \mathbf{x} \in \mathbb{R}_+^n$ such that $\mathbf{Ax} = \mathbf{b}$,
2. $\exists \mathbf{y} \in \mathbb{R}^m$ such that $\mathbf{A}^T \mathbf{y} \geq \mathbf{0}$ and $\mathbf{b}^T \mathbf{y} < 0$.

Proof. 1) \implies 2): For $\mathbf{x} \in \mathbb{R}_+^n$ with $\mathbf{Ax} = \mathbf{b}$ and $\mathbf{y} \in \mathbb{R}^m$ with $\mathbf{A}^T \mathbf{y} \geq 0$ we have $\mathbf{b}^T \mathbf{y} = \mathbf{x}^T \mathbf{A}^T \mathbf{y} \geq 0$.

¬1) \implies 2): $C := \text{cone}(\mathbf{A})$ is a closed convex cone which does not contain the vector \mathbf{b} : by the Separating Hyperplane Theorem (Section 3.5.1) there exists a $\mathbf{y} \in \mathbb{R}^m$ with $\langle \mathbf{y}, \mathbf{x} \rangle \geq 0 > \langle \mathbf{y}, \mathbf{b} \rangle$ for all $\mathbf{x} \in C$, in particular $\mathbf{A}_i^T \mathbf{y} = \langle \mathbf{y}, \mathbf{A}_i \rangle \geq 0, \forall i$, that is, $\mathbf{A}^T \mathbf{y} \geq \mathbf{0}$. \square

3.5.2 Supporting hyperplanes

Suppose $C \subseteq \mathbb{R}^n$, and \mathbf{x}_0 is a point in its boundary $\mathbf{bd} C$, i.e.,

$$\mathbf{x}_0 \in \mathbf{bd} C = \mathbf{cl} C \setminus \mathbf{int} C.$$

If $\mathbf{a} \neq \mathbf{0}$ satisfies $\mathbf{a}^T \mathbf{x} \leq \mathbf{a}^T \mathbf{x}_0$ for all $\mathbf{x} \in C$, then the hyperplane $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = \mathbf{a}^T \mathbf{x}_0\}$ is called a *supporting hyperplane* to C at the point \mathbf{x}_0 . This is equivalent to saying that the point \mathbf{x}_0 and the set C are separated by the hyperplane $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = \mathbf{a}^T \mathbf{x}_0\}$. The geometric interpretation is that the hyperplane $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = \mathbf{a}^T \mathbf{x}_0\}$ is tangent to C at \mathbf{x}_0 , and the halfspace $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} \leq \mathbf{a}^T \mathbf{x}_0\}$ contains C . This is illustrated in Figure 3.21.

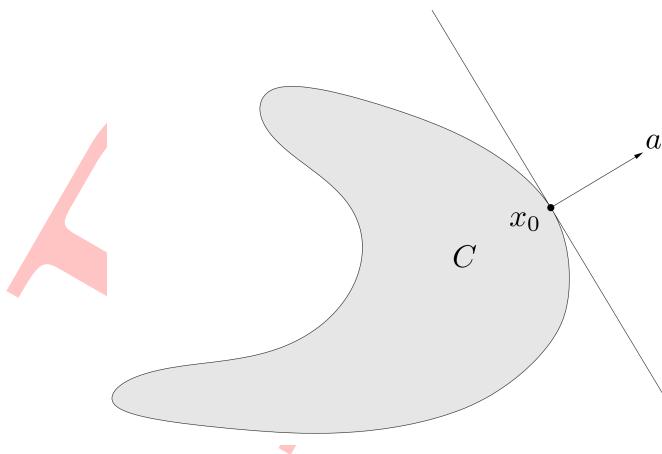


图 3.21: The hyperplane $\{\mathbf{x} | \mathbf{a}^T \mathbf{x} = \mathbf{a}^T \mathbf{x}_0\}$ supports C at \mathbf{x}_0 .

A basic result, called the supporting hyperplane theorem, states that for any nonempty convex set C , and any $\mathbf{x}_0 \in \mathbf{bd} C$, there exists a supporting hyperplane to C at \mathbf{x}_0 . The supporting hyperplane theorem is readily proved from the separating hyperplane theorem. We distinguish two cases. If the interior of C is nonempty, the result follows immediately by applying the separating hyperplane theorem to the sets $\{\mathbf{x}_0\}$ and $\mathbf{int} C$. If the interior of C is empty, then C must lie in an affine set of dimension less than n , and

any hyperplane containing that affine set contains C and \mathbf{x}_0 , and is a (trivial) supporting hyperplane.

There is also a partial converse of the supporting hyperplane theorem: If a set is closed, has nonempty interior, and has a supporting hyperplane at every point in its boundary, then it is convex. (See Exercise 163.)

3.6 Dual cones and generalized inequalities

3.6.1 Dual cones

Let K be a cone. The set

$$K^* = \{\mathbf{y} \mid \mathbf{x}^T \mathbf{y} \geq 0 \text{ for all } \mathbf{x} \in K\} \quad (3.20)$$

is called the dual cone of K . As the name suggests, K^* is a cone, and is always convex, even when the original cone K is not (see Exercise 168).

Geometrically, $\mathbf{y} \in K^*$ if and only if $-\mathbf{y}$ is the normal of a hyperplane that supports K at the origin. This is illustrated in Figure 3.22.

Example 131 (Subspace). The dual cone of a subspace $V \subseteq \mathbb{R}^n$ (which is a cone) is its orthogonal complement

$$V^\perp = \{\mathbf{y} \mid \mathbf{y}^T \mathbf{v} = 0 \text{ for all } \mathbf{v} \in V\}.$$

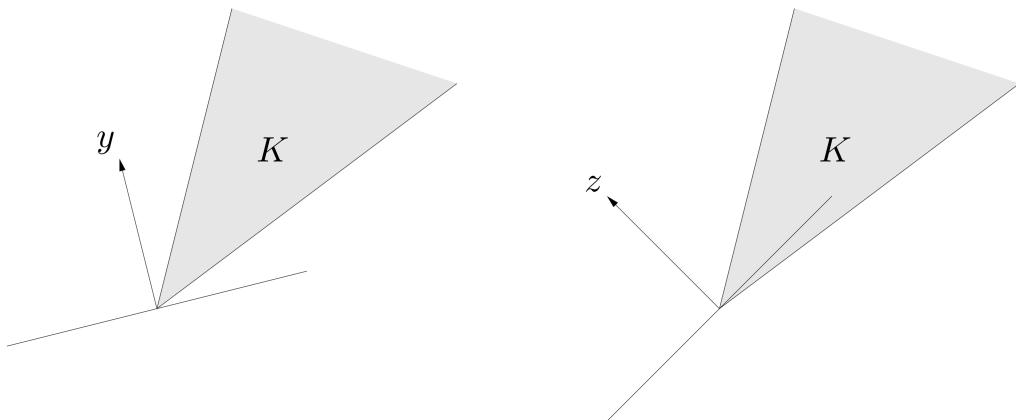


图 3.22: Left. The halfspace with inward normal \mathbf{y} contains the cone K , so $\mathbf{y} \in K^*$. Right. The halfspace with inward normal \mathbf{z} does not contain K , so $\mathbf{z} \notin K^*$.

Example 132 (Nonnegative orthant). *The cone \mathbb{R}_+^n is its own dual:*

$$\mathbf{y}^T \mathbf{x} \geq 0 \text{ for all } \mathbf{x} \succeq \mathbf{0} \iff \mathbf{y} \succeq \mathbf{0}.$$

We call such a cone self-dual.

Example 133 (Positive semidefinite cone). *On the set of symmetric $n \times n$ matrices S^n , we use the standard inner product $\text{tr}(\mathbf{XY}) = \sum_{i,j=1}^n \mathbf{X}_{ij} \mathbf{Y}_{ij}$ (see Section 2.11.1.1). The positive semidefinite cone S_+^n is self-dual, i.e., for $\mathbf{X}, \mathbf{Y} \in S^n$,*

$$\text{tr}(\mathbf{XY}) \geq 0 \text{ for all } \mathbf{X} \succeq \mathbf{0} \iff \mathbf{Y} \succeq \mathbf{0}.$$

We will establish this fact.

Suppose $\mathbf{Y} \notin S_+^n$. Then there exists $\mathbf{q} \in \mathbb{R}^n$ with

$$\mathbf{q}^T \mathbf{Y} \mathbf{q} = \text{tr}(\mathbf{q} \mathbf{q}^T \mathbf{Y}) < 0.$$

Hence the positive semidefinite matrix $\mathbf{X} = \mathbf{q} \mathbf{q}^T$ satisfies $\text{tr}(\mathbf{XY}) < 0$, it follows that $\mathbf{Y} \notin S_+^n$.

Now suppose $\mathbf{X}, \mathbf{Y} \in S_+^n$. We can express \mathbf{X} in terms of its eigenvalue decomposition as $\mathbf{X} = \sum_{i=1}^n \lambda_i \mathbf{q}_i \mathbf{q}_i^T$, where (the eigenvalues) $\lambda_i \geq 0, i = 1, \dots, n$. Then we have

$$\text{tr}(\mathbf{YX}) = \text{tr}\left(\mathbf{Y} \sum_{i=1}^n \lambda_i \mathbf{q}_i \mathbf{q}_i^T\right) = \sum_{i=1}^n \lambda_i \mathbf{q}_i^T \mathbf{Y} \mathbf{q}_i \geq 0.$$

This shows that $\mathbf{Y} \in (S_+^n)^*$.

Example 134 (Dual of a norm cone). Let $\|\cdot\|$ be a norm on \mathbb{R}^n . The dual of the associated cone $K = \{(\mathbf{x}, t) \in \mathbb{R}^{n+1} \mid \|\mathbf{x}\| \leq t\}$ is the cone defined by the dual norm, i.e.,

$$K^* = \{(\mathbf{u}, v) \in \mathbb{R}^{n+1} \mid \|\mathbf{u}\|^* \leq v\},$$

where the dual norm is given by $\|\mathbf{u}\|^* = \sup\{\mathbf{u}^T \mathbf{x} \mid \|\mathbf{x}\| \leq 1\}$ (see Section 2.11.1.9).

To prove the result we have to show that

$$\mathbf{x}^T \mathbf{u} + tv \geq 0 \text{ whenever } \|\mathbf{x}\| \leq t \iff \|\mathbf{u}\|^* \leq v. \quad (3.21)$$

Let us start by showing that the righthand condition on (\mathbf{u}, v) implies the lefthand condition. Suppose $\|\mathbf{u}\|^* \leq v$, and $\|\mathbf{x}\| \leq t$ for some $t > 0$. (If $t = 0$, \mathbf{x} must be zero, so obviously $\mathbf{u}^T \mathbf{x} + vt \geq 0$.) Applying the definition of the dual norm, and the fact that $\|-\mathbf{x}/t\| \leq 1$, we have

$$\mathbf{u}^T (-\mathbf{x}/t) \leq \|\mathbf{u}\|^* \leq v,$$

and therefore $\mathbf{u}^T \mathbf{x} + vt \geq 0$. Next we show that the lefthand condition in (3.21) implies the righthand condition in (3.21). Suppose $\|\mathbf{u}\|^* > v$, i.e., that the righthand condition does not hold. Then by the definition of the dual norm, there exists an \mathbf{x} with $\|\mathbf{x}\| \leq 1$ and $\mathbf{x}^T \mathbf{u} > v$.

Taking $t = 1$, we have

$$\mathbf{u}^T(-\mathbf{x}) + v < 0,$$

which contradicts the lefthand condition in (3.21).

Dual cones satisfy several properties, such as:

- K^* is closed and convex.
- $K_1 \subseteq K_2$ implies $K_2^* \subseteq K_1^*$.
- If K has nonempty interior, then K^* is pointed.
- If the closure of K is pointed then K^* has nonempty interior.
- K^{**} is the closure of the convex hull of K . (Hence if K is convex and closed, $K^{**} = K$.) (See Exercise 168.)

These properties show that if K is a proper cone, then so is its dual K^* , and moreover, that $K^{**} = K$.

3.6.2 Dual generalized inequalities

Now suppose that the convex cone K is proper, so it induces a generalized inequality \preceq_K . Then its dual cone K^* is also proper, and therefore induces a generalized inequality. We refer to the generalized inequality \preceq_{K^*} as the dual of the generalized inequality \preceq_K . Some important properties relating a generalized inequality and its dual are:

- $\mathbf{x} \preceq_K \mathbf{y}$ if and only if $\boldsymbol{\lambda}^T \mathbf{x} \leq \boldsymbol{\lambda}^T \mathbf{y}$ for all $\boldsymbol{\lambda} \succeq_{K^*} \mathbf{0}$.
- $\mathbf{x} \prec_K \mathbf{y}$ if and only if $\boldsymbol{\lambda}^T \mathbf{x} < \boldsymbol{\lambda}^T \mathbf{y}$ for all $\boldsymbol{\lambda} \succeq_{K^*} \mathbf{0}, \boldsymbol{\lambda} \neq \mathbf{0}$.

Since $K = K^{**}$, the dual generalized inequality associated with \preceq_{K^*} is \preceq_K , so these properties hold if the generalized inequality and its dual are swapped. As a specific example, we have $\boldsymbol{\lambda} \preceq_{K^*} \boldsymbol{\mu}$ if and only if $\boldsymbol{\lambda}^T \mathbf{x} \leq \boldsymbol{\mu}^T \mathbf{x}$ for all $\mathbf{x} \succeq_K \mathbf{0}$.

Example 135 (Theorem of alternatives for linear strict generalized inequalities). Suppose $K \subseteq \mathbb{R}^m$ is a proper cone. Consider the strict generalized inequality

$$\mathbf{Ax} \prec_K \mathbf{b}, \quad (3.22)$$

where $\mathbf{x} \in \mathbb{R}^n$.

We will derive a theorem of alternatives for this inequality. Suppose it is infeasible, i.e., the affine set $\{\mathbf{b} - \mathbf{Ax} | \mathbf{x} \in \mathbb{R}^n\}$ does not intersect the open convex set $\text{int } K$. Then there is a separating hyperplane, i.e., a nonzero $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\mu \in \mathbb{R}$ such that $\boldsymbol{\lambda}^T(\mathbf{b} - \mathbf{Ax}) \leq \mu$ for all \mathbf{x} , and $\boldsymbol{\lambda}^T \mathbf{y} \geq \mu$ for all $\mathbf{y} \in \text{int } K$. The first condition implies $\mathbf{A}^T \boldsymbol{\lambda} = 0$ and $\boldsymbol{\lambda}^T \mathbf{b} \leq \mu$. The second condition implies $\boldsymbol{\lambda}^T \mathbf{y} \geq \mu$ for all $\mathbf{y} \in K$, which can only happen if $\boldsymbol{\lambda} \in K^*$ and $\mu \leq 0$.

Putting it all together we find that if (3.22) is infeasible, then there exists $\boldsymbol{\lambda}$ such that

$$\boldsymbol{\lambda} \neq \mathbf{0}, \boldsymbol{\lambda} \succeq_{K^*} \mathbf{0}, \mathbf{A}^T \boldsymbol{\lambda} = 0, \boldsymbol{\lambda}^T \mathbf{b} \leq 0. \quad (3.23)$$

Now we show the converse: if (3.23) holds, then the inequality system (3.22) cannot be feasible. Suppose that both inequality systems hold. Then we have $\boldsymbol{\lambda}^T(\mathbf{b} - \mathbf{Ax}) > 0$, since $\boldsymbol{\lambda} \neq \mathbf{0}$, $\boldsymbol{\lambda} \succeq_{K^*} \mathbf{0}$, and $\mathbf{b} - \mathbf{Ax} \succ_K \mathbf{0}$. But using $\mathbf{A}^T \boldsymbol{\lambda} = 0$ we find that $\boldsymbol{\lambda}^T(\mathbf{b} - \mathbf{Ax}) = \boldsymbol{\lambda}^T \mathbf{b} \leq 0$, which is a contradiction.

Thus, the inequality systems (3.22) and (3.23) are alternatives: for any data \mathbf{A} , \mathbf{b} , exactly one of them is feasible. (This generalizes the alternatives (3.18), (3.19) for the special case $K = \mathbb{R}_+^m$.)

3.6.3 Minimum and minimal elements via dual inequalities

We can use dual generalized inequalities to characterize minimum and minimal elements of a (possibly nonconvex) set $S \subseteq \mathbb{R}^m$ with respect to the generalized inequality induced by a proper cone K .

3.6.3.1 Dual characterization of minimum element

We first consider a characterization of the minimum element: \mathbf{x} is the minimum element of S , with respect to the generalized inequality \preceq_K , if and only if for all $\boldsymbol{\lambda} \succ_{K^*} \mathbf{0}$, \mathbf{x} is the unique minimizer of $\boldsymbol{\lambda}^T \mathbf{z}$ over $\mathbf{z} \in S$. Geometrically, this means that for any $\boldsymbol{\lambda} \succ_{K^*} \mathbf{0}$, the hyperplane

$$\{\mathbf{z} | \boldsymbol{\lambda}^T(\mathbf{z} - \mathbf{x}) = 0\}$$

is a strict supporting hyperplane to S at \mathbf{x} . (By strict supporting hyperplane, we mean that the hyperplane intersects S only at the point \mathbf{x} .) Note that convexity of the set S is not required. This is illustrated in Figure 3.23.

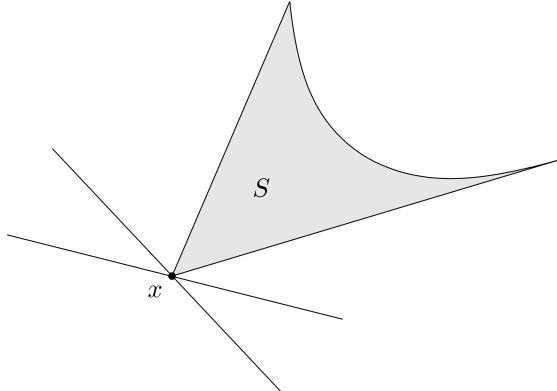


图 3.23: Dual characterization of minimum element. The point \mathbf{x} is the minimum element of the set S with respect to \mathbb{R}_+^2 . This is equivalent to: for every $\boldsymbol{\lambda} \succ \mathbf{0}$, the hyperplane $\{\mathbf{z} | \boldsymbol{\lambda}^T(\mathbf{z} - \mathbf{x}) = 0\}$ strictly supports S at \mathbf{x} , i.e., contains S on one side, and touches it only at \mathbf{x} .

To show this result, suppose \mathbf{x} is the minimum element of S , i.e., $\mathbf{x} \prec_K \mathbf{z}$ for all $\mathbf{z} \in S$, and let $\boldsymbol{\lambda} \succ_{K^*} \mathbf{0}$. Let $\mathbf{z} \in S$, $\mathbf{z} \neq \mathbf{x}$. Since \mathbf{x} is the minimum element of S , we have $\mathbf{z} - \mathbf{x} \succeq_K \mathbf{0}$. From $\boldsymbol{\lambda} \succ_{K^*} \mathbf{0}$ and $\mathbf{z} - \mathbf{x} \succeq_K \mathbf{0}$, $\mathbf{z} - \mathbf{x} \neq \mathbf{0}$, we conclude $\boldsymbol{\lambda}^T(\mathbf{z} - \mathbf{x}) > 0$. Since \mathbf{z} is an arbitrary element of S , not equal to \mathbf{x} , this shows that \mathbf{x} is the unique minimizer of $\boldsymbol{\lambda}^T \mathbf{z}$ over $\mathbf{z} \in S$. Conversely, suppose that for all $\boldsymbol{\lambda} \succ_{K^*} \mathbf{0}$, \mathbf{x} is the unique minimizer of $\boldsymbol{\lambda}^T \mathbf{z}$ over $\mathbf{z} \in S$, but \mathbf{x} is not the minimum element of S . Then there exists $\mathbf{z} \in S$ with $\mathbf{z} \not\succeq_K \mathbf{x}$. Since $\mathbf{z} - \mathbf{x} \not\succeq_K \mathbf{0}$, there exists $\tilde{\boldsymbol{\lambda}} \succeq_{K^*} \mathbf{0}$ with $\tilde{\boldsymbol{\lambda}}^T(\mathbf{z} - \mathbf{x}) < 0$. Hence $\boldsymbol{\lambda}^T(\mathbf{z} - \mathbf{x}) < 0$ for $\boldsymbol{\lambda} \succ_{K^*} \mathbf{0}$ in the neighborhood of $\tilde{\boldsymbol{\lambda}}$. This contradicts the assumption that \mathbf{x} is the unique minimizer of $\boldsymbol{\lambda}^T \mathbf{z}$ over S .

3.6.3.2 Dual characterization of minimal elements

We now turn to a similar characterization of minimal elements. Here there is a gap between the necessary and sufficient conditions. If $\boldsymbol{\lambda} \succ_{K^*} \mathbf{0}$ and \mathbf{x} minimizes $\boldsymbol{\lambda}^T \mathbf{z}$ over $\mathbf{z} \in S$, then \mathbf{x} is minimal. This is illustrated in Figure 3.24.

To show this, suppose that $\boldsymbol{\lambda} \succ_{K^*} \mathbf{0}$, and \mathbf{x} minimizes $\boldsymbol{\lambda}^T \mathbf{z}$ over S , but \mathbf{x} is not minimal, i.e., there exists a $\mathbf{z} \in S$, $\mathbf{z} \neq \mathbf{x}$, and $\mathbf{z} \prec_K \mathbf{x}$. Then $\boldsymbol{\lambda}^T(\mathbf{x} - \mathbf{z}) > 0$, which contradicts our assumption that \mathbf{x} is the minimizer of $\boldsymbol{\lambda}^T \mathbf{z}$ over S .

The converse is in general false: a point \mathbf{x} can be minimal in S , but not a minimizer of $\boldsymbol{\lambda}^T \mathbf{z}$ over $\mathbf{z} \in S$, for any $\boldsymbol{\lambda}$, as shown in Figure 3.25. This figure suggests that convexity plays an important role in the converse, which is correct. Provided the set S is convex, we can say that for any minimal element \mathbf{x} there exists a nonzero $\boldsymbol{\lambda} \succeq_{K^*} \mathbf{0}$ such that \mathbf{x}

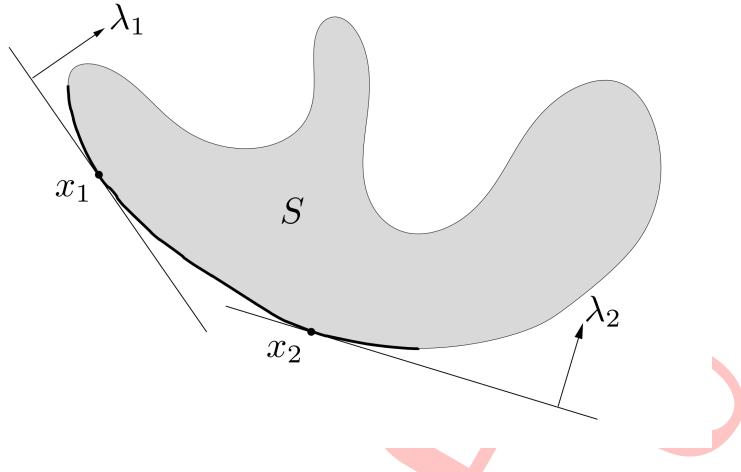


图 3.24: A set $S \subseteq \mathbb{R}^2$. Its set of minimal points, with respect to \mathbb{R}_+^2 , is shown as the darker section of its (lower, left) boundary. The minimizer of $\lambda_1^T z$ over S is x_1 , and is minimal since $\lambda_1 \succ \mathbf{0}$. The minimizer of $\lambda_2^T z$ over S is x_2 , which is another minimal point of S , since $\lambda_2 \succ \mathbf{0}$.

minimizes $\lambda^T z$ over $z \in S$.

To show this, suppose \mathbf{x} is minimal, which means that $((\mathbf{x} - K) \setminus \{\mathbf{x}\}) \cap S = \emptyset$. Applying the separating hyperplane theorem to the convex sets $(\mathbf{x} - K) \setminus \{\mathbf{x}\}$ and S , we conclude that there is a $\lambda \neq \mathbf{0}$ and μ such that $\lambda^T(\mathbf{x} - \mathbf{y}) \leq \mu$ for all $\mathbf{y} \in K$, and $\lambda^T z \geq \mu$ for all $z \in S$. From the first inequality we conclude $\lambda \succeq_{K^*} \mathbf{0}$. Since $\mathbf{x} \in S$ and $\mathbf{x} \in \mathbf{x} - K$, we have $\lambda^T \mathbf{x} = \mu$, so the second inequality implies that μ is the minimum value of $\lambda^T z$ over S . Therefore, \mathbf{x} is a minimizer of $\lambda^T z$ over S , where $\lambda \neq \mathbf{0}$, $\lambda \succeq_{K^*} \mathbf{0}$.

This converse theorem cannot be strengthened to $\lambda \succ_{K^*} \mathbf{0}$. Examples show that a point \mathbf{x} can be a minimal point of a convex set S , but not a minimizer of $\lambda^T z$ over $z \in S$ for any $\lambda \succ_{K^*} \mathbf{0}$. (See Figure 3.26, left.) Nor is it true that any minimizer of $\lambda^T z$ over $z \in S$, with $\lambda \succeq_{K^*} \mathbf{0}$, is minimal (see Figure 3.26, right.)

Example 136 (Pareto optimal production frontier). We consider a product which requires n resources (such as labor, electricity, natural gas, water) to manufacture. The product can be manufactured or produced in many ways. With each production method, we associate a resource vector $\mathbf{x} \in \mathbb{R}^n$, where \mathbf{x}_i denotes the amount of resource i consumed by the method to manufacture the product. We assume that $\mathbf{x}_i \geq 0$ (i.e., resources are consumed by the production methods) and that the resources are valuable (so using less of any resource is preferred).

The production set $P \subseteq \mathbb{R}^n$ is defined as the set of all resource vectors \mathbf{x} that corre-

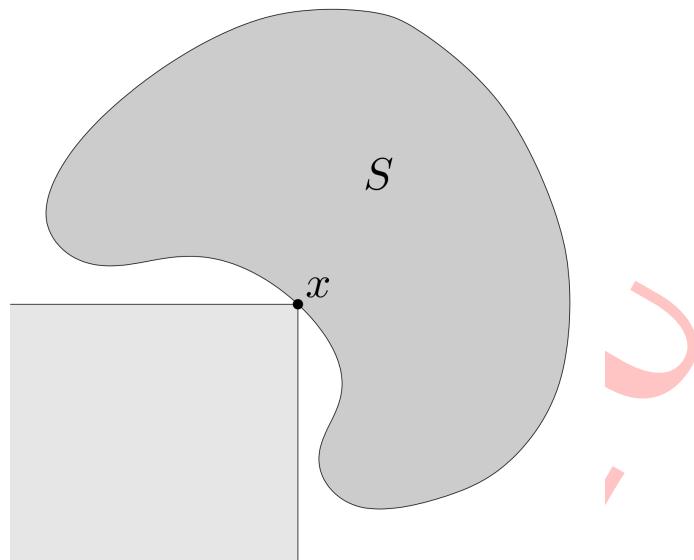


图 3.25: The point \mathbf{x} is a minimal element of $S \subseteq \mathbb{R}^2$ with respect to \mathbb{R}_+^2 . However there exists no $\boldsymbol{\lambda}$ for which \mathbf{x} minimizes $\boldsymbol{\lambda}^T \mathbf{z}$ over $\mathbf{z} \in S$.

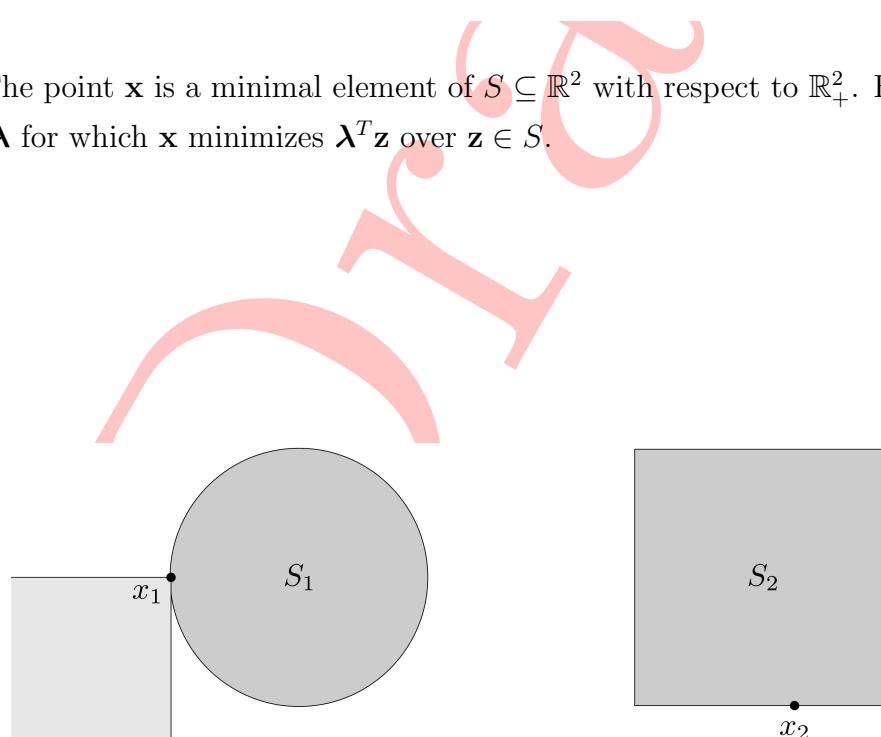


图 3.26: *Left.* The point $\mathbf{x}_1 \in S_1$ is minimal, but is not a minimizer of $\boldsymbol{\lambda}^T \mathbf{z}$ over S_1 for any $\boldsymbol{\lambda} \succ \mathbf{0}$. (It does, however, minimize $\boldsymbol{\lambda}^T \mathbf{z}$ over $\mathbf{z} \in S_1$ for $\boldsymbol{\lambda} = (1, 0)^T$.) *Right.* The point $\mathbf{x}_2 \in S_2$ is not minimal, but it does minimize $\boldsymbol{\lambda}^T \mathbf{z}$ over $\mathbf{z} \in S_2$ for $\boldsymbol{\lambda} = (0, 1)^T \succeq \mathbf{0}$.

spond to some production method.

Production methods with resource vectors that are minimal elements of P , with respect to componentwise inequality, are called Pareto optimal or efficient. The set of minimal elements of P is called the efficient production frontier.

We can give a simple interpretation of Pareto optimality. We say that one production method, with resource vector \mathbf{x} , is better than another, with resource vector \mathbf{y} , if $\mathbf{x}_i \leq \mathbf{y}_i$ for all i , and for some i , $\mathbf{x}_i < \mathbf{y}_i$. In other words, one production method is better than another if it uses no more of each resource than another method, and for at least one resource, actually uses less. This corresponds to $\mathbf{x} \preceq \mathbf{y}$, $\mathbf{x} \neq \mathbf{y}$. Then we can say: A production method is Pareto optimal or efficient if there is no better production method.

We can find Pareto optimal production methods (i.e., minimal resource vectors) by minimizing

$$\boldsymbol{\lambda}^T \mathbf{x} = \lambda_1 \mathbf{x}_1 + \dots + \lambda_n \mathbf{x}_n$$

over the set P of production vectors, using any $\boldsymbol{\lambda}$ that satisfies $\boldsymbol{\lambda} \succ \mathbf{0}$.

Here the vector $\boldsymbol{\lambda}$ has a simple interpretation: λ_i is the price of resource i . By minimizing $\boldsymbol{\lambda}^T \mathbf{x}$ over P we are finding the overall cheapest production method (for the resource prices $\boldsymbol{\lambda}_i$). As long as the prices are positive, the resulting production method is guaranteed to be efficient.

These ideas are illustrated in Figure 3.27.

3.7 Exercises

Definition of convexity

Exercise 137. Let $C \subseteq \mathbb{R}^n$ be a convex set, with $\mathbf{x}_1, \dots, \mathbf{x}_k \in C$, and let $\theta_1, \dots, \theta_k \in \mathbb{R}$ satisfy $\theta_i \geq 0$, $\theta_1 + \dots + \theta_k = 1$. Show that $\theta_1 \mathbf{x}_1 + \dots + \theta_k \mathbf{x}_k \in C$. (The definition of convexity is that this holds for $k = 2$, you must show it for arbitrary k .) Hint. Use induction on k .

Exercise 138. Show that a set is convex if and only if its intersection with any line is convex. Show that a set is affine if and only if its intersection with any line is affine.

Exercise 139 (Midpoint convexity). A set C is midpoint convex if whenever two points \mathbf{a}, \mathbf{b} are in C , the average or midpoint $(\mathbf{a} + \mathbf{b})/2$ is in C . Obviously a convex set is midpoint convex. It can be proved that under mild conditions midpoint convexity implies convexity. As a simple case, prove that if C is closed and midpoint convex, then C is convex.

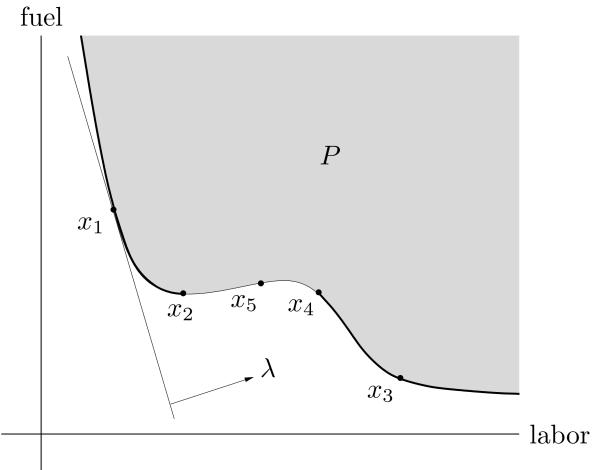


图 3.27: The production set P , for a product that requires labor and fuel to produce, is shown shaded. The two dark curves show the efficient production frontier. The points \mathbf{x}_1 , \mathbf{x}_2 and \mathbf{x}_3 are efficient. The points \mathbf{x}_4 and \mathbf{x}_5 are not (since in particular, \mathbf{x}_2 corresponds to a production method that uses no more fuel, and less labor). The point \mathbf{x}_1 is also the minimum cost production method for the price vector λ (which is positive). The point \mathbf{x}_2 is efficient, but cannot be found by minimizing the total cost $\lambda^T \mathbf{x}$ for any price vector $\lambda \succeq 0$.

Exercise 140. Show that the convex hull of a set S is the intersection of all convex sets that contain S . (The same method can be used to show that the conic, or affine, or linear hull of a set S is the intersection of all conic sets, or affine sets, or subspaces that contain S .)

Examples

Exercise 141. What is the distance between two parallel hyperplanes $\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{a}^T \mathbf{x} = b_1\}$ and $\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{a}^T \mathbf{x} = b_2\}$?

Exercise 142. When does one halfspace contain another? Give conditions under which

$$\{\mathbf{x} \mid \mathbf{a}^T \mathbf{x} \leq b\} \subseteq \{\mathbf{x} \mid \tilde{\mathbf{a}}^T \mathbf{x} \leq \tilde{b}\}$$

(where $\mathbf{a} \neq \mathbf{0}, \tilde{\mathbf{a}} \neq \mathbf{0}$). Also find the conditions under which the two halfspaces are equal.

Exercise 143 (Voronoi description of halfspace). Let a and b be distinct points in \mathbb{R}^n . Show that the set of all points that are closer (in Euclidean norm) to \mathbf{a} than \mathbf{b} , i.e., $\{\mathbf{x} \mid \|\mathbf{x} - \mathbf{a}\|_2 \leq \|\mathbf{x} - \mathbf{b}\|_2\}$, is a halfspace. Describe it explicitly as an inequality of the form $\mathbf{c}^T \mathbf{x} \leq d$. Draw a picture.

Exercise 144. Which of the following sets S are polyhedra? If possible, express S in the form $S = \{\mathbf{x} | \mathbf{Ax} \preceq \mathbf{b}, \mathbf{Fx} = \mathbf{g}\}$.

- (a) $S = \{y_1\mathbf{a}_1 + y_2\mathbf{a}_2 | -1 \leq y_1 \leq 1, -1 \leq y_2 \leq 1\}$, where $\mathbf{a}_1, \mathbf{a}_2 \in \mathbb{R}^n$.
- (b) $S = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{x} \succeq \mathbf{0}, \mathbf{1}^T \mathbf{x} = 1, \sum_{i=1}^n x_i a_i = b_1, \sum_{i=1}^n x_i a_i^2 = b_2\}$, where $a_1, \dots, a_n \in \mathbb{R}$ and $b_1, b_2 \in \mathbb{R}$.
- (c) $S = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{x} \succeq \mathbf{0}, \mathbf{x}^T \mathbf{y} \leq 1 \text{ for all } \mathbf{y} \text{ with } \|\mathbf{y}\|_2 = 1\}$.
- (d) $S = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{x} \succeq \mathbf{0}, \mathbf{x}^T \mathbf{y} \leq 1 \text{ for all } \mathbf{y} \text{ with } \sum_{i=1}^n |y_i| = 1\}$.

Exercise 145 (Voronoi sets and polyhedral decomposition). Let $\mathbf{x}_0, \dots, \mathbf{x}_K \in \mathbb{R}^n$. Consider the set of points that are closer (in Euclidean norm) to \mathbf{x}_0 than the other \mathbf{x}_i , i.e.,

$$V = \{\mathbf{x} \in \mathbb{R}^n | \|\mathbf{x} - \mathbf{x}_0\|_2 \leq \|\mathbf{x} - \mathbf{x}_i\|_2, i = 1, \dots, K\}.$$

V is called the Voronoi region around \mathbf{x}_0 with respect to $\mathbf{x}_1, \dots, \mathbf{x}_K$.

- (a) Show that V is a polyhedron. Express V in the form $V = \{\mathbf{x} | \mathbf{Ax} \preceq \mathbf{b}\}$.
- (b) Conversely, given a polyhedron P with nonempty interior, show how to find $\mathbf{x}_0, \dots, \mathbf{x}_K$ so that the polyhedron is the Voronoi region of \mathbf{x}_0 with respect to $\mathbf{x}_1, \dots, \mathbf{x}_K$.
- (c) We can also consider the sets

$$V_k = \{\mathbf{x} \in \mathbb{R}^n | \|\mathbf{x} - \mathbf{x}_k\|_2 \leq \|\mathbf{x} - \mathbf{x}_i\|_2, i \neq k\}.$$

The set V_k consists of points in \mathbb{R}^n for which the closest point in the set $\{\mathbf{x}_0, \dots, \mathbf{x}_K\}$ is \mathbf{x}_k . The sets V_0, \dots, V_K give a polyhedral decomposition of \mathbb{R}^n . More precisely, the sets V_k are polyhedra, $\bigcup_{k=0}^K V_k = \mathbb{R}^n$, and $\text{int } V_i \cap \text{int } V_j = \emptyset$, for $i \neq j$, i.e., V_i and V_j intersect at most along a boundary. Suppose that P_1, \dots, P_m are polyhedra such that $\bigcap_{i=1}^m P_i = \mathbb{R}^n$, and $\text{int } P_i \cap \text{int } P_j = \emptyset$, for $i \neq j$. Can this polyhedral decomposition of \mathbb{R}^n be described as the Voronoi regions generated by an appropriate set of points?

Exercise 146 (Solution set of a quadratic inequality). Let $C \subseteq \mathbb{R}^n$ be the solution set of a quadratic inequality,

$$C = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{x}^T \mathbf{Ax} + \mathbf{b}^T \mathbf{x} + c \leq 0\},$$

with $\mathbf{A} \in S^n$, $\mathbf{b} \in \mathbb{R}^n$, and $c \in \mathbb{R}$.

- (a) Show that C is convex if $\mathbf{A} \succeq \mathbf{0}$.

- (b) Show that the intersection of C and the hyperplane defined by $\mathbf{g}^T \mathbf{x} + h = 0$ (where $\mathbf{g} \neq \mathbf{0}$) is convex if $\mathbf{A} + \lambda \mathbf{g} \mathbf{g}^T \succeq \mathbf{0}$ for some $\lambda \in \mathbb{R}$.

Are the converses of these statements true?

Exercise 147 (Hyperbolic sets). Show that the hyperbolic set $\{\mathbf{x} \in \mathbb{R}_+^2 \mid \mathbf{x}_1 \mathbf{x}_2 \geq 1\}$ is convex. As a generalization, show that $\{\mathbf{x} \in \mathbb{R}_+^n \mid \prod_{i=1}^n \mathbf{x}_i \geq 1\}$ is convex. Hint. If $a, b \geq 0$ and $0 \leq \theta \leq 1$, then $a^\theta b^{1-\theta} \leq \theta a + (1-\theta)b$, see Section 4.1.10.

Exercise 148. Which of the following sets are convex?

- (a) A slab, i.e., a set of the form $\{\mathbf{x} \in \mathbb{R}^n \mid \alpha \leq \mathbf{a}^T \mathbf{x} \leq \beta\}$.
- (b) A rectangle, i.e., a set of the form $\{\mathbf{x} \in \mathbb{R}^n \mid \alpha_i \leq \mathbf{x}_i \leq \beta_i, i = 1, \dots, n\}$. A rectangle is sometimes called a hyperrectangle when $n > 2$.
- (c) A wedge, i.e., $\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{a}_1^T \mathbf{x} \leq b_1, \mathbf{a}_2^T \mathbf{x} \leq b_2\}$.
- (d) The set of points closer to a given point than a given set, i.e., $\{\mathbf{x} \mid \|\mathbf{x} - \mathbf{x}_0\|_2 \leq \|\mathbf{x} - \mathbf{y}\|_2 \text{ for all } \mathbf{y} \in S\}$ where $S \subseteq \mathbb{R}^n$.
- (e) The set of points closer to one set than another, i.e., $\{\mathbf{x} \mid \mathbf{dist}(\mathbf{x}, S) \leq \mathbf{dist}(\mathbf{x}, T)\}$, where $S, T \subseteq \mathbb{R}^n$, and

$$\mathbf{dist}(\mathbf{x}, S) = \inf\{\|\mathbf{x} - \mathbf{z}\|_2 \mid \mathbf{z} \in S\}.$$
- (f) The set $\{\mathbf{x} \mid \mathbf{x} + S_2 \subseteq S_1\}$, where $S_1, S_2 \subseteq \mathbb{R}^n$ with S_1 convex.
- (g) The set of points whose distance to \mathbf{a} does not exceed a fixed fraction θ of the distance to \mathbf{b} , i.e., the set $\{\mathbf{x} \mid \|\mathbf{x} - \mathbf{a}\|_2 \leq \theta \|\mathbf{x} - \mathbf{b}\|_2\}$. You can assume $\mathbf{a} \neq \mathbf{b}$ and $0 \leq \theta \leq 1$.

Exercise 149 (Conic hull of outer products). Consider the set of rank- k outer products, defined as $\{\mathbf{X} \mathbf{X}^T \mid \mathbf{X} \in \mathbb{R}^{n \times k}, \text{rank } \mathbf{X} = k\}$. Describe its conic hull in simple terms.

Exercise 150 (Expanded and restricted sets). Let $S \subseteq \mathbb{R}^n$, and let $\|\cdot\|$ be a norm on \mathbb{R}^n .

- (a) For $a \geq 0$ we define S_a as $\{\mathbf{x} \mid \mathbf{dist}(\mathbf{x}, S) \leq a\}$, where $\mathbf{dist}(\mathbf{x}, S) = \inf_{\mathbf{y} \in S} \|\mathbf{x} - \mathbf{y}\|$. We refer to S_a as S expanded or extended by a . Show that if S is convex, then S_a is convex.

(b) For $a \geq 0$ we define $S_{-a} = \{\mathbf{x} | B(\mathbf{x}, a) \subseteq S\}$, where $B(\mathbf{x}, a)$ is the ball (in the norm $\|\cdot\|$), centered at \mathbf{x} , with radius a . We refer to S_{-a} as S shrunk or restricted by a , since S_{-a} consists of all points that are at least a distance a from $\mathbb{R}^n \setminus S$. Show that if S is convex, then S_{-a} is convex.

Exercise 151 (Some sets of probability distributions). Let \mathbf{x} be a real-valued random variable with $\text{prob}(x = a_i) = p_i, i = 1, \dots, n$, where $a_1 < a_2 < \dots < a_n$. Of course $\mathbf{p} \in \mathbb{R}^n$ lies in the standard probability simplex $P = \{\mathbf{p} | \mathbf{1}^T \mathbf{p} = 1, \mathbf{p} \succeq \mathbf{0}\}$. Which of the following conditions are convex in \mathbf{p} ? (That is, for which of the following conditions is the set of $\mathbf{p} \in P$ that satisfy the condition convex?)

- (a) $\alpha \leq \mathbb{E}f(x) \leq \beta$, where $\mathbb{E}f(\mathbf{x})$ is the expected value of $f(x)$, i.e., $\mathbb{E}f(x) = \sum_{i=1}^n \mathbf{p}_i f(a_i)$.
(The function $f : \mathbb{R} \rightarrow \mathbb{R}$ is given.)
- (b) $\text{prob}(x > \alpha) \leq \beta$.
- (c) $\mathbb{E}|x^3| \leq \alpha \mathbb{E}|x|$.
- (d) $\mathbb{E}x^2 \leq \alpha$.
- (e) $\mathbb{E}x^2 \geq \alpha$.
- (f) $\text{var}(x) \leq \alpha$, where $\text{var}(x) = \mathbb{E}(x - \mathbb{E}x)^2$ is the variance of x .
- (g) $\text{var}(x) \geq \alpha$.
- (h) $\text{quartile}(x) \geq \alpha$, where $\text{quartile}(x) = \inf\{\beta | \text{prob}(x \leq \beta) \geq 0.25\}$.
- (i) $\text{quartile}(x) \leq \alpha$.

Operations that preserve convexity

Exercise 152. Show that if S_1 and S_2 are convex sets in $\mathbb{R}^{m \times n}$, then so is their partial sum

$$S = \{(\mathbf{x}, \mathbf{y}_1 + \mathbf{y}_2) | \mathbf{x} \in \mathbb{R}^m, \mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^n, (\mathbf{x}, \mathbf{y}_1) \in S_1, (\mathbf{x}, \mathbf{y}_2) \in S_2\}.$$

Exercise 153 (Image of polyhedral sets under perspective function). In this problem we study the image of hyperplanes, halfspaces, and polyhedra under the perspective function $P(\mathbf{x}, t) = \mathbf{x}/t$, with $\text{dom } P = \mathbb{R}^n \times \mathbb{R}_{++}$. For each of the following sets C , give a simple description of

$$P(C) = \{\mathbf{v}/t | (\mathbf{v}, t) \in C, t > 0\}.$$

- (a) The polyhedron $C = \text{conv}\{(\mathbf{v}_1, t_1), \dots, (\mathbf{v}_K, t_K)\}$ where $\mathbf{v}_i \in \mathbb{R}^n$ and $t_i > 0$.

(b) The hyperplane $C = \{(\mathbf{v}, t) | \mathbf{f}^T \mathbf{v} + gt = h\}$ (with \mathbf{f} and g not both zero).

(c) The halfspace $C = \{(\mathbf{v}, t) | \mathbf{f}^T \mathbf{v} + gt \leq h\}$ (with \mathbf{f} and g not both zero).

(d) The polyhedron $C = \{(\mathbf{v}, t) | \mathbf{F}\mathbf{v} + \mathbf{g}t \preceq \mathbf{h}\}$.

Exercise 154 (Invertible linear-fractional functions). Let $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be the linear-fractional function

$$f(\mathbf{x}) = (\mathbf{Ax} + \mathbf{b}) / (\mathbf{c}^T \mathbf{x} + d), \quad \text{dom } f = \{\mathbf{x} | \mathbf{c}^T \mathbf{x} + d > 0\}.$$

Suppose the matrix

$$\mathbf{Q} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix}$$

is nonsingular. Show that f is invertible and that f^{-1} is a linear-fractional mapping. Give an explicit expression for f^{-1} and its domain in terms of \mathbf{A} , \mathbf{b} , \mathbf{c} , and d . Hint. It may be easier to express f^{-1} in terms of \mathbf{Q} .

Exercise 155 (Linear-fractional functions and convex sets). Let $f : \mathbb{R}^m \rightarrow \mathbb{R}^n$ be the linear-fractional function

$$f(\mathbf{x}) = (\mathbf{Ax} + \mathbf{b}) / (\mathbf{c}^T \mathbf{x} + d), \quad \text{dom } f = \{\mathbf{x} | \mathbf{c}^T \mathbf{x} + d > 0\}.$$

In this problem we study the inverse image of a convex set C under f , i.e.,

$$f^{-1}(C) = \{\mathbf{x} \in \text{dom } f | f(\mathbf{x}) \in C\}.$$

For each of the following sets $C \subseteq \mathbb{R}^n$, give a simple description of $f^{-1}(C)$.

(a) The halfspace $C = \{\mathbf{y} | \mathbf{g}^T \mathbf{y} \leq h\}$ (with $\mathbf{g} \neq 0$).

(b) The polyhedron $C = \{\mathbf{y} | \mathbf{G}\mathbf{y} \leq \mathbf{h}\}$.

(c) The ellipsoid $\{\mathbf{y} | \mathbf{y}^T \mathbf{P}^{-1} \mathbf{y} \leq 1\}$ (where $\mathbf{P} \in \mathbb{S}_{++}^n$).

(d) The solution set of a linear matrix inequality, $C = \{\mathbf{y} | \mathbf{y}_1 \mathbf{A}_1 + \dots + \mathbf{y}_n \mathbf{A}_n \preceq \mathbf{B}\}$, where $\mathbf{A}_1, \dots, \mathbf{A}_n, \mathbf{B} \in \mathbb{S}^p$.

Separation theorems and supporting hyperplanes

Exercise 156 (Strictly positive solution of linear equations). Suppose $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$, with $\mathbf{b} \in \mathcal{R}(\mathbf{A})$. Show that there exists an \mathbf{x} satisfying

$$\mathbf{x} > \mathbf{0}, \quad \mathbf{Ax} = \mathbf{b}$$

if and only if there exists no λ with

$$\mathbf{A}^T \lambda \geq \mathbf{0}, \quad \mathbf{A}^T \lambda \neq \mathbf{0}, \quad \mathbf{b}^T \lambda \leq 0.$$

Hint. First prove the following fact from linear algebra: $\mathbf{c}^T \mathbf{x} = d$ for all \mathbf{x} satisfying $\mathbf{Ax} = \mathbf{b}$ if and only if there is a vector λ such that $\mathbf{c} = \mathbf{A}^T \lambda, d = \mathbf{b}^T \lambda$.

Exercise 157 (The set of separating hyperplanes). Suppose that C and D are disjoint subsets of \mathbb{R}^n . Consider the set of $(\mathbf{a}, b) \in \mathbb{R}^{n+1}$ for which $\mathbf{a}^T \mathbf{x} \leq b$ for all $\mathbf{x} \in C$, and $\mathbf{a}^T \mathbf{x} \geq b$ for all $\mathbf{x} \in D$. Show that this set is a convex cone (which is the singleton $\{\mathbf{0}\}$ if there is no hyperplane that separates C and D).

Exercise 158. Finish the proof of the separating hyperplane theorem in Section 3.5.1: Show that a separating hyperplane exists for two disjoint convex sets C and D . You can use the result proved in Section 3.5.1, i.e., that a separating hyperplane exists when there exist points in the two sets whose distance is equal to the distance between the two sets. Hint. If C and D are disjoint convex sets, then the set $\{\mathbf{x} - \mathbf{y} \mid \mathbf{x} \in C, \mathbf{y} \in D\}$ is convex and does not contain the origin.

Exercise 159. Give an example of two closed convex sets that are disjoint but cannot be strictly separated.

Exercise 160 (Supporting hyperplanes). (a) Express the closed convex set $\{\mathbf{x} \in \mathbb{R}_+^2 \mid \mathbf{x}_1 \mathbf{x}_2 \geq 1\}$ as an intersection of halfspaces.

(b) Let $C = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\|_\infty \leq 1\}$, the ℓ_∞ -norm unit ball in \mathbb{R}^n , and let \mathbf{x} be a point in the boundary of C . Identify the supporting hyperplanes of C at \mathbf{x} explicitly.

Exercise 161 (Inner and outer polyhedral approximations). Let $C \subseteq \mathbb{R}^n$ be a closed convex set, and suppose that $\mathbf{x}_1, \dots, \mathbf{x}_K$ are on the boundary of C . Suppose that for each i , $\mathbf{a}_i^T(\mathbf{x} - \mathbf{x}_i) = 0$ defines a supporting hyperplane for C at \mathbf{x}_i , i.e., $C \subseteq \{\mathbf{x} \mid \mathbf{a}_i^T(\mathbf{x} - \mathbf{x}_i) \leq 0\}$. Consider the two polyhedra

$$P_{inner} = \text{conv}\{\mathbf{x}_1, \dots, \mathbf{x}_K\}, \quad P_{outer} = \{\mathbf{x} \mid \mathbf{a}_i^T(\mathbf{x} - \mathbf{x}_i) \leq 0, i = 1, \dots, K\}.$$

Show that $P_{inner} \subseteq C \subseteq P_{outer}$. Draw a picture illustrating this.

Exercise 162 (Support function). The support function of a set $C \subseteq \mathbb{R}^n$ is defined as

$$S_C(\mathbf{y}) = \sup\{\mathbf{y}^T \mathbf{x} \mid \mathbf{x} \in C\}.$$

(We allow $S_C(\mathbf{y})$ to take on the value $+\infty$.) Suppose that C and D are closed convex sets in \mathbb{R}^n . Show that $C = D$ if and only if their support functions are equal.

Exercise 163 (Converse supporting hyperplane theorem). Suppose the set C is closed, has nonempty interior, and has a supporting hyperplane at every point in its boundary. Show that C is convex.

(Added by Zhouchen Lin)

Exercise 164 (Gordan's theorem). Let \mathbf{A} be an $m \times n$ matrix, then $\mathbf{Ax} < \mathbf{0}$ has a solution iff there does not exist a nonzero vector \mathbf{y} such that $\mathbf{y} \geq \mathbf{0}$ and $\mathbf{A}^T \mathbf{y} = \mathbf{0}$.

Convex cones and generalized inequalities

Exercise 165 (Positive semidefinite cone for $n = 1, 2, 3$). Give an explicit description of the positive semidefinite cone \mathbb{S}_+^n , in terms of the matrix coefficients and ordinary inequalities, for $n = 1, 2, 3$. To describe a general element of \mathbb{S}^n , for $n = 1, 2, 3$, use the notation

$$x_1, \begin{bmatrix} x_1 & x_2 \\ x_2 & x_3 \end{bmatrix}, \begin{bmatrix} x_1 & x_2 & x_3 \\ x_2 & x_4 & x_5 \\ x_3 & x_5 & x_6 \end{bmatrix}.$$

Exercise 166 (Cones in \mathbb{R}^2). Suppose $K \subseteq \mathbb{R}^2$ is a closed convex cone.

- (a) Give a simple description of K in terms of the polar coordinates of its elements ($\mathbf{x} = r(\cos \phi, \sin \phi)^T$ with $r \geq 0$).
- (b) Give a simple description of K^* , and draw a plot illustrating the relation between K and K^* .
- (c) When is K pointed?
- (d) When is K proper (hence, defines a generalized inequality)? Draw a plot illustrating what $\mathbf{x} \preceq_K \mathbf{y}$ means when K is proper.

Exercise 167 (Properties of generalized inequalities). Prove the properties of (nonstrict and strict) generalized inequalities listed in Section 3.4.1.

Exercise 168 (Properties of dual cones). Let K^* be the dual cone of a convex cone K , as defined in (3.20). Prove the following.

- (a) K^* is indeed a convex cone.
- (b) $K_1 \subseteq K_2$ implies $K_2^* \subseteq K_1^*$.
- (c) K^* is closed.

(d) The interior of K^* is given by $\text{int } K^* = \{\mathbf{y} \mid \mathbf{y}^T \mathbf{x} > 0 \text{ for all } \mathbf{x} \in K\}$.

(e) If K has nonempty interior then K^* is pointed.

(f) K^{**} is the closure of K . (Hence if K is closed, $K^{**} = K$.)

(g) If the closure of K is pointed then K^* has nonempty interior.

Exercise 169. Find the dual cone of $\{\mathbf{Ax} \mid \mathbf{x} \succeq \mathbf{0}\}$, where $\mathbf{A} \in \mathbb{R}^{m \times n}$.

Exercise 170 (The monotone nonnegative cone). We define the monotone nonnegative cone as

$$K_{m+} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{x}_1 \geq \mathbf{x}_2 \geq \dots \geq \mathbf{x}_n \geq 0\}.$$

i.e., all nonnegative vectors with components sorted in nonincreasing order.

(a) Show that K_{m+} is a proper cone.

(b) Find the dual cone K_{m+}^* . Hint. Use the identity

$$\begin{aligned} \sum_{i=1}^n \mathbf{x}_i \mathbf{y}_i = & (\mathbf{x}_1 - \mathbf{x}_2) \mathbf{y}_1 + (\mathbf{x}_2 - \mathbf{x}_3) (\mathbf{y}_1 + \mathbf{y}_2) + \\ & (\mathbf{x}_3 - \mathbf{x}_4) (\mathbf{y}_1 + \mathbf{y}_2 + \mathbf{y}_3) + \dots + \\ & (\mathbf{x}_{n-1} - \mathbf{x}_n) (\mathbf{y}_1 + \dots + \mathbf{y}_{n-1}) + \\ & \mathbf{x}_n (\mathbf{y}_1 + \dots + \mathbf{y}_n). \end{aligned}$$

Exercise 171 (The lexicographic cone and ordering). The lexicographic cone is defined as

$$K_{lex} = \{\mathbf{0}\} \cup \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{x}_1 = \dots = \mathbf{x}_k = 0, \mathbf{x}_{k+1} > 0, \text{ for some } k, 0 \leq k < n\},$$

i.e., all vectors whose first nonzero coefficient (if any) is positive.

(a) Verify that K_{lex} is a cone, but not a proper cone.

(b) We define the lexicographic ordering on \mathbb{R}^n as follows: $\mathbf{x} \leq_{lex} \mathbf{y}$ if and only if $\mathbf{y} - \mathbf{x} \in K_{lex}$. (Since K_{lex} is not a proper cone, the lexicographic ordering is not a generalized inequality.) Show that the lexicographic ordering is a linear ordering. for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, either $\mathbf{x} \leq_{lex} \mathbf{y}$ or $\mathbf{y} \leq_{lex} \mathbf{x}$. Therefore any set of vectors can be sorted with respect to the lexicographic cone, which yields the familiar sorting used in dictionaries.

(c) Find K_{lex}^* .

Exercise 172 (Copositive matrices). A matrix $\mathbf{X} \in \mathbb{S}^n$ is called copositive if $\mathbf{z}^T \mathbf{X} \mathbf{z} \geq 0$ for all $\mathbf{z} \geq \mathbf{0}$. Verify that the set of copositive matrices is a proper cone. Find its dual cone.

Exercise 173 (Euclidean distance matrices). Let $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^k$. The matrix $\mathbf{D} \in \mathbb{S}^n$ defined by $D_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2^2$ is called a Euclidean distance matrix. It satisfies some obvious properties such as $D_{ij} = D_{ji}$, $D_{ii} = 0$, $D_{ij} \geq 0$, and (from the triangle inequality) $D_{ik}^{1/2} \leq D_{ij}^{1/2} + D_{jk}^{1/2}$. We now pose the question: When is a matrix $\mathbf{D} \in \mathbb{S}^n$ a Euclidean distance matrix (for some points in \mathbb{R}^k , for some k)? A famous result answers this question: $\mathbf{D} \in \mathbb{S}^n$ is a Euclidean distance matrix if and only if $D_{ii} = 0$ and $\mathbf{x}^T \mathbf{D} \mathbf{x} \leq 0$ for all \mathbf{x} with $\mathbf{1}^T \mathbf{x} = 0$. Show that the set of Euclidean distance matrices is a convex cone. Find the dual cone.

Exercise 174 (Nonnegative polynomials and Hankel LMIs). Let K_{pol} be the set of (coefficients of) nonnegative polynomials of degree $2k$ on \mathbb{R} :

$$K_{pol} = \{\mathbf{x} \in \mathbb{R}^{2k+1} \mid \mathbf{x}_1 + \mathbf{x}_2 t + \mathbf{x}_3 t^2 + \dots + \mathbf{x}_{2k+1} t^{2k} \geq 0 \text{ for all } t \in \mathbb{R}\}.$$

(a) Show that K_{pol} is a proper cone.

(b) A basic result states that a polynomial of degree $2k$ is nonnegative on \mathbb{R} if and only if it can be expressed as the sum of squares of two polynomials of degree k or less. In other words, $\mathbf{x} \in K_{pol}$ if and only if the polynomial

$$p(t) = x_1 + x_2 t + x_3 t^2 + \dots + x_{2k+1} t^{2k}$$

can be expressed as

$$p(t) = r(t)^2 + s(t)^2,$$

where r and s are polynomials of degree k . Use this result to show that

$$K_{pol} = \left\{ \mathbf{x} \in \mathbb{R}^{2k+1} \mid \mathbf{x}_i = \sum_{m+n=i+1} \mathbf{Y}_{mn} \text{ for some } \mathbf{Y} \in \mathbb{S}_+^{k+1} \right\}.$$

In other words, $p(t) = x_1 + x_2 t + x_3 t^2 + \dots + x_{2k+1} t^{2k}$ is nonnegative if and only if there exists a matrix $\mathbf{Y} \in \mathbb{S}_+^{k+1}$ such that

$$x_1 = Y_{11}$$

$$x_2 = Y_{12} + Y_{21}$$

$$x_3 = Y_{13} + Y_{22} + Y_{31}$$

⋮

$$x_{2k+1} = Y_{k+1,k+1}.$$

(c) Show that $K_{pol}^* = K_{han}$ where $K_{han} = \{\mathbf{z} \in \mathbb{R}^{2k+1} | H(\mathbf{z}) \succeq \mathbf{0}\}$ and

$$H(\mathbf{z}) = \begin{bmatrix} z_1 & z_2 & z_3 & \cdots & z_k & z_{k+1} \\ z_2 & z_3 & z_4 & \cdots & z_{k+1} & z_{k+2} \\ z_3 & z_4 & z_5 & \cdots & z_{k+2} & z_{k+4} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ z_k & z_{k+1} & z_{k+2} & \cdots & z_{2k-1} & z_{2k} \\ z_{k+1} & z_{k+2} & z_{k+3} & \cdots & z_{2k} & z_{2k+1} \end{bmatrix}.$$

(This is the Hankel matrix with coefficients z_1, \dots, z_{2k+1} .)

(d) Let K_{mom} be the conic hull of the set of all vectors of the form $(1, t, t^2, \dots, t^{2k})$, where $t \in \mathbb{R}$. Show that $y \in K_{mom}$ if and only if $y_1 \geq 0$ and $y = y_1(1, Eu, Eu^2, \dots, Eu^{2k})$ for some random variable u . In other words, the elements of K_{mom} are nonnegative multiples of the moment vectors of all possible distributions on \mathbb{R} . Show that $K_{pol} = K_{mom}^*$.

(e) Combining the results of (c) and (d), conclude that $K_{han} = \text{cl } K_{mom}$. As an example illustrating the relation between K_{mom} and K_{han} , take $k = 2$ and $\mathbf{z} = (1, 0, 0, 0, 1)$. Show that $\mathbf{z} \in K_{han}, \mathbf{z} \notin K_{mom}$. Find an explicit sequence of points in K_{mom} which converge to \mathbf{z} .

Exercise 175 (Convex cones constructed from sets). (a) The barrier cone of a set C is defined as the set of all vectors \mathbf{y} such that $\mathbf{y}^T \mathbf{x}$ is bounded above over $\mathbf{x} \in C$. In other words, a nonzero vector \mathbf{y} is in the barrier cone if and only if it is the normal vector of a halfspace $\{\mathbf{x} | \mathbf{y}^T \mathbf{x} \leq \alpha\}$ that contains C . Verify that the barrier cone is a convex cone (with no assumptions on C).

(b) The recession cone (also called asymptotic cone) of a set C is defined as the set of all vectors \mathbf{y} such that for each $\mathbf{x} \in C$, $\mathbf{x} - t\mathbf{y} \in C$ for all $t \geq 0$. Show that the recession cone of a convex set is a convex cone. Show that if C is nonempty, closed, and convex, then the recession cone of C is the dual of the barrier cone.

(c) The normal cone of a set C at a boundary point \mathbf{x}_0 is the set of all vectors \mathbf{y} such that $\mathbf{y}^T(\mathbf{x} - \mathbf{x}_0) \leq 0$ for all $\mathbf{x} \in C$ (i.e., the set of vectors that define a supporting hyperplane to C at \mathbf{x}_0). Show that the normal cone is a convex cone (with no assumptions on C). Give a simple description of the normal cone of a polyhedron $\{\mathbf{x} | \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ at a point in its boundary.

Exercise 176 (Separation of cones). Let K and \tilde{K} be two convex cones whose interiors are nonempty and disjoint. Show that there is a nonzero \mathbf{y} such that $\mathbf{y} \in K^*, -\mathbf{y} \in \tilde{K}^*$.

第四章 Convex Functions

(Taken from Chapter 3 of [18])

4.1 Basic properties and examples

4.1.1 Definition

A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is *convex* if $\text{dom } f$ is a convex set and if for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$, and θ with $0 \leq \theta \leq 1$, we have

$$f(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \leq \theta f(\mathbf{x}) + (1 - \theta)f(\mathbf{y}). \quad (4.1)$$

Geometrically, this inequality means that the line segment between $(\mathbf{x}, f(\mathbf{x}))$ and $(\mathbf{y}, f(\mathbf{y}))$, which is the *chord* from \mathbf{x} to \mathbf{y} , lies above the graph of f (Figure 4.1). A function f is *strictly convex* if strict inequality holds in (4.1) whenever $\mathbf{x} \neq \mathbf{y}$ and $0 < \theta < 1$. We say f is *concave* if $-f$ is convex, and *strictly concave* if $-f$ is strictly convex.

For an affine function we always have equality in (4.1), so all affine (and therefore also linear) functions are both convex and concave. Conversely, any function that is convex and concave is affine.

A function is convex if and only if it is convex when restricted to any line that intersects its domain. In other words f is convex if and only if for all $\mathbf{x} \in \text{dom } f$ and all \mathbf{v} , the function $g(t) = f(\mathbf{x} + t\mathbf{v})$ is convex (on its domain, $\{t \mid \mathbf{x} + t\mathbf{v} \in \text{dom } f\}$). This property is very useful, since it allows us to check whether a function is convex by restricting it to a line.

The *analysis* of convex functions is a well developed field, which we will not pursue in any depth. One simple result, for example, is that a convex function is continuous on the relative interior of its domain; it can have discontinuities only on its relative boundary.

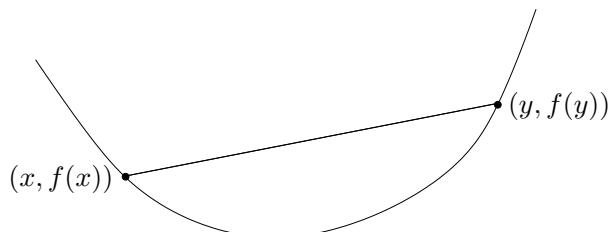


图 4.1: Graph of a convex function. The chord (*i.e.*, line segment) between any two points on the graph lies above the graph.

4.1.2 Extended-value extensions

It is often convenient to extend a convex function to all of \mathbb{R}^n by defining its value to be ∞ outside its domain. If f is convex we define its *extended-value extension* $\tilde{f} : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ by

$$\tilde{f}(\mathbf{x}) = \begin{cases} f(\mathbf{x}), & \mathbf{x} \in \text{dom } f \\ \infty, & \mathbf{x} \notin \text{dom } f. \end{cases}$$

The extension \tilde{f} is defined on all \mathbb{R}^n , and takes values in $\mathbb{R} \cup \{\infty\}$. We can recover the domain of the original function f from the extension \tilde{f} as $\text{dom } f = \{\mathbf{x} \mid \tilde{f}(\mathbf{x}) < \infty\}$.

The extension can simplify notation, since we do not need to explicitly describe the domain, or add the qualifier ‘for all $\mathbf{x} \in \text{dom } f$ ’ every time we refer to $f(\mathbf{x})$. Consider, for example, the basic defining inequality (4.1). In terms of the extension \tilde{f} , we can express it as: for $0 < \theta < 1$,

$$\tilde{f}(\theta \mathbf{x} + (1 - \theta) \mathbf{y}) \leq \theta \tilde{f}(\mathbf{x}) + (1 - \theta) \tilde{f}(\mathbf{y})$$

for *any* \mathbf{x} and \mathbf{y} . (For $\theta = 0$ or $\theta = 1$ the inequality always holds.) Of course here we must interpret the inequality using extended arithmetic and ordering. For \mathbf{x} and \mathbf{y} both in $\text{dom } f$, this inequality coincides with (4.1); if either is outside $\text{dom } f$, then the righthand side is ∞ , and the inequality therefore holds. As another example of this notational device, suppose f_1 and f_2 are two convex functions on \mathbb{R}^n . The pointwise sum $f = f_1 + f_2$ is the function with domain $\text{dom } f = \text{dom } f_1 \cap \text{dom } f_2$, with $f(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x})$ for any $\mathbf{x} \in \text{dom } f$. Using extended-value extensions we can simply say that for any \mathbf{x} , $\tilde{f}(\mathbf{x}) = \tilde{f}_1(\mathbf{x}) + \tilde{f}_2(\mathbf{x})$. In this equation the domain of f has been automatically defined as $\text{dom } f = \text{dom } f_1 \cap \text{dom } f_2$, since $\tilde{f}(\mathbf{x}) = \infty$ whenever $\mathbf{x} \notin \text{dom } f_1$ or $\mathbf{x} \notin \text{dom } f_2$. In this example we are relying on extended arithmetic to automatically define the domain.

In this book we will use the same symbol to denote a convex function and its extension, whenever there is no harm from the ambiguity. This is the same as assuming that all convex functions are implicitly extended, *i.e.*, are defined as ∞ outside their domains.

Example 177 (Indicator function of a convex set). *Let $C \subseteq \mathbb{R}^n$ be a convex set, and consider the (convex) function I_C with domain C and $I_C(\mathbf{x}) = 0$ for all $\mathbf{x} \in C$. In other words, the function is identically zero on the set C . Its extended-value extension is given by*

$$\tilde{I}_C(\mathbf{x}) = \begin{cases} 0, & \mathbf{x} \in C \\ \infty, & \mathbf{x} \notin C \end{cases}$$

The convex function \tilde{I}_C is called the indicator function of the set C .

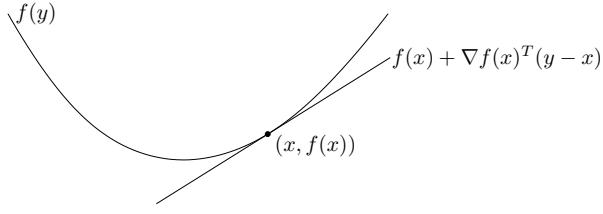


图 4.2: If f is convex and differentiable, then $f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) \leq f(\mathbf{y})$ for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$.

We can play several notational tricks with the indicator function \tilde{I}_C . For example the problem of minimizing a function f (defined on all of \mathbb{R}^n , say) on the set C is the same as minimizing the function $f + \tilde{I}_C$ over all of \mathbb{R}^n . Indeed, the function $f + \tilde{I}_C$ is (by our convention) f restricted to the set C .

In a similar way we can extend a concave function by defining it to be $-\infty$ outside its domain.

4.1.3 First-order conditions

Suppose f is differentiable (*i.e.*, its gradient ∇f exists at each point in $\text{dom } f$, which is open). Then f is convex if and only if $\text{dom } f$ is convex and

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \quad (4.2)$$

holds for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$. This inequality is illustrated in Figure 4.2.

The affine function of \mathbf{y} given by $f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$ is, of course, the first-order Taylor approximation of f near \mathbf{x} . The inequality (4.2) states that for a convex function, the first-order Taylor approximation is in fact a *global underestimator* of the function. Conversely, if the first-order Taylor approximation of a function is always a global underestimator of the function, then the function is convex.

The inequality (4.2) shows that from *local information* about a convex function (*i.e.*, its value and derivative at a point) we can derive *global information* (*i.e.*, a global underestimator of it). This is perhaps the most important property of convex functions, and explains some of the remarkable properties of convex functions and convex optimization problems. As one simple example, the inequality (4.2) shows that if $\nabla f(\mathbf{x}) = \mathbf{0}$, then for all $\mathbf{y} \in \text{dom } f$, $f(\mathbf{y}) \geq f(\mathbf{x})$, *i.e.*, \mathbf{x} is a global minimizer of the function f .

Strict convexity can also be characterized by a first-order condition: f is strictly convex if and only if $\text{dom } f$ is convex and for $\mathbf{x}, \mathbf{y} \in \text{dom } f$, $\mathbf{x} \neq \mathbf{y}$, we have

$$f(\mathbf{y}) > f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle. \quad (4.3)$$

For concave functions we have the corresponding characterization: f is concave if and only if $\text{dom } f$ is convex and

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$$

for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$.

Proof of first-order convexity condition

To prove (4.2), we first consider the case $n = 1$: We show that a differentiable function $f : \mathbb{R} \rightarrow \mathbb{R}$ is convex if and only if

$$f(y) \geq f(x) + f'(x)(y - x) \quad (4.4)$$

for all x and y in $\text{dom } f$.

Assume first that f is convex and $x, y \in \text{dom } f$. Since $\text{dom } f$ is convex (*i.e.*, an interval), we conclude that for all $0 < t \leq 1$, $x + t(y - x) \in \text{dom } f$, and by convexity of f ,

$$f(x + t(y - x)) \leq (1 - t)f(x) + tf(y).$$

If we divide both sides by t , we obtain

$$f(y) \geq f(x) + \frac{f(x + t(y - x)) - f(x)}{t},$$

and taking the limit as $t \rightarrow 0$ yields (4.4).

To show sufficiency, assume the function satisfies (4.4) for all x and y in $\text{dom } f$ (which is an interval). Choose any $x \neq y$, and $0 \leq \theta \leq 1$, and let $z = \theta x + (1 - \theta)y$. Applying (4.4) twice yields

$$f(x) \geq f(z) + f'(z)(x - z), \quad f(y) \geq f(z) + f'(z)(y - z).$$

Multiplying the first inequality by θ , the second by $1 - \theta$, and adding them yields

$$\theta f(x) + (1 - \theta)f(y) \geq f(z),$$

which proves that f is convex.

Now we can prove the general case, with $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and consider f restricted to the line passing through them, *i.e.*, the function defined by $g(t) = f(t\mathbf{y} + (1 - t)\mathbf{x})$, so $g'(t) = \langle \nabla f(t\mathbf{y} + (1 - t)\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$.

First assume f is convex, which implies g is convex, so by the argument above we have $g(1) \geq g(0) + g'(0)$, which means

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle.$$

Now assume that this inequality holds for any \mathbf{x} and \mathbf{y} , so if $t\mathbf{y} + (1-t)\mathbf{x} \in \text{dom } f$ and $\tilde{t}\mathbf{y} + (1-\tilde{t})\mathbf{x} \in \text{dom } f$, we have

$$f(t\mathbf{y} + (1-t)\mathbf{x}) \geq f(\tilde{t}\mathbf{y} + (1-\tilde{t})\mathbf{x}) + \langle \nabla f(\tilde{t}\mathbf{y} + (1-\tilde{t})\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle (t - \tilde{t}),$$

i.e., $g(t) \geq g(\tilde{t}) + g'(\tilde{t})(t - \tilde{t})$. We have seen that this implies that g is convex.

(A much more concise proof. Added by Zhouchen Lin)

Proof. If f is convex, then $f((1-\alpha)\mathbf{x} + \alpha\mathbf{y}) \leq (1-\alpha)f(\mathbf{x}) + \alpha f(\mathbf{y})$, which can be rewritten as

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \frac{f(\mathbf{x} + \alpha(\mathbf{y} - \mathbf{x})) - f(\mathbf{x})}{\alpha}, \quad \forall \alpha \in (0, 1].$$

Letting $\alpha \rightarrow 0^+$, we have (1). If (1) holds, we have

$$\begin{aligned} f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) &\leq f(\mathbf{x}) - (1-\alpha)\langle \nabla f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle, \\ f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) &\leq f(\mathbf{y}) + \alpha\langle \nabla f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle. \end{aligned}$$

Multiplying the first inequality with α and the second with $(1-\alpha)$ and adding them together, we obtain $f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha)f(\mathbf{y})$.

4.1.4 Second-order conditions

We now assume that f is twice differentiable, that is, its *Hessian* or second derivative $\nabla^2 f$ exists at each point in $\text{dom } f$, which is open. Then f is convex if and only if $\text{dom } f$ is convex and its Hessian is positive semidefinite: for all $\mathbf{x} \in \text{dom } f$,

$$\nabla^2 f(\mathbf{x}) \succeq \mathbf{0}.$$

For a function on \mathbb{R} , this reduces to the simple condition $f''(x) \geq 0$ (and $\text{dom } f$ convex, i.e., an interval), which means that the derivative is nondecreasing. The condition $\nabla^2 f(\mathbf{x}) \succeq \mathbf{0}$ can be interpreted geometrically as the requirement that the graph of the function have positive (upward) curvature at \mathbf{x} . We leave the proof of the second-order condition as an exercise (Exercise 273).

Similarly, f is concave if and only if $\text{dom } f$ is convex and $\nabla^2 f(\mathbf{x}) \preceq \mathbf{0}$ for all $\mathbf{x} \in \text{dom } f$. Strict convexity can be partially characterized by second-order conditions. If $\nabla^2 f(\mathbf{x}) \succ \mathbf{0}$ for all $\mathbf{x} \in \text{dom } f$, then f is strictly convex. The converse, however, is not true: for example, the function $f : \mathbb{R} \rightarrow \mathbb{R}$ given by $f(x) = x^4$ is strictly convex but has zero second derivative at $x = 0$.

Example 178 (Quadratic functions). Consider the quadratic function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, with $\text{dom } f = \mathbb{R}^n$, given by

$$f(\mathbf{x}) = (1/2)\mathbf{x}^T \mathbf{P} \mathbf{x} + \langle \mathbf{q}, \mathbf{x} \rangle + r,$$

with $\mathbf{P} \in \mathbb{S}^n$, $\mathbf{q} \in \mathbb{R}^n$, and $\mathbf{r} \in \mathbb{R}$. Since $\nabla^2 f(\mathbf{x}) = \mathbf{P}$ for all \mathbf{x} , f is convex if and only if $\mathbf{P} \succeq \mathbf{0}$ (and concave if and only if $\mathbf{P} \preceq \mathbf{0}$).

For quadratic functions, strict convexity is easily characterized: f is strictly convex if and only if $\mathbf{P} \succ \mathbf{0}$ (and strictly concave if and only if $\mathbf{P} \prec \mathbf{0}$).

Remark 179. The separate requirement that $\text{dom } f$ be convex cannot be dropped from the first- or second-order characterizations of convexity and concavity. For example, the function $f(x) = 1/x^2$, with $\text{dom } f = \{x \in \mathbb{R} \mid x \neq 0\}$, satisfies $f''(x) > 0$ for all $x \in \text{dom } f$, but is not a convex function.

4.1.5 Examples

We have already mentioned that all linear and affine functions are convex (and concave), and have described the convex and concave quadratic functions. In this section we give a few more examples of convex and concave functions. We start with some functions on \mathbb{R} , with variable x .

- *Exponential.* e^{ax} is convex on \mathbb{R} , for any $a \in \mathbb{R}$.
- *Powers.* x^a is convex on \mathbb{R}_{++} when $a \geq 1$ or $a \leq 0$, and concave for $0 \leq a \leq 1$.
- *Powers of absolute value.* $|x|^p$, for $p \geq 1$, is convex on \mathbb{R} .
- *Logarithm.* $\log x$ is concave on \mathbb{R}_{++} .
- *Negative entropy.* $x \log x$ (either on \mathbb{R}_{++} , or on \mathbb{R}_+ , defined as 0 for $x = 0$) is convex.

Convexity or concavity of these examples can be shown by verifying the basic inequality (4.1), or by checking that the second derivative is nonnegative or nonpositive. For example, with $f(x) = x \log x$ we have

$$f'(x) = \log x + 1, \quad f''(x) = 1/x,$$

so that $f''(x) > 0$ for $x > 0$. This shows that the negative entropy function is (strictly) convex.

We now give a few interesting examples of functions on \mathbb{R}^n .

- *Norms.* Every norm on \mathbb{R}^n is convex.
- *Max function.* $f(\mathbf{x}) = \max\{x_1, \dots, x_n\}$ is convex on \mathbb{R}^n .

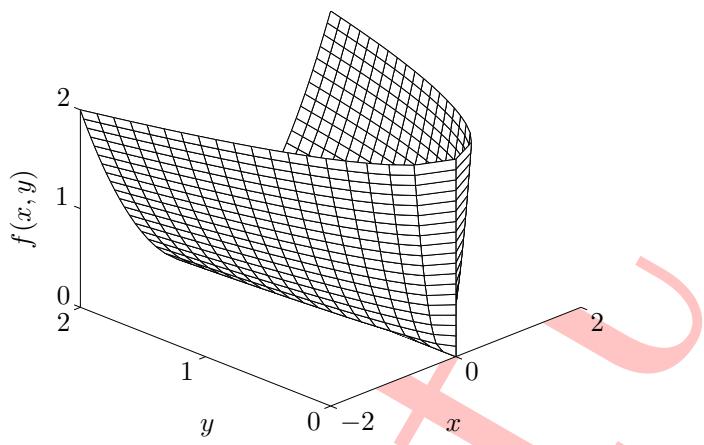


图 4.3: Graph of $f(x, y) = x^2/y$

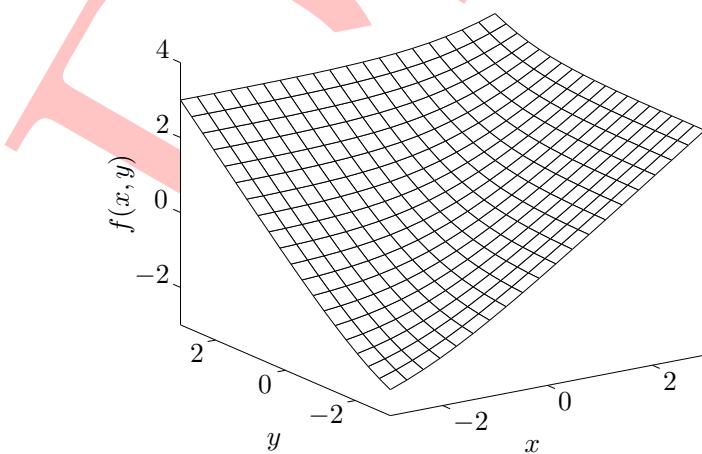


图 4.4: Graph of $f(x, y) = \log(e^x + e^y)$

- *Quadratic-over-linear function.* The function $f(x, y) = x^2/y$, with

$$\text{dom } f = \mathbb{R} \times \mathbb{R}_{++} = \{(x, y) \in \mathbb{R}^2 \mid y > 0\},$$

is convex (Figure 4.3).

- *Log-sum-exp.* The function $f(\mathbf{x}) = \log(e^{x_1} + \dots + e^{x_n})$ is convex on \mathbb{R}^n . This function can be interpreted as a differentiable (in fact, analytic) approximation of the max function, since

$$\max\{x_1, \dots, x_n\} \leq f(\mathbf{x}) \leq \max\{x_1, \dots, x_n\} + \log n$$

for all \mathbf{x} . (The second inequality is tight when all components of \mathbf{x} are equal.) Figure 4.4 shows f for $n = 2$.

- *Geometric mean.* The geometric mean $f(\mathbf{x}) = (\prod_{i=1}^n x_i)^{1/n}$ is concave on $\text{dom } f = \mathbb{R}_{++}^n$.
- *Log-determinant.* The function $f(\mathbf{X}) = \log \det \mathbf{X}$ is concave on $\text{dom } f = \mathbb{S}_{++}^n$.

Convexity (or concavity) of these examples can be verified in several ways, such as directly verifying the inequality (4.1), verifying that the Hessian is positive semidefinite, or restricting the function to an arbitrary line and verifying convexity of the resulting function of one variable.

Norms. If $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a norm, and $0 \leq \theta \leq 1$, then

$$f(\theta \mathbf{x} + (1 - \theta) \mathbf{y}) \leq f(\theta \mathbf{x}) + f((1 - \theta) \mathbf{y}) = \theta f(\mathbf{x}) + (1 - \theta) f(\mathbf{y}).$$

The inequality follows from the triangle inequality, and the equality follows from homogeneity of a norm.

Max function. The function $f(\mathbf{x}) = \max_i x_i$ satisfies, for $0 \leq \theta \leq 1$,

$$\begin{aligned} f(\theta \mathbf{x} + (1 - \theta) \mathbf{y}) &= \max_i (\theta x_i + (1 - \theta) y_i) \\ &\leq \theta \max_i x_i + (1 - \theta) \max_i y_i \\ &= \theta f(\mathbf{x}) + (1 - \theta) f(\mathbf{y}). \end{aligned}$$

Quadratic-over-linear function. To show that the quadratic-over-linear function $f(x, y) = x^2/y$ is convex, we note that (for $y > 0$),

$$\nabla^2 f(x, y) = \frac{2}{y^3} \begin{bmatrix} y^2 & -xy \\ -xy & x^2 \end{bmatrix} = \frac{2}{y^3} \begin{bmatrix} y \\ -x \end{bmatrix} \begin{bmatrix} y \\ -x \end{bmatrix}^T \succeq \mathbf{0}.$$

Log-sum-exp. The Hessian of the log-sum-exp function is

$$\nabla^2 f(\mathbf{x}) = \frac{1}{\langle \mathbf{1}, \mathbf{z} \rangle^2} (\langle \mathbf{1}, \mathbf{z} \rangle \operatorname{diag}(\mathbf{z}) - \mathbf{z} \mathbf{z}^T),$$

where $\mathbf{z} = (e^{x_1}, \dots, e^{x_n})$. To verify that $\nabla^2 f(\mathbf{x}) \succeq \mathbf{0}$ we must show that for all \mathbf{v} , $\mathbf{v}^T \nabla^2 f(\mathbf{x}) \mathbf{v} \geq 0$, i.e.,

$$\mathbf{v}^T \nabla^2 f(\mathbf{x}) \mathbf{v} = \frac{1}{\langle \mathbf{1}, \mathbf{z} \rangle^2} \left(\left(\sum_{i=1}^n z_i \right) \left(\sum_{i=1}^n v_i^2 z_i \right) - \left(\sum_{i=1}^n v_i z_i \right)^2 \right) \geq 0.$$

But this follows from the Cauchy-Schwarz inequality $\langle \mathbf{a}, \mathbf{a} \rangle \cdot \langle \mathbf{b}, \mathbf{b} \rangle \geq \langle \mathbf{a}, \mathbf{b} \rangle^2$ applied to the vectors with components $a_i = v_i \sqrt{z_i}$, $b_i = \sqrt{z_i}$.

Geometric mean. In a similar way we can show that the geometric mean $f(\mathbf{x}) = (\prod_{i=1}^n x_i)^{1/n}$ is concave on $\operatorname{dom} f = \mathbb{R}_{++}^n$. Its Hessian $\nabla^2 f(\mathbf{x})$ is given by

$$\begin{aligned} \frac{\partial^2 f(\mathbf{x})}{\partial x_k^2} &= -(n-1) \frac{(\prod_{i=1}^n x_i)^{1/n}}{n^2 x_k^2}, \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_k \partial x_l} &= \frac{(\prod_{i=1}^n x_i)^{1/n}}{n^2 x_k x_l} \quad \text{for } k \neq l, \end{aligned}$$

and can be expressed as

$$\nabla^2 f(\mathbf{x}) = -\frac{\prod_{i=1}^n x_i^{1/n}}{n^2} \left(n \operatorname{diag}(1/x_1^2, \dots, 1/x_n^2) - \mathbf{q} \mathbf{q}^T \right),$$

where $q_i = 1/x_i$. We must show that $\nabla^2 f(\mathbf{x}) \preceq \mathbf{0}$, i.e., that

$$\mathbf{v}^T \nabla^2 f(\mathbf{x}) \mathbf{v} = -\frac{\prod_{i=1}^n x_i^{1/n}}{n^2} \left(n \sum_{i=1}^n \frac{v_i^2}{x_i^2} - \left(\sum_{i=1}^n \frac{v_i}{x_i} \right)^2 \right) \leq 0$$

for all \mathbf{v} . Again this follows from the Cauchy-Schwarz inequality $\langle \mathbf{a}, \mathbf{a} \rangle \cdot \langle \mathbf{b}, \mathbf{b} \rangle \geq \langle \mathbf{a}, \mathbf{b} \rangle^2$, applied to the vectors $\mathbf{a} = \mathbf{1}$ and $b_i = v_i/x_i$.

Log-determinant. For the function $f(\mathbf{X}) = \log \det \mathbf{X}$, we can verify concavity by considering an arbitrary line, given by $\mathbf{X} = \mathbf{Z} + t\mathbf{V}$, where $\mathbf{Z}, \mathbf{V} \in \mathbb{S}^n$. We define $g(t) = f(\mathbf{Z} + t\mathbf{V})$, and restrict g to the interval of values of t for which $\mathbf{Z} + t\mathbf{V} \succeq \mathbf{0}$. Without loss of generality, we can assume that $t = 0$ is inside this interval, i.e., $\mathbf{Z} \succeq \mathbf{0}$. We have

$$\begin{aligned} g(t) &= \log \det(\mathbf{Z} + t\mathbf{V}) \\ &= \log \det(\mathbf{Z}^{1/2} (\mathbf{I} + t\mathbf{Z}^{-1/2} \mathbf{V} \mathbf{Z}^{-1/2}) \mathbf{Z}^{1/2}) \\ &= \sum_{i=1}^n \log(1 + t\lambda_i) + \log \det \mathbf{Z} \end{aligned}$$

where $\lambda_1, \dots, \lambda_n$ are the eigenvalues of $\mathbf{Z}^{-1/2}\mathbf{V}\mathbf{Z}^{-1/2}$. Therefore we have

$$g'(t) = \sum_{i=1}^n \frac{\lambda_i}{1+t\lambda_i}, \quad g''(t) = -\sum_{i=1}^n \frac{\lambda_i^2}{(1+t\lambda_i)^2}.$$

Since $g''(t) \leq 0$, we conclude that f is concave.

4.1.6 Sublevel sets

The α -sublevel set of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is defined as

$$C_\alpha = \{\mathbf{x} \in \text{dom } f \mid f(\mathbf{x}) \leq \alpha\}.$$

Sublevel sets of a convex function are convex, for any value of α . The proof is immediate from the definition of convexity: if $\mathbf{x}, \mathbf{y} \in C_\alpha$, then $f(\mathbf{x}) \leq \alpha$ and $f(\mathbf{y}) \leq \alpha$, and so $f(\theta\mathbf{x} + (1-\theta)\mathbf{y}) \leq \alpha$ for $0 \leq \theta \leq 1$, and hence $\theta\mathbf{x} + (1-\theta)\mathbf{y} \in C_\alpha$.

The converse is not true: a function can have all its sublevel sets convex, but not be a convex function. For example, $f(x) = -e^x$ is not convex on \mathbb{R} (indeed, it is strictly concave) but all its sublevel sets are convex.

If f is concave, then its α -sublevel set, given by $\{\mathbf{x} \in \text{dom } f \mid f(\mathbf{x}) \geq \alpha\}$, is a convex set. The sublevel set property is often a good way to establish convexity of a set, by expressing it as a sublevel set of a convex function, or as the superlevel set of a concave function.

Example 180. The geometric and arithmetic means of $\mathbf{x} \in \mathbb{R}_+^n$ are, respectively,

$$G(\mathbf{x}) = \left(\prod_{i=1}^n x_i \right)^{1/n}, \quad A(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n x_i,$$

(where we take $0^{1/n} = 0$ in our definition of G). The arithmetic-geometric mean inequality states that $G(\mathbf{x}) \leq A(\mathbf{x})$.

Suppose $0 \leq \alpha \leq 1$, and consider the set

$$\{\mathbf{x} \in \mathbb{R}_+^n \mid G(\mathbf{x}) \geq \alpha A(\mathbf{x})\},$$

i.e., the set of vectors with geometric mean at least as large as a factor α times the arithmetic mean. This set is convex, since it is the 0-superlevel set of the function $G(\mathbf{x}) - \alpha A(\mathbf{x})$, which is concave. In fact, the set is positively homogeneous, so it is a convex cone.

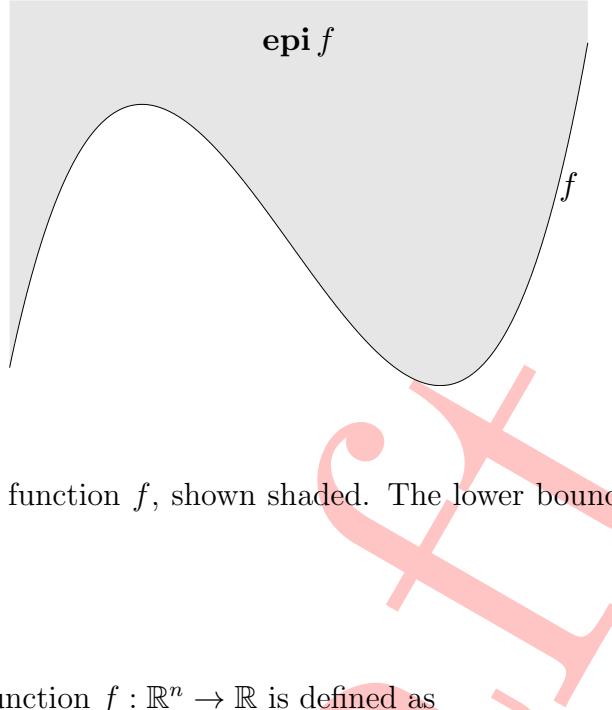


图 4.5: Epigraph of a function f , shown shaded. The lower boundary, shown darker, is the graph of f .

4.1.7 Epigraph

The graph of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is defined as

$$\{(\mathbf{x}, f(\mathbf{x})) \mid \mathbf{x} \in \text{dom } f\},$$

which is a subset of \mathbb{R}^{n+1} . The *epigraph* of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is defined as

$$\text{epi } f = \{(\mathbf{x}, t) \mid \mathbf{x} \in \text{dom } f, f(\mathbf{x}) \leq t\},$$

which is a subset of \mathbb{R}^{n+1} . ('Epi' means 'above' so epigraph means 'above the graph'.) The definition is illustrated in Figure 4.5.

The link between convex sets and convex functions is via the epigraph: A function is convex if and only if its epigraph is a convex set. A function is concave if and only if its *hypograph*, defined as

$$\text{hypo } f = \{(\mathbf{x}, t) \mid t \leq f(\mathbf{x})\},$$

is a convex set.

Example 181 (Matrix fractional function). *The function $f : \mathbb{R}^n \times \mathbb{S}^n \rightarrow \mathbb{R}$, defined as*

$$f(\mathbf{x}, \mathbf{Y}) = \mathbf{x}^T \mathbf{Y}^{-1} \mathbf{x}$$

is convex on $\text{dom } f = \mathbb{R}^n \times \mathbb{S}_{++}^n$. (This generalizes the quadratic-over-linear function $f(x, y) = x^2/y$, with $\text{dom } f = \mathbb{R} \times \mathbb{R}_{++}$.)

One easy way to establish convexity of f is via its epigraph:

$$\begin{aligned} \text{epi } f &= \{(\mathbf{x}, \mathbf{Y}, t) \mid \mathbf{Y} \succ \mathbf{0}, \mathbf{x}^T \mathbf{Y}^{-1} \mathbf{x} \leq t\} \\ &= \left\{ (\mathbf{x}, \mathbf{Y}, t) \mid \begin{bmatrix} \mathbf{Y} & \mathbf{x} \\ \mathbf{x}^T & t \end{bmatrix} \succeq \mathbf{0}, \mathbf{Y} \succ \mathbf{0} \right\}, \end{aligned}$$

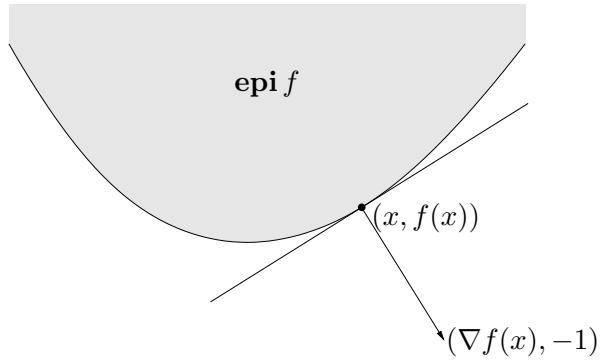


图 4.6: For a differentiable convex function f , the vector $(\nabla f(x), -1)$ defines a supporting hyperplane to the epigraph of f at x .

using the Schur complement condition for positive semidefiniteness of a block matrix (see Section 2.11.7.5). The last condition is a linear matrix inequality in $(\mathbf{x}, \mathbf{Y}, t)$, and therefore $\text{epi } f$ is convex.

For the special case $n = 1$, the matrix fractional function reduces to the quadratic-over-linear function x^2/y , and the associated LMI representation is

$$\begin{bmatrix} y & x \\ x & t \end{bmatrix} \succeq 0, \quad y > 0$$

(the graph of which is shown in Figure 4.3).

Many results for convex functions can be proved (or interpreted) geometrically using epigraphs, and applying results for convex sets. As an example, consider the first-order condition for convexity:

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle,$$

where f is convex and $\mathbf{x}, \mathbf{y} \in \text{dom } f$. We can interpret this basic inequality geometrically in terms of $\text{epi } f$. If $(\mathbf{y}, t) \in \text{epi } f$, then

$$t \geq f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle.$$

We can express this as:

$$(\mathbf{y}, t) \in \text{epi } f \implies \begin{bmatrix} \nabla f(\mathbf{x}) \\ -1 \end{bmatrix}^T \left(\begin{bmatrix} \mathbf{y} \\ t \end{bmatrix} - \begin{bmatrix} \mathbf{x} \\ f(\mathbf{x}) \end{bmatrix} \right) \leq 0.$$

This means that the hyperplane defined by $(\nabla f(\mathbf{x}), -1)$ supports $\text{epi } f$ at the boundary point $(\mathbf{x}, f(\mathbf{x}))$; see Figure 4.6.

4.1.8 Proper function

f is called *proper* if $f(\mathbf{x}) < \infty$ for at least one $\mathbf{x} \in \mathcal{X}$ and $f(\mathbf{x}) > -\infty$ for all $\mathbf{x} \in \mathcal{X}$, and we say that f is *improper* if it is not proper. In words, a function is proper if and only if its epigraph is nonempty and does not contain a vertical line.

4.1.9 Jensen's inequality and extensions

The basic inequality (4.1), *i.e.*,

$$f(\theta\mathbf{x} + (1-\theta)\mathbf{y}) \leq \theta f(\mathbf{x}) + (1-\theta)f(\mathbf{y}),$$

is sometimes called *Jensen's inequality*. It is easily extended to convex combinations of more than two points: If f is convex, $\mathbf{x}_1, \dots, \mathbf{x}_k \in \text{dom } f$, and $\theta_1, \dots, \theta_k \geq 0$ with $\theta_1 + \dots + \theta_k = 1$, then

$$f(\theta_1\mathbf{x}_1 + \dots + \theta_k\mathbf{x}_k) \leq \theta_1 f(\mathbf{x}_1) + \dots + \theta_k f(\mathbf{x}_k).$$

As in the case of convex sets, the inequality extends to infinite sums, integrals, and expected values. For example, if $p(\mathbf{x}) \geq 0$ on $S \subseteq \text{dom } f$, $\int_S p(\mathbf{x}) d\mathbf{x} = 1$, then

$$f\left(\int_S p(\mathbf{x})\mathbf{x} d\mathbf{x}\right) \leq \int_S f(\mathbf{x})p(\mathbf{x}) d\mathbf{x},$$

provided the integrals exist. In the most general case we can take any probability measure with support in $\text{dom } f$. If \mathbf{x} is a random variable such that $\mathbf{x} \in \text{dom } f$ with probability one, and f is convex, then we have

$$f(\mathbb{E} \mathbf{x}) \leq \mathbb{E} f(\mathbf{x}), \quad (4.5)$$

provided the expectations exist. We can recover the basic inequality (4.1) from this general form, by taking the random variable \mathbf{x} to have support $\{\mathbf{x}_1, \mathbf{x}_2\}$, with $\Pr(\mathbf{x} = \mathbf{x}_1) = \theta$, $\Pr(\mathbf{x} = \mathbf{x}_2) = 1 - \theta$. Thus the inequality (4.5) characterizes convexity: If f is not convex, there is a random variable \mathbf{x} , with $\mathbf{x} \in \text{dom } f$ with probability one, such that $f(\mathbb{E} \mathbf{x}) > \mathbb{E} f(\mathbf{x})$.

All of these inequalities are now called *Jensen's inequality*, even though the inequality studied by Jensen was the very simple one

$$f\left(\frac{x+y}{2}\right) \leq \frac{f(x) + f(y)}{2}.$$

Remark 182. We can interpret (4.5) as follows. Suppose $\mathbf{x} \in \text{dom } f \subseteq \mathbb{R}^n$ and \mathbf{z} is any zero mean random vector in \mathbb{R}^n . Then we have

$$\mathbb{E} f(\mathbf{x} + \mathbf{z}) \geq f(\mathbf{x}).$$

Thus, randomization or dithering (i.e., adding a zero mean random vector to the argument) cannot decrease the value of a convex function on average.

4.1.10 Inequalities

Many famous inequalities can be derived by applying Jensen's inequality to some appropriate convex function. (Indeed, convexity and Jensen's inequality can be made the foundation of a theory of inequalities.) As a simple example, consider the arithmetic-geometric mean inequality:

$$\sqrt{ab} \leq (a + b)/2 \quad (4.6)$$

for $a, b \geq 0$. The function $-\log x$ is convex; Jensen's inequality with $\theta = 1/2$ yields

$$-\log\left(\frac{a+b}{2}\right) \leq \frac{-\log a - \log b}{2}.$$

Taking the exponential of both sides yields (4.6).

As a less trivial example we prove Hölder's inequality: for $p, q > 1$, $1/p + 1/q = 1$, and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$,

$$\sum_{i=1}^n x_i y_i \leq \left(\sum_{i=1}^n |x_i|^p \right)^{1/p} \left(\sum_{i=1}^n |y_i|^q \right)^{1/q}.$$

By convexity of $-\log x$, and Jensen's inequality with general θ , we obtain the more general arithmetic-geometric mean inequality

$$a^\theta b^{1-\theta} \leq \theta a + (1 - \theta)b,$$

valid for $a, b \geq 0$ and $0 \leq \theta \leq 1$. Applying this with

$$a = \frac{|x_i|^p}{\sum_{j=1}^n |x_j|^p}, \quad b = \frac{|y_i|^q}{\sum_{j=1}^n |y_j|^q}, \quad \theta = 1/p,$$

yields

$$\left(\frac{|x_i|^p}{\sum_{j=1}^n |x_j|^p} \right)^{1/p} \left(\frac{|y_i|^q}{\sum_{j=1}^n |y_j|^q} \right)^{1/q} \leq \frac{|x_i|^p}{p \sum_{j=1}^n |x_j|^p} + \frac{|y_i|^q}{q \sum_{j=1}^n |y_j|^q}.$$

Summing over i then yields Hölder's inequality.

(Added by Zhouchen Lin)

4.1.11 Bregman distance

Given a differentiable strictly convex function $f : C \rightarrow \mathbb{R}$, where $C \subset \mathbb{R}^n$ is a convex set, the Bregman distance is defined as:

$$B_f(\mathbf{y}, \mathbf{x}) = f(\mathbf{y}) - f(\mathbf{x}) - \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle. \quad (4.7)$$

It is clear that $B_f(\mathbf{y}, \mathbf{x}) \geq 0$ for all $\mathbf{x}, \mathbf{y} \in C$ due to the convexity of f . However, the Bregman distance is not symmetric: $B_f(\mathbf{y}, \mathbf{x}) \neq B_f(\mathbf{x}, \mathbf{y})$.

Strong convex w.r.t. Bregman distance:

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} B_f(\mathbf{y}, \mathbf{x}).$$

Exercise 183. Does the Bregman distance satisfy the triangular inequality?

(Taken from Page 297 of [20])

Exercise 184. Prove the identity:

$$\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{z} \rangle = B_f(\mathbf{z}, \mathbf{x}) + B_f(\mathbf{x}, \mathbf{y}) - B_f(\mathbf{z}, \mathbf{y}). \quad (4.8)$$

(Taken from https://en.wikipedia.org/wiki/Bregman_divergence)

Exercise 185 (Convexity). $B_f(\mathbf{x}, \mathbf{y})$ is convex in \mathbf{x} but not necessarily in \mathbf{y} .

Exercise 186 (Linearity). If f_1 and f_2 are two differentiable strictly convex functions, and $\lambda \geq 0$, then

$$B_{f_1+\lambda f_2}(\mathbf{x}, \mathbf{y}) = B_{f_1}(\mathbf{x}, \mathbf{y}) + \lambda B_{f_2}(\mathbf{x}, \mathbf{y}).$$

Exercise 187 (Duality).

$$B_{f^*}(\mathbf{y}^*, \mathbf{x}^*) = B_f(\mathbf{x}, \mathbf{y}),$$

where f^* is the conjugate of f and $\mathbf{x}^* = \nabla f(\mathbf{x})$ and $\mathbf{y}^* = \nabla f(\mathbf{y})$ are the dual points corresponding to \mathbf{x} and \mathbf{y} . The above can be more concisely written as

$$B_f(\mathbf{x}, \mathbf{y}) = B_{f^*}(\nabla f(\mathbf{y}), \nabla f(\mathbf{x})).$$

(Added by Zhouchen Lin)

4.1.12 Subgradient

(Taken from Section 4.2 of [14])

The *subgradient* of a convex function f is defined as:

$$\partial f(\mathbf{x}) = \{\mathbf{g} | f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \mathbf{g}, \mathbf{y} - \mathbf{x} \rangle, \forall \mathbf{x}, \mathbf{y} \in \text{dom } f\}. \quad (4.9)$$

Subgradient is a generalization of gradient if f is differentiable. A subgradient admits an intuitive geometric interpretation: it can be identified with a non-vertical supporting hyperplane to the epigraph of f at $(\mathbf{x}, f(\mathbf{x}))$. Such a hyperplane provides a linear approximation to the function f , which is an underestimate of f . See Figure 4.7.

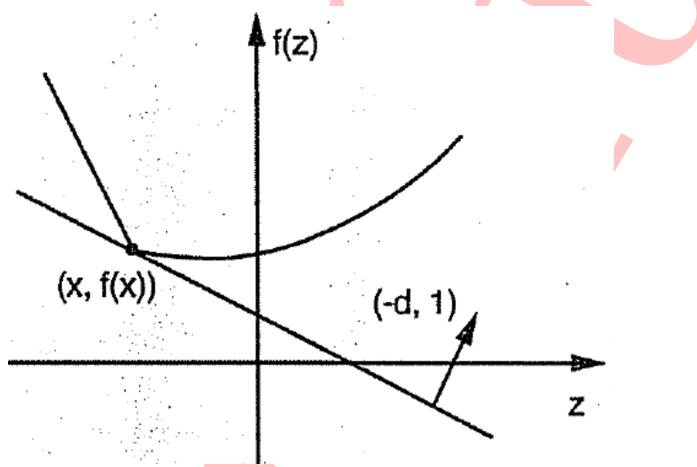


图 4.7: Illustration of a subgradient of a convex function f .

Proposition 188. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a proper convex function. The subgradient $\partial f(\mathbf{x})$ is nonempty, convex, and compact for all $\mathbf{x} \in (\text{dom } f)^\circ$.

Proof. Since f is convex, its epigraph $\text{epi } f$ is convex. Thus for every boundary point $\tilde{\mathbf{x}} = (\mathbf{x}^T, f(\mathbf{x}))^T$ of $\text{epi } f$ there is at least one supporting hyperplane $\tilde{\mathbf{g}}^T(\tilde{\mathbf{y}} - \tilde{\mathbf{x}}) = 0$ such that $\tilde{\mathbf{g}}^T(\tilde{\mathbf{y}} - \tilde{\mathbf{x}}) \geq 0$ for all $\tilde{\mathbf{y}} = (\mathbf{y}^T, t)^T \in \text{epi } f$, i.e., $t \geq f(\mathbf{y})$, where $\tilde{\mathbf{g}} = (\mathbf{g}^T, g_{n+1})^T$. Then

$$\mathbf{g}^T(\mathbf{y} - \mathbf{x}) + g_{n+1}(t - f(\mathbf{x})) \geq 0, \quad \forall t \geq f(\mathbf{y}), \mathbf{y} \in \text{dom } f. \quad (4.10)$$

Since $\mathbf{x} \in (\text{dom } f)^\circ$, the supporting hyperplane cannot be vertical. Thus $g_{n+1} \neq 0$. Then since t can be arbitrarily large, g_{n+1} must be positive so that (4.10) can hold. Then we can divide both sides of (4.10) with g_{n+1} and let $t = f(\mathbf{y})$ to obtain

$$\hat{\mathbf{g}}^T(\mathbf{y} - \mathbf{x}) + (f(\mathbf{y}) - f(\mathbf{x})) \geq 0, \quad \mathbf{y} \in \text{dom } f,$$

where $\hat{\mathbf{g}} = \mathbf{g}/g_{n+1}$, which means that $-\hat{\mathbf{g}} \in \partial f(\mathbf{x})$ and hence $\partial f(\mathbf{x}) \neq \emptyset$.

The convexity and compactness of $\partial f(\mathbf{x})$ is easy to prove. \square

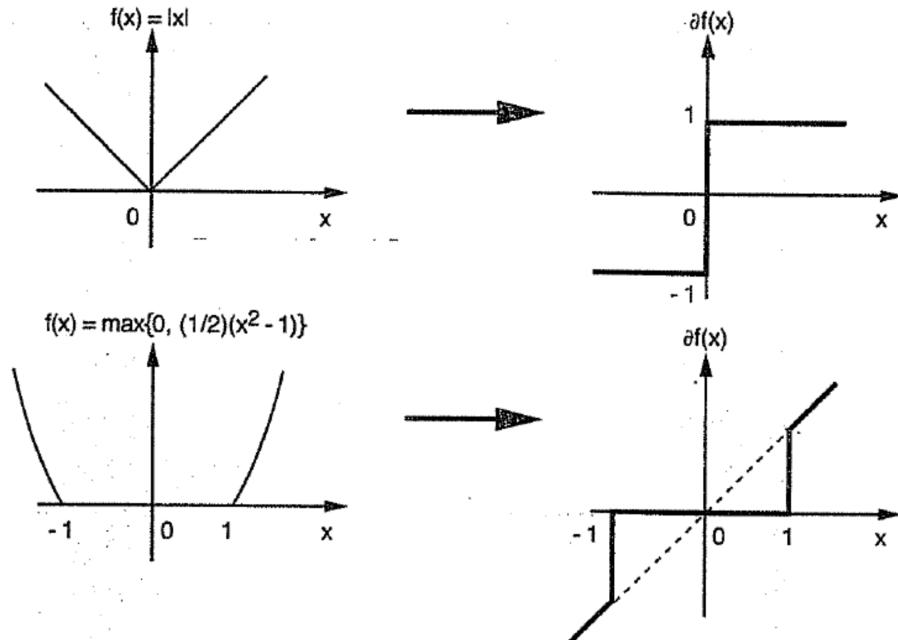


图 4.8: The subgradient of some scalar convex functions as a function of the argument x .

Note that $\partial f(\mathbf{x})$ may be empty when $\mathbf{x} \in \partial(\text{dom } f)$.

The directional derivative and the subgradient of a convex function are closely related. To see this, note that the subgradient inequality (4.9) is equivalent to

$$\frac{f(\mathbf{x} + \alpha\mathbf{y}) - f(\mathbf{x})}{\alpha} \geq \langle \mathbf{g}, \mathbf{y} \rangle, \quad \forall \mathbf{y} \in \mathbb{R}^n, \forall \alpha > 0.$$

Since the quotient on the left above decreases monotonically to $f'(\mathbf{x}; \mathbf{y})$, the directional derivative in the direction \mathbf{y} , as $\alpha \downarrow 0$, we conclude that the subgradient inequality (4.9) is equivalent to $f'(\mathbf{x}; \mathbf{y}) \geq \langle \mathbf{y}, \mathbf{g} \rangle$ for all $\mathbf{y} \in \mathbb{R}^n$. Therefore, we obtain

$$\mathbf{g} \in \partial f(\mathbf{x}) \iff f'(\mathbf{x}; \mathbf{y}) \geq \langle \mathbf{y}, \mathbf{g} \rangle, \quad \forall \mathbf{y} \in \mathbb{R}^n, \quad (4.11)$$

and it follows that

$$f'(\mathbf{x}; \mathbf{y}) \geq \max_{\mathbf{g} \in \partial f(\mathbf{x})} \langle \mathbf{y}, \mathbf{g} \rangle.$$

The following proposition shows that in fact equality holds in the above relation.

Proposition 189. *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a convex function. For every $\mathbf{x} \in \mathbb{R}^n$, we have*

$$f'(\mathbf{x}; \mathbf{y}) = \max_{\mathbf{g} \in \partial f(\mathbf{x})} \langle \mathbf{y}, \mathbf{g} \rangle, \quad \forall \mathbf{y} \in \mathbb{R}^n. \quad (4.12)$$

In particular, f is differentiable at \mathbf{x} with gradient $\nabla f(\mathbf{x})$ if and only if it has $\nabla f(\mathbf{x})$ as its unique subgradient at \mathbf{x} .

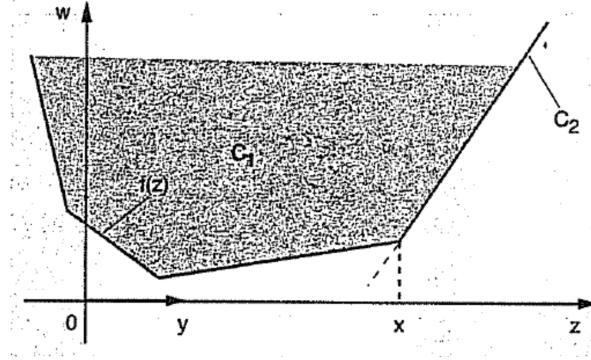


图 4.9: Illustration of the sets C_1 and C_2 used in the hyperplane separation argument of the proof of Proposition 189.

Proof. We have already shown that $f'(\mathbf{x}; \mathbf{y}) \geq \max_{\mathbf{g} \in \partial f(\mathbf{x})} \langle \mathbf{y}, \mathbf{g} \rangle$ for all $\mathbf{y} \in \mathbb{R}^n$. To show the reverse inequality, we take any \mathbf{x} and \mathbf{y} in \mathbb{R}^n and consider the subset of \mathbb{R}^{n+1}

$$C_1 = \{(\mathbf{z}, w) | f(\mathbf{z}) < w\},$$

and the half-line

$$C_2 = \{(\mathbf{z}, w) | \mathbf{z} = \mathbf{x} + \alpha \mathbf{y}, w = f(\mathbf{x}) + \alpha f'(\mathbf{x}; \mathbf{y}), \alpha \geq 0\};$$

see Figure 4.8. Using the definition of directional derivative and the convexity of f , it follows that these two sets are nonempty, convex and disjoint. Thus we can use the Separating Hyperplane Theorem to assert the existence of a nonzero vector $(\boldsymbol{\mu}, \gamma) \in \mathbb{R}^{n+1}$ such that

$$\gamma w + \langle \boldsymbol{\mu}, \mathbf{z} \rangle \geq \gamma(f(\mathbf{x}) + \alpha f'(\mathbf{x}; \mathbf{y})) + \langle \boldsymbol{\mu}, \mathbf{x} + \alpha \mathbf{y} \rangle, \quad \forall \alpha \geq 0, \mathbf{z} \in \mathbb{R}^n, w > f(\mathbf{z}). \quad (4.13)$$

We cannot have $\gamma < 0$ since then the left-hand side above could be made arbitrarily small by choosing w sufficiently large. Also if $\gamma = 0$, then (4.13) implies that $\boldsymbol{\mu} = \mathbf{0}$, which is a contradiction. Therefore, $\gamma > 0$ and by dividing with γ in (4.13) if necessary, we may assume that $\gamma = 1$, i.e.,

$$w + \langle \boldsymbol{\mu}, \mathbf{z} - \mathbf{x} \rangle \geq f(\mathbf{x}) + \alpha f'(\mathbf{x}; \mathbf{y}) + \alpha \langle \boldsymbol{\mu}, \mathbf{y} \rangle, \quad \forall \alpha \geq 0, \mathbf{z} \in \mathbb{R}^n, w > f(\mathbf{z}). \quad (4.14)$$

By setting $\alpha = 0$ in the above relation and by taking the limit as $w \downarrow f(\mathbf{z})$, we obtain

$$f(\mathbf{z}) \geq f(\mathbf{x}) - \langle \boldsymbol{\mu}, \mathbf{z} - \mathbf{x} \rangle, \quad \forall \mathbf{z} \in \mathbb{R}^n$$

implying that $-\boldsymbol{\mu} \in \partial f(\mathbf{x})$. By setting $\mathbf{z} = \mathbf{x}$ and $\alpha = 1$ in (4.14), and by taking the limit as $w \downarrow f(\mathbf{x})$, we obtain $-\langle \mathbf{y}, \boldsymbol{\mu} \rangle \geq f'(\mathbf{x}; \mathbf{y})$, which implies that

$$\max_{\mathbf{g} \in \partial f(\mathbf{x})} \langle \mathbf{y}, \mathbf{g} \rangle \geq f'(\mathbf{x}; \mathbf{y}),$$

and completes the proof of (4.12).

From the definition of directional derivative, we see that f is differentiable at \mathbf{x} if and only if the directional derivative $f'(\mathbf{x}; \mathbf{y})$ is a linear function of the form $f'(\mathbf{x}; \mathbf{y}) = \langle \nabla f(\mathbf{x}), \mathbf{y} \rangle$. Thus, from (4.12), f is differentiable at \mathbf{x} if and only if it has $\nabla f(\mathbf{x})$ as its unique subgradient at \mathbf{x} . \square

The following proposition provides some important boundedness and continuity properties of the subgradient.

Proposition 190. *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a convex function.*

(a) *If \mathcal{X} is a bounded set, then the set $\cup_{\mathbf{x} \in \mathcal{X}} \partial f(\mathbf{x})$ is bounded.*

(b) *If a sequence $\{\mathbf{x}_k\}$ converges to a vector $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$ for all k , then the sequence $\{\mathbf{g}_k\}$ is bounded and each of its accumulation points is a subgradient of f at \mathbf{x} .*

Proof. (a) Assume the contrary, i.e., that there exists a sequence $\{\mathbf{x}_k\} \subset \mathcal{X}$, and an unbounded sequence \mathbf{g}_k with $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$ for all k . Without loss of generality, we assume that $\mathbf{g}_k \neq \mathbf{0}$ for all k , and we denote $\mathbf{y}_k = \mathbf{g}_k / \|\mathbf{g}_k\|$. Since both \mathbf{x}_k and \mathbf{y}_k are bounded, they must contain convergent subsequences. We assume without loss of generality that $\{\mathbf{x}_k\}$ converges to some \mathbf{x} and $\{\mathbf{y}_k\}$ converges to some \mathbf{y} . Since $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$, we have

$$f(\mathbf{x}_k + \mathbf{y}_k) - f(\mathbf{x}_k) \geq \langle \mathbf{g}_k, \mathbf{y}_k \rangle = \|\mathbf{g}_k\|.$$

Since $\{\mathbf{x}_k\}$ and $\{\mathbf{y}_k\}$ converge, by the continuity of f the left-hand side above is bounded. This implied that the right-hand side is bounded, thereby contradicting the unboundedness of $\{\mathbf{g}_k\}$.

(b) By Proposition 189, we have

$$\langle \mathbf{y}, \mathbf{g}_k \rangle \leq f'(\mathbf{x}; \mathbf{y}), \quad \forall \mathbf{y} \in \mathbb{R}^n.$$

By part (a), the sequence $\{\mathbf{g}_k\}$ is bounded, so let \mathbf{g} be an accumulation point of $\{\mathbf{g}_k\}$. By taking limit along the relevant subsequence in the above relation and by using Proposition 4.1.2 of [14], it follows that

$$\langle \mathbf{y}, \mathbf{g} \rangle \leq \limsup_{k \rightarrow \infty} f'(\mathbf{x}; \mathbf{y}) \leq f'(\mathbf{x}; \mathbf{y}), \quad \mathbf{y} \in \mathbb{R}^n.$$

Therefore, by (4.11), we have $\mathbf{g} \in \partial f(\mathbf{x})$. \square

The subgradient of the sum of convex functions is obtained as the vector sum of the corresponding subgradients, as shown in the following proposition.

Proposition 191. *Let $f_j : \mathbb{R}^n \rightarrow \mathbb{R}$, $j = 1, \dots, m$, be convex functions and let $f = f_1 + \dots + f_m$. Then*

$$\partial f(\mathbf{x}) = \partial f_1(\mathbf{x}) + \dots + \partial f_m(\mathbf{x}).$$

Proof. It suffices to prove that results of the case where $f = f_1 + f_2$. If $\mathbf{g}_i \in \partial f_i(\mathbf{x})$, $i = 1, 2$, then from the subgradient inequality (4.9), we have

$$f_i(\mathbf{z}) \geq f_i(\mathbf{x}) + \langle \mathbf{g}_i, \mathbf{z} - \mathbf{x} \rangle, \quad \forall \mathbf{z} \in \mathbb{R}^n,$$

so by adding, we obtain

$$f(\mathbf{z}) \geq f(\mathbf{x}) + \langle \mathbf{g}_1 + \mathbf{g}_2, \mathbf{z} - \mathbf{x} \rangle, \quad \forall \mathbf{z} \in \mathbb{R}^n.$$

Hence $\mathbf{g}_1 + \mathbf{g}_2 \in \partial f(\mathbf{x})$, implying that $\partial f_1(\mathbf{x}) + \partial f_2(\mathbf{x}) \subset \partial f(\mathbf{x})$.

To prove the reverse inclusion, assume to arrive at a contradiction, that there exists a $\mathbf{g} \in \partial f(\mathbf{x})$ such that $\mathbf{g} \notin \partial f_1(\mathbf{x}) + \partial f_2(\mathbf{x})$. Since by Proposition 188, the set $\partial f_1(\mathbf{x})$ and $\partial f_2(\mathbf{x})$ are compact, the set $\partial f_1(\mathbf{x}) + \partial f_2(\mathbf{x})$ is compact, and by the Strict Separation Theorem (cf. Proposition 2.4.3 of [14]), there exists a hyperplane strictly separating \mathbf{g} from $\partial f_1(\mathbf{x}) + \partial f_2(\mathbf{x})$, i.e., a vector \mathbf{y} and a scalar b such that

$$\langle \mathbf{y}, \mathbf{g}_1 + \mathbf{g}_2 \rangle < b < \langle \mathbf{y}, \mathbf{g} \rangle, \quad \forall \mathbf{g}_1 \in \partial f_1(\mathbf{x}), \forall \mathbf{g}_2 \in \partial f_2(\mathbf{x}).$$

Therefore,

$$\sup_{\mathbf{g}_1 \in \partial f_1(\mathbf{x})} \langle \mathbf{y}, \mathbf{g}_1 \rangle + \sup_{\mathbf{g}_2 \in \partial f_2(\mathbf{x})} \langle \mathbf{y}, \mathbf{g}_2 \rangle < \langle \mathbf{y}, \mathbf{g} \rangle,$$

and by Proposition 189,

$$f'_1(\mathbf{x}; \mathbf{y}) + f'_2(\mathbf{x}; \mathbf{y}) < \langle \mathbf{y}, \mathbf{g} \rangle.$$

By using the definition of directional derivative, we have $f'_1(\mathbf{x}; \mathbf{y}) + f'_2(\mathbf{x}; \mathbf{y}) = f'(\mathbf{x}; \mathbf{y})$, so that

$$f'(\mathbf{x}; \mathbf{y}) < \langle \mathbf{y}, \mathbf{g} \rangle,$$

which contradicts the assumption $\mathbf{g} \in \partial f(\mathbf{x})$, in view of Proposition 189. \square

Finally, we present some versions of the chain rule for directional derivatives and subgradients.

Proposition 192 (Chain Rule). (a) Let $f : \mathbb{R}^m \rightarrow \mathbb{R}$ be a convex function, and let \mathbf{A} be an $m \times n$ matrix. Then the subgradient of the function F , defined by

$$F(\mathbf{x}) = f(\mathbf{Ax}),$$

is given by

$$\partial F(\mathbf{x}) = \mathbf{A}^T \partial f(\mathbf{Ax}).$$

(b) Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a convex function and let $h : \mathbb{R} \rightarrow \mathbb{R}$ be a differentiable scalar function. Then the function F , defined by

$$F(\mathbf{x}) = h(f(\mathbf{x})),$$

is directionally differentiable at all \mathbf{x} , and its directional derivative is given by

$$F'(\mathbf{x}; \mathbf{y}) = h'(f(\mathbf{x})) f'(\mathbf{x}; \mathbf{y}), \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n.$$

Furthermore, if h is convex and monotonically nondecreasing, then F is convex and its subgradient is given by

$$\partial F(\mathbf{x}) = \partial h(f(\mathbf{x})) \partial f(\mathbf{x}) = \{g\mathbf{g} | g \in \partial h(f(\mathbf{x})), \mathbf{g} \in \partial f(\mathbf{x})\}, \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

(Taken from Section 4.5 of [14])

Theorem 193 (Danskin's Theorem). Let \mathcal{Z} be a compact subset of \mathbb{R}^m , and let $\phi : \mathbb{R}^n \times \mathcal{Z} \rightarrow \mathbb{R}$ be continuous and such that $\phi(\cdot, \mathbf{z}) : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex for each $\mathbf{z} \in \mathcal{Z}$.

(a) The function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ given by

$$f(\mathbf{x}) = \max_{\mathbf{z} \in \mathcal{Z}} \phi(\mathbf{x}, \mathbf{z}) \tag{4.15}$$

is convex and has directional derivative given by

$$f'(\mathbf{x}; \mathbf{y}) = \max_{\mathbf{z} \in \mathcal{Z}(\mathbf{x})} \phi'(\mathbf{x}, \mathbf{z}; \mathbf{y}), \tag{4.16}$$

where $\phi'(\mathbf{x}, \mathbf{z}; \mathbf{y})$ is the directional derivative of the function $\phi(\cdot, \mathbf{z})$ at \mathbf{x} in the direction \mathbf{y} and $\mathcal{Z}(\mathbf{x})$ is the set of maximizing points in (4.15)

$$\mathcal{Z}(\mathbf{x}) = \left\{ \bar{\mathbf{z}} \mid \phi(\mathbf{x}, \bar{\mathbf{z}} = \max_{\mathbf{z} \in \mathcal{Z}} \phi(\mathbf{x}, \mathbf{z})) \right\}.$$

(b) If $\phi(\cdot, \mathbf{z})$ is differentiable for all $\mathbf{z} \in \mathcal{Z}$ and $\nabla_x \phi(\mathbf{x}, \cdot)$ is continuous on \mathcal{Z} for each \mathbf{x} , then

$$\partial f(\mathbf{x}) = \text{conv}\{\nabla_x \phi(\mathbf{x}, \mathbf{z}) | \mathbf{z} \in \mathcal{Z}(\mathbf{x})\}, \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

Proof. (b) By part (a), we have

$$f'(\mathbf{x}; \mathbf{y}) = \max_{\mathbf{z} \in \mathcal{Z}(\mathbf{x})} \langle \nabla_x \phi(\mathbf{x}, \mathbf{z}), \mathbf{y} \rangle,$$

while by Proposition 189,

$$f'(\mathbf{x}; \mathbf{y}) = \max_{\mathbf{z} \in \partial f(\mathbf{x})} \langle \mathbf{z}, \mathbf{y} \rangle.$$

For all $\bar{\mathbf{z}} \in \mathcal{Z}(\mathbf{x})$ and $\mathbf{y} \in \mathbb{R}^n$, we have

$$\begin{aligned} f(\mathbf{y}) &= \max_{\mathbf{z} \in \mathcal{Z}} \phi(\mathbf{y}, \mathbf{z}) \\ &\geq \phi(\mathbf{y}, \bar{\mathbf{z}}) \\ &\geq \phi(\mathbf{x}, \bar{\mathbf{z}}) + \langle \nabla_x \phi(\mathbf{x}, \bar{\mathbf{z}}), \mathbf{y} - \mathbf{x} \rangle \\ &= f(\mathbf{x}) + \langle \nabla_x \phi(\mathbf{x}, \bar{\mathbf{z}}), \mathbf{y} - \mathbf{x} \rangle. \end{aligned}$$

Therefore, $\nabla_x \phi(\mathbf{x}, \bar{\mathbf{z}})$ is a subgradient of f at \mathbf{x} , implying that

$$\text{conv} \{ \nabla_x \phi(\mathbf{x}, \mathbf{z}) | \mathbf{z} \in \mathcal{Z}(\mathbf{x}) \} \subset \partial f(\mathbf{x}).$$

To prove the converse inclusion, we use a hyperplane separation argument. Since $\phi(\mathbf{x}, \cdot)$ is continuous and \mathcal{Z} is compact, from Weierstrass' Theorem it follows that $\mathcal{Z}(\mathbf{x})$ is compact, so since $\nabla_x \phi(\mathbf{x}, \cdot)$ is continuous, the set $\{ \nabla_x \phi(\mathbf{x}, \mathbf{z}) | \mathbf{z} \in \mathcal{Z}(\mathbf{x}) \}$ is compact. By Proposition 1.3.2 of [14], it follows that $\text{conv} \{ \nabla_x \phi(\mathbf{x}, \mathbf{z}) | \mathbf{z} \in \mathcal{Z}(\mathbf{x}) \}$ is compact. If $\mathbf{g} \in \partial f(\mathbf{x})$ while $\mathbf{d} \notin \text{conv} \{ \nabla_x \phi(\mathbf{x}, \mathbf{z}) | \mathbf{z} \in \mathcal{Z}(\mathbf{x}) \}$, by the Strict Separation Theorem, there exist $\mathbf{y} \neq \mathbf{0}$ and $\gamma \in \mathbb{R}$, such that

$$\langle \mathbf{g}, \mathbf{y} \rangle > \gamma > \langle \nabla_x \phi(\mathbf{x}, \mathbf{z}), \mathbf{y} \rangle, \quad \forall \mathbf{z} \in \mathcal{Z}(\mathbf{x}).$$

Therefore, we have

$$\langle \mathbf{g}, \mathbf{y} \rangle > \max_{\mathbf{z} \in \mathcal{Z}(\mathbf{x})} \langle \nabla_x \phi(\mathbf{x}, \mathbf{z}), \mathbf{y} \rangle = f'(\mathbf{x}; \mathbf{y}),$$

contradicting Proposition 189. Thus, $\partial f(\mathbf{x}) \subset \text{conv} \{ \nabla_x \phi(\mathbf{x}, \mathbf{z}) | \mathbf{z} \in \mathcal{Z}(\mathbf{x}) \}$ and the proof is complete. \square

Theorem 194 (Subgradient of norms [45]). *Let \mathcal{H} be a real Hilbert space endowed with an inner product $\langle \cdot, \cdot \rangle$ and a norm $\|\cdot\|$. Then $\partial \|\mathbf{x}\| = \{ \mathbf{y} | \langle \mathbf{y}, \mathbf{x} \rangle = \|\mathbf{x}\| \text{ and } \|\mathbf{y}\|^* \leq 1 \}$, where $\|\cdot\|^*$ is the dual norm of $\|\cdot\|$.*

Proof. Let $S = \{ \mathbf{y} | \langle \mathbf{y}, \mathbf{x} \rangle = \|\mathbf{x}\| \text{ and } \|\mathbf{y}\|^* \leq 1 \}$.

For every $\mathbf{y} \in \partial \|\mathbf{x}\|$, we have

$$\|\mathbf{w} - \mathbf{x}\| \geq \|\mathbf{w}\| - \|\mathbf{x}\| \geq \langle \mathbf{y}, \mathbf{w} - \mathbf{x} \rangle, \quad \forall \mathbf{w} \in \mathcal{H}. \quad (4.17)$$

Choosing $\mathbf{w} = \mathbf{0}$ and $\mathbf{w} = 2\mathbf{x}$ for the second inequality above, which results from the convexity of norm $\|\cdot\|$, we can deduce that

$$\|\mathbf{x}\| = \langle \mathbf{y}, \mathbf{x} \rangle. \quad (4.18)$$

On the other hand, (4.17) gives

$$\|\mathbf{w} - \mathbf{x}\| \geq \langle \mathbf{y}, \mathbf{w} - \mathbf{x} \rangle, \quad \forall \mathbf{w} \in \mathcal{H}. \quad (4.19)$$

So

$$\left\langle \mathbf{y}, \frac{\mathbf{w} - \mathbf{x}}{\|\mathbf{w} - \mathbf{x}\|} \right\rangle \leq 1, \quad \forall \mathbf{w} \neq \mathbf{x}.$$

Therefore $\|\mathbf{y}\|^* \leq 1$. Thus $\partial\|\mathbf{x}\| \subset S$.

For every $\mathbf{y} \in S$, we have

$$\langle \mathbf{y}, \mathbf{w} - \mathbf{x} \rangle = \langle \mathbf{y}, \mathbf{w} \rangle - \langle \mathbf{y}, \mathbf{x} \rangle = \langle \mathbf{y}, \mathbf{w} \rangle - \|\mathbf{x}\| \leq \|\mathbf{y}\|^* \|\mathbf{w}\| - \|\mathbf{x}\| \leq \|\mathbf{w}\| - \|\mathbf{x}\|, \quad \forall \mathbf{w} \in \mathcal{H}, \quad (4.20)$$

where the second equality utilizes $\langle \mathbf{y}, \mathbf{x} \rangle = \|\mathbf{x}\|$ and the first inequality is by the definition of dual norm. Thus, $\mathbf{y} \in \partial\|\mathbf{x}\|$. So $S \subset \partial\|\mathbf{x}\|$. \square

Example 195. $\partial\|\mathbf{x}\|_1, \partial I_C(\mathbf{x})$.

Example 196 (Subgradient of nuclear norm). *The subgradient of nuclear norm of a matrix \mathbf{X} is:*

$$\partial\|\mathbf{X}\|_* = \{ \mathbf{U}\mathbf{V}^T + \mathbf{W} | \mathbf{U}^T\mathbf{W} = \mathbf{0}, \mathbf{W}\mathbf{V} = \mathbf{0}, \|\mathbf{W}\| \leq 1 \}, \quad (4.21)$$

where $\mathbf{U}\Sigma\mathbf{V}^T$ is the skinny SVD of \mathbf{X} .

Proof. Based on Theorem 194, we provide a different proof from that in [22].

By Theorem 194, $\partial\|\mathbf{X}\|_* = S \triangleq \{ \mathbf{Y} | \langle \mathbf{X}, \mathbf{Y} \rangle = \|\mathbf{X}\|_*, \|\mathbf{Y}\| \leq 1 \}$. Let $T = \{ \mathbf{U}\mathbf{V}^T + \mathbf{W} | \mathbf{U}^T\mathbf{W} = \mathbf{0}, \mathbf{W}\mathbf{V} = \mathbf{0}, \|\mathbf{W}\| \leq 1 \}$. We need to prove that $S = T$.

For every $\mathbf{Y} \in T$, it can be written as $\mathbf{Y} = \mathbf{U}\mathbf{V}^T + \mathbf{W}$, where $\mathbf{U}^T\mathbf{W} = \mathbf{0}$, $\mathbf{W}\mathbf{V} = \mathbf{0}$, $\|\mathbf{W}\| \leq 1$. So \mathbf{W} can be represented as $\mathbf{W} = \mathbf{U}^\perp \mathbf{Q} (\mathbf{V}^\perp)^T$, where $\|\mathbf{Q}\| \leq 1$, and \mathbf{U}^\perp and \mathbf{V}^\perp are the orthogonal complements of \mathbf{U} and \mathbf{V} , respectively. Then $\mathbf{Y} = (\mathbf{U}, \mathbf{U}^\perp) \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q} \end{pmatrix} (\mathbf{V}, \mathbf{V}^\perp)^T$. It can be verified that $\mathbf{Y} \in S$.

For every $\mathbf{Y} \in S$, since

$$\|\mathbf{X}\|_* = \langle \mathbf{X}, \mathbf{Y} \rangle \leq \sum_{i=1}^r \sigma_i(\mathbf{X}) \sigma_i(\mathbf{Y}) \leq \sum_{i=1}^r \sigma_i(\mathbf{X}) = \|\mathbf{X}\|_*,$$

where $r = \text{rank}(\mathbf{X})$ and the first inequality comes from Theorem 761, we have $\langle \mathbf{X}, \mathbf{Y} \rangle = \sum_{i=1}^r \sigma_i(\mathbf{X})\sigma_i(\mathbf{Y})$. By Theorem 761, there are common column orthonormal matrices $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ such that

$$\mathbf{X} = \hat{\mathbf{U}} \begin{pmatrix} \Sigma & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \hat{\mathbf{V}}^T, \quad \text{and} \quad \mathbf{Y} = \hat{\mathbf{U}} \begin{pmatrix} \Sigma_{Y,1} & \mathbf{0} \\ \mathbf{0} & \Sigma_{Y,2} \end{pmatrix} \hat{\mathbf{V}}^T,$$

where $\Sigma_{Y,1}$ and $\Sigma_{Y,2}$ consist of the singular values of \mathbf{Y} , ordered from large to small. Since $\|\mathbf{Y}\| \leq 1$ it is easy to see that $\Sigma_{Y,1} = \mathbf{I}$ in order to fulfil $\|\mathbf{X}\|_* = \langle \mathbf{X}, \mathbf{Y} \rangle$, and all the singular values in $\Sigma_{Y,2}$ do not exceed 1. Partitioning $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ into the first r columns and the remaining ones:

$$\hat{\mathbf{U}} = (\hat{\mathbf{U}}_1, \hat{\mathbf{U}}_2) \quad \text{and} \quad \hat{\mathbf{V}} = (\hat{\mathbf{V}}_1, \hat{\mathbf{V}}_2),$$

we have $\mathbf{X} = \hat{\mathbf{U}}_1 \Sigma \hat{\mathbf{V}}_1^T$. As the singular subspaces are unique, $\hat{\mathbf{U}}_1$ and \mathbf{U} can only differ by an orthogonal matrix \mathbf{O} . So do $\hat{\mathbf{V}}_1$ and \mathbf{V} , with the same \mathbf{O} . Thus $\hat{\mathbf{U}}_1 \hat{\mathbf{V}}_1^T = \mathbf{U} \mathbf{V}^T$, and $\hat{\mathbf{U}}_2$ and $\hat{\mathbf{V}}_2$ are still the orthogonal complements of \mathbf{U} and \mathbf{V} , respectively. Finally, $\mathbf{Y} = \mathbf{U} \mathbf{V}^T + \mathbf{W}$, where $\mathbf{W} = \hat{\mathbf{U}}_2 \Sigma_{Y,2} \hat{\mathbf{V}}_2^T$ satisfies $\mathbf{U}^T \mathbf{W} = \mathbf{0}$, $\mathbf{W} \mathbf{V}^T = \mathbf{0}$, and $\|\mathbf{W}\| \leq 1$. Hence $\mathbf{Y} \in T$. \square

Example 197 (Subgradient of matrix 2-norm). Compute $\partial \|\mathbf{X}\|_2$.

We first recall that $\|\mathbf{X}\|_2 = \max_{\|\mathbf{u}\|=\|\mathbf{v}\|=1} \mathbf{u}^T \mathbf{X} \mathbf{v} = \max_{\|\mathbf{u}\|=\|\mathbf{v}\|=1} \text{tr}(\mathbf{X} \mathbf{v} \mathbf{u}^T)$. Then by Danskin theorem,

$$\partial \|\mathbf{X}\|_2 = \text{conv}\{\mathbf{u} \mathbf{v}^T \mid \mathbf{u}^T \mathbf{X} \mathbf{v} = \|\mathbf{X}\|_2, \|\mathbf{u}\| = \|\mathbf{v}\| = 1\}.$$

If $\mathbf{X} \neq \mathbf{0}$, then the (\mathbf{u}, \mathbf{v}) that satisfy $\mathbf{u}^T \mathbf{X} \mathbf{v} = \|\mathbf{X}\|_2$ are simply the unit left and right singular vectors associated to the largest singular value σ_1 of \mathbf{X} and $\mathbf{u} = \sigma_1^{-1} \mathbf{X} \mathbf{v}$.

Suppose the dimension of the principal singular subspace is d , then there are orthonormal right singular vectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$, and left singular vectors $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_d$, such that $\mathbf{u}_i = \sigma_1^{-1} \mathbf{X} \mathbf{v}_i$, $i = 1, 2, \dots, d$.

Then every pair of unit left and right singular vectors associated to the largest singular value σ_1 of \mathbf{X} can be written as $\mathbf{u} = \mathbf{U} \boldsymbol{\alpha}$ and $\mathbf{v} = \mathbf{V} \boldsymbol{\alpha}$, where $\mathbf{U} = (\mathbf{u}_1, \dots, \mathbf{u}_d)$, $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_d)$, $\|\boldsymbol{\alpha}\| = 1$. So

$$\partial \|\mathbf{X}\|_2 = \text{conv}\{\mathbf{U} \boldsymbol{\alpha} \boldsymbol{\alpha}^T \mathbf{V}^T \mid \|\boldsymbol{\alpha}\| = 1\} = \{\mathbf{U} \mathbf{W} \mathbf{V}^T \mid \mathbf{W} \succcurlyeq \mathbf{0}, \text{tr } \mathbf{W} = 1\}. \quad (4.22)$$

If $\mathbf{X} = \mathbf{0}$, then

$$\partial \|\mathbf{X}\|_2 = \text{conv}\{\mathbf{u} \mathbf{v}^T \mid \|\mathbf{u}\| = \|\mathbf{v}\| = 1\} = \{\mathbf{W} \mid \|\mathbf{W}\|_* \leq 1\}. \quad (4.23)$$

4.1.12.1 Exercises

Exercise 198. Prove that the directional derivative $f'(\mathbf{x}; \mathbf{y})$ is a convex function of \mathbf{y} .

Exercise 199. Compute the subgradients of $f(\mathbf{x}) = \frac{1}{2}x_1^2 + |x_2|$, $\|\mathbf{x}\|_2$, $\|\mathbf{x}\|_\infty$, $\|\mathbf{x}\|_{\mathbf{M}} = \sqrt{\mathbf{x}^T \mathbf{M} \mathbf{x}}$, $\|\mathbf{X}\|_{2,1}$, and $\|\mathbf{D} \text{Diag}(\mathbf{x})\|_*$.

Exercise 200. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a convex function. Show that $\partial f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a monotone mapping, i.e.,

$$\langle \mathbf{g}_1 - \mathbf{g}_2, \mathbf{x}_1 - \mathbf{x}_2 \rangle \geq 0, \quad \forall \mathbf{g}_i \in \partial f(\mathbf{x}_i), i = 1, 2. \quad (4.24)$$

Further, if f is μ -strongly convex, then the above inequality can be strengthened as

$$\langle \mathbf{g}_1 - \mathbf{g}_2, \mathbf{x}_1 - \mathbf{x}_2 \rangle \geq \mu \|\mathbf{x}_1 - \mathbf{x}_2\|^2, \quad \forall \mathbf{g}_i \in \partial f(\mathbf{x}_i), i = 1, 2. \quad (4.25)$$

Exercise 201. Which of the commonly used loss functions are convex?

4.2 Operations that preserve convexity

In this section we describe some operations that preserve convexity or concavity of functions, or allow us to construct new convex and concave functions. We start with some simple operations such as addition, scaling, and pointwise supremum, and then describe some more sophisticated operations (some of which include the simple operations as special cases).

4.2.1 Nonnegative weighted sums

Evidently if f is a convex function and $\alpha \geq 0$, then the function αf is convex. If f_1 and f_2 are both convex functions, then so is their sum $f_1 + f_2$. Combining nonnegative scaling and addition, we see that the set of convex functions is itself a convex cone: a nonnegative weighted sum of convex functions,

$$f = w_1 f_1 + \cdots + w_m f_m,$$

is convex. Similarly, a nonnegative weighted sum of concave functions is concave. A nonnegative, nonzero weighted sum of strictly convex (concave) functions is strictly convex (concave).

These properties extend to infinite sums and integrals. For example if $f(\mathbf{x}, \mathbf{y})$ is convex in \mathbf{x} for each $\mathbf{y} \in \mathcal{A}$, and $w(\mathbf{y}) \geq 0$ for each $\mathbf{y} \in \mathcal{A}$, then the function g defined as

$$g(\mathbf{x}) = \int_A w(\mathbf{y}) f(\mathbf{x}, \mathbf{y}) d\mathbf{y}$$

is convex in \mathbf{x} (provided the integral exists).

The fact that convexity is preserved under nonnegative scaling and addition is easily verified directly, or can be seen in terms of the associated epigraphs. For example, if $w \geq 0$ and f is convex, we have

$$\text{epi}(wf) = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0}^T & w \end{bmatrix} \text{epi } f,$$

which is convex because the image of a convex set under a linear mapping is convex.

4.2.2 Composition with an affine mapping

Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $\mathbf{A} \in \mathbb{R}^{n \times m}$, and $\mathbf{b} \in \mathbb{R}^n$. Define $g : \mathbb{R}^m \rightarrow \mathbb{R}$ by

$$g(\mathbf{x}) = f(\mathbf{Ax} + \mathbf{b}),$$

with $\text{dom } g = \{\mathbf{x} \mid \mathbf{Ax} + \mathbf{b} \in \text{dom } f\}$. Then if f is convex, so is g ; if f is concave, so is g .

4.2.3 Pointwise maximum and supremum

If f_1 and f_2 are convex functions then their *pointwise maximum* f , defined by

$$f(\mathbf{x}) = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\},$$

with $\text{dom } f = \text{dom } f_1 \cap \text{dom } f_2$, is also convex. This property is easily verified: if $0 \leq \theta \leq 1$ and $\mathbf{x}, \mathbf{y} \in \text{dom } f$, then

$$\begin{aligned} & f(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \\ &= \max\{f_1(\theta\mathbf{x} + (1 - \theta)\mathbf{y}), f_2(\theta\mathbf{x} + (1 - \theta)\mathbf{y})\} \\ &\leq \max\{\theta f_1(\mathbf{x}) + (1 - \theta)f_1(\mathbf{y}), \theta f_2(\mathbf{x}) + (1 - \theta)f_2(\mathbf{y})\} \\ &\leq \theta \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\} + (1 - \theta) \max\{f_1(\mathbf{y}), f_2(\mathbf{y})\} \\ &= \theta f(\mathbf{x}) + (1 - \theta)f(\mathbf{y}), \end{aligned}$$

which establishes convexity of f . It is easily shown that if f_1, \dots, f_m are convex, then their pointwise maximum

$$f(\mathbf{x}) = \max\{f_1(\mathbf{x}), \dots, f_m(\mathbf{x})\}$$

is also convex.

Example 202 (Piecewise-linear functions). *The function*

$$f(x) = \max\{\langle \mathbf{a}_1, \mathbf{x} \rangle + \mathbf{b}_1, \dots, \langle \mathbf{a}_L, \mathbf{x} \rangle + \mathbf{b}_L\}$$

defines a piecewise-linear (or really, affine) function (with L or fewer regions). It is convex since it is the pointwise maximum of affine functions.

The converse can also be shown: any piecewise-linear convex function with L or fewer regions can be expressed in this form. (See Exercise 294.)

Example 203 (Sum of r largest components). For $\mathbf{x} \in \mathbb{R}^n$ we denote by $x_{[i]}$ the i th largest component of \mathbf{x} , i.e.,

$$x_{[1]} \geq x_{[2]} \geq \cdots \geq x_{[n]}$$

are the components of \mathbf{x} sorted in nonincreasing order. Then the function

$$f(\mathbf{x}) = \sum_{i=1}^r x_{[i]},$$

i.e., the sum of the r largest elements of \mathbf{x} , is a convex function. This can be seen by writing it as

$$f(\mathbf{x}) = \sum_{i=1}^r x_{[i]} = \max\{x_{i_1} + \cdots + x_{i_r} \mid 1 \leq i_1 < i_2 < \cdots < i_r \leq n\},$$

i.e., the maximum of all possible sums of r different components of \mathbf{x} . Since it is the pointwise maximum of $n!/(r!(n-r)!)$ linear functions, it is convex.

As an extension it can be shown that the function $\sum_{i=1}^r w_i x_{[i]}$ is convex, provided $w_1 \geq w_2 \geq \cdots \geq w_r \geq 0$. (See Exercise 284.)

The pointwise maximum property extends to the pointwise supremum over an infinite set of convex functions. If for each $\mathbf{y} \in \mathcal{A}$, $f(\mathbf{x}, \mathbf{y})$ is convex in \mathbf{x} , then the function g , defined as

$$g(\mathbf{x}) = \sup_{\mathbf{y} \in \mathcal{A}} f(\mathbf{x}, \mathbf{y}) \tag{4.26}$$

is convex in \mathbf{x} . Here the domain of g is

$$\text{dom } g = \{\mathbf{x} \mid (\mathbf{x}, \mathbf{y}) \in \text{dom } f \text{ for all } \mathbf{y} \in \mathcal{A}, \sup_{\mathbf{y} \in \mathcal{A}} f(\mathbf{x}, \mathbf{y}) < \infty\}.$$

Similarly, the pointwise infimum of a set of concave functions is a concave function.

In terms of epigraphs, the pointwise supremum of functions corresponds to the intersection of epigraphs: with f , g , and \mathcal{A} as defined in (4.26), we have

$$\text{epi } g = \bigcap_{\mathbf{y} \in \mathcal{A}} \text{epi } f(\cdot, \mathbf{y}).$$

Thus, the result follows from the fact that the intersection of a family of convex sets is convex.

Example 204 (Support function of a set). Let $C \subseteq \mathbb{R}^n$, with $C \neq \emptyset$. The support function S_C associated with the set C is defined as

$$S_C(\mathbf{x}) = \sup\{\langle \mathbf{x}, \mathbf{y} \rangle \mid \mathbf{y} \in C\}$$

(and, naturally, $\text{dom } S_C = \{\mathbf{x} \mid \sup_{\mathbf{y} \in C} \langle \mathbf{x}, \mathbf{y} \rangle < \infty\}$). For each $\mathbf{y} \in C$, $\langle \mathbf{x}, \mathbf{y} \rangle$ is a linear function of \mathbf{x} , so S_C is the pointwise supremum of a family of linear functions, hence convex.

Example 205 (Distance to farthest point of a set). Let $C \subseteq \mathbb{R}^n$. The distance (in any norm) to the farthest point of C ,

$$f(\mathbf{x}) = \sup_{\mathbf{y} \in C} \|\mathbf{x} - \mathbf{y}\|,$$

is convex. To see this, note that for any \mathbf{y} , the function $\|\mathbf{x} - \mathbf{y}\|$ is convex in \mathbf{x} . Since f is the pointwise supremum of a family of convex functions (indexed by $\mathbf{y} \in C$), it is a convex function of \mathbf{x} .

Example 206 (Least-squares cost as a function of weights). Let $\mathbf{a}_1, \dots, \mathbf{a}_n \in \mathbb{R}^m$. In a weighted least-squares problem we minimize the objective function $\sum_{i=1}^n w_i (\langle \mathbf{a}_i, \mathbf{x} \rangle - b_i)^2$ over $\mathbf{x} \in \mathbb{R}^m$. We refer to w_i as weights, and allow negative w_i (which opens the possibility that the objective function is unbounded below).

We define the (optimal) weighted least-squares cost as

$$g(\mathbf{w}) = \inf_{\mathbf{x}} \sum_{i=1}^n w_i (\langle \mathbf{a}_i, \mathbf{x} \rangle - b_i)^2,$$

with domain

$$\text{dom } g = \left\{ \mathbf{w} \mid \inf_{\mathbf{x}} \sum_{i=1}^n w_i (\langle \mathbf{a}_i, \mathbf{x} \rangle - b_i)^2 > -\infty \right\}.$$

Since g is the infimum of a family of linear functions of \mathbf{w} (indexed by $\mathbf{x} \in \mathbb{R}^m$), it is a concave function of \mathbf{w} .

We can derive an explicit expression for g , at least on part of its domain. Let $\mathbf{W} = \text{diag}(\mathbf{w})$, the diagonal matrix with elements w_1, \dots, w_n , and let $\mathbf{A} \in \mathbb{R}^{n \times m}$ have rows \mathbf{a}_i^T , so we have

$$\begin{aligned} g(\mathbf{w}) &= \inf_{\mathbf{x}} (\mathbf{Ax} - \mathbf{b})^T \mathbf{W} (\mathbf{Ax} - \mathbf{b}) \\ &= \inf_{\mathbf{x}} (\mathbf{x}^T \mathbf{A}^T \mathbf{W} \mathbf{A} \mathbf{x} - 2\mathbf{b}^T \mathbf{W} \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{W} \mathbf{b}). \end{aligned}$$

From this we see that if $\mathbf{A}^T \mathbf{W} \mathbf{A} \not\succeq \mathbf{0}$, the quadratic function is unbounded below in \mathbf{x} , so $g(\mathbf{w}) = -\infty$, i.e., $\mathbf{w} \notin \text{dom } g$. We can give a simple expression for g when $\mathbf{A}^T \mathbf{W} \mathbf{A} \succ \mathbf{0}$

(which defines a strict linear matrix inequality), by analytically minimizing the quadratic function:

$$\begin{aligned} g(\mathbf{w}) &= \mathbf{b}^T \mathbf{W} \mathbf{b} - \mathbf{b}^T \mathbf{W} \mathbf{A} (\mathbf{A}^T \mathbf{W} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{W} \mathbf{b} \\ &= \sum_{i=1}^n w_i b_i^2 - \sum_{i=1}^n w_i^2 b_i^2 \mathbf{a}_i^T \left(\sum_{j=1}^n w_j \mathbf{a}_j \mathbf{a}_j^T \right)^{-1} \mathbf{a}_i. \end{aligned}$$

Concavity of g from this expression is not immediately obvious (but does follow, for example, from convexity of the matrix fractional function; see Example 181).

Example 207 (Maximum eigenvalue of a symmetric matrix). The function $f(\mathbf{X}) = \lambda_{\max}(\mathbf{X})$, with $\text{dom } f = \mathbb{S}^m$, is convex. To see this, we express f as

$$f(\mathbf{X}) = \sup\{\mathbf{y}^T \mathbf{X} \mathbf{y} \mid \|\mathbf{y}\|_2 = 1\},$$

i.e., as the pointwise supremum of a family of linear functions of \mathbf{X} (i.e., $\mathbf{y}^T \mathbf{X} \mathbf{y}$) indexed by $\mathbf{y} \in \mathbb{R}^m$.

Example 208 (Norm of a matrix). Consider $f(\mathbf{X}) = \|\mathbf{X}\|_2$ with $\text{dom } f = \mathbb{R}^{p \times q}$, where $\|\cdot\|_2$ denotes the spectral norm or maximum singular value. Convexity of f follows from

$$f(\mathbf{X}) = \sup\{\mathbf{u}^T \mathbf{X} \mathbf{v} \mid \|\mathbf{u}\|_2 = 1, \|\mathbf{v}\|_2 = 1\},$$

which shows it is the pointwise supremum of a family of linear functions of \mathbf{X} .

As a generalization suppose $\|\cdot\|_a$ and $\|\cdot\|_b$ are norms on \mathbb{R}^p and \mathbb{R}^q , respectively. The induced norm of a matrix $\mathbf{X} \in \mathbb{R}^{p \times q}$ is defined as

$$\|\mathbf{X}\|_{a,b} = \sup_{\mathbf{v} \neq \mathbf{0}} \frac{\|\mathbf{X}\mathbf{v}\|_a}{\|\mathbf{v}\|_b}.$$

(This reduces to the spectral norm when both norms are Euclidean.) The induced norm can be expressed as

$$\begin{aligned} \|\mathbf{X}\|_{a,b} &= \sup\{\|\mathbf{X}\mathbf{v}\|_a \mid \|\mathbf{v}\|_b = 1\} \\ &= \sup\{\mathbf{u}^T \mathbf{X} \mathbf{v} \mid \|\mathbf{u}\|_{a*} = 1, \|\mathbf{v}\|_b = 1\}, \end{aligned}$$

where $\|\cdot\|_{a*}$ is the dual norm of $\|\cdot\|_a$, and we use the fact that

$$\|\mathbf{z}\|_a = \sup\{\langle \mathbf{u}, \mathbf{z} \rangle \mid \|\mathbf{u}\|_{a*} = 1\}.$$

Since we have expressed $\|\mathbf{X}\|_{a,b}$ as a supremum of linear functions of \mathbf{X} , it is a convex function.

Representation as pointwise supremum of affine functions

The examples above illustrate a good method for establishing convexity of a function: by expressing it as the pointwise supremum of a family of affine functions. Except for a technical condition, a converse holds: almost every convex function can be expressed as the pointwise supremum of a family of affine functions. For example, if $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex, with $\text{dom } f = \mathbb{R}^n$, then we have

$$f(\mathbf{x}) = \sup\{g(\mathbf{x}) \mid g \text{ affine}, g(\mathbf{z}) \leq f(\mathbf{z}) \text{ for all } \mathbf{z}\}.$$

In other words, f is the pointwise supremum of the set of all affine global underestimators of it. We give the proof of this result below, and leave the case where $\text{dom } f \neq \mathbb{R}^n$ as an exercise (Exercise 226).

Suppose f is convex with $\text{dom } f = \mathbb{R}^n$. The inequality

$$f(\mathbf{x}) \geq \{g(\mathbf{x}) \mid g \text{ affine}, g(\mathbf{z}) \leq f(\mathbf{z}) \text{ for all } \mathbf{z}\}$$

is clear, since if g is any affine underestimator of f , we have $g(\mathbf{x}) \leq f(\mathbf{x})$. To establish equality, we will show that for each $\mathbf{x} \in \mathbb{R}^n$, there is an affine function g , which is a global underestimator of f , and satisfies $g(\mathbf{x}) = f(\mathbf{x})$.

The epigraph of f is, of course, a convex set. Hence we can find a supporting hyperplane to it at $(\mathbf{x}, f(\mathbf{x}))$, i.e., $\mathbf{a} \in \mathbb{R}^n$ and $b \in \mathbb{R}$ with $(\mathbf{a}, b) \neq \mathbf{0}$ and

$$\left\langle \begin{bmatrix} \mathbf{a} \\ b \end{bmatrix}, \begin{bmatrix} \mathbf{x} - \mathbf{z} \\ f(\mathbf{x}) - t \end{bmatrix} \right\rangle \leq 0$$

for all $(\mathbf{z}, t) \in \text{epi } f$. This means that

$$\langle \mathbf{a}, \mathbf{x} - \mathbf{z} \rangle + b(f(\mathbf{x}) - f(\mathbf{z}) - s) \leq 0 \quad (4.27)$$

for all $\mathbf{z} \in \text{dom } f = \mathbb{R}^n$ and all $s \geq 0$ (since $(\mathbf{z}, t) \in \text{epi } f$ means $t = f(\mathbf{z}) + s$ for some $s \geq 0$). For the inequality (4.27) to hold for all $s \geq 0$, we must have $b \geq 0$. If $b = 0$, then the inequality (4.27) reduces to $\langle \mathbf{a}, \mathbf{x} - \mathbf{z} \rangle \leq 0$ for all $\mathbf{z} \in \mathbb{R}^n$, which implies $\mathbf{a} = \mathbf{0}$ and contradicts $(\mathbf{a}, b) \neq \mathbf{0}$. We conclude that $b > 0$, i.e., that the supporting hyperplane is not vertical.

Using the fact that $b > 0$ we rewrite (4.27) for $s = 0$ as

$$g(\mathbf{z}) = f(\mathbf{x}) + \langle \mathbf{a}/b, \mathbf{x} - \mathbf{z} \rangle \leq f(\mathbf{z})$$

for all \mathbf{z} . The function g is an affine underestimator of f , and satisfies $g(\mathbf{x}) = f(\mathbf{x})$.

4.2.4 Composition

In this section we examine conditions on $h : \mathbb{R}^k \rightarrow \mathbb{R}$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}^k$ that guarantee convexity or concavity of their composition $f = h \circ g : \mathbb{R}^n \rightarrow \mathbb{R}$, defined by

$$f(\mathbf{x}) = h(g(\mathbf{x})), \quad \text{dom } f = \{\mathbf{x} \in \text{dom } g \mid g(\mathbf{x}) \in \text{dom } h\}.$$

Scalar composition We first consider the case $k = 1$, so $h : \mathbb{R} \rightarrow \mathbb{R}$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}$. We can restrict ourselves to the case $n = 1$ (since convexity is determined by the behavior of a function on arbitrary lines that intersect its domain).

To discover the composition rules, we start by assuming that h and g are twice differentiable, with $\text{dom } g = \text{dom } h = \mathbb{R}$. In this case, convexity of f reduces to $f'' \geq 0$ (meaning, $f''(x) \geq 0$ for all $x \in \mathbb{R}$).

The second derivative of the composition function $f = h \circ g$ is given by

$$f''(x) = h''(g(x))g'(x)^2 + h'(g(x))g''(x). \quad (4.28)$$

Now suppose, for example, that g is convex (so $g'' \geq 0$) and h is convex and nondecreasing (so $h'' \geq 0$ and $h' \geq 0$). It follows from (4.28) that $f'' \geq 0$, i.e., f is convex. In a similar way, the expression (4.28) gives the results:

- f is convex if h is convex and nondecreasing, and g is convex,
- f is convex if h is convex and nonincreasing, and g is concave,
- f is concave if h is concave and nondecreasing, and g is concave,
- f is concave if h is concave and nonincreasing, and g is convex.

These statements are valid when the functions g and h are twice differentiable and have domains that are all of \mathbb{R} . It turns out that very similar composition rules hold in the general case $n > 1$, without assuming differentiability of h and g , or that $\text{dom } g = \mathbb{R}^n$ and $\text{dom } h = \mathbb{R}$:

- f is convex if h is convex, \tilde{h} is nondecreasing, and g is convex,
- f is convex if h is convex, \tilde{h} is nonincreasing, and g is concave,
- f is concave if h is concave, \tilde{h} is nondecreasing, and g is concave,
- f is concave if h is concave, \tilde{h} is nonincreasing, and g is convex.

Here \tilde{h} denotes the extended-value extension of the function h , which assigns the value ∞ ($-\infty$) to points not in $\text{dom } h$ for h convex (concave). The only difference between these results, and the results in (4.29), is that we require that the *extended-value extension* function \tilde{h} be nonincreasing or nondecreasing, on all of \mathbb{R} .

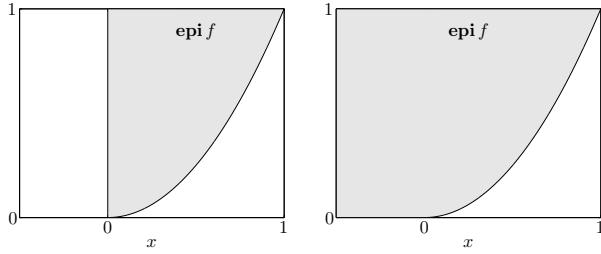


图 4.10: *Left.* The function x^2 , with domain \mathbb{R}_+ , is convex and nondecreasing on its domain, but its extended-value extension is *not* nondecreasing. *Right.* The function $\max\{x, 0\}^2$, with domain \mathbb{R} , is convex, and its extended-value extension is nondecreasing.

To understand what this means, suppose h is convex, so \tilde{h} takes on the value ∞ outside $\text{dom } h$. To say that \tilde{h} is nondecreasing means that for *any* $x, y \in \mathbb{R}$, with $x < y$, we have $\tilde{h}(x) \leq \tilde{h}(y)$. In particular, this means that if $y \in \text{dom } h$, then $x \in \text{dom } h$. In other words, the domain of h extends infinitely in the negative direction; it is either \mathbb{R} , or an interval of the form $(-\infty, a)$ or $(-\infty, a]$. In a similar way, to say that h is convex and \tilde{h} is nonincreasing means that h is nonincreasing and $\text{dom } h$ extends infinitely in the positive direction. This is illustrated in Figure 4.10.

Example 209. Some simple examples will illustrate the conditions on h that appear in the composition theorems.

- The function $h(x) = \log x$, with $\text{dom } h = \mathbb{R}_{++}$, is concave and satisfies \tilde{h} nondecreasing.
- The function $h(x) = x^{1/2}$, with $\text{dom } h = \mathbb{R}_+$, is concave and satisfies the condition \tilde{h} nondecreasing.
- The function $h(x) = x^{3/2}$, with $\text{dom } h = \mathbb{R}_+$, is convex but does not satisfy the condition \tilde{h} nondecreasing. For example, we have $\tilde{h}(-1) = \infty$, but $\tilde{h}(1) = 1$.
- The function $h(x) = x^{3/2}$ for $x \geq 0$, and $h(x) = 0$ for $x < 0$, with $\text{dom } h = \mathbb{R}$, is convex and does satisfy the condition \tilde{h} nondecreasing.

The composition results (4.30) can be proved directly, without assuming differentiability, or using the formula (4.28). As an example, we will prove the following composition theorem: if g is convex, h is convex, and \tilde{h} is nondecreasing, then $f = h \circ g$ is convex. Assume that $x, y \in \text{dom } f$, and $0 \leq \theta \leq 1$. Since $x, y \in \text{dom } f$, we have that $x, y \in \text{dom } g$ and $g(x), g(y) \in \text{dom } h$. Since $\text{dom } g$ is convex, we conclude that $\theta x + (1 - \theta)y \in \text{dom } g$,

and from convexity of g , we have

$$g(\theta x + (1 - \theta)y) \leq \theta g(x) + (1 - \theta)g(y). \quad (4.31)$$

Since $g(x), g(y) \in \text{dom } h$, we conclude that $\theta g(x) + (1 - \theta)g(y) \in \text{dom } h$, i.e., the righthand side of (4.31) is in $\text{dom } h$. Now we use the assumption that \tilde{h} is nondecreasing, which means that its domain extends infinitely in the negative direction. Since the righthand side of (4.31) is in $\text{dom } h$, we conclude that the lefthand side, i.e., $g(\theta x + (1 - \theta)y) \in \text{dom } h$. This means that $\theta x + (1 - \theta)y \in \text{dom } f$. At this point, we have shown that $\text{dom } f$ is convex.

Now using the fact that \tilde{h} is nondecreasing and the inequality (4.31), we get

$$h(g(\theta x + (1 - \theta)y)) \leq h(\theta g(x) + (1 - \theta)g(y)). \quad (4.32)$$

From convexity of h , we have

$$h(\theta g(x) + (1 - \theta)g(y)) \leq \theta h(g(x)) + (1 - \theta)h(g(y)). \quad (4.33)$$

Putting (4.32) and (4.33) together, we have

$$h(g(\theta x + (1 - \theta)y)) \leq \theta h(g(x)) + (1 - \theta)h(g(y)).$$

which proves the composition theorem.

Example 210 (Simple composition results).

- If g is convex then $\exp g(x)$ is convex.
- If g is concave and positive, then $\log g(x)$ is concave.
- If g is concave and positive, then $1/g(x)$ is convex.
- If g is convex and nonnegative and $p \geq 1$, then $g(x)^p$ is convex.
- If g is convex then $-\log(-g(x))$ is convex on $\{x \mid g(x) < 0\}$.

Remark 211. The requirement that monotonicity hold for the extended-value extension \tilde{h} , and not just the function h , cannot be removed. For example, consider the function $g(x) = x^2$, with $\text{dom } g = \mathbb{R}$, and $h(x) = 0$, with $\text{dom } h = [1, 2]$. Here g is convex, and h is convex and nondecreasing. But the function $f = h \circ g$, given by

$$f(x) = 0, \quad \text{dom } f = [-\sqrt{2}, -1] \cup [1, \sqrt{2}],$$

is not convex, since its domain is not convex. Here, of course, the function \tilde{h} is not nondecreasing.

Vector composition We now turn to the more complicated case when $k \geq 1$. Suppose

$$f(\mathbf{x}) = h(g(\mathbf{x})) = h(g_1(\mathbf{x}), \dots, g_k(\mathbf{x})),$$

with $h : \mathbb{R}^k \rightarrow \mathbb{R}$, $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$. Again without loss of generality we can assume $n = 1$. As in the case $k = 1$, we start by assuming the functions are twice differentiable, with $\text{dom } g = \mathbb{R}$ and $\text{dom } h = \mathbb{R}^k$, in order to discover the composition rules. We have

$$f''(\mathbf{x}) = \mathbf{g}'(\mathbf{x})^T \nabla^2 h(\mathbf{g}(\mathbf{x})) \mathbf{g}'(\mathbf{x}) + \nabla h(\mathbf{g}(\mathbf{x}))^T \mathbf{g}''(\mathbf{x}), \quad (4.34)$$

which is the vector analog of (4.28). Again the issue is to determine conditions under which $f''(\mathbf{x}) \geq 0$ for all \mathbf{x} (or $f''(\mathbf{x}) \leq 0$ for all \mathbf{x} for concavity). From (4.34) we can derive many rules, for example:

- f is convex if h is convex, h is nondecreasing in each argument, and g_i are convex,
- f is convex if h is convex, h is nondecreasing in each argument, and g_i are concave,
- f is concave if h is concave, h is nondecreasing in each argument, and g_i are concave.

As in the scalar case, similar composition results hold in general, with $n > 1$, no assumption of differentiability of h or g , and general domains. For the general results, the monotonicity condition on h must hold for the extended-value extension \tilde{h} .

To understand the meaning of the condition that the extended-value extension \tilde{h} be monotonic, we consider the case where $h : \mathbb{R}^k \rightarrow \mathbb{R}$ is convex, and \tilde{h} nondecreasing, i.e., whenever $\mathbf{u} \preceq \mathbf{v}$, we have $\tilde{h}(\mathbf{u}) \leq \tilde{h}(\mathbf{v})$. This implies that if $\mathbf{v} \in \text{dom } h$, then so is \mathbf{u} : the domain of h must extend infinitely in the $-\mathbb{R}_+^k$ directions. We can express this compactly as $\text{dom } h - \mathbb{R}_+^k = \text{dom } h$.

Example 212 (Vector composition examples).

- Let $h(\mathbf{z}) = z_{[1]} + \dots + z_{[r]}$, the sum of the r largest components of $\mathbf{z} \in \mathbb{R}^k$. Then h is convex and nondecreasing in each argument. Suppose g_1, \dots, g_k are convex functions on \mathbb{R}^n . Then the composition function $f = h \circ \mathbf{g}$, i.e., the pointwise sum of the r largest g_i 's, is convex.
- The function $h(\mathbf{z}) = \log(\sum_{i=1}^k e^{z_i})$ is convex and nondecreasing in each argument, so $\log(\sum_{i=1}^k e^{g_i(\mathbf{x})})$ is convex whenever g_i are.
- For $0 < p \leq 1$, the function $h(\mathbf{z}) = (\sum_{i=1}^k z_i^p)^{1/p}$ on \mathbb{R}_+^k is concave, and its extension (which has the value $-\infty$ for $\mathbf{z} \not\geq \mathbf{0}$) is nondecreasing in each component. So if g_i are concave and nonnegative, we conclude that $f(\mathbf{x}) = (\sum_{i=1}^k g_i(\mathbf{x})^p)^{1/p}$ is concave.

- Suppose $p \geq 1$, and g_1, \dots, g_k are convex and nonnegative. Then the function $f(\mathbf{x}) = (\sum_{i=1}^k g_i(\mathbf{x})^p)^{1/p}$ is convex. To show this, we consider the function $h : \mathbb{R}^k \rightarrow \mathbb{R}$ defined as

$$h(\mathbf{z}) = \left(\sum_{i=1}^k \max\{z_i, 0\}^p \right)^{1/p},$$

with $\text{dom } h = \mathbb{R}^k$, so $h = \tilde{h}$. This function is convex, and nondecreasing, so we conclude $h(g(\mathbf{x}))$ is a convex function of \mathbf{x} . For $\mathbf{z} \succeq \mathbf{0}$, we have $h(\mathbf{z}) = (\sum_{i=1}^k z_i^p)^{1/p}$, so our conclusion is that $(\sum_{i=1}^k g_i(\mathbf{x})^p)^{1/p}$ is convex.

- The geometric mean $h(\mathbf{z}) = (\prod_{i=1}^k z_i)^{1/k}$ on \mathbb{R}_+^k is concave and its extension is nondecreasing in each argument. It follows that if g_1, \dots, g_k are nonnegative concave functions, then so is their geometric mean, $(\prod_{i=1}^k g_i)^{1/k}$.

4.2.5 Minimization

We have seen that the maximum or supremum of an arbitrary family of convex functions is convex. It turns out that some special forms of minimization also yield convex functions. If f is convex in (\mathbf{x}, \mathbf{y}) , and C is a convex nonempty set, then the function

$$g(\mathbf{x}) = \inf_{\mathbf{y} \in C} f(\mathbf{x}, \mathbf{y}) \quad (4.35)$$

is convex in \mathbf{x} , provided $g(\mathbf{x}) > -\infty$ for some \mathbf{x} (which implies $g(\mathbf{x}) > -\infty$ for all \mathbf{x}). The domain of g is the projection of $\text{dom } f$ on its \mathbf{x} -coordinates, i.e.,

$$\text{dom } g = \{\mathbf{x} \mid (\mathbf{x}, \mathbf{y}) \in \text{dom } f \text{ for some } \mathbf{y} \in C\}.$$

We prove this by verifying Jensen's inequality for $\mathbf{x}_1, \mathbf{x}_2 \in \text{dom } g$. Let $\epsilon > 0$. Then there are $\mathbf{y}_1, \mathbf{y}_2 \in C$ such that $f(\mathbf{x}_i, \mathbf{y}_i) \leq g(\mathbf{x}_i) + \epsilon$ for $i = 1, 2$. Now let $\theta \in [0, 1]$. We have

$$\begin{aligned} g(\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2) &= \inf_{\mathbf{y} \in C} f(\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2, \mathbf{y}) \\ &\leq f(\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2, \theta\mathbf{y}_1 + (1 - \theta)\mathbf{y}_2) \\ &\leq \theta f(\mathbf{x}_1, \mathbf{y}_1) + (1 - \theta)f(\mathbf{x}_2, \mathbf{y}_2) \\ &\leq \theta g(\mathbf{x}_1) + (1 - \theta)g(\mathbf{x}_2) + \epsilon. \end{aligned}$$

Since this holds for any $\epsilon > 0$, we have

$$g(\theta\mathbf{x}_1 + (1 - \theta)\mathbf{x}_2) \leq \theta g(\mathbf{x}_1) + (1 - \theta)g(\mathbf{x}_2).$$

The result can also be seen in terms of epigraphs. With f , g , and C defined as in (4.35), and assuming the infimum over $\mathbf{y} \in C$ is attained for each \mathbf{x} , we have

$$\text{epi } g = \{(\mathbf{x}, t) \mid (\mathbf{x}, \mathbf{y}, t) \in \text{epi } f \text{ for some } \mathbf{y} \in C\}.$$

Thus $\text{epi } g$ is convex, since it is the projection of a convex set on some of its components.

Example 213 (Schur complement). Suppose the quadratic function

$$f(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{A} \mathbf{x} + 2\mathbf{x}^T \mathbf{B} \mathbf{y} + \mathbf{y}^T \mathbf{C} \mathbf{y},$$

(where \mathbf{A} and \mathbf{C} are symmetric) is convex in (\mathbf{x}, \mathbf{y}) , which means

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix} \succeq \mathbf{0}.$$

We can express $g(\mathbf{x}) = \inf_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$ as

$$g(\mathbf{x}) = \mathbf{x}^T (\mathbf{A} - \mathbf{B} \mathbf{C}^\dagger \mathbf{B}^T) \mathbf{x},$$

where \mathbf{C}^\dagger is the pseudo-inverse of \mathbf{C} (see Section 2.11.7.4). By the minimization rule, g is convex, so we conclude that $\mathbf{A} - \mathbf{B} \mathbf{C}^\dagger \mathbf{B}^T \succeq \mathbf{0}$.

If \mathbf{C} is invertible, i.e., $\mathbf{C} \succ \mathbf{0}$, then the matrix $\mathbf{A} - \mathbf{B} \mathbf{C}^{-1} \mathbf{B}^T$ is called the Schur complement of \mathbf{C} in the matrix

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix}$$

(see Section 2.11.7.5).

Example 214 (Distance to a set). The distance of a point \mathbf{x} to a set $S \subseteq \mathbb{R}^n$, in the norm $\|\cdot\|$, is defined as

$$\text{dist}(\mathbf{x}, S) = \inf_{\mathbf{y} \in S} \|\mathbf{x} - \mathbf{y}\|.$$

The function $\|\mathbf{x} - \mathbf{y}\|$ is convex in (\mathbf{x}, \mathbf{y}) , so if the set S is convex, the distance function $\text{dist}(\mathbf{x}, S)$ is a convex function of \mathbf{x} .

Example 215. Suppose h is convex. Then the function g defined as

$$g(\mathbf{x}) = \inf\{h(\mathbf{y}) \mid \mathbf{A}\mathbf{y} = \mathbf{x}\}$$

is convex. To see this, we define f by

$$f(\mathbf{x}, \mathbf{y}) = \begin{cases} h(\mathbf{y}), & \text{if } \mathbf{A}\mathbf{y} = \mathbf{x} \\ \infty, & \text{otherwise,} \end{cases}$$

which is convex in (\mathbf{x}, \mathbf{y}) . Then g is the minimum of f over \mathbf{y} , and hence is convex. (It is not hard to show directly that g is convex.)

4.2.6 Perspective of a function

If $f : \mathbb{R}^n \rightarrow \mathbb{R}$, then the *perspective* of f is the function $g : \mathbb{R}^{n+1} \rightarrow \mathbb{R}$ defined by

$$g(\mathbf{x}, t) = tf(\mathbf{x}/t),$$

with domain

$$\text{dom } g = \{(\mathbf{x}, t) \mid \mathbf{x}/t \in \text{dom } f, t > 0\}.$$

The perspective operation preserves convexity: If f is a convex function, then so is its perspective function g . Similarly, if f is concave, then so is g .

This can be proved several ways, for example, direct verification of the defining inequality (see Exercise 245). We give a short proof here using epigraphs and the perspective mapping on \mathbb{R}^{n+1} described in Section 3.3.3 (which will also explain the name ‘perspective’). For $t > 0$ we have

$$\begin{aligned} (\mathbf{x}, t, s) \in \text{epi } g &\iff tf(\mathbf{x}/t) \leq s \\ &\iff f(\mathbf{x}/t) \leq s/t \\ &\iff (\mathbf{x}/t, s/t) \in \text{epi } f. \end{aligned}$$

Therefore $\text{epi } g$ is the inverse image of $\text{epi } f$ under the perspective mapping that takes (u, v, w) to $(u, w)/v$. It follows (see Section 3.3.3) that $\text{epi } g$ is convex, so the function g is convex.

Example 216 (Euclidean norm squared). *The perspective of the convex function $f(\mathbf{x}) = \langle \mathbf{x}, \mathbf{x} \rangle$ on \mathbb{R}^n is*

$$g(\mathbf{x}, t) = t \langle \mathbf{x}/t, \mathbf{x}/t \rangle = \frac{\langle \mathbf{x}, \mathbf{x} \rangle}{t},$$

which is convex in (\mathbf{x}, t) for $t > 0$.

We can deduce convexity of g using several other methods. First, we can express g as the sum of the quadratic-over-linear functions x_i^2/t , which were shown to be convex in Section 4.1.5. We can also express g as a special case of the matrix fractional function $\mathbf{x}^T(t\mathbf{I})^{-1}\mathbf{x}$ (see Example 181).

Example 217 (Negative logarithm). *Consider the convex function $f(x) = -\log x$ on \mathbb{R}_{++} . Its perspective is*

$$g(x, t) = -t \log(x/t) = t \log(t/x) = t \log t - t \log x,$$

and is convex on \mathbb{R}_{++}^2 . The function g is called the relative entropy of t and x . For $x = 1$, g reduces to the negative entropy function.

From convexity of g we can establish convexity or concavity of several interesting related functions. First, the relative entropy of two vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}_{++}^n$, defined as

$$\sum_{i=1}^n u_i \log(u_i/v_i),$$

is convex in (\mathbf{u}, \mathbf{v}) , since it is a sum of relative entropies of u_i, v_i .

A closely related function is the Kullback-Leibler divergence between $\mathbf{u}, \mathbf{v} \in \mathbb{R}_{++}^n$, given by

$$D_{kl}(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^n (u_i \log(u_i/v_i) - u_i + v_i), \quad (4.36)$$

which is convex, since it is the relative entropy plus a linear function of (\mathbf{u}, \mathbf{v}) . The Kullback-Leibler divergence satisfies $D_{kl}(\mathbf{u}, \mathbf{v}) \geq 0$, and $D_{kl}(\mathbf{u}, \mathbf{v}) = 0$ if and only if $\mathbf{u} = \mathbf{v}$, and so can be used as a measure of deviation between two positive vectors; see Exercise 278. (Note that the relative entropy and the Kullback-Leibler divergence are the same when \mathbf{u} and \mathbf{v} are probability vectors, i.e., satisfy $\langle \mathbf{1}, \mathbf{u} \rangle = \langle \mathbf{1}, \mathbf{v} \rangle = 1$.)

If we take $v_i = \langle \mathbf{1}, \mathbf{u} \rangle$ in the relative entropy function, we obtain the concave (and homogeneous) function of $\mathbf{u} \in \mathbb{R}_{++}^n$ given by

$$\sum_{i=1}^n u_i \log(\langle \mathbf{1}, \mathbf{u} \rangle / u_i) = \langle \mathbf{1}, \mathbf{u} \rangle \sum_{i=1}^n z_i \log(1/z_i),$$

where $\mathbf{z} = \mathbf{u} / \langle \mathbf{1}, \mathbf{u} \rangle$, which is called the normalized entropy function. The vector $\mathbf{z} = \mathbf{u} / \langle \mathbf{1}, \mathbf{u} \rangle$ is a normalized vector or probability distribution, since its components sum to one; the normalized entropy of \mathbf{u} is $\langle \mathbf{1}, \mathbf{u} \rangle$ times the entropy of this normalized distribution.

Example 218. Suppose $f : \mathbb{R}^m \rightarrow \mathbb{R}$ is convex, and $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{c} \in \mathbb{R}^n$, and $d \in \mathbb{R}$. We define

$$g(\mathbf{x}) = (\langle \mathbf{c}, \mathbf{x} \rangle + d)f((\mathbf{Ax} + \mathbf{b}) / (\langle \mathbf{c}, \mathbf{x} \rangle + d)),$$

with

$$\text{dom } g = \{\mathbf{x} \mid \langle \mathbf{c}, \mathbf{x} \rangle + d > 0, (\mathbf{Ax} + \mathbf{b}) / (\langle \mathbf{c}, \mathbf{x} \rangle + d) \in \text{dom } f\}.$$

Then g is convex.

4.3 The conjugate function

In this section we introduce an operation that will play an important role in later chapters.

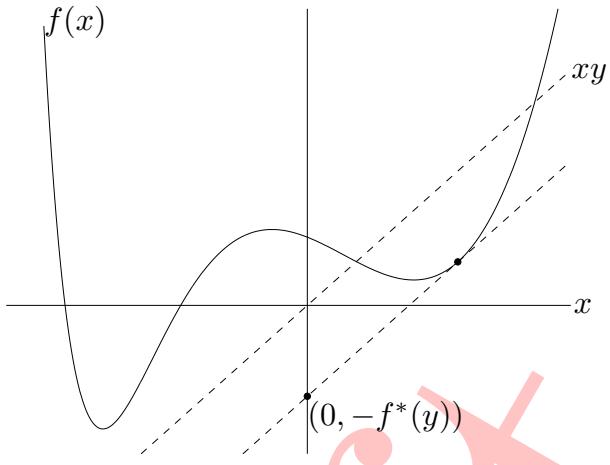


图 4.11: A function $f : \mathbb{R} \rightarrow \mathbb{R}$, and a value $y \in \mathbb{R}$. The conjugate function $f^*(y)$ is the maximum gap between the linear function yx and $f(x)$, as shown by the dashed line in the figure. If f is differentiable, this occurs at a point x where $f'(x) = y$.

4.3.1 Definition and examples

Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$. The function

$$f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \text{dom } f} (\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x})), \quad (4.37)$$

is called the *conjugate* of the function f . The domain of the conjugate function consists of $\mathbf{y} \in \mathbb{R}^n$ for which the supremum is finite, *i.e.*, for which the difference $\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x})$ is bounded above on $\text{dom } f$. This definition is illustrated in Figure 4.11.

We see immediately that f^* is a convex function, since it is the pointwise supremum of a family of convex (indeed, affine) functions of \mathbf{y} . This is true whether or not f is convex. (Note that when f is convex, the subscript $\mathbf{x} \in \text{dom } f$ is not necessary since, by convention, $\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x}) = -\infty$ for $\mathbf{x} \notin \text{dom } f$.)

We start with some simple examples, and then describe some rules for conjugating functions. This allows us to derive an analytical expression for the conjugate of many common convex functions.

Example 219. We derive the conjugates of some convex functions on \mathbb{R} .

- Affine function. $f(x) = ax + b$. As a function of x , $yx - ax - b$ is bounded if and only if $y = a$, in which case it is constant. Therefore the domain of the conjugate function f^* is the singleton $\{a\}$, and $f^*(a) = -b$.
- Negative logarithm. $f(x) = -\log x$, with $\text{dom } f = \mathbb{R}_{++}$. The function $xy + \log x$ is unbounded above if $y \geq 0$ and reaches its maximum at $x = -1/y$ otherwise. Therefore, $\text{dom } f^* = \{y \mid y < 0\} = -\mathbb{R}_{++}$ and $f^*(y) = -\log(-y) - 1$ for $y < 0$.

- Exponential. $f(x) = e^x$. $xy - e^x$ is unbounded if $y < 0$. For $y > 0$, $xy - e^x$ reaches its maximum at $x = \log y$, so we have $f^*(y) = y \log y - y$. For $y = 0$, $f^*(y) = \sup_x -e^x = 0$. In summary, $\text{dom } f^* = \mathbb{R}_+$ and $f^*(y) = y \log y - y$ (with the interpretation $0 \log 0 = 0$).
- Negative entropy. $f(x) = x \log x$, with $\text{dom } f = \mathbb{R}_+$ (and $f(0) = 0$). The function $xy - x \log x$ is bounded above on \mathbb{R}_+ for all y , hence $\text{dom } f^* = \mathbb{R}$. It attains its maximum at $x = e^{y-1}$, and substituting we find $f^*(y) = e^{y-1}$.
- Inverse. $f(x) = 1/x$ on \mathbb{R}_{++} . For $y > 0$, $yx - 1/x$ is unbounded above. For $y = 0$ this function has supremum 0; for $y < 0$ the supremum is attained at $x = (-y)^{-1/2}$. Therefore we have $f^*(y) = -2(-y)^{1/2}$, with $\text{dom } f^* = -\mathbb{R}_+$.

Example 220 (Strictly convex quadratic function). Consider $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q}\mathbf{x}$, with $\mathbf{Q} \in \mathbb{S}_{++}^n$. The function $\langle \mathbf{y}, \mathbf{x} \rangle - \frac{1}{2}\mathbf{x}^T \mathbf{Q}\mathbf{x}$ is bounded above as a function of \mathbf{x} for all \mathbf{y} . It attains its maximum at $\mathbf{x} = \mathbf{Q}^{-1}\mathbf{y}$, so

$$f^*(\mathbf{y}) = \frac{1}{2}\mathbf{y}^T \mathbf{Q}^{-1}\mathbf{y}.$$

Example 221 (Log-determinant). We consider $f(\mathbf{X}) = \log \det \mathbf{X}^{-1}$ on \mathbb{S}_{++}^n . The conjugate function is defined as

$$f^*(\mathbf{Y}) = \sup_{\mathbf{X} > 0} (\text{tr}(\mathbf{Y}\mathbf{X}) + \log \det \mathbf{X}),$$

since $\text{tr}(\mathbf{Y}\mathbf{X})$ is the standard inner product on \mathbb{S}^n . We first show that $\text{tr}(\mathbf{Y}\mathbf{X}) + \log \det \mathbf{X}$ is unbounded above unless $\mathbf{Y} \prec \mathbf{0}$. If $\mathbf{Y} \not\prec \mathbf{0}$, then \mathbf{Y} has an eigenvector \mathbf{v} , with $\|\mathbf{v}\|_2 = 1$, and eigenvalue $\lambda \geq 0$. Taking $\mathbf{X} = \mathbf{I} + t\mathbf{v}\mathbf{v}^T$ we find that

$$\begin{aligned} \text{tr}(\mathbf{Y}\mathbf{X}) + \log \det \mathbf{X} &= \text{tr}(\mathbf{Y}) + t\lambda + \log \det(\mathbf{I} + t\mathbf{v}\mathbf{v}^T) \\ &= \text{tr}(\mathbf{Y}) + t\lambda + \log(1 + t), \end{aligned}$$

which is unbounded above as $t \rightarrow \infty$.

Now consider the case $\mathbf{Y} \prec \mathbf{0}$. We can find the maximizing \mathbf{X} by setting the gradient with respect to \mathbf{X} equal to zero:

$$\nabla_{\mathbf{X}} (\text{tr}(\mathbf{Y}\mathbf{X}) + \log \det \mathbf{X}) = \mathbf{Y} + \mathbf{X}^{-1} = \mathbf{0}$$

(see Section 2.11.4.1), which yields $\mathbf{X} = -\mathbf{Y}^{-1}$ (which is, indeed, positive definite). Therefore we have

$$f^*(\mathbf{Y}) = \log \det(-\mathbf{Y})^{-1} - n,$$

with $\text{dom } f^* = -\mathbb{S}_{++}^n$.

Example 222 (Indicator function). Let I_S be the indicator function of a (not necessarily convex) set $S \subseteq \mathbb{R}^n$, i.e., $I_S(\mathbf{x}) = 0$ on $\text{dom } I_S = S$. Its conjugate is

$$I_S^*(\mathbf{y}) = \sup_{\mathbf{x} \in S} \langle \mathbf{y}, \mathbf{x} \rangle,$$

which is the support function of the set S .

Example 223 (Log-sum-exp function). To derive the conjugate of the log-sum-exp function $f(\mathbf{x}) = \log(\sum_{i=1}^n e^{x_i})$, we first determine the values of \mathbf{y} for which the maximum over \mathbf{x} of $\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x})$ is attained. By setting the gradient with respect to \mathbf{x} equal to zero, we obtain the condition

$$y_i = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}, \quad i = 1, \dots, n.$$

These equations are solvable for \mathbf{x} if and only if $\mathbf{y} \succ \mathbf{0}$ and $\langle \mathbf{1}, \mathbf{y} \rangle = 1$. By substituting the expression for y_i into $\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x})$ we obtain $f^*(\mathbf{y}) = \sum_{i=1}^n y_i \log y_i$. This expression for f^* is still correct if some components of \mathbf{y} are zero, as long as $\mathbf{y} \succeq \mathbf{0}$ and $\langle \mathbf{1}, \mathbf{y} \rangle = 1$, and we interpret $0 \log 0$ as 0.

In fact the domain of f^* is exactly given by $\langle \mathbf{1}, \mathbf{y} \rangle = 1$, $\mathbf{y} \succeq \mathbf{0}$. To show this, suppose that a component of \mathbf{y} is negative, say, $y_k < 0$. Then we can show that $\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x})$ is unbounded above by choosing $x_k = -t$, and $x_i = 0$, $i \neq k$, and letting t go to infinity.

If $\mathbf{y} \succeq \mathbf{0}$ but $\langle \mathbf{1}, \mathbf{y} \rangle \neq 1$, we choose $\mathbf{x} = t\mathbf{1}$, so that

$$\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x}) = t \langle \mathbf{1}, \mathbf{y} \rangle - t - \log n.$$

If $\langle \mathbf{1}, \mathbf{y} \rangle > 1$, this grows unboundedly as $t \rightarrow \infty$; if $\langle \mathbf{1}, \mathbf{y} \rangle < 1$, it grows unboundedly as $t \rightarrow -\infty$.

In summary,

$$f^*(\mathbf{y}) = \begin{cases} \sum_{i=1}^n y_i \log y_i, & \text{if } \mathbf{y} \succeq \mathbf{0} \text{ and } \langle \mathbf{1}, \mathbf{y} \rangle = 1 \\ \infty, & \text{otherwise.} \end{cases}$$

In other words, the conjugate of the log-sum-exp function is the negative entropy function, restricted to the probability simplex.

Example 224 (Norm). Let $\|\cdot\|$ be a norm on \mathbb{R}^n , with dual norm $\|\cdot\|_*$. We will show that the conjugate of $f(\mathbf{x}) = \|\mathbf{x}\|$ is

$$f^*(\mathbf{y}) = \begin{cases} 0, & \|\mathbf{y}\|_* \leq 1 \\ \infty, & \text{otherwise,} \end{cases}$$

i.e., the conjugate of a norm is the indicator function of the dual norm unit ball.

If $\|\mathbf{y}\|_* > 1$, then by definition of the dual norm, there is a $\mathbf{z} \in \mathbb{R}^n$ with $\|\mathbf{z}\| \leq 1$ and $\langle \mathbf{y}, \mathbf{z} \rangle > 1$. Taking $\mathbf{x} = t\mathbf{z}$ and letting $t \rightarrow \infty$, we have

$$\langle \mathbf{y}, \mathbf{x} \rangle - \|\mathbf{x}\| = t(\langle \mathbf{y}, \mathbf{z} \rangle - \|\mathbf{z}\|) \rightarrow \infty,$$

which shows that $f^*(\mathbf{y}) = \infty$. Conversely, if $\|\mathbf{y}\|_* \leq 1$, then we have $\langle \mathbf{y}, \mathbf{x} \rangle \leq \|\mathbf{x}\| \|\mathbf{y}\|_*$ for all \mathbf{x} , which implies for all \mathbf{x} , $\langle \mathbf{y}, \mathbf{x} \rangle - \|\mathbf{x}\| \leq 0$. Therefore $\mathbf{x} = \mathbf{0}$ is the value that maximizes $\langle \mathbf{y}, \mathbf{x} \rangle - \|\mathbf{x}\|$, with maximum value 0.

Example 225 (Norm squared). Now consider the function $f(\mathbf{x}) = (1/2) \|\mathbf{x}\|^2$, where $\|\cdot\|$ is a norm, with dual norm $\|\cdot\|_*$. We will show that its conjugate is $f^*(\mathbf{y}) = (1/2) \|\mathbf{y}\|_*^2$. From $\langle \mathbf{y}, \mathbf{x} \rangle \leq \|\mathbf{y}\|_* \|\mathbf{x}\|$, we conclude

$$\langle \mathbf{y}, \mathbf{x} \rangle - (1/2) \|\mathbf{x}\|^2 \leq \|\mathbf{y}\|_* \|\mathbf{x}\| - (1/2) \|\mathbf{x}\|^2$$

for all \mathbf{x} . The righthand side is a quadratic function of $\|\mathbf{x}\|$, which has maximum value $(1/2) \|\mathbf{y}\|_*^2$. Therefore for all \mathbf{x} , we have

$$\langle \mathbf{y}, \mathbf{x} \rangle - (1/2) \|\mathbf{x}\|^2 \leq (1/2) \|\mathbf{y}\|_*^2,$$

which shows that $f^*(\mathbf{y}) \leq (1/2) \|\mathbf{y}\|_*^2$.

To show the other inequality, let \mathbf{x} be any vector with $\langle \mathbf{y}, \mathbf{x} \rangle = \|\mathbf{y}\|_* \|\mathbf{x}\|$, scaled so that $\|\mathbf{x}\| = \|\mathbf{y}\|_*$. Then we have, for this \mathbf{x} ,

$$\langle \mathbf{y}, \mathbf{x} \rangle - (1/2) \|\mathbf{x}\|^2 = (1/2) \|\mathbf{y}\|_*^2,$$

which shows that $f^*(\mathbf{y}) \geq (1/2) \|\mathbf{y}\|_*^2$.

Example 226 (Revenue and profit functions). We consider a business or enterprise that consumes n resources and produces a product that can be sold. We let $\mathbf{r} = (r_1, \dots, r_n)$ denote the vector of resource quantities consumed, and $S(\mathbf{r})$ denote the sales revenue derived from the product produced (as a function of the resources consumed). Now let p_i denote the price (per unit) of resource i , so the total amount paid for resources by the enterprise is $\langle \mathbf{p}, \mathbf{r} \rangle$. The profit derived by the firm is then $S(\mathbf{r}) - \langle \mathbf{p}, \mathbf{r} \rangle$. Let us fix the prices of the resources, and ask what is the maximum profit that can be made, by wisely choosing the quantities of resources consumed. This maximum profit is given by

$$M(\mathbf{p}) = \sup_{\mathbf{r}} (S(\mathbf{r}) - \langle \mathbf{p}, \mathbf{r} \rangle).$$

The function $M(\mathbf{p})$ gives the maximum profit attainable, as a function of the resource prices. In terms of conjugate functions, we can express M as

$$M(\mathbf{p}) = (-S)^*(-\mathbf{p}).$$

Thus the maximum profit (as a function of resource prices) is closely related to the conjugate of gross sales (as a function of resources consumed).

4.3.2 Basic properties

Fenchel's inequality

From the definition of conjugate function, we immediately obtain the inequality

$$f(\mathbf{x}) + f^*(\mathbf{y}) \geq \langle \mathbf{x}, \mathbf{y} \rangle$$

for all \mathbf{x}, \mathbf{y} . This is called *Fenchel's inequality* (or *Young's inequality* when f is differentiable).

For example with $f(\mathbf{x}) = (1/2)\mathbf{x}^T \mathbf{Q} \mathbf{x}$, where $\mathbf{Q} \in \mathbb{S}_{++}^n$, we obtain the inequality

$$\langle \mathbf{x}, \mathbf{y} \rangle \leq (1/2)\mathbf{x}^T \mathbf{Q} \mathbf{x} + (1/2)\mathbf{y}^T \mathbf{Q}^{-1} \mathbf{y}.$$

Conjugate of the conjugate

The examples above, and the name ‘conjugate’, suggest that the conjugate of the conjugate of a convex function is the original function. This is the case provided a technical condition holds: if f is convex, and f is closed (*i.e.*, $\text{epi } f$ is a closed set; see Section 2.11.3.3), then $f^{**} = f$. For example, if $\text{dom } f = \mathbb{R}^n$, then we have $f^{**} = f$, *i.e.*, the conjugate of the conjugate of f is f again (see Exercise 304).

Differentiable functions

The conjugate of a differentiable function f is also called the *Legendre transform* of f . (To distinguish the general definition from the differentiable case, the term *Fenchel conjugate* is sometimes used instead of conjugate.)

Suppose f is convex and differentiable, with $\text{dom } f = \mathbb{R}^n$. Any maximizer \mathbf{x}^* of $\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x})$ satisfies $\mathbf{y} = \nabla f(\mathbf{x}^*)$, and conversely, if \mathbf{x}^* satisfies $\mathbf{y} = \nabla f(\mathbf{x}^*)$, then \mathbf{x}^* maximizes $\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x})$. Therefore, if $\mathbf{y} = \nabla f(\mathbf{x}^*)$, we have

$$f^*(\mathbf{y}) = \mathbf{x}^{*T} \nabla f(\mathbf{x}^*) - f(\mathbf{x}^*).$$

This allows us to determine $f^*(\mathbf{y})$ for any \mathbf{y} for which we can solve the gradient equation $\mathbf{y} = \nabla f(\mathbf{z})$ for \mathbf{z} .

We can express this another way. Let $\mathbf{z} \in \mathbb{R}^n$ be arbitrary and define $\mathbf{y} = \nabla f(\mathbf{z})$. Then we have

$$f^*(\mathbf{y}) = \mathbf{z}^T \nabla f(\mathbf{z}) - f(\mathbf{z}).$$

Scaling and composition with affine transformation

For $a > 0$ and $b \in \mathbb{R}$, the conjugate of $g(x) = af(x) + b$ is $g^*(y) = af^*(y/a) - b$.

Suppose $\mathbf{A} \in \mathbb{R}^{n \times n}$ is nonsingular and $\mathbf{b} \in \mathbb{R}^n$. Then the conjugate of $g(\mathbf{x}) = f(\mathbf{Ax} + \mathbf{b})$ is

$$g^*(\mathbf{y}) = f^*(\mathbf{A}^{-T}\mathbf{y}) - \mathbf{b}^T \mathbf{A}^{-T}\mathbf{y},$$

with $\text{dom } g^* = \mathbf{A}^T \text{dom } f^*$.

Separable functions

If $f(\mathbf{u}, \mathbf{v}) = f_1(\mathbf{u}) + f_2(\mathbf{v})$, where f_1 and f_2 are convex functions with conjugates f_1^* and f_2^* , respectively, then

$$f^*(\mathbf{w}, \mathbf{z}) = f_1^*(\mathbf{w}) + f_2^*(\mathbf{z}).$$

In other words, the conjugate of a *separable* convex function is the sum of the conjugates. ('separable' means the component functions are of different variables.)

4.4 Envelope function and proximal mapping

(Taken from page 160 of [14])

Definition 227. Let $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ be a closed proper function. For a scalar $c > 0$, define the corresponding envelope function $E_c f$ and the proximal mapping $P_c f$ by

$$\begin{aligned} E_c f(\mathbf{x}) &= \inf_{\mathbf{w}} \left\{ f(\mathbf{w}) + \frac{1}{2c} \|\mathbf{w} - \mathbf{x}\|^2 \right\}, \\ P_c f(\mathbf{x}) &= \underset{\mathbf{w}}{\operatorname{argmin}} \left\{ f(\mathbf{w}) + \frac{1}{2c} \|\mathbf{w} - \mathbf{x}\|^2 \right\}. \end{aligned} \quad (4.38)$$

Example: $f(\mathbf{x}) = \chi_C(\mathbf{x})$, $f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x} + \mathbf{c}$, $f(\mathbf{x}) = \|\mathbf{x}\|_1$, $f(\mathbf{x}) = \sum_{i=1}^n \log x_i$, $f(\mathbf{X}) = \|\mathbf{X}\|_*$ (see [22]).

Theorem 228. For each $\tau > 0$ and $\mathbf{Y} \in \mathbb{R}^{n_1 \times n_2}$,

$$\mathcal{D}_\tau(\mathbf{Y}) = \underset{\mathbf{X}}{\operatorname{argmin}} \left\{ \tau \|\mathbf{X}\|_* + \frac{1}{2} \|\mathbf{X} - \mathbf{Y}\|_F^2 \right\}, \quad (4.39)$$

where $\mathcal{D}_\tau(\mathbf{Y})$ is the singular value thresholding operator defined as

$$\mathcal{D}_\tau(\mathbf{Y}) = \mathbf{U} \operatorname{diag}(\{(\sigma_i - \tau)_+\}) \mathbf{V}^T, \quad (4.40)$$

in which $\mathbf{U} \operatorname{diag}(\{\sigma_i\}) \mathbf{V}^T$ is the SVD of \mathbf{Y} .

Proof. Since the function $h_0(\mathbf{X}) \triangleq \tau\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X} - \mathbf{Y}\|_F^2$ is strictly convex, it is easy to see that there exists a unique minimizer, and we thus need to prove that it is equal to $\mathcal{D}_\tau(\mathbf{Y})$.

$\hat{\mathbf{X}}$ minimizes h_0 if and only if 0 is a subgradient of the functional h_0 at the point $\hat{\mathbf{X}}$, i.e.

$$0 \in \tau\partial\|\hat{\mathbf{X}}\|_* + \hat{\mathbf{X}} - \mathbf{Y}. \quad (4.41)$$

We need to prove that $\hat{\mathbf{X}} = \mathcal{D}_\tau(\mathbf{Y})$ fulfils the above condition.

Let the SVD of \mathbf{Y} be

$$\mathbf{Y} = \mathbf{U}_0 \Sigma_0 \mathbf{V}_0^T + \mathbf{U}_1 \Sigma_1 \mathbf{V}_1^T,$$

where \mathbf{U}_0 , \mathbf{V}_0 (resp. \mathbf{U}_1 , \mathbf{V}_1) are the singular vectors associated with singular values greater than τ (resp. smaller than or equal to τ). With these notations, we may write

$$\hat{\mathbf{X}} = \mathbf{U}_0 (\Sigma_0 - \tau \mathbf{I}) \mathbf{V}_0^T,$$

and, therefore,

$$\mathbf{Y} - \hat{\mathbf{X}} = \tau(\mathbf{U}_0 \mathbf{V}_0^T + \mathbf{W}), \text{ where } \mathbf{W} = \tau^{-1} \mathbf{U}_1 \Sigma_1 \mathbf{V}_1^T.$$

We can verify that $\mathbf{Y} - \hat{\mathbf{X}} \in \tau\partial\|\hat{\mathbf{X}}\|_*$. □

Remark 229. 1. The envelope function $E_c f$ is an underestimate of the function f , i.e., $E_c f(\mathbf{x}) \leq f(\mathbf{x})$, $\forall \mathbf{x} \in \mathbb{R}^n$. Furthermore, $E_c f$ is a real-valued continuous function, whereas f itself may only be extended real-valued and lower semi-continuous.
 2. It is obvious that $\mathbf{u} = P_c f(\mathbf{x}) \Leftrightarrow \mathbf{x} - \mathbf{u} \in \partial f(\mathbf{u})$.

Theorem 230. For any function f , $P_c f(\mathbf{x})$ is a monotonic function of \mathbf{x} in the sense that:

$$\langle \mathbf{x}_1 - \mathbf{x}_2, \mathbf{y}_1 - \mathbf{y}_2 \rangle \geq 0, \quad \forall \mathbf{y}_i \in P_c f(\mathbf{x}_i), i = 1, 2. \quad (4.42)$$

Proof. By the definition of \mathbf{y}_i ,

$$\begin{aligned} f(\mathbf{y}_1) + \frac{1}{2c} \|\mathbf{y}_1 - \mathbf{x}_1\|^2 &\leq f(\mathbf{y}_2) + \frac{1}{2c} \|\mathbf{y}_2 - \mathbf{x}_1\|^2, \\ f(\mathbf{y}_2) + \frac{1}{2c} \|\mathbf{y}_2 - \mathbf{x}_2\|^2 &\leq f(\mathbf{y}_1) + \frac{1}{2c} \|\mathbf{y}_1 - \mathbf{x}_2\|^2. \end{aligned} \quad (4.43)$$

Adding them together yields (4.42). □

Proposition 231. Let $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ be a closed proper convex function and $c > 0$. The envelope function $E_c f$ is convex and smooth and its gradient is given by

$$\nabla E_c f(\mathbf{x}) = \frac{1}{c}(\mathbf{x} - P_c f(\mathbf{x})). \quad (4.44)$$

Remark 232. The envelope function $E_c f$ is smooth, regardless of whether f is smooth.

Proposition 233. Let $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ be a closed proper convex function and $c > 0$. The proximal mapping $P_c f$ is single-valued and is continuous in the sense that $P_c f(x) \rightarrow P_{c^*} f(x^*)$ whenever $(x, c) \rightarrow (x^*, c^*)$, with $c^* > 0$.

Proposition 234. Let $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ be L -smooth, $F(\mathbf{x}) = f(\mathbf{x}) + g(\mathbf{x})$, $\mathbf{x}_{k+1} \in P_c g(\mathbf{x}^k - c\nabla f(\mathbf{x}^k))$, where $0 < c < L^{-1}$. Prove

$$F(\mathbf{x}^{k+1}) - F(\mathbf{x}^k) \leq \frac{1}{2}(c^{-1} - L)\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2,$$

i.e., $F(\mathbf{x})$ has sufficient descent on $\{\mathbf{x}_k\}$.

Theorem 235 (Moreau Decomposition). $\mathbf{x} = P_c f(\mathbf{x}) + P_{c^*} f^*(\mathbf{x})$, $\forall \mathbf{x}$.

Exercise 236. 1. Prove the last equalities in (4.22) and (4.23).

2. Find the envelope function and proximal mapping of $f(\mathbf{x}) = \|\mathbf{x}\|_2$.
3. Prove that if $f(\mathbf{x}) = g(\lambda\mathbf{x} + \mathbf{a})$, where $\lambda \neq 0$, then $P_c f(\mathbf{x}) = \frac{1}{\lambda} [P_c(\lambda^2 g)(\lambda\mathbf{x} + \mathbf{a}) - \mathbf{a}]$.
4. Prove that if $f(\mathbf{x}) = \lambda g(\mathbf{x}/\lambda)$, where $\lambda > 0$, then $P_c f(\mathbf{x}) = \lambda P_c(\lambda^{-1} g)(\mathbf{x}/\lambda)$.
5. Let $f(\mathbf{X})$ be the indicator function on the set S_+ of psd matrices. Compute its proximal mapping.
6. Use Moreau decomposition to prove that $\mathbf{x} = P_L(\mathbf{x}) + P_{L^\perp}(\mathbf{x})$, where L is a subspace and L^\perp is its orthogonal complement.

Exercise 237. Let

$$\mathbf{A} = \begin{pmatrix} 1 & -1 & & \\ & 1 & -1 & \\ & & \ddots & \ddots \\ & & & 1 & -1 \end{pmatrix}.$$

Find the proximal mapping of function $f(\mathbf{x}) = \lambda_1 \|\mathbf{x}\|_1 + \lambda_2 \|\mathbf{Ax}\|_1$.

Exercise 238. Prove that:

1. If $f(\mathbf{x}) = g(\mathbf{x}) + \mathbf{a}^T \mathbf{x}$, then $P_c f(\mathbf{x}) = P_c g(\mathbf{x} + c\mathbf{a})$.
2. If $f(\mathbf{x}) = g(\mathbf{x}) + \frac{1}{2\mu} \|\mathbf{x} - \mathbf{a}\|^2$, then $P_c f(\mathbf{x}) = P_\lambda g(\lambda(c^{-1}\mathbf{x} + \mu^{-1}\mathbf{a}))$, where $\lambda^{-1} = \mu^{-1} + c^{-1}$.

Exercise 239. Find the projection onto the second order cone.

Exercise 240. Let $d(\mathbf{x})$ be the distance from \mathbf{x} to a closed convex set \mathcal{C} : $d(\mathbf{x}) = \inf_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x} - \mathbf{y}\|_2$. Find the proximal mappings of $d(\mathbf{x})$ and $\frac{1}{2}d^2(\mathbf{x})$.

4.5 Quasiconvex functions

4.5.1 Definition and examples

A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is called *quasiconcave* (or *unimodal*) if its domain and all its sublevel sets

$$\mathbb{S}_\alpha = \{\mathbf{x} \in \text{dom } f \mid f(\mathbf{x}) \leq \alpha\},$$

for $\alpha \in \mathbb{R}$, are convex. A function is *quasiconcave* if $-f$ is quasiconvex, i.e., every superlevel set $\{\mathbf{x} \mid f(\mathbf{x}) \geq \alpha\}$ is convex. A function that is both quasiconvex and quasiconcave is called *quasilinear*. A function is *quasilinear* if its domain, and every level set $\{\mathbf{x} \mid f(\mathbf{x}) = \alpha\}$ is convex.

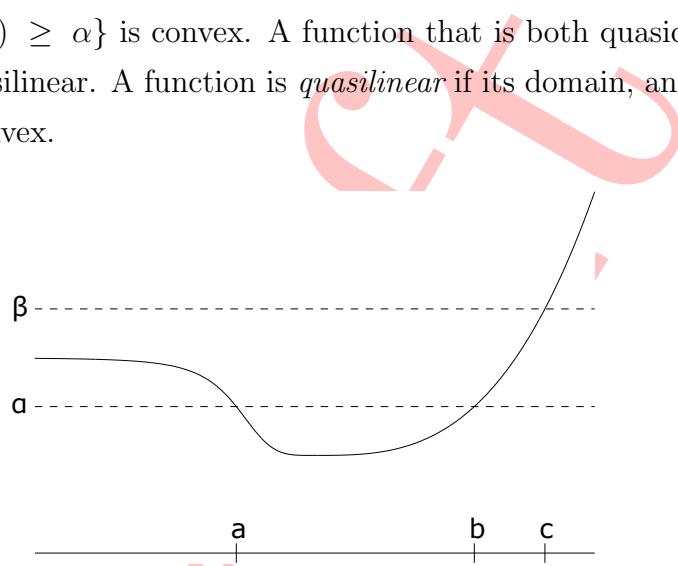


图 4.12: A quasiconvex function on \mathbb{R} . For each α , the α -sublevel set \mathbb{S}_α is convex, i.e., an interval. The sublevel set \mathbb{S}_α is the interval $[a, b]$. The sublevel set \mathbb{S}_β is the interval $(-1, c]$.

For a function on \mathbb{R} , quasiconvexity requires that each sublevel set be an interval (including, possibly, an infinite interval). An example of a quasiconvex function on \mathbb{R} is shown in Figure 4.12.

Convex functions have convex sublevel sets, and so are quasiconvex. But simple examples, such as the one shown in Figure 4.12, show that the converse is not true.

Example 241. Some examples on \mathbb{R} :

- *Logarithm.* $\log x$ on \mathbb{R}_{++} is quasiconvex (and quasiconcave, hence quasilinear).
- *Ceiling function.* $\text{ceil}(x) = \inf \{z \in \mathbb{Z} \mid z \geq x\}$ is quasiconvex (and quasiconcave).

These examples show that quasiconvex functions can be concave, or discontinuous. We now give some examples on \mathbb{R}^n .

Example 242. *Length of a vector.* We define the length of $\mathbf{x} \in \mathbb{R}^n$ as the largest index of a nonzero component, *i.e.*,

$$f(\mathbf{x}) = \max \{i \mid x_i \neq 0\}.$$

We define the length of the zero vector to be zero. This function is quasiconvex on \mathbb{R}^n , since its sublevel sets are subspaces:

$$f(\mathbf{x}) \geq \alpha \Leftrightarrow x_i = 0 \text{ for } \lfloor \alpha \rfloor + 1, \dots, n.$$

Example 243. Consider $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, with $\text{dom } f = \mathbb{R}_+$ and $f(x_1, x_2) = x_1 x_2$. This function is neither convex nor concave since its Hessian

$$\nabla^2 f(\mathbf{x}) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

is indefinite; it has one positive and one negative eigenvalue. The function f is quasiconcave, however, since the superlevel sets

$$\{\mathbf{x} \in \mathbb{R}_+^2 \mid x_1 x_2 \geq \alpha\}$$

are convex sets for all α . (Note, however, that f is not quasiconcave on \mathbb{R}^2 .)

Example 244. *Linear-fractional function.* The function

$$f(\mathbf{x}) = \frac{\mathbf{a}^T \mathbf{x} + b}{\mathbf{c}^T \mathbf{x} + d}$$

with $\text{dom } f = \{\mathbf{c}^T \mathbf{x} + d \geq 0\}$, is quasiconvex, and quasiconcave, *i.e.*, quasilinear. Its α -sublevel set is

$$\begin{aligned} \mathbb{S}_\alpha &= \{\mathbf{x} \mid \mathbf{c}^T \mathbf{x} + d > 0, (\mathbf{a}^T \mathbf{x} + b)/(\mathbf{c}^T \mathbf{x} + d) \leq \alpha\} \\ &= \{\mathbf{x} \mid \mathbf{c}^T \mathbf{x} + d > 0, \mathbf{a}^T \mathbf{x} + b \leq \alpha(\mathbf{c}^T \mathbf{x} + d)\} \end{aligned}$$

which is convex, since it is the intersection of an open halfspace and a closed halfspace. (The same method can be used to show its superlevel sets are convex.)

Example 245. *Distance ratio function.* Suppose $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$, and define

$$f(\mathbf{x}) = \frac{\|\mathbf{x} - \mathbf{a}\|_2}{\|\mathbf{x} - \mathbf{b}\|_2}$$

i.e., the ratio of the Euclidean distance to \mathbf{a} to the distance to \mathbf{b} . Then f is quasiconvex on the halfspace $\{\mathbf{x} \mid \|\mathbf{x} - \mathbf{a}\|_2 \leq \alpha \|\mathbf{x} - \mathbf{b}\|_2\}$. To see this, we consider the α -sublevel

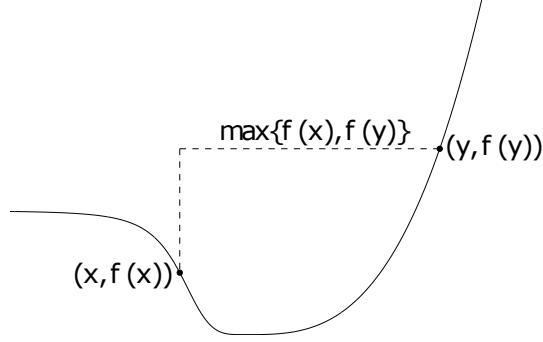


图 4.13: A quasiconvex function on \mathbb{R} . The value of f between x and y is no more than $\max\{f(x), f(y)\}$.

set of f , with $\alpha \leq 1$ since $f(\mathbf{x}) \leq 1$ on the halfspace $\{\mathbf{x} \mid \|\mathbf{x} - \mathbf{a}\|_2 \leq \alpha \|\mathbf{x} - \mathbf{b}\|_2\}$. This sublevel set is the set of points satisfying

$$\|\mathbf{x} - \mathbf{a}\|_2 \leq \alpha \|\mathbf{x} - \mathbf{b}\|_2.$$

Squaring both sides, and rearranging terms, we see that this is equivalent to

$$(1 - \alpha^2)\mathbf{x}^T \mathbf{x} - 2(\mathbf{a} - \alpha^2 \mathbf{b})^T \mathbf{x} + \mathbf{a}^T \mathbf{a} - \alpha^2 \mathbf{b}^T \mathbf{b} \leq 0.$$

This describes a convex set (in fact a Euclidean ball) if $\alpha \leq 1$.

Example 246. *Internal rate of return.* Let $\mathbf{x} = (x_0, x_1, \dots, x_n)^T$ denote a cash flow sequence over n periods, where $x_i > 0$ means a payment to us in period i , and $x_i < 0$ means a payment by us in period i . We define the present value of a cash flow, with interest rate $r \geq 0$, to be

$$PV(\mathbf{x}, r) = \sum_{i=0}^n (1+r)^{-i} x_i.$$

(The factor $(1+r)^{-i}$ is a discount factor for a payment by or to us in period i .) Now we consider cash flows for which $x_0 < 0$ and $x_0 + x_1 + \dots + x_n > 0$. This means that we start with an investment of $|x_0|$ in period 0, and that the total of the remaining cash flow, $x_1 + \dots + x_n$, (not taking any discount factors into account) exceeds our initial investment.

For such a cash flow, $PV(\mathbf{x}, 0) \geq 0$ and $PV(\mathbf{x}, r) \rightarrow x_0 < 0$ as $r \rightarrow \infty$, so it follows that for at least one $r \geq 0$, we have $PV(\mathbf{x}, r) = 0$. We define the *internal rate of return* of the cash flow as the smallest interest rate $r \geq 0$ for which the present value is zero:

$$IRR(\mathbf{x}) = \inf \{r \geq 0 \mid PV(\mathbf{x}, r) = 0\}.$$

Internal rate of return is a quasiconcave function of \mathbf{x} (restricted to $x_0 < 0, x_1 + \dots + x_n > 0$). To see this, we note that

$$IRR(\mathbf{x}) \geq R \Leftrightarrow PV(\mathbf{x}, r) \geq 0 \text{ for } 0 \leq r \leq R.$$

The lefthand side defines the R -superlevel set of IRR. The righthand side is the intersection of the sets $\{\mathbf{x} \mid \text{PV}(\mathbf{x}, r) \geq 0\}$, indexed by r , over the range $0 \leq r \leq R$. For each r , $\text{PV}(\mathbf{x}, r) \geq 0$ defines a halfspace, so the righthand side defines a convex set.

4.5.2 Basic properties

The examples above show that quasiconvexity is a considerable generalization of convexity. Still, many of the properties of convex functions hold, or have analogs, for quasiconvex functions. For example, there is a variation on Jensen's inequality that characterizes quasiconvexity: A function f is quasiconvex if and only if $\text{dom } f$ is convex and for any $\mathbf{x}, \mathbf{y} \in \text{dom } f$ and $0 \leq \theta \leq 1$,

$$f(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \leq \max\{f(\mathbf{x}), f(\mathbf{y})\}, \quad (4.45)$$

i.e., the value of the function on a segment does not exceed the maximum of its values at the endpoints. The inequality (4.45) is sometimes called Jensen's inequality for quasiconvex functions, and is illustrated in Figure 4.13.

Example 247. *Cardinality of a nonnegative vector.* The *cardinality* or *size* of a vector $\mathbf{x} \in \mathbb{R}^n$ is the number of nonzero components, and denoted $\text{card}(\mathbf{x})$. The function card is quasiconcave on \mathbb{R}_+^n (but not \mathbb{R}^n). This follows immediately from the modified Jensen inequality

$$\text{card}(\mathbf{x} + \mathbf{y}) \geq \min\{\text{card}(\mathbf{x}), \text{card}(\mathbf{y})\},$$

which holds for $\mathbf{x}, \mathbf{y} \succeq 0$.

Example 248. *Rank of positive semidefinite matrix.* The function $\text{rank}(\mathbf{X})$ is quasiconcave on \mathbb{S}_+^n . This follows from the modified Jensen inequality (4.45)

$$\text{rank}(\mathbf{X} + \mathbf{Y}) \geq \min\{\text{rank}(\mathbf{X}), \text{rank}(\mathbf{Y})\},$$

which holds for $\mathbf{X}, \mathbf{Y} \in \mathbb{S}_+^n$. (This can be considered an extension of the previous example, since $\text{rank}(\text{diag}(\mathbf{x})) = \text{card}(\mathbf{x})$ for $\mathbf{x} \succeq 0$.)

Like convexity, quasiconvexity is characterized by the behavior of a function f on lines: f is quasiconvex if and only if its restriction to any line intersecting its domain is quasiconvex. In particular, quasiconvexity of a function can be verified by restricting it to an arbitrary line, and then checking quasiconvexity of the resulting function on \mathbb{R} .

Quasiconvex functions on \mathbb{R} .

We can give a simple characterization of quasiconvex functions on \mathbb{R} . We consider continuous functions, since stating the conditions in the general case is cumbersome. A

continuous function $f : \mathbb{R} \rightarrow \mathbb{R}$ is quasiconvex if and only if at least one of the following conditions holds:

- f is nondecreasing
- f is nonincreasing
- there is a point $c \in \text{dom } f$ such that for $t \leq c$ (and $t \in \text{dom } f$), f is nonincreasing, and for $t \geq c$ (and $t \in \text{dom } f$), f is nondecreasing.

The point c can be chosen as any point which is a global minimizer of f . Figure 4.14 illustrates this.

4.5.3 Differentiable quasiconvex functions

First-order conditions

Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is differentiable. Then f is quasiconvex if and only if $\text{dom } f$ is convex and for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$

$$f(\mathbf{y}) \leq f(\mathbf{x}) \Rightarrow \nabla f(\mathbf{x})^T (\mathbf{y} - \mathbf{x}) \leq 0. \quad (4.46)$$

This is the analog of inequality (4.2), for quasiconvex functions. We leave the proof as an exercise (Exercise 308).

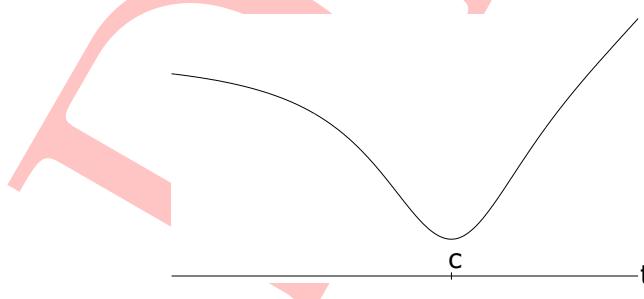


图 4.14: A quasiconvex function on \mathbb{R} . The function is nonincreasing for $t \leq c$ and nondecreasing for $t \geq c$.

The condition (4.46) has a simple geometric interpretation when $\nabla f(\mathbf{x}) \neq \mathbf{0}$. It states that $\nabla f(\mathbf{x})$ defines a supporting hyperplane to the sublevel set $\{\mathbf{y} \mid f(\mathbf{y}) \leq f(\mathbf{x})\}$, at the point \mathbf{x} , as illustrated in Figure 4.15.

While the first-order condition for convexity (4.2), and the first-order condition for quasiconvexity (4.46) are similar, there are some important differences. For example, if f is convex and $\nabla f(\mathbf{x}) = \mathbf{0}$, then \mathbf{x} is a global minimizer of f . But this statement is false

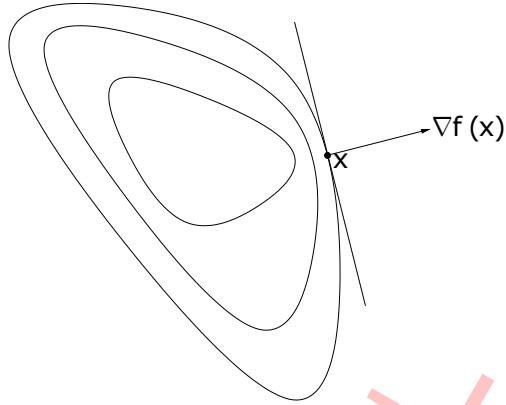


图 4.15: Three level curves of a quasiconvex function f are shown. The vector $\nabla f(\mathbf{x})$ defines a supporting hyperplane to the sublevel set $\{\mathbf{z} \mid f(\mathbf{z}) \leq f(\mathbf{x})\}$ at \mathbf{x} .

for quasiconvex functions: it is possible that $\nabla f(\mathbf{x}) = \mathbf{0}$, but \mathbf{x} is not a global minimizer of f .

Second-order conditions

Now suppose f is twice differentiable. If f is quasiconvex, then for all $\mathbf{x} \in \text{dom } f$, and all $\mathbf{y} \in \mathbb{R}^n$, we have

$$\mathbf{y}^T \nabla f(\mathbf{x}) = 0 \Rightarrow \mathbf{y}^T \nabla^2 f(\mathbf{x}) \mathbf{y} \geq 0. \quad (4.47)$$

For a quasiconvex function on \mathbb{R} , this reduces to the simple condition

$$f'(x) = 0 \Rightarrow f''(x) \geq 0,$$

i.e., at any point with zero slope, the second derivative is nonnegative. For a quasiconvex function on \mathbb{R}^n , the interpretation of the condition (4.47) is a bit more complicated. As in the case $n = 1$, we conclude that whenever $\nabla f(\mathbf{x}) = \mathbf{0}$, we must have $\nabla^2 f(\mathbf{x}) \succeq \mathbf{0}$. When $\nabla f(\mathbf{x}) \neq \mathbf{0}$, the condition (4.47) means that $\nabla^2 f(\mathbf{x})$ is positive semidefinite on the $(n - 1)$ -dimensional subspace $\nabla f(\mathbf{x})^\perp$. This implies that $\nabla^2 f(\mathbf{x})$ can have at most one negative eigenvalue.

As a (partial) converse, if f satisfies

$$\mathbf{y}^T \nabla f(\mathbf{x}) = 0 \Rightarrow \mathbf{y}^T \nabla^2 f(\mathbf{x}) \mathbf{y} > 0 \quad (4.48)$$

for all $\mathbf{x} \in \text{dom } f$ and all $\mathbf{y} \in \mathbb{R}^n, \mathbf{y} \neq 0$, then f is quasiconvex. This condition is the same as requiring $\nabla^2 f(\mathbf{x})$ to be positive definite for any point with $\nabla f(\mathbf{x}) = \mathbf{0}$, and for all other points, requiring $\nabla^2 f(\mathbf{x})$ to be positive definite on the $(n - 1)$ -dimensional subspace $\nabla f(\mathbf{x})^\perp$.

Proof of second-order conditions for quasiconvexity

By restricting the function to an arbitrary line, it suffices to consider the case in which $f : \mathbb{R} \rightarrow \mathbb{R}$.

We first show that if $f : \mathbb{R} \rightarrow \mathbb{R}$ is quasiconvex on an interval (a, b) , then it must satisfy (4.47), i.e., if $f'(c) = 0$ with $c \in (a, b)$, then we must have $f''(c) \geq 0$. If $f'(c) = 0$ with $c \in (a, b)$, $f''(c) < 0$, then for small positive ϵ we have $f(c - \epsilon) < f(c)$ and $f(c + \epsilon) < f(c)$. It follows that the sublevel set $\{x \mid f(x) \leq f(c) - \epsilon\}$ is disconnected for small positive ϵ , and therefore not convex, which contradicts our assumption that f is quasiconvex.

Now we show that if the condition (4.48) holds, then f is quasiconvex. Assume that (4.48) holds, i.e., for each $c \in (a, b)$ with $f'(c) = 0$, we have $f''(c) > 0$. This means that whenever the function f' crosses the value 0, it is strictly increasing. Therefore it can cross the value 0 at most once. If f' does not cross the value 0 at all, then f is either nonincreasing or nondecreasing on (a, b) , and therefore quasiconvex. Otherwise it must cross the value 0 exactly once, say at $c \in (a, b)$. Since $f''(c) > 0$, it follows that $f'(t) \leq 0$ for $a < t \leq c$, and $f'(t) \geq 0$ for $c \leq t < b$. This shows that f is quasiconvex.

4.5.4 Operations that preserve quasiconvexity

Nonnegative weighted maximum

A nonnegative weighted maximum of quasiconvex functions, i.e.,

$$f = \max \{\omega_1 f_1, \dots, \omega_m f_m\},$$

with $\omega_i \geq 0$ and f_i quasiconvex, is quasiconvex. The property extends to the general pointwise supremum

$$f(\mathbf{x}) = \sup_{\mathbf{y} \in C} \{\omega(\mathbf{y})g(\mathbf{x}, \mathbf{y})\},$$

where $\omega(\mathbf{y}) \geq 0$ and $g(\mathbf{x}, \mathbf{y})$ is quasiconvex in \mathbf{x} for each \mathbf{y} . This fact can be easily verified: $f(\mathbf{x}) \leq \alpha$ if and only if

$$\omega(\mathbf{y})g(\mathbf{x}, \mathbf{y}) \leq \alpha \text{ for all } \mathbf{y} \in C,$$

i.e., the α -sublevel set of f is the intersection of the α -sublevel sets of the functions $\omega(\mathbf{y})g(\mathbf{x}, \mathbf{y})$ in the variable \mathbf{x} .

Example 249. *Generalized eigenvalue.* The maximum generalized eigenvalue of a pair of symmetric matrices (\mathbf{X}, \mathbf{Y}) , with $\mathbf{Y} \succ \mathbf{0}$, is defined as

$$\lambda_{\max}(\mathbf{X}, \mathbf{Y}) = \sup_{\mathbf{u} \neq 0} \frac{\mathbf{u}^T \mathbf{X} \mathbf{u}}{\mathbf{u}^T \mathbf{Y} \mathbf{u}} = \sup \{\lambda \mid \det(\lambda \mathbf{Y} - \mathbf{X}) = 0\}$$

(See Section 2.11.7.3). This function is quasiconvex on $\text{dom } f = \mathbb{S}^n \times \mathbb{S}_{++}^n$.

To see this we consider the expression

$$\lambda_{\max}(\mathbf{X}, \mathbf{Y}) = \sup_{\mathbf{u} \neq 0} \frac{\mathbf{u}^T \mathbf{X} \mathbf{u}}{\mathbf{u}^T \mathbf{Y} \mathbf{u}}.$$

For each $\mathbf{u} \neq 0$, the function $\mathbf{u}^T \mathbf{X} \mathbf{u} = \mathbf{u}^T \mathbf{Y} \mathbf{u}$ is linear-fractional in (\mathbf{X}, \mathbf{Y}) , hence a quasiconvex function of (\mathbf{X}, \mathbf{Y}) . We conclude that λ_{\max} is quasiconvex, since it is the supremum of a family of quasiconvex functions.

Composition

If $g : \mathbb{R}^n \rightarrow \mathbb{R}$ is quasiconvex and $h : \mathbb{R} \rightarrow \mathbb{R}$ is nondecreasing, then $f = h \circ g$ is quasiconvex.

The composition of a quasiconvex function with an affine or linear-fractional transformation yields a quasiconvex function. If f is quasiconvex, then $g(\mathbf{x}) = f(\mathbf{Ax} + \mathbf{b})$ is quasiconvex, and $\tilde{g}(\mathbf{x}) = f((\mathbf{Ax} + \mathbf{b}) / (\mathbf{c}^T \mathbf{x} + d))$ is quasiconvex on the set

$$\{\mathbf{x} \mid \mathbf{c}^T \mathbf{x} + d > 0, (\mathbf{Ax} + \mathbf{b}) / (\mathbf{c}^T \mathbf{x} + d) \in \text{dom } f\}.$$

Minimization

If $f(\mathbf{x}, \mathbf{y})$ is quasiconvex jointly in \mathbf{x} and \mathbf{y} and C is a convex set, then the function

$$g(\mathbf{x}) = \inf_{\mathbf{y} \in C} f(\mathbf{x}, \mathbf{y})$$

is quasiconvex.

To show this, we need to show that $\{\mathbf{x} \mid g(\mathbf{x}) \leq \alpha\}$ is convex, where $\alpha \in \mathbb{R}$ is arbitrary. From the definition of g , $g(\mathbf{x}) \leq \alpha$ if and only if for any $\epsilon > 0$ there exists a $\mathbf{y} \in C$ with $f(\mathbf{x}, \mathbf{y}) \leq \alpha + \epsilon$. Now let \mathbf{x}_1 and \mathbf{x}_2 be two points in the α -sublevel set of g . Then for any $\epsilon > 0$, there exists $\mathbf{y}_1, \mathbf{y}_2 \in C$ with

$$f(\mathbf{x}_1, \mathbf{y}_1) \leq \alpha + \epsilon, \quad f(\mathbf{x}_2, \mathbf{y}_2) \leq \alpha + \epsilon$$

and since f is quasiconvex in \mathbf{x} and \mathbf{y} , we also have

$$f(\theta \mathbf{x}_1 + (1 - \theta) \mathbf{x}_2, \theta \mathbf{y}_1 + (1 - \theta) \mathbf{y}_2) \leq \alpha + \epsilon$$

for $0 \leq \theta \leq 1$. Hence $g(\theta \mathbf{x}_1 + (1 - \theta) \mathbf{x}_2) \leq \alpha$, which proves that $\{\mathbf{x} \mid g(\mathbf{x}) \leq \alpha\}$ is convex.

4.5.5 Representation via family of convex functions

In the sequel, it will be convenient to represent the sublevel sets of a quasiconvex function f (which are convex) via inequalities of convex functions. We seek a family of convex functions $\phi_t : \mathbb{R}^n \rightarrow \mathbb{R}$, indexed by $t \in \mathbb{R}$, with

$$f(\mathbf{x}) \leq t \Leftrightarrow \phi_t(\mathbf{x}) \leq 0, \tag{4.49}$$

i.e., the t -sublevel set of the quasiconvex function f is the 0-sublevel set of the convex function ϕ_t . Evidently ϕ_t must satisfy the property that for all $\mathbf{x} \in \mathbb{R}^n$, $\phi_t(\mathbf{x}) \leq 0 \Rightarrow$

$\phi_s(\mathbf{x}) \leq 0$ for $s \geq t$. This is satisfied if for each \mathbf{x} , $\phi_t(\mathbf{x})$ is a nonincreasing function of t , i.e., $\phi_s(\mathbf{x}) \leq \phi_t(\mathbf{x})$ whenever $s \geq t$.

To see that such a representation always exists, we can take

$$\phi_t(\mathbf{x}) = \begin{cases} 0, & f(\mathbf{x}) \leq t \\ \infty, & \text{otherwise,} \end{cases}$$

i.e., ϕ_t is the indicator function of the t -sublevel of f . Obviously this representation is not unique; for example if the sublevel sets of f are closed, we can take

~~$$\phi_t(\mathbf{x}) = \text{dist}(\mathbf{x}, \{\mathbf{z} \mid f(\mathbf{z}) \leq t\}).$$~~

We are usually interested in a family ϕ_t with nice properties, such as differentiability.

Example 250. *Convex over concave function.* Suppose p is a convex function, q is a concave function, with $p(\mathbf{x}) \geq 0$ and $q(\mathbf{x}) > 0$ on a convex set C . Then the function f defined by $f(\mathbf{x}) = p(\mathbf{x})/q(\mathbf{x})$, on C , is quasiconvex.

Here we have

$$f(\mathbf{x}) \leq t \Leftrightarrow p(\mathbf{x}) - tq(\mathbf{x}) \leq 0.$$

So we can take $\phi_t(\mathbf{x}) = p(\mathbf{x}) - tq(\mathbf{x})$ for $t \geq 0$. For each t , ϕ_t is convex and for each \mathbf{x} , $\phi_t(\mathbf{x})$ is decreasing in t .

4.6 Log-concave and log-convex functions

4.6.1 Definition

A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is *logarithmically concave* or *log-concave* if $f(\mathbf{x}) > 0$ for all $\mathbf{x} \in \text{dom } f$ and $\log f$ is concave. It is said to be *logarithmically convex* or *log-convex* if $\log f$ is convex. Thus f is log-convex if and only if $1/f$ is log-concave. It is convenient to allow f to take on the value zero, in which case we take $\log f(\mathbf{x}) = -\infty$. In this case we say f is log-concave if the extended-value function $\log f$ is concave.

We can express log-concavity directly, without logarithms: a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, with convex domain and $f(\mathbf{x}) > 0$ for all $\mathbf{x} \in \text{dom } f$, is log-concave if and only if for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$ and $0 \leq \theta \leq 1$, we have

$$f(\theta\mathbf{x} + (1-\theta)\mathbf{y}) \geq f(\mathbf{x})^\theta f(\mathbf{y})^{1-\theta}.$$

In particular, the value of a log-concave function at the average of two points is at least the *geometric mean* of the values at the two points.

From the composition rules we know that e^h is convex if h is convex, so a log-convex function is convex. Similarly, a nonnegative concave function is log-concave. It is also clear that a log-convex function is quasiconvex and a log-concave function is quasiconcave, since the logarithm is monotone increasing.

Example 251. Some simple examples of log-concave and log-convex functions.

- *Affine function.* $f(\mathbf{x}) = \mathbf{a}^T \mathbf{x} + b$ is log-concave on $\{\mathbf{x} \mid \mathbf{a}^T \mathbf{x} + b > 0\}$.
- *Powers.* $f(x) = x^\alpha$, on \mathbb{R}_{++} , is log-convex for $a \leq 0$, and log-concave for $a \geq 0$.
- *Exponentials.* $f(x) = e^{\alpha x}$ is log-convex and log-concave.
- The cumulative distribution function of a Gaussian density,

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-u^2/2} du,$$

is log-concave (see Exercise 319).

- *Gamma function.* The Gamma function,

$$\Gamma(x) = \int_0^\infty u^{x-1} e^{-u} du,$$

is log-convex for $x \geq 1$ (see Exercise 317).

- *Determinant.* $\det \mathbf{X}$ is log concave on \mathbb{S}_{++}^n .
- *Determinant over trace.* $\det \mathbf{X} / \text{tr } \mathbf{X}$ is log concave on \mathbb{S}_{++}^n (see Exercise 314).

Example 252. Log-concave density functions. Many common probability density functions are log-concave. Two examples are the multivariate normal distribution,

$$f(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \det \Sigma}} e^{-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}})^T \Sigma^{-1} (\mathbf{x}-\bar{\mathbf{x}})}$$

(where $\bar{\mathbf{x}} \in \mathbb{R}^n$ and $\Sigma \in \mathbb{S}_{++}^n$), and the exponential distribution on \mathbb{R}_+^n ,

$$f(\mathbf{x}) = \left(\prod_{i=1}^n \lambda_i \right) e^{-\boldsymbol{\lambda}^T \mathbf{x}}$$

(where $\boldsymbol{\lambda} \succ \mathbf{0}$). Another example is the uniform distribution over a convex set C ,

$$f(\mathbf{x}) = \begin{cases} 1/\alpha, & \mathbf{x} \in C \\ 0, & \mathbf{x} \notin C \end{cases}$$

where $\alpha = \text{vol}(C)$ is the volume (Lebesgue measure) of C . In this case $\log f$ takes on the value $-\infty$ outside C , and $-\log \alpha$ on C , hence is concave.

As a more exotic example consider the Wishart distribution, defined as follows. Let $\mathbf{x}_1, \dots, \mathbf{x}_p \in \mathbb{R}^n$ be independent Gaussian random vectors with zero mean and covariance $\Sigma \in \mathbb{S}^n$, with $p > n$. The random matrix $\mathbf{X} = \sum_{i=1}^p \mathbf{x}_i \mathbf{x}_i^T$ has the Wishart density

$$f(\mathbf{X}) = a(\det \mathbf{X})^{(p-n-1)/2} e^{-\frac{1}{2} \text{tr}(\Sigma^{-1} \mathbf{X})},$$

with $\text{dom } f = \mathbb{S}_{++}^n$, and a is a positive constant. The Wishart density is log-concave, since

$$\log f(\mathbf{X}) = \log a + \frac{p-n-1}{2} \log \det \mathbf{X} - \frac{1}{2} \text{tr}(\Sigma^{-1} \mathbf{X}),$$

which is a concave function of \mathbf{X} .

4.6.2 Properties

Twice differentiable log-convex/concave functions

Suppose f is twice differentiable, with $\text{dom } f$ convex, so

$$\nabla^2 \log f(\mathbf{x}) = \frac{1}{f(\mathbf{x})} \nabla^2 f(\mathbf{x}) - \frac{1}{f(\mathbf{x})^2} \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^T.$$

We conclude that f is log-convex if and only if for all $\mathbf{x} \in \text{dom } f$,

$$f(\mathbf{x}) \nabla^2 f(\mathbf{x}) \succeq \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^T,$$

and log-concave if and only if for all $\mathbf{x} \in \text{dom } f$,

$$f(\mathbf{x}) \nabla^2 f(\mathbf{x}) \preceq \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^T.$$

Multiplication, addition, and integration

Log-convexity and log-concavity are closed under multiplication and positive scaling. For example, if f and g are log-concave, then so is the pointwise product $h(\mathbf{x}) = f(\mathbf{x})g(\mathbf{x})$, since $\log h(\mathbf{x}) = \log f(\mathbf{x}) + \log g(\mathbf{x})$, and $\log f(\mathbf{x})$ and $\log g(\mathbf{x})$ are concave functions of \mathbf{x} .

Simple examples show that the sum of log-concave functions is not, in general, log-concave. Log-convexity, however, is preserved under sums. Let f and g be log-convex functions, *i.e.*, $F = \log f$ and $G = \log g$ are convex. From the composition rules for convex functions, it follows that

$$\log(\exp F + \exp G) = \log(f + g)$$

is convex. Therefore the sum of two log-convex functions is log-convex.

More generally, if $f(\mathbf{x}, \mathbf{y})$ is log-convex in \mathbf{x} for each $\mathbf{y} \in C$ then

$$g(\mathbf{x}) = \int_C f(\mathbf{x}, \mathbf{y}) d\mathbf{y}$$

is log-convex.

Example 253. *Laplace transform of a nonnegative function and the moment and cumulant generating functions.* Suppose $p : \mathbb{R}^n \rightarrow \mathbb{R}$ satisfies $p(\mathbf{x}) \geq 0$ for all \mathbf{x} . The Laplace transform of p ,

$$P(\mathbf{z}) = \int p(\mathbf{x}) e^{-\mathbf{z}^T \mathbf{x}} d\mathbf{x},$$

is log-convex on \mathbb{R}^n . (Here $\text{dom } P$ is, naturally, $\{\mathbf{z} \mid P(\mathbf{z}) < \infty\}$.)

Now suppose p is a density, *i.e.*, satisfies $\int p(\mathbf{x}) d\mathbf{x} = 1$. The function $M(\mathbf{z}) = P(-\mathbf{z})$ is called the *moment generating function* of the density. It gets its name from the fact that the moments of the density can be found from the derivatives of the moment generating function, evaluated at $\mathbf{z} = \mathbf{0}$, *e.g.*,

$$\nabla M(\mathbf{0}) = \mathbb{E}\mathbf{v}, \quad \nabla^2 M(\mathbf{0}) = \mathbb{E}\mathbf{v}\mathbf{v}^T,$$

where \mathbf{v} is a random variable with density p .

The function $\log M(\mathbf{z})$, which is convex, is called the *cumulant generating function* for p , since its derivatives give the cumulants of the density. For example, the first and second derivatives of the cumulant generating function, evaluated at zero, are the mean and covariance of the associated random variable:

$$\nabla \log M(\mathbf{0}) = \mathbb{E}\mathbf{v}, \quad \nabla \log M(\mathbf{0}) = \mathbb{E}(\mathbf{v} - \mathbb{E}\mathbf{v})(\mathbf{v} - \mathbb{E}\mathbf{v})^T.$$

4.6.3 Integration of log-concave functions

In some special cases log-concavity is preserved by integration. If $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ is log-concave, then

$$g(\mathbf{x}) = \int f(\mathbf{x}, \mathbf{y}) d\mathbf{y}$$

is a log-concave function of \mathbf{x} (on \mathbb{R}^n). (The integration here is over \mathbb{R}^m .) A proof of this result is not simple; see the references.

This result has many important consequences, some of which we describe in the rest of this section. It implies, for example, that marginal distributions of log-concave probability densities are log-concave. It also implies that log-concavity is closed under convolution, *i.e.*, if f and g are log-concave on \mathbb{R}^n , then so is the convolution

$$(f * g)(\mathbf{x}) = \int f(\mathbf{x} - \mathbf{y}) g(\mathbf{y}) d\mathbf{y}.$$

(To see this, note that $g(\mathbf{y})$ and $f(\mathbf{x} - \mathbf{y})$ are log-concave in (\mathbf{x}, \mathbf{y}) , hence the product $f(\mathbf{x} - \mathbf{y})g(\mathbf{y})$ is; then the integration result applies.)

Suppose $C \subseteq \mathbb{R}^n$ is a convex set and $\boldsymbol{\omega}$ is a random vector in \mathbb{R}^n with log-concave probability density p . Then the function

$$f(\mathbf{x}) = \text{prob}(\mathbf{x} + \boldsymbol{\omega} \in C)$$

is log-concave in \mathbf{x} . To see this, express f as

$$f(\mathbf{x}) = \int g(\mathbf{x} + \boldsymbol{\omega})p(\boldsymbol{\omega})d\boldsymbol{\omega},$$

where g is defined as

$$g(\mathbf{u}) = \begin{cases} 1, & \mathbf{u} \in C \\ 0, & \mathbf{u} \notin C, \end{cases}$$

(which is log-concave) and apply the integration result.

Example 254. The *cumulative distribution function* of a probability density function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is defined as

$$F(\mathbf{x}) = \text{prob}(\boldsymbol{\omega} \preceq \mathbf{x}) = \int_{-\infty}^{x_n} \dots \int_{-\infty}^{x_1} f(\mathbf{z})dz_1 \dots dz_n,$$

where $\boldsymbol{\omega}$ is a random variable with density f . If f is log-concave, then F is log-concave. We have already encountered a special case: the cumulative distribution function of a Gaussian random variable,

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt,$$

is log-concave. (See Example 251 and Exercise 319)

Example 255. *Yield function.* Let $\mathbf{x} \in \mathbb{R}^n$ denote the nominal or target value of a set of parameters of a product that is manufactured. Variation in the manufacturing process causes the parameters of the product, when manufactured, to have the value $\mathbf{x} + \boldsymbol{\omega}$, where $\boldsymbol{\omega} \in \mathbb{R}^n$ is a random vector that represents manufacturing variation, and is usually assumed to have zero mean. The *yield* of the manufacturing process, as a function of the nominal parameter values, is given by

$$Y(\mathbf{x}) = \text{prob}(\mathbf{x} + \boldsymbol{\omega} \in \mathbb{S})$$

where $\mathbb{S} \subseteq \mathbb{R}^n$ denotes the set of acceptable parameter values for the product, *i.e.*, the product specifications.

If the density of the manufacturing error ω is log-concave (for example, Gaussian) and the set \mathbb{S} of product specifications is convex, then the yield function Y is log-concave. This implies that the α -yield region, defined as the set of nominal parameters for which the yield exceeds α , is convex. For example, the 95% yield region

$$\{\mathbf{x} \mid Y(\mathbf{x}) \geq 0.95\} = \{\mathbf{x} \mid \log Y(\mathbf{x}) \geq \log 0.95\}$$

is convex, since it is a superlevel set of the concave function $\log Y$.

Example 256. *Volume of polyhedron.* Let $\mathbf{A} \in \mathbb{R}^{m \times n}$. Define

$$P_{\mathbf{u}} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{u}\}.$$

Then its volume $\text{vol } P_{\mathbf{u}}$ is a log-concave function of \mathbf{u} . To prove this, note that the function

$$\Psi(\mathbf{x}, \mathbf{u}) = \begin{cases} 1, & \mathbf{A}\mathbf{x} \leq \mathbf{u} \\ 0, & \text{otherwise,} \end{cases}$$

is log-concave. By the integration result, we conclude that

$$\int \Psi(\mathbf{x}, \mathbf{u}) d\mathbf{x} = \text{vol } P_{\mathbf{u}}$$

is log-concave.

4.7 Convexity with respect to generalized inequalities

We now consider generalizations of the notions of monotonicity and convexity, using generalized inequalities instead of the usual ordering on \mathbb{R} .

4.7.1 Monotonicity with respect to a generalized inequality

Suppose $K \subseteq \mathbb{R}^n$ is a proper cone with associated generalized inequality \preceq_K . A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is called *K-nondecreasing* if

$$\mathbf{x} \preceq_K \mathbf{y} \Rightarrow f(\mathbf{x}) \leq f(\mathbf{y})$$

and *K-increasing* if

$$\mathbf{x} \preceq_K \mathbf{y}, \mathbf{x} \neq \mathbf{y} \Rightarrow f(\mathbf{x}) < f(\mathbf{y}).$$

We define *K-nonincreasing* and *K-decreasing* functions in a similar way.

Example 257. *Monotone vector functions.* A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is nondecreasing with respect to \mathbb{R}_+^n if and only if

$$x_1 \leq y_1, \dots, x_n \leq y_n \Rightarrow f(\mathbf{x}) \leq f(\mathbf{y})$$

for all \mathbf{x}, \mathbf{y} . This is the same as saying that f , when restricted to any component x_i (*i.e.*, x_i is considered the variable while x_j for $j \neq i$ are fixed), is nondecreasing.

Example 258. *Matrix monotone functions.* A function $f : \mathbb{S}^n \rightarrow \mathbb{R}$ is called *matrix monotone* (increasing, decreasing) if it is monotone with respect to the positive semidefinite cone. Some examples of matrix monotone functions of the variable $\mathbf{X} \in \mathbb{S}^n$:

- $\text{tr}(\mathbf{W}\mathbf{X})$, where $\mathbf{W} \in \mathbb{S}^n$, is matrix nondecreasing if $\mathbf{W} \succ \mathbf{0}$, and matrix increasing if $\mathbf{W} \preceq \mathbf{0}$ (it is matrix nonincreasing if $\mathbf{W} \succ \mathbf{0}$, and matrix decreasing if $\mathbf{W} \prec \mathbf{0}$).
- $\text{tr}(\mathbf{X}^{-1})$ is matrix decreasing on \mathbb{S}_{++}^n .
- $\det \mathbf{X}$ is matrix increasing on \mathbb{S}_{++}^n .

Gradient conditions for monotonicity

Recall that a differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, with convex (*i.e.*, interval) domain, is nondecreasing if and only if $f'(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in \text{dom } f$, and increasing if $f'(\mathbf{x}) > 0$ for all $\mathbf{x} \in \text{dom } f$ (but the converse is not true). These conditions are readily extended to the case of monotonicity with respect to a generalized inequality. A differentiable function f , with convex domain, is K -nondecreasing if and only if

$$\nabla f(\mathbf{x}) \succeq_{K^*} \mathbf{0} \tag{4.50}$$

for all $\mathbf{x} \in \text{dom } f$. Note the difference with the simple scalar case: the gradient must be nonnegative in the *dual* inequality. For the strict case, we have the following: If

$$\nabla f(\mathbf{x}) \succ_{K^*} \mathbf{0} \tag{4.51}$$

for all $\mathbf{x} \in \text{dom } f$, then f is K -increasing. As in the scalar case, the converse is not true.

Let us prove these first-order conditions for monotonicity. First, assume that f satisfies (4.50) for all \mathbf{x} , but is not K -nondecreasing, *i.e.*, there exist \mathbf{x}, \mathbf{y} with $\mathbf{x} \succeq_K \mathbf{y}$ and $f(\mathbf{y}) < f(\mathbf{x})$. By differentiability of f there exists a $t \in [0, 1]$ with

$$\frac{d}{dt} f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) = \nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x}))^T (\mathbf{y} - \mathbf{x}) < 0.$$

Since $\mathbf{y} - \mathbf{x} \in K$ this means

$$\nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) \notin K^*$$

which contradicts our assumption that (4.50) is satisfied everywhere. In a similar way it can be shown that (4.51) implies f is K -increasing.

It is also straightforward to see that it is necessary that (4.50) hold everywhere. Assume (4.50) does not hold for $\mathbf{x} = \mathbf{z}$. By the definition of dual cone this means there exists a $\mathbf{v} \in K$ with

$$\nabla f(\mathbf{z})^T \mathbf{v} < 0.$$

Now consider $h(t) = f(\mathbf{z} + t\mathbf{v})$ as a function of t . We have $h'(0) = \nabla f(\mathbf{z})^T \mathbf{v} < 0$, and therefore there exists $t > 0$ with $h(t) = f(\mathbf{z} + t\mathbf{v}) < h(0) = f(\mathbf{z})$, which means f is not K -nondecreasing.

4.7.2 Convexity with respect to a generalized inequality

Suppose $K \subseteq \mathbb{R}^m$ is a proper cone with associated generalized inequality \preceq_K . We say $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is K -convex if for all \mathbf{x}, \mathbf{y} , and $0 \leq \theta \leq 1$,

$$f(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \preceq_K \theta f(\mathbf{x}) + (1 - \theta)f(\mathbf{y}).$$

The function is *strictly K-convex* if

$$f(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \prec_K \theta f(\mathbf{x}) + (1 - \theta)f(\mathbf{y})$$

for all $\mathbf{x} \neq \mathbf{y}$ and $0 < \theta < 1$. These definitions reduce to ordinary convexity and strict convexity when $m = 1$ (and $K = \mathbb{R}_+$).

Example 259. *Convexity with respect to componentwise inequality.* A function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is convex with respect to componentwise inequality (*i.e.*, the generalized inequality induced by \mathbb{R}_+^m) if and only if for all \mathbf{x}, \mathbf{y} and $0 \leq \theta \leq 1$,

$$f(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \preceq \theta f(\mathbf{x}) + (1 - \theta)f(\mathbf{y}),$$

i.e., each component f_i is a convex function. The function f is strictly convex with respect to componentwise inequality if and only if each component f_i is strictly convex.

Example 260. *Matrix convexity.* Suppose f is a symmetric matrix valued function, *i.e.*, $f : \mathbb{R}^n \rightarrow \mathbb{S}^m$. The function f is convex with respect to matrix inequality if

$$f(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \preceq \theta f(\mathbf{x}) + (1 - \theta)f(\mathbf{y})$$

for any \mathbf{x} and \mathbf{y} , and for $\theta \in [0, 1]$. This is sometimes called *matrix convexity*. An equivalent definition is that the scalar function $\mathbf{z}^T f(\mathbf{x}) \mathbf{z}$ is convex for all vectors \mathbf{z} . (This is often a good way to prove matrix convexity). A matrix function is strictly matrix convex if

$$f(\theta\mathbf{x} + (1 - \theta)\mathbf{y}) \prec \theta f(\mathbf{x}) + (1 - \theta)f(\mathbf{y})$$

when $\mathbf{x} \neq \mathbf{y}$ and $0 < \theta < 1$, or, equivalently, if $\mathbf{z}^T f(\mathbf{x}) \mathbf{z}$ is strictly convex for every $\mathbf{z} \neq \mathbf{0}$. Some examples:

- The function $f(\mathbf{X}) = \mathbf{X}\mathbf{X}^T$ where $\mathbf{X} \in \mathbb{R}^{n \times m}$ is matrix convex, since for fixed \mathbf{z} the function $\mathbf{z}^T \mathbf{X}\mathbf{X}^T \mathbf{z} = \|\mathbf{X}^T \mathbf{z}\|_2^2$ is a convex quadratic function of (the components of) \mathbf{X} . For the same reason, $f(\mathbf{X}) = \mathbf{X}^2$ is matrix convex on \mathbb{S}^n .

- The function \mathbf{X}_p is matrix convex on \mathbb{S}_{++}^n for $1 \leq p \leq 2$ or $-1 \leq p \leq 0$, and matrix concave for $0 \leq p \leq 1$.
- The function $f(\mathbf{X}) = e^{\mathbf{X}}$ is not matrix convex on \mathbb{S}^n , for $n \geq 2$.

Many of the results for convex functions have extensions to K -convex functions. As a simple example, a function is K -convex if and only if its restriction to any line in its domain is K -convex. In the rest of this section we list a few results for K -convexity that we will use later; more results are explored in the exercises.

Dual characterization of K -convexity

A function f is K -convex if and only if for every $\boldsymbol{\omega} \succeq_{K^*} \mathbf{0}$, the (real-valued) function $\boldsymbol{\omega}^T f$ is convex (in the ordinary sense); f is strictly K -convex if and only if for every nonzero $\boldsymbol{\omega} \succeq_{K^*} \mathbf{0}$ the function $\boldsymbol{\omega}^T f$ is strictly convex. (These follow directly from the definitions and properties of dual inequality.)

Differentiable K -convex functions

A differentiable function f is K -convex if and only if its domain is convex, and for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$,

$$f(\mathbf{y}) \succeq_K f(\mathbf{x}) + Df(\mathbf{x})(\mathbf{y} - \mathbf{x}).$$

(Here $Df(\mathbf{x}) \in \mathbb{R}^{m \times n}$ is the derivative or Jacobian matrix of f at \mathbf{x} ; see Section 2.11.4.1.) The function f is strictly K -convex if and only if for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$ with $\mathbf{x} \neq \mathbf{y}$,

$$f(\mathbf{y}) >_K f(\mathbf{x}) + Df(\mathbf{x})(\mathbf{y} - \mathbf{x}).$$

Composition theorem

Many of the results on composition can be generalized to K -convexity. For example, if $g : \mathbb{R}^n \rightarrow \mathbb{R}^p$ is K -convex, $h : \mathbb{R}^p \rightarrow \mathbb{R}$ is convex, and \tilde{h} (the extended-value extension of h) is K -nondecreasing, then $h \circ g$ is convex. This generalizes the fact that a nondecreasing convex function of a convex function is convex. The condition that \tilde{h} be K -nondecreasing implies that $\text{dom } h - K = \text{dom } h$.

Example 261. The quadratic matrix function $g : \mathbb{R}^{m \times n} \rightarrow \mathbb{S}^n$ defined by

$$g(\mathbf{X}) = \mathbf{X}^T \mathbf{A} \mathbf{X} + \mathbf{B}^T \mathbf{X} + \mathbf{X}^T \mathbf{B} + \mathbf{C},$$

where $\mathbf{A} \in \mathbb{S}^m$, $\mathbf{B} \in \mathbb{R}^{m \times n}$, and $\mathbf{C} \in \mathbb{S}^n$, is convex when $\mathbf{A} \succeq \mathbf{0}$.

The function $h : S^n \rightarrow \mathbb{R}$ defined by $h(\mathbf{Y}) = -\log \det(-\mathbf{Y})$ is convex and increasing on $\text{dom } h = -\mathbb{S}_{++}^n$. By the composition theorem, we conclude that

$$f(\mathbf{X}) = -\log \det(-(\mathbf{X}^T \mathbf{A} \mathbf{X} + \mathbf{B}^T \mathbf{X} + \mathbf{X}^T \mathbf{B} + \mathbf{C}))$$

is convex on

$$\text{dom } f = \{\mathbf{X} \in \mathbb{R}^{m \times n} \mid \mathbf{X}^T \mathbf{A} \mathbf{X} + \mathbf{B}^T \mathbf{X} + \mathbf{X}^T \mathbf{B} + \mathbf{C} \prec \mathbf{0}\}.$$

This generalizes the fact that

$$-\log(-(ax^2 + bx + c))$$

is convex on

$$\{x \in \mathbb{R} \mid ax^2 + bx + c < 0\},$$

provided $a \geq 0$.

4.8 A little about nonconvex analysis

(Taken from [16, 113])

Definition 262. A function $g : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ is said to be proper if $\text{dom } g \neq \emptyset$, where $\text{dom } g = \{\mathbf{x} \in \mathbb{R} : g(\mathbf{x}) < +\infty\}$. g is lower semicontinuous at point \mathbf{x}_0 if

$$\liminf_{x \rightarrow \mathbf{x}_0} g(\mathbf{x}) \geq g(\mathbf{x}_0). \quad (4.52)$$

Definition 263. A function $F(\mathbf{x})$ is called coercive if F is bounded from below and

$$F(\mathbf{x}) \rightarrow \infty \quad \text{when} \quad \|\mathbf{x}\| \rightarrow \infty, \quad (4.53)$$

where $\|\cdot\|$ is the l_2 -norm.

Definition 264. Let g be a proper and lower semicontinuous function.

1. For a given $\mathbf{x} \in \text{dom } g$, the Frechet subdifferential of g at \mathbf{x} , written as $\hat{\partial}g(\mathbf{x})$, is the set of all vectors $\mathbf{u} \in \mathbb{R}^n$ which satisfy

$$\liminf_{\mathbf{y} \neq \mathbf{x}, \mathbf{y} \rightarrow \mathbf{x}} \frac{g(\mathbf{y}) - g(\mathbf{x}) - \langle \mathbf{u}, \mathbf{y} - \mathbf{x} \rangle}{\|\mathbf{y} - \mathbf{x}\|} \geq 0. \quad (4.54)$$

2. The limiting-subdifferential, or simply the subdifferential, of g at $\mathbf{x} \in \mathbb{R}^n$, written as $\partial g(\mathbf{x})$, is defined through the following closure process

$$\partial f(\mathbf{x}) := \{\mathbf{u} \in \mathbb{R}^n : \exists \mathbf{x}_k \rightarrow \mathbf{x}, g(\mathbf{x}_k) \rightarrow g(\mathbf{x}), \mathbf{u}_k \in \hat{\partial}g(\mathbf{x}_k) \rightarrow \mathbf{u}, k \rightarrow \infty\}. \quad (4.55)$$

Proposition 265.

1. In the nonsmooth context, the Fermat's rule remains unchanged: If $\mathbf{x} \in \mathbb{R}^n$ is a local minimizer of g , then $0 \in \partial g(\mathbf{x})$.
2. Let $(\mathbf{x}_k, \mathbf{u}_k)$ be a sequence such that $\mathbf{x}_k \rightarrow \mathbf{x}$, $\mathbf{u}_k \rightarrow \mathbf{u}$, $g(\mathbf{x}_k) \rightarrow g(\mathbf{x})$ and $\mathbf{u}_k \in \partial g(\mathbf{x}_k)$, then $\mathbf{u} \in \partial g(\mathbf{x})$.
3. If f is a continuously differentiable function, then $\partial(f + g)(\mathbf{x}) = \nabla f(\mathbf{x}) + \partial g(\mathbf{x})$.

Recall that points whose subdifferential contains $\mathbf{0}$ are called critical points.

4.9 Exercises

Definition of convexity

Exercise 266. Suppose $f : \mathbb{R} \rightarrow \mathbb{R}$ is convex, and $a, b \in \text{dom } f$ with $a < b$.

(a) Show that

$$f(x) \leq \frac{b-x}{a-x}f(a) + \frac{x-a}{b-a}f(b)$$

for all $x \in [a, b]$.

(b) Show that

$$\frac{f(x) - f(a)}{x - a} \leq \frac{f(b) - f(a)}{b - a} \leq \frac{f(b) - f(x)}{b - x}$$

for all $x \in (a, b)$. Draw a sketch that illustrates this inequality.

(c) Suppose f is differentiable. Use the result in (b) to show that

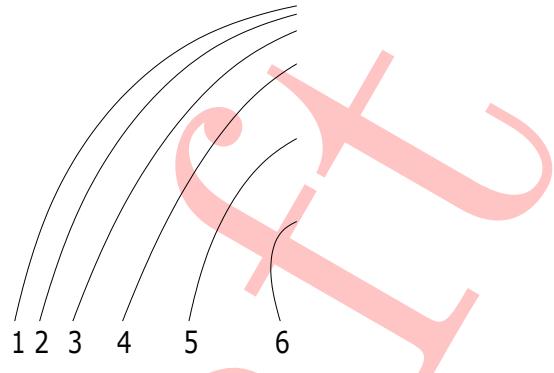
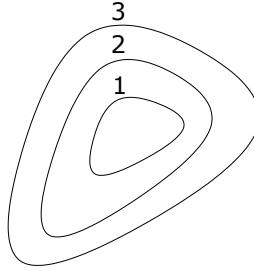
$$f'(a) \leq \frac{f(b) - f(a)}{b - a} \leq f'(b).$$

Note that these inequalities also follow from (4.2):

$$\begin{aligned} f(b) &\geq f(a) + f'(a)(b - a), \\ f(a) &\geq f(b) + f'(b)(a - b). \end{aligned}$$

(d) Suppose f is twice differentiable. Use the result in (c) to show that $f''(a) \geq 0$ and $f''(b) \geq 0$.

Exercise 267. Level sets of convex, concave, quasiconvex, and quasiconcave functions. One level sets of a function f are shown below. The curve labeled 1 shows $\{x \mid f(x = 1)\}$, etc. Could f be convex (concave, quasiconvex, quasiconcave)? Explain your answer. Repeat for the level curves shown below.



Exercise 268. *Inverse of an increasing convex function.* Suppose $f : \mathbb{R} \rightarrow \mathbb{R}$ is increasing and convex on its domain (a, b) . Let g denote its inverse, i.e., the function with domain $(f(a), f(b))$ and $g(f(x)) = x$ for $a < x < b$. What can you say about convexity or concavity of g ?

Exercise 269. Show that a continuous function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex if and only if for every line segment, its average value on the segment is less than or equal to the average of its values at the endpoints of the segment: For every $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$,

$$\int_0^1 f(\mathbf{x} + \lambda(\mathbf{y} - \mathbf{x})) d\lambda \leq \frac{f(\mathbf{x}) + f(\mathbf{y})}{2}.$$

Exercise 270. *Running average of a convex function.* Suppose $f : \mathbb{R} \rightarrow \mathbb{R}$ is convex, with $\mathbb{R}_+ \subseteq \text{dom } f$. Show that its *running average* F , defined as

$$F(x) = \frac{1}{x} \int_0^x f(t) dt, \text{dom } F = \mathbb{R}_{++}$$

is convex. You can assume f is differentiable.

Exercise 271. *Functions and epigraphs.* When is the epigraph of a function a halfspace? When is the epigraph of a function a convex cone? When is the epigraph of a function a polyhedron?

Exercise 272. Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex with $\text{dom } f = \mathbb{R}^n$, and bounded above on \mathbb{R}^n . Show that f is constant.

Exercise 273. *Second-order condition for convexity.* Prove that a twice differentiable function f is convex if and only if its domain is convex and $\nabla^2 f(\mathbf{x}) \succeq \mathbf{0}$ for all $\mathbf{x} \in \text{dom } f$. Hint. First consider the case $f : \mathbb{R} \rightarrow \mathbb{R}$. You can use the first-order condition for convexity (see Section 4.1.3).

Exercise 274. *Second-order conditions for convexity on an affine set.* Let $\mathbf{F} \in \mathbb{R}^{n \times m}$, $\hat{\mathbf{x}} \in \mathbb{R}^n$. The restriction of $f : \mathbb{R}^n \rightarrow \mathbb{R}$ to the affine set $\{\mathbf{F}\mathbf{z} + \hat{\mathbf{x}} \mid \mathbf{z} \in \mathbb{R}^m\}$ is defined as the function $\tilde{f} : \mathbb{R}^m \rightarrow \mathbb{R}$ with

$$\tilde{f}(\mathbf{z}) = f(\mathbf{F}\mathbf{z} + \hat{\mathbf{x}}), \quad \text{dom } \tilde{f} = \{\mathbf{z} \mid \mathbf{F}\mathbf{z} + \hat{\mathbf{x}} \in \text{dom } f\}.$$

Suppose f is twice differentiable with a convex domain.

- (a) Show that \tilde{f} is convex if and only if for all $\mathbf{z} \in \text{dom } f$

$$\mathbf{F}^T \nabla^2 f(\mathbf{F}\mathbf{z} + \hat{\mathbf{x}}) \mathbf{F} \succeq \mathbf{0}.$$

- (b) Suppose $\mathbf{A} \in \mathbb{R}^{p \times n}$ is a matrix whose nullspace is equal to the range of \mathbf{F} , i.e., $\mathbf{A}\mathbf{F} = \mathbf{0}$ and $\text{rank A} = n - \text{rank F}$. Show that \tilde{f} is convex if and only if for all $\mathbf{z} \in \text{dom } \tilde{f}$ there exists a $\lambda \in \mathbb{R}$ such that

$$\nabla^2 f(\mathbf{F}\mathbf{z} + \hat{\mathbf{x}}) + \lambda \mathbf{A}^T \mathbf{A} \succeq \mathbf{0}.$$

Hint. Use the following result: If $B \in \mathbb{S}^n$ and $\mathbf{A} \in \mathbb{R}^{p \times n}$, then $\mathbf{x}^T \mathbf{B} \mathbf{x} \geq 0$ for all $\mathbf{x} \in \mathcal{N}(\mathbf{A})$ if and only if there exists a λ such that $\mathbf{B} + \lambda \mathbf{A}^T \mathbf{A} \succeq \mathbf{0}$.

Exercise 275. *An extension of Jensen's inequality.* One interpretation of Jensen's inequality is that randomization or dithering hurts, i.e., raises the average value of a convex function: For f convex and v a zero mean random variable, we have $\mathbb{E}f(x_0 + v) \geq f(x_0)$. This leads to the following conjecture. If f_0 is convex, then the larger the variance of v , the larger $\mathbb{E}f(x_0 + v)$.

- (a) Give a counterexample that shows that this conjecture is false. Find zero mean random variables v and w , with $\text{var}(v) > \text{var}(w)$, a convex function f , and a point x_0 , such that $\mathbb{E}f(x_0 + v) < \mathbb{E}f(x_0 + w)$.
- (b) The conjecture is true when v and w are scaled versions of each other. Show that $\mathbb{E}f(x_0 + tv)$ is monotone increasing in $t \geq 0$, when f is convex and v is zero mean.

Exercise 276. *Monotone mappings.* A function $\phi : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is called monotone if for all $\mathbf{x}, \mathbf{y} \in \text{dom } \phi$,

$$(\phi(\mathbf{x}) - \phi(\mathbf{y}))^T (\mathbf{x} - \mathbf{y}) \geq 0.$$

(Note that ‘monotone’ as defined here is not the same as the definition given in Section 4.7.1. Both definitions are widely used.) Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a differentiable convex function. Show that its gradient ∇f is monotone. Is the converse true, i.e., is every monotone mapping the gradient of a convex function?

Exercise 277. Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex, $g : \mathbb{R}^n \rightarrow \mathbb{R}$ is concave, $\text{dom } f = \text{dom } g = \mathbb{R}^n$, and for all \mathbf{x} , $g(\mathbf{x}) \leq f(\mathbf{x})$. Show that there exists an affine function h such that for all \mathbf{x} , $g(\mathbf{x}) \leq h(\mathbf{x}) \leq f(\mathbf{x})$. In other words, if a concave function g is an underestimator of a convex function f , then we can fit an affine function between f and g .

Exercise 278. *Kullback-Leibler divergence and the information inequality.* Let D_{kl} be the KullbackLeibler divergence, as defined in (4.36). Prove the information inequality: $D_{kl}(\mathbf{u}, \mathbf{v}) \geq 0$ for all $\mathbf{u}, \mathbf{v} \in \mathbb{R}_{++}^n$. Also show that $D_{kl}(\mathbf{u}, \mathbf{v}) = 0$ if and only if $\mathbf{u} = \mathbf{v}$. *Hint.* The Kullback-Leibler divergence can be expressed as

$$D_{kl}(\mathbf{u}, \mathbf{v}) = f(\mathbf{u}) - f(\mathbf{v}) - \nabla f(\mathbf{u})^T(\mathbf{u} - \mathbf{v}),$$

where $f(\mathbf{v}) = \sum_{i=1}^n v_i \log v_i$ is the negative entropy of \mathbf{v} .

Exercise 279. *Convex-concave functions and saddle-points.* We say the function $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ is convex-concave if $f(\mathbf{x}, \mathbf{z})$ is a concave function of \mathbf{z} , for each fixed \mathbf{x} , and a convex function of \mathbf{x} , for each fixed \mathbf{z} . We also require its domain to have the product form $\text{dom } f = \mathbf{A} \times \mathbf{B}$, where $\mathbf{A} \subseteq \mathbb{R}^n$ and $\mathbf{B} \subseteq \mathbb{R}^m$ are convex.

- (a) Give a second-order condition for a twice differentiable function $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ to be convex-concave, in terms of its Hessian $\nabla^2 f(\mathbf{x}, \mathbf{z})$.
- (b) Suppose that $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ is convex-concave and differentiable, with $\nabla f(\tilde{\mathbf{x}}, \tilde{\mathbf{z}}) = \mathbf{0}$. Show that the saddle-point property holds: for all \mathbf{x}, \mathbf{z} , we have

$$f(\tilde{\mathbf{x}}, \mathbf{z}) \leq f(\tilde{\mathbf{x}}, \tilde{\mathbf{z}}) \leq f(\mathbf{x}, \tilde{\mathbf{z}})$$

Show that this implies that f satisfies the *strong max-min property*:

$$\sup_{\mathbf{x}} \inf_{\mathbf{z}} f(\mathbf{x}, \mathbf{z}) = \inf_{\mathbf{z}} \sup_{\mathbf{x}} f(\mathbf{x}, \mathbf{z}).$$

(And their common value is $f(\tilde{\mathbf{x}}, \tilde{\mathbf{z}})$).

- (c) Now suppose that $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ is differentiable, but not necessarily convex-concave, and the saddle-point property holds at $\tilde{\mathbf{x}}, \tilde{\mathbf{z}}$:

$$f(\tilde{\mathbf{x}}, \mathbf{z}) \leq f(\tilde{\mathbf{x}}, \tilde{\mathbf{z}}) \leq f(\mathbf{x}, \tilde{\mathbf{z}})$$

for all \mathbf{x}, \mathbf{z} . Show that $\nabla f(\tilde{\mathbf{x}}, \tilde{\mathbf{z}}) = \mathbf{0}$.

Examples

Exercise 280. A family of concave utility functions. For $0 < \alpha \leq 1$ let

$$u_\alpha = \frac{x^\alpha - 1}{\alpha}$$

with $\text{dom } u_\alpha = \mathbb{R}_+$. We also define $u_0(x) = \log x$ (with $\text{dom } u_0 = R_{++}$).

(a) Show that for $x > 0$, $u_0(x) = \lim_{\alpha \rightarrow 0} u_\alpha(x)$.

(b) Show that u_α are concave, monotone increasing, and all satisfy $u_\alpha(1) = 0$.

These functions are often used in economics to model the benefit or utility of some quantity of goods or money. Concavity of u_α means that the marginal utility (*i.e.*, the increase in utility obtained for a fixed increase in the goods) decreases as the amount of goods increases. In other words, concavity models the effect of satiation.

Exercise 281. For each of the following functions determine whether it is convex, concave, quasiconvex, or quasiconcave.

(a) $f(x) = e^x - 1$ on \mathbb{R} .

(b) $f(x_1, x_2) = x_1 x_2$ on \mathbb{R}_{++}^2 .

(c) $f(x_1, x_2) = 1/(x_1 x_2)$ on \mathbb{R}_{++}^2 .

(d) $f(x_1, x_2) = x_1/x_2$ on \mathbb{R}_{++}^2 .

(e) $f(x_1, x_2) = x_1^2/x_2$ on $\mathbb{R} \times \mathbb{R}_{++}$.

(f) $f(x_1, x_2) = x_1^\alpha x_2^{1-\alpha}$, where $0 \leq \alpha \leq 1$ on \mathbb{R}_{++}^2 .

Exercise 282. Suppose $p < 1, p \neq 0$. Show that the function

$$f(\mathbf{x}) = \left(\sum_{i=1}^n \mathbf{x}_i^p \right)^{1/p}$$

with $\text{dom } f = \mathbb{R}_{++}^n$ is concave. This includes as special cases $f(\mathbf{x}) = (\sum_{i=1}^n x_i^{1/2})^2$ and the harmonic mean $f(\mathbf{x}) = (\sum_{i=1}^n 1/x_i)^{-1}$. Hint. Adapt the proofs for the log-sum-exp function and the geometric mean in Section 4.1.5.

Exercise 283. Adapt the proof of concavity of the log-determinant function in Section 4.1.5 to show the following.

(a) $f(\mathbf{X}) = \text{tr}(\mathbf{X}^{-1})$ is convex on $\text{dom } f = \mathbb{S}_{++}^n$.

(b) $f(\mathbf{X}) = (\det \mathbf{X})^{1/n}$ is concave on $\text{dom } f = \mathbb{S}_{++}^n$.

Exercise 284. *Nonnegative weighted sums and integrals.*

(a) Show that $f(\mathbf{x}) = \sum_{i=1}^r \alpha_i x_{[i]}$ is a convex function of \mathbf{x} , where $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_r \geq 0$, and $x_{[i]}$ denotes the i th largest component of \mathbf{x} . (You can use the fact that $f(\mathbf{x}) = \sum_{i=1}^k x_{[i]}$ is convex on \mathbb{R}^n .)

(b) Let $T(\mathbf{x}, \omega)$ denote the trigonometric polynomial

$$T(\mathbf{x}, \omega) = x_1 + x_2 \cos \omega + x_3 \cos 2\omega + \dots + x_n \cos(n-1)\omega.$$

Show that the function

$$f(\mathbf{x}) = - \int_0^{2\pi} \log T(\mathbf{x}, \omega) d\omega$$

is convex on $\{\mathbf{x} \in \mathbb{R}^n \mid T(\mathbf{x}, \omega) > 0, 0 \leq \omega \leq 2\pi\}$.

Exercise 285. *Composition with an affine function.* Show that the following functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$ are convex.

(a) $f(\mathbf{x}) = \|\mathbf{Ax} - \mathbf{b}\|$ where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$, and $\|\cdot\|$ is a norm on \mathbb{R}^m .

(b) $f(\mathbf{x}) = -(\det(\mathbf{A}_0 + x_1 \mathbf{A}_1 + \dots + x_n \mathbf{A}_n))^{1/m}$, on $\{\mathbf{x} \mid \mathbf{A}_0 + x_1 \mathbf{A}_1 + \dots + x_n \mathbf{A}_n \succeq \mathbf{0}\}$, where $\mathbf{A}_i \in \mathbb{S}^m$.

(c) $f(\mathbf{X}) = \text{tr}(\mathbf{A}_0 + x_1 \mathbf{A}_1 + \dots + x_n \mathbf{A}_n)^{-1}$, on $\{\mathbf{x} \mid \mathbf{A}_0 + x_1 \mathbf{A}_1 + \dots + x_n \mathbf{A}_n \succ \mathbf{0}\}$, where $\mathbf{A}_i \in \mathbb{S}^m$. (Use the fact that $\text{tr}(\mathbf{X}^{-1})$ is convex on \mathbb{S}_{++}^m ; see Exercise 283.)

Exercise 286. *Pointwise maximum and supremum.* Show that the following functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$ are convex.

(a) $f(\mathbf{x}) = \max_{i=1, \dots, k} \|\mathbf{A}^{(i)} \mathbf{x} - \mathbf{b}^{(i)}\|$, where $\mathbf{A}^{(i)} \in \mathbb{R}^{m \times n}$, $\mathbf{b}^{(i)} \in \mathbb{R}^m$ and $\|\cdot\|$ is a norm on \mathbb{R}^m .

(b) $f(\mathbf{x}) = \sum_{i=1}^r |\mathbf{x}|_{[i]}$ on \mathbb{R}^n , where $|\mathbf{x}|$ denotes the vector with $|\mathbf{x}|_i = |\mathbf{x}_i|$ (i.e., $|\mathbf{x}|$ is the absolute value of \mathbf{x} , componentwise), and $|\mathbf{x}|_{[i]}$ is the i th largest component of $|\mathbf{x}|$. In other words, $|\mathbf{x}|_{[1]}, |\mathbf{x}|_{[2]}, \dots, |\mathbf{x}|_{[n]}$ are the absolute values of the components of \mathbf{x} , sorted in nonincreasing order.

(c) $f(\mathbf{x}) = \text{tr}(\mathbf{A}_0 + x_1 \mathbf{A}_1 + \dots + x_n \mathbf{A}_n)^{-1}$, on $\{\mathbf{x} \mid \mathbf{A}_0 + x_1 \mathbf{A}_1 + \dots + x_n \mathbf{A}_n \succeq \mathbf{0}\}$, where $\mathbf{A}_i \in \mathbb{S}^m$. (Use the fact that $\text{tr}(\mathbf{X}^{-1})$ is convex on \mathbb{S}_{++}^m ; see Exercise 283.)

Exercise 287. *Composition.* Show that the following functions are convex.

- (a) $f(\mathbf{x}) = -\log \left(-\log \left(\sum_{i=1}^m e^{\mathbf{a}_i^T \mathbf{x} + b_i} \right) \right)$ on $\text{dom } f = \{\mathbf{x} \mid \sum_{i=1}^m e^{\mathbf{a}_i^T \mathbf{x} + b_i} < 1\}$. You can use the fact that $\log(\sum_{i=1}^n e^y_i)$ is convex.
- (b) $f(\mathbf{x}, u, v) = -\sqrt{uv - \mathbf{x}^T \mathbf{x}}$ on $\text{dom } f = \{(\mathbf{x}, u, v) \mid uv > \mathbf{x}^T \mathbf{x}, u, v > 0\}$. Use the fact that $\mathbf{x}^T \mathbf{x} = u$ is convex in (\mathbf{x}, u) for $u > 0$, and that $-\sqrt{u_1 u_2}$ is convex on R_{++}^2 .
- (c) $f(\mathbf{x}, u, v) = -\log(uv - \mathbf{x}^T \mathbf{x})$ on $\text{dom } f = \{(\mathbf{x}, u, v) \mid uv > \mathbf{x}^T \mathbf{x}, u, v > 0\}$.
- (d) $f(\mathbf{x}, t) = -(t^p - \|\mathbf{x}\|_p^p)^{1/p}$ where $p > 1$ and $\text{dom } f = \{(\mathbf{x}, t) \mid t \geq \|\mathbf{x}\|_p\}$. You can use the fact that $\|\mathbf{x}\|_p^p/u^{p-1}$ is convex in (\mathbf{x}, u) for $u > 0$ (see Exercise 288), and that $-x^{1/p}y^{1-1/p}$ is convex on R_+^2 (see Exercise 281).
- (e) $f(\mathbf{x}, t) = -\log(t^p - \|\mathbf{x}\|_p^p)$ where $p > 1$ and $\text{dom } f = \{(\mathbf{x}, t) \mid t \geq \|\mathbf{x}\|_p\}$. You can use the fact that $\|\mathbf{x}\|_p^p/u^{p-1}$ is convex in (\mathbf{x}, u) for $u > 0$ (see Exercise 288).

Exercise 288. *Perspective of a function.*

- (a) Show that for $p > 1$,

$$f(\mathbf{x}, t) = \frac{|x_1|^p + \dots + |x_n|^p}{t^{p-1}} = \frac{\|\mathbf{x}\|_p^p}{t^{p-1}}$$

is convex on $\{(\mathbf{x}, t) \mid t > 0\}$.

- (b) Show that

$$f(\mathbf{x}) = \frac{\|\mathbf{Ax} + \mathbf{b}\|_2^2}{\mathbf{c}^T \mathbf{x} + d}$$

is convex on $\{\mathbf{x} \mid \mathbf{c}^T \mathbf{x} + d > 0\}$, where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{c} \in \mathbb{R}^n$ and $d \in \mathbb{R}$.

Exercise 289. *Some functions on the probability simplex.* Let x be a real-valued random variable which takes values in $\{a_1, \dots, a_n\}$ where $a_1 < a_2 < \dots < a_n$, with $\text{prob}(x = a_i) = p_i$, $i = 1, \dots, n$. For each of the following functions of \mathbf{p} (on the probability simplex $\{p \in R_+^n \mid \mathbf{1}^T \mathbf{p} = 1\}$), determine if the function is convex, concave, quasiconvex, or quasiconcave.

- (a) $\mathbb{E}x$.

- (b) $\text{prob}(x \geq \alpha)$.

- (c) $\text{prob}(\alpha \leq x \leq \beta)$.

- (d) $\sum_{i=1}^n p_i \log p_i$, the negative entropy of the distribution.

(e) $\text{var } x = \mathbb{E}(x - \mathbb{E}x)^2$.

(f) $\text{quartile}(x) = \inf\{\beta \mid \text{prob}(x \leq \beta) \geq 0.25\}$.

(g) The cardinality of the smallest set $A \subseteq \{1, \dots, n\}$ with probability $\geq 90\%$. (By cardinality we mean the number of elements in A .)

(h) The minimum width interval that contains 90% of the probability, *i.e.*,

$$\inf\{\beta - \alpha \mid \text{prob}(\alpha \leq \beta) \geq 0.9\}.$$

Exercise 290. *Maximum probability distance between distributions.* Let $\mathbf{p}, \mathbf{q} \in \mathbb{R}^n$ represent two probability distributions on $\{1, \dots, n\}$ (so $\mathbf{p}, \mathbf{q} \succeq 0, \mathbf{1}^T \mathbf{p} = \mathbf{1}^T \mathbf{q} = 1$). We define the maximum probability distance $d_{mp}(\mathbf{p}, \mathbf{q})$ between \mathbf{p} and \mathbf{q} as the maximum difference in probability assigned by \mathbf{p} and \mathbf{q} , over all events:

$$d_{mp}(\mathbf{p}, \mathbf{q}) = \max\{|\text{prob}(\mathbf{p}, \mathbf{C}) - \text{prob}(\mathbf{q}, \mathbf{C})| \mid \mathbf{C} \subseteq \{1, \dots, n\}\}.$$

Here $\text{prob}(\mathbf{p}, \mathbf{C})$ is the probability of \mathbf{C} , under the distribution \mathbf{p} , *i.e.*, $\text{prob}(\mathbf{p}, \mathbf{C}) = \sum_{i \in \mathbf{C}} p_i$.

Find a simple expression for d_{mp} , involving $\|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^n |p_i - q_i|$, and show that d_{mp} is a convex function on $\mathbb{R}^n \times \mathbb{R}^n$. (Its domain is $\{(\mathbf{p}, \mathbf{q}) \mid \mathbf{p}, \mathbf{q} \succeq 0, \mathbf{1}^T \mathbf{p} = \mathbf{1}^T \mathbf{q} = 1\}$, but it has a natural extension to all of $\mathbb{R}^n \times \mathbb{R}^n$.)

Exercise 291. *More functions of eigenvalues.* Let $\lambda_1(\mathbf{X}) \geq \lambda_2(\mathbf{X}) \geq \dots \geq \lambda_n(\mathbf{X})$ denote the eigenvalues of a matrix $\mathbf{X} \in \mathbb{S}^n$. We have already seen several functions of the eigenvalues that are convex or concave functions of \mathbf{X} .

- The maximum eigenvalue $\lambda_1(\mathbf{X})$ is convex (Example 207). The minimum eigenvalue $\lambda_n(\mathbf{X})$ is concave.
- The sum of the eigenvalues (or trace), $\text{tr} \mathbf{X} = \lambda_1(\mathbf{X}) + \dots + \lambda_n(\mathbf{X})$, is linear.
- The sum of the inverses of the eigenvalues (or trace of the inverse), $\text{tr}(\mathbf{X}^{-1}) = \sum_{i=1}^n 1/\lambda_i(\mathbf{X})$, is convex on \mathbb{S}_{++}^n (Exercise 283).
- The geometric mean of the eigenvalues, $(\det \mathbf{X})^{1/n} = (\prod_{i=1}^n \lambda_i(\mathbf{X}))^{1/n}$, and the logarithm of the product of the eigenvalues, $\log \det \mathbf{X} = \sum_{i=1}^n \log \lambda_i(\mathbf{X})$, are concave on $\mathbf{X} \in \mathbb{S}_{++}^n$ (Exercise 283 and Section 4.1.5).

In this problem we explore some more functions of eigenvalues, by exploiting variational characterizations.

- (a) *Sum of k largest eigenvalues.* Show that $\sum_{i=1}^k \lambda_i(\mathbf{X})$ is convex on \mathbb{S}^n . Hint. Use the variational characterization

$$\sum_{i=1}^k \lambda_i(\mathbf{X}) = \sup\{\text{tr}(\mathbf{V}^T \mathbf{X} \mathbf{V}) \mid \mathbf{V} \in \mathbb{R}^{n \times k}, \mathbf{V}^T \mathbf{V} = \mathbf{I}\}$$

- (b) *Geometric mean of k smallest eigenvalues.* Show that $(\prod_{i=n-k+1}^n \lambda_i(\mathbf{X}))^{1/k}$ is concave on \mathbb{S}_{++}^n . Hint. For $\mathbf{X} \succ \mathbf{0}$, we have

$$\left(\prod_{i=n-k+1}^n \lambda_i(\mathbf{X}) \right)^{1/k} = \frac{1}{k} \inf\{\text{tr}(\mathbf{V}^T \mathbf{X} \mathbf{V}) \mid \mathbf{V} \in \mathbb{R}^{n \times k}, \det \mathbf{V}^T \mathbf{V} = 1\}.$$

- (c) *Log of product of k smallest eigenvalues.* Show that $\sum_{i=n-k+1}^n \log \lambda_i(\mathbf{X})$ is concave on \mathbb{S}_{++}^n . Hint. For $\mathbf{X} \succ \mathbf{0}$,

$$\prod_{i=n-k+1}^n \lambda_i(\mathbf{X}) = \inf \left\{ \prod_{i=1}^k (\mathbf{V}^T \mathbf{X} \mathbf{V})_{ii} \mid \mathbf{V} \in \mathbb{R}^{n \times k}, \mathbf{V}^T \mathbf{V} = \mathbf{I} \right\}.$$

Exercise 292. *Diagonal elements of Cholesky factor.* Each $\mathbf{X} \in \mathbb{S}_{++}^n$ has a unique Cholesky factorization $\mathbf{X} = \mathbf{L} \mathbf{L}^T$, where \mathbf{L} is lower triangular, with $L_{ii} > 0$. Show that L_{ii} is a concave function of \mathbf{X} (with domain \mathbb{S}_{++}^n). Hint. L_{ii} can be expressed as $L_{ii} = (w - \mathbf{z}^T \mathbf{Y}^{-1} \mathbf{z})^{1/2}$, where

$$\begin{bmatrix} \mathbf{Y} & \mathbf{z} \\ \mathbf{z}^T & w \end{bmatrix}$$

is the leading $i \times i$ submatrix of \mathbf{X} .

Operations that preserve convexity

Exercise 293. *Expressing a convex function as the pointwise supremum of a family of affine functions.* In this problem we extend the result proved in Section 4.2.3 to the case where $\text{dom } f \neq \mathbb{R}^n$. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a convex function. Define $\tilde{f} : \mathbb{R}^n \rightarrow \mathbb{R}$ as the pointwise supremum of all affine functions that are global underestimators of f :

$$\tilde{f}(x) = \sup\{g(x) \mid g \text{ affine, } g(\mathbf{z}) \leq f(\mathbf{z}) \text{ for all } \mathbf{z}\}.$$

- (a) Show that $f(x) = \tilde{f}(x)$ for $x \in \text{int dom } f$.

- (b) Show that $f = \tilde{f}$ if f is closed (i.e., $\text{epi } f$ is a closed set; see Section 2.11.3.3).

Exercise 294. *Representation of piecewise-linear convex functions.* A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, with $\text{dom } f = \mathbb{R}^n$, is called piecewise-linear if there exists a partition of \mathbb{R}^n as

$$\mathbb{R}^n = \mathbf{X}_1 \cup \mathbf{X}_2 \cup \dots \cup \mathbf{X}_L,$$

where $\text{int}\mathbf{X}_i \neq \emptyset$ and $\text{int}\mathbf{X}_i \cap \text{int}\mathbf{X}_j = \emptyset$, for $i \neq j$, and a family of affine functions $\mathbf{a}_1^T \mathbf{x} + b_1, \dots, \mathbf{a}_L^T \mathbf{x} + b_L$ such that $f(\mathbf{x}) = \mathbf{a}_i^T \mathbf{x} + b_i$ for $\mathbf{x} \in \mathbf{X}_i$. Show that this means that $f(\mathbf{x}) = \max\{\mathbf{a}_1^T \mathbf{x} + b_1, \dots, \mathbf{a}_L^T \mathbf{x} + b_L\}$.

Exercise 295. *Convex hull or envelope of a function.* The convex hull or convex envelope of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is defined as

$$g(\mathbf{x}) = \inf\{t \mid (\mathbf{x}, t) \in \text{conv epi } f\}.$$

Geometrically, the epigraph of g is the convex hull of the epigraph of f . Show that g is the largest convex underestimator of f . In other words, show that if h is convex and satisfies $h(\mathbf{x}) \leq f(\mathbf{x})$ for all \mathbf{x} , then $h(\mathbf{x}) \leq g(\mathbf{x})$ for all \mathbf{x} .

Exercise 296. *Largest homogeneous underestimator.* Let f be a convex function. Define the function g as

$$g(\mathbf{x}) = \inf_{\alpha > 0} \frac{f(\alpha \mathbf{x})}{\alpha}.$$

- (a) Show that g is homogeneous ($g(t\mathbf{x}) = tg(\mathbf{x})$ for all $t \geq 0$).
- (b) Show that g is the largest homogeneous underestimator of f : If h is homogeneous and $h(\mathbf{x}) \leq f(\mathbf{x})$ for all \mathbf{x} , then we have $h(\mathbf{x}) \leq g(\mathbf{x})$ for all \mathbf{x} .
- (c) Show that g is convex.

Exercise 297. *Products and ratios of convex functions.* In general the product or ratio of two convex functions is not convex. However, there are some results that apply to functions on \mathbb{R} . Prove the following.

- (a) If f and g are convex, both nondecreasing (or nonincreasing), and positive functions on an interval, then f_g is convex.
- (b) If f and g are concave, positive, with one nondecreasing and the other nonincreasing, then f_g is concave.
- (c) If f is convex, nondecreasing, and positive, and g is concave, nonincreasing, and positive, then f/g is convex.

Exercise 298. *Direct proof of perspective theorem.* Give a direct proof that the perspective function g , as defined in Section 4.2.6, of a convex function f is convex: Show that $\mathbf{dom} g$ is a convex set, and that for $(\mathbf{x}, t), (\mathbf{y}, s) \in \mathbf{dom} g$, and $0 \leq \theta \leq 1$, we have

$$g(\theta\mathbf{x} + (1 - \theta)\mathbf{y}, \theta t + (1 - \theta)s) \leq \theta g(\mathbf{x}, t) + (1 - \theta)g(\mathbf{y}, s).$$

Exercise 299. *The Minkowski function.* The Minkowski function of a convex set C is defined as

$$M_C(\mathbf{x}) = \inf\{t > 0 \mid t^{-1}\mathbf{x} \in C\}.$$

- (a) Draw a picture giving a geometric interpretation of how to find $M_C(x)$.
- (b) Show that M_C is homogeneous, i.e., $M_C(\alpha\mathbf{x}) = \alpha M_C(\mathbf{x})$ for $\alpha \geq 0$.
- (c) What is $\mathbf{dom} M_C$?
- (d) Show that M_C is a convex function.
- (e) Suppose C is also closed, symmetric (if $\mathbf{x} \in C$ then $-\mathbf{x} \in C$), and has nonempty interior. Show that M_C is a norm. What is the corresponding unit ball?

Exercise 300. *Support function calculus.* Recall that the support function of a set $C \subseteq \mathbb{R}^n$ is defined as $S_C(\mathbf{y}) = \sup\{\mathbf{y}^T \mathbf{x} \mid \mathbf{x} \in C\}$. In Example 204 we showed that S_C is a convex function.

- (a) Show that $S_B = S_{\text{conv}(B)}$.
- (b) Show that $S_{A+B} = S_A + S_B$.
- (c) Show that $S_{A \cup B} = \max\{S_A, S_B\}$.
- (d) Let B be closed and convex. Show that $A \subseteq B$ if and only if $S_A(\mathbf{y}) \leq S_B(\mathbf{y})$ for all \mathbf{y} .

Conjugate functions

Exercise 301. Derive the conjugates of the following functions.

- (a) *Max function.* $f(\mathbf{x}) = \max_{i=1,\dots,n} x_i$ on \mathbb{R}^n .
- (b) *Sum of largest elements.* $f(\mathbf{x}) = \sum_{i=1}^r x_{[i]}$ on \mathbb{R}^n .

(c) Piecewise-linear function on \mathbb{R} . $f(x) = \max_{i=1,\dots,n}(a_i x + b_i)$ on \mathbb{R} . You can assume that the a_i are sorted in increasing order, i.e., $a_1 \leq \dots \leq a_m$, and that none of the functions $a_i x + b_i$ is redundant, i.e., for each k there is at least one x with $f(x) = a_k x + b_k$.

(d) Power function. $f(x) = x^p$ on \mathbb{R}_{++} , where $p > 1$. Repeat for $p < 0$.

(e) Geometric mean. $f(\mathbf{x}) = -(\prod x_i)^{1/n}$ on \mathbb{R}_{++}^n .

(f) Negative generalized logarithm for second-order cone. $f(\mathbf{x}, t) = -\log(t^2 - \mathbf{x}^T \mathbf{x})$ on $\{(\mathbf{x}, t) \in \mathbb{R}^n \times \mathbb{R} \mid \|\mathbf{x}\|_2 < t\}$.

Exercise 302. Show that the conjugate of $f(\mathbf{X}) = \text{tr}(\mathbf{X}^{-1})$ with $\text{dom } f = \mathbb{S}_{++}^n$ is given by

$$f^*(\mathbf{Y}) = -2\text{tr}(-\mathbf{Y})^{1/2}, \quad \text{dom } f^* = -\mathbb{S}_+^n.$$

Hint. The gradient of f is $\nabla f(\mathbf{X}) = -\mathbf{X}^{-2}$.

Exercise 303. *Young's inequality.* Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be an increasing function, with $f(0) = 0$, and let g be its inverse. Define F and G as

$$F(x) = \int_0^x f(a)da, \quad G(y) = \int_0^y g(a)da.$$

Show that F and G are conjugates. Give a simple graphical interpretation of Young's inequality,

$$xy \leq F(x) + G(y).$$

Exercise 304. Properties of conjugate functions.

(a) Conjugate of convex plus affine function. Define $g(\mathbf{x}) = f(\mathbf{x}) + \mathbf{c}^T \mathbf{x} + d$, where f is convex. Express g^* in terms of f^* (and \mathbf{c}, d).

(b) Conjugate of perspective. Express the conjugate of the perspective of a convex function f in terms of f^* .

(c) Conjugate and minimization. Let $f(\mathbf{x}, \mathbf{z})$ be convex in (\mathbf{x}, \mathbf{z}) and define $g(\mathbf{x}) = \inf_z f(\mathbf{x}, \mathbf{z})$. Express the conjugate g^* in terms of f^* . As an application, express the conjugate of $g(\mathbf{x}) = \inf\{h(\mathbf{z}) \mid \mathbf{A}\mathbf{z} + \mathbf{b} = \mathbf{x}\}$, where h is convex, in terms of h^* , \mathbf{A} , and \mathbf{b} .

(d) Conjugate of conjugate. Show that the conjugate of the conjugate of a closed convex function is itself: $f = f^{**}$ if f is closed and convex. (A function is closed if

its epigraph is closed; see Section 2.11.3.3.) *Hint.* Show that f^{**} is the pointwise supremum of all affine global underestimators of f . Then apply the result of Exercise 293.

Exercise 305. *Gradient and Hessian of conjugate function.* Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex and twice continuously differentiable. Suppose $\bar{\mathbf{y}}$ and $\bar{\mathbf{x}}$ are related by $\bar{\mathbf{y}} = \nabla f(\bar{\mathbf{x}})$, and that $\nabla^2 f(\bar{\mathbf{x}}) \succ \mathbf{0}$.

- (a) Show that $\nabla f^*(\bar{\mathbf{y}}) = \bar{\mathbf{x}}$.
- (b) Show that $\nabla^2 f^*(\bar{\mathbf{y}}) = \nabla^2 f(\bar{\mathbf{x}})^{-1}$.

Exercise 306. *Domain of conjugate function.* Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a twice differentiable convex function and $\mathbf{x} \in \text{dom } f$. Show that for small enough \mathbf{u} we have

$$\mathbf{y} = \nabla f(\mathbf{x}) + \nabla^2 f(\mathbf{x})\mathbf{u} \in \text{dom } f^*,$$

i.e., $\mathbf{y}^T \mathbf{x} - f(\mathbf{x})$ is bounded above. It follows that $\dim(\text{dom } f) \geq \text{rank } \nabla^2 f(\mathbf{x})$. *Hint.* Consider $\nabla f(\mathbf{x} + t\mathbf{v})$, where t is small, and \mathbf{v} is any vector in \mathbb{R}^n .

Quasiconvex functions

Exercise 307. *Approximation width.* Let $f_0, \dots, f_n : \mathbb{R} \rightarrow \mathbb{R}$ be given continuous functions. We consider the problem of approximating f_0 as a linear combination of f_1, \dots, f_n . For $\mathbf{x} \in \mathbb{R}^n$, we say that $f = x_1 f_1 + \dots + x_n f_n$ approximates f_0 with tolerance $\epsilon > 0$ over the interval $[0, T]$ if $|f(t) - f_0(t)| \leq \epsilon$ for $0 \leq t \leq T$. Now we choose a fixed tolerance $\epsilon > 0$ and define the approximation width as the largest T such that f approximates f_0 over the interval $[0, T]$:

$$W(\mathbf{x}) = \sup\{T \mid |x_1 f_1(t) + \dots + x_n f_n(t) - f_0(t)| \leq \epsilon \text{ for } 0 \leq t \leq T\}.$$

Show that W is quasiconcave.

Exercise 308. *First-order condition for quasiconvexity.* Prove the first-order condition for quasiconvexity given in Section 4.5.3: A differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, with $\text{dom } f$ convex, is quasiconvex if and only if for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$,

$$f(\mathbf{y}) \leq f(\mathbf{x}) \Rightarrow \nabla f(\mathbf{x})^T (\mathbf{y} - \mathbf{x}) \leq 0.$$

Hint. It suffices to prove the result for a function on \mathbb{R} ; the general result follows by restriction to an arbitrary line.

Exercise 309. *Second-order conditions for quasiconvexity.* In this problem we derive alternate representations of the second-order conditions for quasiconvexity given in Section 4.5.3. Prove the following.

- (a) A point $\mathbf{x} \in \text{dom } f$ satisfies (4.47) if and only if there exists a σ such that

$$\nabla^2 f(\mathbf{x}) + \sigma \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^T \succeq \mathbf{0}. \quad (4.56)$$

It satisfies (4.48) for all $\mathbf{y} \neq \mathbf{0}$ if and only if there exists a σ such that

~~$$\nabla^2 f(\mathbf{x}) + \sigma \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^T \succ \mathbf{0}. \quad (4.57)$$~~

Hint. We can assume without loss of generality that $\nabla^2 f(\mathbf{x})$ is diagonal.

- (b) A point $\mathbf{x} \in \text{dom } f$ satisfies (4.47) if and only if either $\nabla f(\mathbf{x}) = \mathbf{0}$ and $\nabla f(\mathbf{x}) \succeq \mathbf{0}$, or $\nabla f(\mathbf{x}) \neq \mathbf{0}$ and the matrix

$$H(\mathbf{x}) = \begin{bmatrix} \nabla^2 f(\mathbf{x}) & \nabla f(\mathbf{x}) \\ \nabla f(\mathbf{x})^T & 0 \end{bmatrix} \quad (4.58)$$

has exactly one negative eigenvalue. It satisfies (4.48) for all $\mathbf{y} \neq \mathbf{0}$ if and only if $H(\mathbf{x})$ has exactly one nonpositive eigenvalue. *Hint.* You can use the result of part (a). The following result, which follows from the eigenvalue interlacing theorem in linear algebra, may also be useful: If $\mathbf{B} \in \mathbb{S}^n$ and $\mathbf{a} \in \mathbb{R}^n$, then

$$\lambda_n \left(\begin{bmatrix} \mathbf{B} & \mathbf{a} \\ \mathbf{a}^T & 0 \end{bmatrix} \right) \geq \lambda_n(\mathbf{B}).$$

Exercise 310. Use the first and second-order conditions for quasiconvexity given in Section 4.5.3 to verify quasiconvexity of the function $f(\mathbf{x}) = -x_1 x_2$, with $\text{dom } f = \mathbb{R}_{++}^2$.

Exercise 311. *Quasilinear functions with domain \mathbb{R}^n .* A function on \mathbb{R} that is quasilinear (*i.e.*, quasiconvex and quasiconcave) is monotone, *i.e.*, either nondecreasing or nonincreasing. In this problem we consider a generalization of this result to functions on \mathbb{R}^n . Suppose the function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is quasilinear and continuous with $\text{dom } f = \mathbb{R}^n$. Show that it can be expressed as $f(\mathbf{x}) = g(\mathbf{a}^T \mathbf{x})$, where $g : \mathbb{R} \rightarrow \mathbb{R}$ is monotone and $\mathbf{a} \in \mathbb{R}^n$. In other words, a quasilinear function with domain \mathbb{R}^n must be a monotone function of a linear function. (The converse is also true.)

Log-concave and log-convex functions

Exercise 312. Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is differentiable, $\text{dom } f$ is convex, and $f(\mathbf{x}) > 0$ for all $\mathbf{x} \in \text{dom } f$. Show that f is log-concave if and only if for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$,

$$\frac{f(\mathbf{y})}{f(\mathbf{x})} \leq \exp \frac{\nabla f(\mathbf{x})^T (\mathbf{y} - \mathbf{x})}{f(\mathbf{x})}.$$

Exercise 313. Show that if $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is log-concave and $a \geq 0$, then the function $g = f - a$ is log-concave, where $\text{dom } g = \{x \in \text{dom } f \mid f(\mathbf{x}) > a\}$.

Exercise 314. Show that the following functions are log-concave.

(a) *Logistic function:* $f(x) = e^x / (1 + e^x)$ with $\text{dom } f = \mathbb{R}$.

(b) *Harmonic mean:*

$$f(\mathbf{x}) = \frac{1}{1/x_1 + \dots + 1/x_n}, \text{dom } f = \mathbb{R}_{++}^n.$$

(c) *Product over sum:*

$$f(\mathbf{x}) = \frac{\prod_{i=1}^n x_i}{\sum_{i=1}^n x_i}, \text{dom } f = \mathbb{R}_{++}^n.$$

(d) *Determinant over trace:*

$$f(\mathbf{X}) = \frac{\det \mathbf{X}}{\text{tr } \mathbf{X}}, \text{dom } f = \mathbb{S}_{++}^n.$$

Exercise 315. *Coefficients of a polynomial as a function of the roots.* Show that the coefficients of a polynomial with real negative roots are log-concave functions of the roots. In other words, the functions $a_i : \mathbb{R}^n \rightarrow \mathbb{R}$, defined by the identity

$$s^n + a_1(\lambda)s^{n-1} + \dots + a_{n-1}(\lambda)s + a_n(\lambda) = (s - \lambda_1)(s - \lambda_2)\dots(s - \lambda_n)$$

are log-concave on $-\mathbb{R}_{++}^n$. *Hint.* The function

$$S_k(\mathbf{x}) = \sum_{1 \leq i_1 < i_2 < \dots < i_k \leq n} x_{i_1} x_{i_2} \dots x_{i_k},$$

with $\text{dom } S_k \in \mathbb{R}_+^n$ and $1 \leq k \leq n$, is called the k th elementary symmetric function on \mathbb{R}^n . It can be shown that $S_k^{1/k}$ is concave.

Exercise 316. Let p be a polynomial on \mathbb{R} , with all its roots real. Show that it is log-concave on any interval on which it is positive.

Exercise 317. Log-convexity of moment functions. Suppose $f : \mathbb{R} \rightarrow \mathbb{R}$ is nonnegative with $\mathbb{R}_+ \subseteq \text{dom } f$. For $x \geq 0$ define

$$\phi(x) = \int_0^{-\infty} u^x f(u) du.$$

Show that ϕ is a log-convex function. (If x is a positive integer, and f is a probability density function, then $\phi(x)$ is the x th moment of the distribution.) Use this to show that the Gamma function,

$$\Gamma(x) = \int_0^{-\infty} u^{x-1} e^{-u} du,$$

is log-convex for $x \geq 1$.

Exercise 318. Suppose \mathbf{x} and \mathbf{y} are independent random vectors in \mathbb{R}^n , with log-concave probability density functions f and g , respectively. Show that the probability density function of the sum $\mathbf{z} = \mathbf{x} + \mathbf{y}$ is log-concave.

Exercise 319. *Log-concavity of Gaussian cumulative distribution function.* The cumulative distribution function of a Gaussian random variable,

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt,$$

is log-concave. This follows from the general result that the convolution of two log-concave functions is log-concave. In this problem we guide you through a simple self-contained proof that f is log-concave. Recall that f is log-concave if and only if $f''(x)f(x) \leq f'(x)^2$ for all x .

(a) Verify that $f''(x)f(x) \leq f'(x)^2$ for $x \geq 0$. That leaves us the hard part, which is to show the inequality for $x < 0$.

(b) Verify that for any t and x we have $t^2/2 \geq -x^2/2 + xt$.

(c) Using part (b) show that $e^{-t^2/2} \leq e^{x^2/2-xt}$. Conclude that

$$\int_{-\infty}^x e^{-t^2/2} dt \leq e^{x^2/2} \int_{-\infty}^x e^{-xt} dt.$$

(d) Use part (c) to verify that $f''(x)f(x) \leq f'(x)^2$ for $x \leq 0$.

Exercise 320. *Log-concavity of the cumulative distribution function of a log-concave probability density.* In this problem we extend the result of Exercise 319. Let $g(t) = \exp(-h(t))$ be a differentiable log-concave probability density function, and let

$$f(x) = \int_{-\infty}^x g(t) dt = \int_{-\infty}^x e^{h(t)} dt$$

be its cumulative distribution. We will show that f is log-concave, i.e., it satisfies $f''(x)f(x) \leq (f'(x))^2$ for all x .

(a) Express the derivatives of f in terms of the function h . Verify that $f''(x)f(x) \leq (f'(x))^2$ if $h(x) \geq 0$.

(b) Assume that $h'(x) < 0$. Use the inequality

$$h(t) \geq h(x) + h'(x)(t - x)$$

(which follows from convexity of h), to show that

$$\int_{-\infty}^x e^{-h(t)} dt \leq \frac{e^{-h(x)}}{-h'(x)}.$$

Use this inequality to verify that $f''(x)f(x) \leq (f'(x))^2$ if $h'(x) \geq 0$.

Exercise 321. More log-concave densities. Show that the following densities are log-concave.

(a) The *gamma density*, defined by

$$f(x) = \frac{\alpha}{\Gamma(\lambda)} x^{\lambda-1} e^{-\alpha x},$$

with $\text{dom } f = R_+$. The parameters λ and α satisfy $\lambda \geq 1, \alpha > 0$.

(b) The *Dirichlet density*

$$f(\mathbf{x}) = \frac{\Gamma(\mathbf{1}^T \boldsymbol{\lambda})}{\Gamma(\lambda_1) \dots \Gamma(\lambda_{n+1})} x_1^{\lambda_1-1} \dots x_n^{\lambda_n-1} \left(1 - \sum_{i=1}^n x_i\right)^{\lambda_{n+1}-1}$$

with $\text{dom } f = \{\mathbf{x} \in \mathbb{R}_{++}^n \mid \mathbf{1}^T \mathbf{x} < 1\}$. The parameter $\boldsymbol{\lambda}$ satisfies $\boldsymbol{\lambda} \succeq 1$.

Convexity with respect to a generalized inequality

Exercise 322. Show that the function $f(\mathbf{X}) = \mathbf{X}^{-1}$ is matrix convex on \mathbb{S}_{++}^n .

Exercise 323. *Schur complement.* Suppose $\mathbf{X} \in \mathbb{S}^n$ partitioned as

$$\mathbf{X} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix},$$

where $\mathbf{A} \in \mathbb{S}^k$. The Schur complement of \mathbf{X} (with respect to \mathbf{A}) is $\mathbf{S} = \mathbf{C} - \mathbf{B}^T \mathbf{A}^{-1} \mathbf{B}$ (see Section 2.11.7.5). Show that the Schur complement, viewed as function from \mathbb{S}^n into \mathbb{S}^{n-k} , is matrix concave on \mathbb{S}_{++}^n .

Exercise 324. *Second-order conditions for K -convexity.* Let $K \in \mathbb{R}^m$ be a proper convex cone, with associated generalized inequality \preceq_K . Show that a twice differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, with convex domain, is K -convex if and only if for all $\mathbf{x} \in \text{dom } f$ and all $\mathbf{y} \in \mathbb{R}^n$,

$$\sum_{i,j=1}^n \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} y_i y_j \succeq_K \mathbf{0},$$

i.e., the second derivative is a K -nonnegative bilinear form. (Here $\partial^2 f / \partial x_i \partial x_j \in \mathbb{R}^m$, with components $\partial^2 f_k / \partial x_i \partial x_j$, for $k = 1, \dots, m$; see Section 2.11.4.1.)

Exercise 325. *Sublevel sets and epigraph of K -convex functions.* Let $K \subseteq \mathbb{R}^m$ be a proper convex cone with associated generalized inequality \preceq_K , and let $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$. For $\alpha \in \mathbb{R}^m$, the α -sublevel set of f (with respect to \preceq_K) is defined as

$$C_\alpha = \{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) \preceq_K \alpha\}.$$

The epigraph of f , with respect to \preceq_K , is defined as the set

$$\text{epi}_K f = \{(\mathbf{x}, \mathbf{t}) \in \mathbb{R}^{n+m} \mid f(\mathbf{x}) \preceq_K \mathbf{t}\}.$$

Show the following:

- (a) If f is K -convex, then its sublevel sets C_α are convex for all α .
- (b) f is K -convex if and only if $\text{epi}_K f$ is a convex set.

Interplay between Convexity and Smoothness (Taken from [20])

Exercise 326. Let π_C be the projection operator onto a convex set C . Prove:

$$\langle \pi_C(\mathbf{y}) - \mathbf{x}, \pi_C(\mathbf{y}) - \mathbf{y} \rangle \leq 0, \quad \forall \mathbf{x} \in C.$$

Further show that

$$\|\pi_C(\mathbf{y}) - \mathbf{x}\|^2 + \|\pi_C(\mathbf{y}) - \mathbf{y}\|^2 \leq \|\mathbf{x} - \mathbf{y}\|^2, \quad \forall \mathbf{x} \in C.$$

Exercise 327. If f is a β -smooth function, prove

$$f\left(\mathbf{x} - \frac{1}{\beta} \nabla f(\mathbf{x})\right) - f(\mathbf{x}) \leq -\frac{1}{2\beta} \|\nabla f(\mathbf{x})\|^2. \quad (4.59)$$

Exercise 328. Let f satisfy

$$0 \leq f(\mathbf{x}) - f(\mathbf{y}) - \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \leq \frac{\beta}{2} \|\mathbf{x} - \mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y}. \quad (4.60)$$

Prove that

$$f(\mathbf{x}) - f(\mathbf{y}) \leq \langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y} \rangle - \frac{1}{2\beta} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2, \quad \forall \mathbf{x}, \mathbf{y}. \quad (4.61)$$

Exercise 329. Let f be L -smooth and μ -strongly convex on \mathbb{R}^n . Then

$$\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \frac{L\mu}{L + \mu} \|\mathbf{x} - \mathbf{y}\|^2 + \frac{1}{L + \mu} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2, \quad \forall \mathbf{x}, \mathbf{y}. \quad (4.62)$$

Exercise 330. Let f be L -smooth and μ -strongly convex and L -smooth and \mathbf{x}^* be its minimizer. Then

$$2\mu(f(\mathbf{x}) - f(\mathbf{x}^*)) \leq \|\nabla f(\mathbf{x})\|^2 \leq 2L(f(\mathbf{x}) - f(\mathbf{x}^*)). \quad (4.63)$$

Exercise 331. Let f_i be L_i -smooth and μ_i -strongly convex on \mathbb{R}^n , $i = 1, \dots, n$. Then $\tilde{f}(\mathbf{x}) = \sum_{i=1}^n f_i(\mathbf{x})$ is also \tilde{L} -smooth and $\tilde{\mu}$ -strongly convex. What are the values of \tilde{L} and $\tilde{\mu}$? What are the results for $\hat{f}(\mathbf{x}) = \sum_{i=1}^n f_i(\mathbf{x}_i)$, where $\mathbf{x} = (\mathbf{x}_1^T, \dots, \mathbf{x}_n^T)^T$?

Exercise 332. Let f be μ -strongly convex and differentiable on \mathbb{R}^n and $\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x}} f(\mathbf{x})$. Then for all $\mathbf{x} \in \mathbb{R}^n$, $\langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{x}^* \rangle \geq \|\mathbf{x} - \mathbf{x}^*\|^2$.

Exercise 333. Let f^* be the conjugate of f , $f_\delta^*(\mathbf{y}) = f^*(\mathbf{y}) + \frac{\delta}{2} \|\mathbf{y}\|^2$, and $f_\delta(\mathbf{x}) = \max_{\mathbf{y}} \langle \mathbf{x}, \mathbf{y} \rangle - f_\delta^*(\mathbf{y})$. $f_\delta(\mathbf{x})$ is called a δ -smoothing of $f(\mathbf{x})$. Show that

- If f is μ -strongly convex, then f_δ is also μ -strongly convex, and if f is not strongly convex, then f_δ is also not strongly convex.
- If $\delta_1 \geq \delta_2$, then $f_{\delta_1}(\mathbf{x}) \leq f_{\delta_2}(\mathbf{x})$.

Exercise 334. Prove that if $c_1 \geq c_2 > 0$, then $E_{c_1}f(\mathbf{x}) \geq E_{c_2}f(\mathbf{x})(\mathbf{x})$, where $E_c f(\mathbf{x})$ is the envelope function of f .

(Taken from Chapter 9 of [21])

Exercise 335. Let $g(t)$ be a strictly convex function for $t > 0$. For $x > 0$ and $y > 0$, define the function

$$f(x, y) = xg(y/x).$$

Use induction to prove that

$$\sum_{n=1}^N f(x_n, y_n) \geq f(x_+, y_+),$$

for any positive numbers $\{x_n\}$ and $\{y_n\}$, where $x_+ = \sum_{n=1}^N x_n$ and $y_+ = \sum_{n=1}^N y_n$. Also show that equality is attained if and only if the finite sequences $\{x_n\}$ and $\{y_n\}$ are proportional.

Exercise 336. Use Exercise 335 to prove Cauchy's inequality. Hint: $g(t) = -\sqrt{t}$.

Exercise 337. Use Exercise 335 to prove Hölder's inequality. Hint: $g(t) = -t^{1/q}$.

Exercise 338. Use Exercise 335 to prove Minkowski's inequality. Hint: $g(t) = -(t^{1/p} + 1)^p$.

Exercise 339. Use Exercise 335 to prove Milne's inequality:

$$x_+y_+ \geq \left(\sum_{n=1}^N (x_n + y_n) \right) \left(\sum_{n=1}^N \frac{x_n y_n}{x_n + y_n} \right).$$

Hint: $g(t) = -\frac{t}{1+t}$.

Exercise 340. For $x > 0$ and $y > 0$, let $f(x, y)$ be the Kullback-Leibler function:

$$f(x, y) = KL(x, y) = x \log \frac{x}{y} + y - x.$$

Use Exercise 335 to prove:

$$\sum_{n=1}^N KL(x_n, y_n) \geq KL(x_+, y_+).$$

Exercise 341. For $\mathbf{x} > 0$ and $\mathbf{y} > 0$ be vectors in \mathbb{R}^N , and let

$$KL(\mathbf{x}, \mathbf{y}) = \sum_{n=1}^N KL(x_n, y_n)$$

be the Kullback-Leibler distance from \mathbf{y} to \mathbf{x} . Let $y_+ = \sum_{n=1}^N y_n > 0$. Show that:

$$KL(\mathbf{x}, \mathbf{y}) = KL(x_+, y_+) + KL\left(\mathbf{x}, \frac{x_+}{y_+}\mathbf{y}\right).$$

第五章 Unconstrained Optimization

5.1 Unconstrained minimization problems

(Taken from Chapter 9.1 of [18])

In this chapter we discuss methods for solving the unconstrained optimization problem

$$\min_{\mathbf{x}} f(\mathbf{x}), \quad (5.1)$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex and twice continuously differentiable (which implies that $\text{dom } f$ is open). We will assume that the problem is solvable, *i.e.*, there exists an optimal point \mathbf{x}^* . (More precisely, the assumptions later in the chapter will imply that \mathbf{x}^* exists and is unique.) We denote the optimal value, $\inf_{\mathbf{x}} f(\mathbf{x}) = f(\mathbf{x}^*)$, as p^* . Since f is differentiable and convex, a necessary and sufficient condition for a point \mathbf{x}^* to be optimal is

$$\nabla f(\mathbf{x}^*) = \mathbf{0}. \quad (5.2)$$

Thus, solving the unconstrained minimization problem (5.1) is the same as finding a solution of (5.2), which is a set of n equations in the n variables x_1, \dots, x_n . In a few special cases, we can find a solution to the problem (5.1) by analytically solving the optimality equation (5.2), but usually the problem must be solved by an iterative algorithm. By this we mean an algorithm that computes a sequence of points $\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots \in \text{dom } f$ with $f(\mathbf{x}^{(k)}) \rightarrow p^*$ as $k \rightarrow \infty$. Such a sequence of points is called a minimizing sequence for the problem (5.1). The algorithm is terminated when $f(\mathbf{x}^{(k)}) - p^* \leq \epsilon$, where $\epsilon > 0$ is some specified tolerance.

Initial point and sublevel set

The methods described in this chapter require a suitable starting point $\mathbf{x}^{(0)}$. The starting point must lie in $\text{dom } f$, and in addition the sublevel set

$$S = \{\mathbf{x} \in \text{dom } f \mid f(\mathbf{x}) \leq f(\mathbf{x}^{(0)})\} \quad (5.3)$$

must be closed. This condition is satisfied for all $\mathbf{x}^{(0)} \in \text{dom } f$ if the function f is closed, *i.e.*, all its sublevel sets are closed (see Section 2.11.3.3). Continuous functions with $\text{dom } f = \mathbb{R}^n$ are closed, so if $\text{dom } f = \mathbb{R}^n$, the initial sublevel set condition is satisfied by any $\mathbf{x}^{(0)}$. Another important class of closed functions are continuous functions with open domains, for which $f(\mathbf{x})$ tends to infinity as \mathbf{x} approaches the boundary of $\text{dom } f$.

5.1.1 Examples

Quadratic minimization and least-squares

The general convex quadratic minimization problem has the form

$$\min_{\mathbf{x}} (1/2) \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r, \quad (5.4)$$

where $\mathbf{P} \in \mathbb{S}_+^n$, $\mathbf{q} \in \mathbb{R}^n$, and $r \in \mathbb{R}$. This problem can be solved via the optimality conditions, $\mathbf{P}\mathbf{x}^* + \mathbf{q} = \mathbf{0}$, which is a set of linear equations. When $\mathbf{P} \succ \mathbf{0}$, there is a unique solution, $\mathbf{x}^* = -\mathbf{P}^{-1}\mathbf{q}$. In the more general case when \mathbf{P} is not positive definite, any solution of $\mathbf{P}\mathbf{x}^* = -\mathbf{q}$ is optimal for (5.4); if $\mathbf{P}\mathbf{x}^* = -\mathbf{q}$ does not have a solution, then the problem (5.4) is unbounded below (as an exercise). Our ability to analytically solve the quadratic minimization problem (5.4) is the basis for Newton's method, a powerful method for unconstrained minimization described in Section 5.3. One special case of the quadratic minimization problem that arises very frequently is the least-squares problem.

Unconstrained geometric programming

As a second example, we consider an unconstrained geometric program in convex form,

$$\min_{\mathbf{x}} f(\mathbf{x}) = \log \sum_{i=1}^m \exp(\mathbf{a}_i^T \mathbf{x} + b_i).$$

The optimality condition is

$$\nabla f(\mathbf{x}^*) = \frac{1}{\sum_{j=1}^m \exp(\mathbf{a}_j^T \mathbf{x}^* + b_j)} \sum_{i=1}^m \exp(\mathbf{a}_i^T \mathbf{x}^* + b_i) \mathbf{a}_i = \mathbf{0},$$

which in general has no analytical solution, so here we must resort to an iterative algorithm. For this problem, $\text{dom } f = \mathbb{R}^n$, so any point can be chosen as the initial point $\mathbf{x}^{(0)}$.

Analytic center of linear inequalities

We consider the optimization problem

$$\min_{\mathbf{x}} f(\mathbf{x}) = - \sum_{i=1}^m \log(b_i - \mathbf{a}_i^T \mathbf{x}), \quad (5.5)$$

where the domain of f is the open set $\text{dom } f = \{\mathbf{x} | \mathbf{a}_i^T \mathbf{x} < b_i, i = 1, \dots, m\}$. The objective function f in this problem is called the *logarithmic barrier* for the inequalities $\mathbf{a}_i^T \mathbf{x} \leq b_i$. The solution of (5.5), if it exists, is called the analytic center of the inequalities. The initial point $\mathbf{x}^{(0)}$ must satisfy the strict inequalities $\mathbf{a}_i^T \mathbf{x}^{(0)} < b_i, i = 1, \dots, m$. Since f is closed, the sublevel set S for any such point is closed.

Analytic center of a linear matrix inequality

A closely related problem is

$$\min_{\mathbf{x}} f(\mathbf{x}) = \log \det F(\mathbf{x})^{-1}, \quad (5.6)$$

where $F : \mathbb{R}^n \rightarrow \mathbb{S}^p$ is affine, i.e.,

$$F(\mathbf{x}) = \mathbf{F}_0 + x_1 \mathbf{F}_1 + \cdots + x_n \mathbf{F}_n,$$

with $\mathbf{F}_i \in \mathbb{S}^p$. Here the domain of f is $\text{dom } f = \{\mathbf{x} | F(\mathbf{x}) \succ \mathbf{0}\}$. The objective function f is called the logarithmic barrier for the linear matrix inequality $F(\mathbf{x}) \succ \mathbf{0}$, and the solution (if it exists) is called the analytic center of the linear matrix inequality. The initial point $\mathbf{x}^{(0)}$ must satisfy the strict linear matrix inequality $F(\mathbf{x}^{(0)}) \succ \mathbf{0}$. As in the previous example, the sublevel set of any such point will be closed, since f is closed.

5.1.2 Strong convexity and implications

In much of this chapter we assume that the objective function is strongly convex on S , which means that there exists an $m > 0$ such that

$$\nabla^2 f(\mathbf{x}) \succeq m\mathbf{I}. \quad (5.7)$$

for all $\mathbf{x} \in S$. Strong convexity has several interesting consequences. For $\mathbf{x}, \mathbf{y} \in S$ we have

$$f(\mathbf{y}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) + \frac{1}{2}(\mathbf{y} - \mathbf{x})^T \nabla^2 f(\mathbf{z})(\mathbf{y} - \mathbf{x})$$

for some \mathbf{z} on the line segment $[\mathbf{x}, \mathbf{y}]$. By the strong convexity assumption (5.7), the last term on the righthand side is at least $(m/2)\|\mathbf{y} - \mathbf{x}\|_2^2$, so we have the inequality

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) + \frac{m}{2}\|\mathbf{y} - \mathbf{x}\|_2^2, \quad (5.8)$$

for all \mathbf{x} and \mathbf{y} in S . When $m = 0$, we recover the basic inequality characterizing convexity; for $m > 0$ we obtain a better lower bound on $f(\mathbf{y})$ than follows from convexity alone.

We will first show that the inequality (5.8) can be used to bound $f(\mathbf{x}) - p^*$, which is the suboptimality of the point \mathbf{x} , in terms of $\|\nabla f(\mathbf{x})\|_2$. The righthand side of (5.8) is a convex quadratic function of \mathbf{y} (for fixed \mathbf{x}). Setting the gradient with respect to \mathbf{y} equal to zero, we find that $\tilde{\mathbf{y}} = \mathbf{x} - (1/m)\nabla f(\mathbf{x})$ minimizes the righthand side. Therefore we have

$$\begin{aligned} f(\mathbf{y}) &\geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) + \frac{m}{2}\|\mathbf{y} - \mathbf{x}\|_2^2 \\ &\geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\tilde{\mathbf{y}} - \mathbf{x}) + \frac{m}{2}\|\tilde{\mathbf{y}} - \mathbf{x}\|_2^2 \\ &= f(\mathbf{x}) - \frac{1}{2m}\|\nabla f(\mathbf{x})\|_2^2. \end{aligned}$$

Since this holds for any $\mathbf{y} \in S$, we have

$$p^* \geq f(\mathbf{x}) - \frac{1}{2m} \|\nabla f(\mathbf{x})\|_2^2. \quad (5.9)$$

This inequality shows that if the gradient is small at a point, then the point is nearly optimal. The inequality (5.9) can also be interpreted as a condition for suboptimality which generalizes the optimality condition (5.2):

$$\|\nabla f(\mathbf{x})\|_2 \leq (2m\epsilon)^{1/2} \Rightarrow f(\mathbf{x}) - p^* \leq \epsilon \quad (5.10)$$

We can also derive a bound on $\|\mathbf{x} - \mathbf{x}^*\|_2$, the distance between \mathbf{x} and any optimal point \mathbf{x}^* , in terms of $\|\nabla f(\mathbf{x})\|_2$.

$$\|\mathbf{x} - \mathbf{x}^*\|_2 \leq \|\nabla f(\mathbf{x})\|_2. \quad (5.11)$$

To see this, we apply (5.8) with $\mathbf{y} = \mathbf{x}^*$ to obtain

$$\begin{aligned} p^* &= f(\mathbf{x}^*) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T (\mathbf{x}^* - \mathbf{x}) + \frac{m}{2} \|\mathbf{x}^* - \mathbf{x}\|_2^2 \\ &\geq f(\mathbf{x}) - \|\nabla f(\mathbf{x})\|_2 \|\mathbf{x}^* - \mathbf{x}\|_2 + \frac{m}{2} \|\mathbf{x}^* - \mathbf{x}\|_2^2, \end{aligned}$$

where we use the Cauchy-Schwarz inequality in the second inequality. Since $p^* \leq f(\mathbf{x})$, we must have

$$-\|\nabla f(\mathbf{x})\|_2 \|\mathbf{x}^* - \mathbf{x}\|_2 + \frac{m}{2} \|\mathbf{x}^* - \mathbf{x}\|_2^2 \leq 0,$$

from which (5.11) follows. One consequence of (5.11) is that the optimal point \mathbf{x}^* is unique.

Upper bound on $\nabla^2 f(\mathbf{x})$

The inequality (5.8) implies that the sublevel sets contained in S are bounded, so in particular, S is bounded. Therefore the maximum eigenvalue of $\nabla^2 f(\mathbf{x})$, which is a continuous function of \mathbf{x} on S , is bounded above on S , *i.e.*, there exists a constant M such that

$$\nabla^2 f(\mathbf{x}) \preceq M\mathbf{I}, \quad (5.12)$$

for all $\mathbf{x} \in S$. This upper bound on the Hessian implies for any $\mathbf{x}, \mathbf{y} \in S$,

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \nabla f(\mathbf{x})^T (\mathbf{y} - \mathbf{x}) + \frac{M}{2} \|\mathbf{y} - \mathbf{x}\|_2^2, \quad (5.13)$$

which is analogous to (5.8). Minimizing each side over \mathbf{y} yields

$$p^* \leq f(\mathbf{x}) - \frac{1}{2M} \|\nabla f(\mathbf{x})\|_2^2, \quad (5.14)$$

the counterpart of (5.9).

Condition number of sublevel sets

From the strong convexity inequality (5.7) and the inequality (5.12), we have

$$m\mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq M\mathbf{I}, \quad (5.15)$$

for all $\mathbf{x} \in S$. The ratio $\kappa = M/m$ is thus an upper bound on the condition number of the matrix $\nabla^2 f(\mathbf{x})$, *i.e.*, the ratio of its largest eigenvalue to its smallest eigenvalue. We can also give a geometric interpretation of (5.15) in terms of the sublevel sets of f .

We define the *width* of a convex set $C \subseteq \mathbb{R}^n$, in the direction \mathbf{q} , where $\|\mathbf{q}\|_2 = 1$, as

$$W(C, \mathbf{q}) = \sup_{\mathbf{z} \in C} \mathbf{q}^T \mathbf{z} - \inf_{\mathbf{z} \in C} \mathbf{q}^T \mathbf{z}.$$

The *minimum width* and maximum width of C are given by

$$W_{\min} = \inf_{\|\mathbf{q}\|_2=1} W(C, \mathbf{q}), \quad W_{\max} = \sup_{\|\mathbf{q}\|_2=1} W(C, \mathbf{q}).$$

The *condition number* of the convex set C is defined as

$$\mathbf{cond}(C) = \frac{W_{\max}^2}{W_{\min}^2},$$

i.e., the square of the ratio of its maximum width to its minimum width. The condition number of C gives a measure of its *anisotropy* or *eccentricity*. If the condition number of a set C is small (say, near one) it means that the set has approximately the same width in all directions, *i.e.*, it is nearly spherical. If the condition number is large, it means that the set is far wider in some directions than in others.

Example 9.1 *Condition number of an ellipsoid.* Let \mathcal{E} be the ellipsoid

$$\mathcal{E} = \{\mathbf{x} | (\mathbf{x} - \mathbf{x}_0)^T \mathbf{A}^{-1} (\mathbf{x} - \mathbf{x}_0) \leq 1\},$$

where $\mathbf{A} \in \mathbb{S}_{++}^n$. The width of \mathcal{E} in the direction \mathbf{q} is

$$\sup_{\mathbf{z} \in \mathcal{E}} \mathbf{q}^T \mathbf{z} - \inf_{\mathbf{z} \in \mathcal{E}} \mathbf{q}^T \mathbf{z} = (\|\mathbf{A}^{1/2} \mathbf{q}\|_2 + \mathbf{q}^T \mathbf{x}_0) - (-\|\mathbf{A}^{1/2} \mathbf{q}\|_2 + \mathbf{q}^T \mathbf{x}_0) \quad (5.16)$$

$$= 2\|\mathbf{A}^{1/2} \mathbf{q}\|_2. \quad (5.17)$$

It follows that its minimum and maximum width are

$$W_{\min} = 2\lambda_{\min}(\mathbf{A})^{1/2}, \quad W_{\max} = 2\lambda_{\max}(\mathbf{A})^{1/2}.$$

and its condition number is

$$\mathbf{cond}(\mathcal{E}) = \frac{\lambda_{\max}(\mathbf{A})}{\lambda_{\min}(\mathbf{A})} = \kappa(\mathbf{A}),$$

where $\kappa(\mathbf{A})$ denotes the condition number of the matrix \mathbf{A} , *i.e.*, the ratio of its maximum singular value to its minimum singular value. Thus the condition number of the ellipsoid \mathcal{E} is the same as the condition number of the matrix \mathbf{A} that defines it.

Now suppose f satisfies $m\mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq M\mathbf{I}$ for all $\mathbf{x} \in S$. We will derive a bound on the condition number of the α -sublevel $C_\alpha = \{\mathbf{x} | f(\mathbf{x}) \leq \alpha\}$, where $p^* < \alpha \leq f(\mathbf{x}^{(0)})$. Applying (5.13) and (5.8) with $\mathbf{x} = \mathbf{x}^*$, we have

$$p^* + (M/2)\|\mathbf{y} - \mathbf{x}^*\|_2^2 \geq f(\mathbf{y}) \geq p^* + (m/2)\|\mathbf{y} - \mathbf{x}^*\|_2^2.$$

This implies that $B_{inner} \subseteq C_\alpha \subseteq B_{outer}$ where

$$B_{inner} = \{\mathbf{y} | \|\mathbf{y} - \mathbf{x}^*\|_2 \leq (2(\alpha - p^*)/M)^{1/2}\}, \quad (5.18)$$

$$B_{outer} = \{\mathbf{y} | \|\mathbf{y} - \mathbf{x}^*\|_2 \leq (2(\alpha - p^*)/m)^{1/2}\}. \quad (5.19)$$

In other words, the α -sublevel set contains B_{inner} , and is contained in B_{outer} , which are balls with radii

$$(2(\alpha p^*)/M)^{1/2}, \quad (2(\alpha p^*)/m)^{1/2},$$

respectively. The ratio of the radii squared gives an upper bound on the condition number of C_α :

$$\text{cond}(C_\alpha) \leq \frac{M}{m}.$$

We can also give a geometric interpretation of the condition number $\kappa(\nabla^2(\mathbf{x}^*))$ of the Hessian at the optimum. From the Taylor series expansion of f around \mathbf{x}^* ,

$$f(\mathbf{y}) \approx p^* + \frac{1}{2}(\mathbf{y} - \mathbf{x}^*)^T \nabla^2 f(\mathbf{x}^*)(\mathbf{y} - \mathbf{x}^*),$$

we see that, for α close to p^* ,

$$C_\alpha \approx \{\mathbf{y} | (\mathbf{y} - \mathbf{x}^*)^T \nabla^2 f(\mathbf{x}^*)(\mathbf{y} - \mathbf{x}^*) \leq 2(\alpha - p^*)\},$$

i.e., the sublevel set is well approximated by an ellipsoid with center \mathbf{x}^* . Therefore

$$\lim_{\alpha \rightarrow p^*} \text{cond}(C_\alpha) = \kappa(\nabla^2 f(\mathbf{x}^*)).$$

We will see that the condition number of the sublevel sets of f (which is bounded by M/m) has a strong effect on the efficiency of some common methods for unconstrained minimization.

The strong convexity constants

It must be kept in mind that the constants m and M are known only in rare cases, so the inequality (5.10) cannot be used as a practical stopping criterion. It can be considered

a conceptual stopping criterion; it shows that if the gradient of f at \mathbf{x} is small enough, then the difference between $f(\mathbf{x})$ and p^* is small. If we terminate an algorithm when $\|\nabla f(\mathbf{x}^{(k)})\|_2 \leq \eta$, where η is chosen small enough to be (very likely) smaller than $(m\epsilon)^{1/2}$, then we have $f(\mathbf{x}^{(k)}) - p^* \leq \epsilon$ (very likely).

In the following sections we give convergence proofs for algorithms, which include bounds on the number of iterations required before $f(\mathbf{x}^{(k)}) - p^* \leq \epsilon$, where ϵ is some positive tolerance. Many of these bounds involve the (usually unknown) constants m and M , so the same comments apply. These results are at least conceptually useful; they establish that the algorithm converges, even if the bound on the number of iterations required to reach a given accuracy depends on constants that are unknown.

We will encounter one important exception to this situation.

5.2 Descent methods

(Taken from Chapter 9.2 of [18])

The algorithms described in this chapter produce a minimizing sequence $\mathbf{x}^{(k)}, k = 1, \dots$, where

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + t^{(k)} \Delta \mathbf{x}^{(k)}$$

and $t^{(k)} > 0$ (except when $\mathbf{x}^{(k)}$ is optimal). Here the concatenated symbols Δ and \mathbf{x} that form $\Delta \mathbf{x}$ are to be read as a single entity, a vector in \mathbb{R}^n called the step or search direction (even though it need not have unit norm), and $k = 0, 1, \dots$ denotes the iteration number. The scalar $t^{(k)} \geq 0$ is called the step size or step length at iteration k (even though it is not equal to $\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|$ unless $\|\Delta \mathbf{x}^{(k)}\| = 1$). The terms ‘search step’ and ‘scale factor’ are more accurate, but ‘search direction’ and ‘step length’ are the ones widely used. When we focus on one iteration of an algorithm, we sometimes drop the superscripts and use the lighter notation $\mathbf{x}^+ = \mathbf{x} + t\Delta \mathbf{x}$, or $\mathbf{x} := \mathbf{x} + t\Delta \mathbf{x}$, in place of $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + t^{(k)} \Delta \mathbf{x}^{(k)}$.

All the methods we study are *descent methods*, which means that

$$f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)}),$$

except when $\mathbf{x}^{(k)}$ is optimal. This implies that for all k we have $\mathbf{x}^{(k)} \in S$, the initial sublevel set, and in particular we have $\mathbf{x}^{(k)} \in \text{dom } f$. From convexity we know that $\nabla f(\mathbf{x}^{(k)})^T (\mathbf{y} - \mathbf{x}^{(k)}) \geq 0$ implies $f(\mathbf{y}) \geq f(\mathbf{x}^{(k)})$, so the search direction in a descent method must satisfy

$$\nabla f(\mathbf{x}^{(k)})^T \Delta \mathbf{x}^{(k)} < 0,$$

i.e., it must make an acute angle with the negative gradient. We call such a direction a *descent direction* (for f , at $\mathbf{x}^{(k)}$).

The outline of a general descent method is as follows. It alternates between two steps: determining a descent direction $\Delta\mathbf{x}$, and the selection of a step size t .

Algorithm 9.1 *General descent method.*

given a starting point $\mathbf{x} \in \text{dom } f$.

repeat

1. Determine a descent direction $\Delta\mathbf{x}$.
2. Line search. Choose a step size $t > 0$.
3. Update. $\mathbf{x} := \mathbf{x} + t\Delta\mathbf{x}$.

until stopping criterion is satisfied.

The second step is called the line search since selection of the step size t determines where along the line $\{\mathbf{x} + t\Delta\mathbf{x} | t \in \mathbb{R}_+\}$ the next iterate will be. (A more accurate term might be ray search.) A practical descent method has the same general structure, but might be organized differently. For example, the stopping criterion is often checked while, or immediately after, the descent direction $\Delta\mathbf{x}$ is computed. The stopping criterion is often of the form $\|\nabla f(\mathbf{x})\|_2 \leq \eta$, where η is small and positive, as suggested by the suboptimality condition (5.9).

Exact line search

One line search method sometimes used in practice is exact line search, in which t is chosen to minimize f along the ray $\{\mathbf{x} + t\Delta\mathbf{x} | t \leq 0\}$:

$$t = \underset{s \geq 0}{\operatorname{argmin}} f(\mathbf{x} + s\Delta\mathbf{x}). \quad (5.20)$$

An exact line search is used when the cost of the minimization problem with one variable, required in (5.20), is low compared to the cost of computing the search direction itself. In some special cases the minimizer along the ray can be found analytically, and in others it can be computed efficiently. (This is discussed in Section 9.7.1 of [18].)

Backtracking line search

Most line searches used in practice are inexact: the step length is chosen to approximately minimize f along the ray $\{\mathbf{x} + t\Delta\mathbf{x} | t \geq 0\}$, or even to just reduce f ‘enough’. Many inexact line search methods have been proposed. One inexact line search method that is very simple and quite effective is called backtracking line search. It depends on two constants α, β with $0 < \alpha < 0.5, 0 < \beta < 1$.

Algorithm 9.2 *Backtracking line search.*

given a descent direction $\Delta\mathbf{x}$ for f at $\mathbf{x} \in \text{dom } f$, $\alpha \in (0, 0.5)$, $\beta \in (0, 1)$.

$t := 1$.

while $f(\mathbf{x} + t\Delta\mathbf{x}) > f(\mathbf{x}) + \alpha t \nabla f(\mathbf{x})^T \Delta\mathbf{x}$, $t := \beta t$.

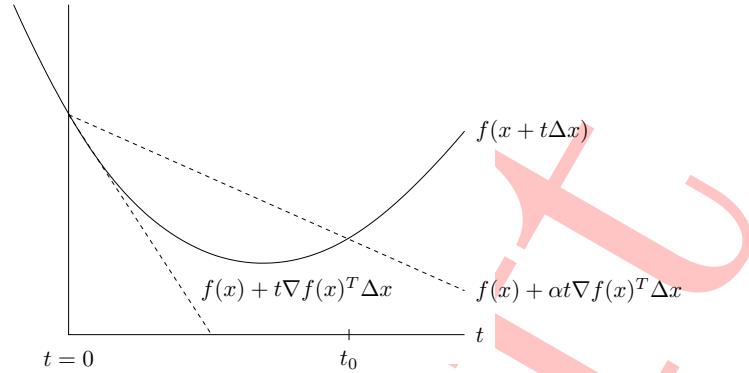


图 5.1: *Backtracking line search*. The curve shows f , restricted to the line over which we search. The lower dashed line shows the linear extrapolation of f , and the upper dashed line has a slope a factor of α smaller. The backtracking condition is that f lies below the upper dashed line, *i.e.*, $0 \leq t \leq t_0$.

The line search is called backtracking because it starts with unit step size and then reduces it by the factor β until the stopping condition $f(\mathbf{x} + t\Delta\mathbf{x}) \leq f(\mathbf{x}) + \alpha t \nabla f(\mathbf{x})^T \Delta\mathbf{x}$ holds. Since $\Delta\mathbf{x}$ is a descent direction, we have $\nabla f(\mathbf{x})^T \Delta\mathbf{x} < 0$, so for small enough t we have

$$f(\mathbf{x} + t\Delta\mathbf{x}) \approx f(\mathbf{x}) + t \nabla f(\mathbf{x})^T \Delta\mathbf{x} < f(\mathbf{x}) + \alpha t \nabla f(\mathbf{x})^T \Delta\mathbf{x},$$

which shows that the backtracking line search eventually terminates. The constant α can be interpreted as the fraction of the decrease in f predicted by linear extrapolation that we will accept. (The reason for requiring α to be smaller than 0.5 will become clear later.)

The backtracking condition is illustrated in Figure 5.1. This figure suggests, and it can be shown, that the backtracking exit inequality $f(\mathbf{x} + t\Delta\mathbf{x}) \leq f(\mathbf{x}) + \alpha t \nabla f(\mathbf{x})^T \Delta\mathbf{x}$ holds for $t \geq 0$ in an interval $(0, t_0]$. It follows that the backtracking line search stops with a step length t that satisfies

$$t = 1, \text{ or } t \in (\beta t_0, t_0].$$

The first case occurs when the step length $t = 1$ satisfies the backtracking condition, *i.e.*, $1 \leq t_0$. In particular, we can say that the step length obtained by backtracking line

search satisfies

$$t \geq \min\{1, \beta t_0\}.$$

When $\text{dom } f$ is not all of \mathbb{R}^n , the condition $f(\mathbf{x} + t\Delta\mathbf{x}) \leq f(\mathbf{x}) + \alpha t \nabla f(\mathbf{x})^T \Delta\mathbf{x}$ in the backtracking line search must be interpreted carefully. By our convention that f is infinite outside its domain, the inequality implies that $\mathbf{x} + t\Delta\mathbf{x} \in \text{dom } f$. In a practical implementation, we first multiply t by β until $\mathbf{x} + t\Delta\mathbf{x} \in \text{dom } f$; then we start to check whether the inequality $f(\mathbf{x} + t\Delta\mathbf{x}) \leq f(\mathbf{x}) + \alpha t \nabla f(\mathbf{x})^T \Delta\mathbf{x}$ holds.

The parameter α is typically chosen between 0.01 and 0.3, meaning that we accept a decrease in f between 1% and 30% of the prediction based on the linear extrapolation. The parameter β is often chosen to be between 0.1 (which corresponds to a very crude search) and 0.8 (which corresponds to a less crude search).

5.2.1 Gradient descent method

(Taken from Chapter 9.3 of [18])

A natural choice for the search direction is the negative gradient $\Delta\mathbf{x} = -\nabla f(\mathbf{x})$. The resulting algorithm is called the *gradient algorithm* or *gradient descent method*.

Algorithm 9.3 *Gradient descent method.*

given a starting point $\mathbf{x} \in \text{dom } f$.

repeat

1. $\Delta\mathbf{x} := -\nabla f(\mathbf{x})$.
2. Line search. Choose step size t via exact or backtracking line search.
3. Update. $\mathbf{x} := \mathbf{x} + t\Delta\mathbf{x}$.

until stopping criterion is satisfied.

The stopping criterion is usually of the form $\|\nabla f(\mathbf{x})\|_2 \leq \eta$, where η is small and positive. In most implementations, this condition is checked after step 1, rather than after the update.

5.2.1.1 Convergence analysis

In this section we present a simple convergence analysis for the gradient method, using the lighter notation $\mathbf{x}^+ = \mathbf{x} + t\Delta\mathbf{x}$ for $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + t^{(k)}\Delta\mathbf{x}^{(k)}$, where $\Delta\mathbf{x} = -\nabla f(\mathbf{x})$. We assume f is strongly convex on S , so there are positive constants m and M such that $m\mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq M\mathbf{I}$ for all $\mathbf{x} \in S$. Define the function $\tilde{f} : \mathbb{R} \rightarrow \mathbb{R}$ by $\tilde{f}(t) = f(\mathbf{x} - t\nabla f(\mathbf{x}))$, i.e., f as a function of the step length t in the negative gradient direction. In the following

discussion we will only consider t for which $\mathbf{x} - t\nabla f(\mathbf{x}) \in S$. From the inequality (5.13), with $\mathbf{y} = \mathbf{x} - t\nabla f(\mathbf{x})$, we obtain a quadratic upper bound on \tilde{f} :

$$\tilde{f}(t) \leq f(\mathbf{x}) - t\|\nabla f(\mathbf{x})\|_2^2 + \frac{Mt^2}{2}\|\nabla f(\mathbf{x})\|_2^2. \quad (5.21)$$

Analysis for exact line search

We now assume that an exact line search is used, and minimize over t both sides of the inequality (5.21). On the lefthand side we get $\tilde{f}(t_{exact})$, where t_{exact} is the step length that minimizes \tilde{f} . The righthand side is a simple quadratic, which is minimized by $t = 1/M$, and has minimum value $f(x) - (1/(2M))\|\nabla f(x)\|_2^2$.

Therefore we have

$$f(\mathbf{x}^+) = \tilde{f}(t_{exact}) \leq f(\mathbf{x}) - \frac{1}{2M}\|\nabla f(\mathbf{x})\|_2^2.$$

Subtracting p^* from both sides, we get

$$f(\mathbf{x}^+) - p^* \leq f(\mathbf{x}) - p^* - \frac{1}{2M}\|\nabla f(\mathbf{x})\|_2^2.$$

We combine this with $\|\nabla f(\mathbf{x})\|_2^2 \leq 2m(f(\mathbf{x}) - p^*)$ (which follows from (5.9)) to conclude

$$f(\mathbf{x}^+) - p^* \leq (1 - m/M)(f(\mathbf{x}) - p^*).$$

Applying this inequality recursively, we find that

$$f(\mathbf{x}^{(k)}) - p^* \leq c^k(f(\mathbf{x}^{(0)}) - p^*), \quad (5.22)$$

where $c = 1 - m/M < 1$, which shows that $f(\mathbf{x}^{(k)})$ converges to p^* as $k \rightarrow \infty$. In particular, we must have $f(\mathbf{x}^{(k)}) - p^* \leq \epsilon$ after at most

$$\frac{\log((f(\mathbf{x}^{(0)}) - p^*)/\epsilon)}{\log(1/c)} \quad (5.23)$$

iterations of the gradient method with exact line search.

This bound on the number of iterations required, even though crude, can give some insight into the gradient method. The numerator,

$$\log((f(\mathbf{x}^{(0)}) - p^*)/\epsilon)$$

can be interpreted as the log of the ratio of the initial suboptimality (*i.e.*, gap between $f(\mathbf{x}^{(0)})$ and p^*), to the final suboptimality (*i.e.*, less than ϵ). This term suggests that the number of iterations depends on how good the initial point is, and what the final required accuracy is.

The denominator appearing in the bound (5.23), $\log(1/c)$, is a function of M/m , which we have seen is a bound on the condition number of $\nabla^2 f(\mathbf{x})$ over S , or the condition number of the sublevel sets $\{\mathbf{z} | f(\mathbf{z}) \leq \alpha\}$. For large condition number bound M/m , we have

$$\log(1/c) = -\log(1 - m/M) \approx m/M.$$

So our bound on the number of iterations required increases approximately linearly with increasing M/m .

We will see that the gradient method does in fact require a large number of iterations when the Hessian of f , near \mathbf{x}^* , has a large condition number. Conversely, when the sublevel sets of f are relatively isotropic, so that the condition number bound M/m can be chosen to be relatively small, the bound (5.22) shows that convergence is rapid, since c is small, or at least not too close to one.

The bound (5.22) shows that the error $f(\mathbf{x}^{(k)}) - p^*$ converges to zero at least as fast as a geometric series. In the context of iterative numerical methods, this is called *linear convergence*, since the error lies below a line on a log-linear plot of error versus iteration number.

Analysis for backtracking line search

Now we consider the case where a backtracking line search is used in the gradient descent method. We will show that the backtracking exit condition,

$$\tilde{f}(t) \leq f(\mathbf{x}) - \alpha t \|\nabla f(\mathbf{x})\|_2^2,$$

is satisfied whenever $0 \leq t \leq 1/M$. First note that

$$0 \leq t \leq 1/M \implies -t + \frac{Mt^2}{2} \leq -t/2,$$

(which follows from convexity of $-t + Mt^2/2$). Using this result and the bound (5.21), we have, for $0 \leq t \leq 1/M$,

$$\begin{aligned} \tilde{f}(t) &\leq f(\mathbf{x}) - t \|\nabla f(\mathbf{x})\|_2^2 + \frac{Mt^2}{2} \|\nabla f(\mathbf{x})\|_2^2 \\ &\leq f(\mathbf{x}) - (t/2) \|\nabla f(\mathbf{x})\|_2^2 \\ &\leq f(\mathbf{x}) - \alpha t \|\nabla f(\mathbf{x})\|_2^2 \end{aligned}$$

since $\alpha < 1/2$. Therefore the backtracking line search terminates either with $t = 1$ or with a value $t \geq \beta/M$. This provides a lower bound on the decrease in the objective function. In the first case we have

$$f(\mathbf{x}^+) \leq f(\mathbf{x}) - \alpha \|\nabla f(\mathbf{x})\|_2^2,$$

and in the second case we have

$$f(\mathbf{x}^+) \leq f(\mathbf{x}) - (\beta\alpha/M)\|\nabla f(\mathbf{x})\|_2^2.$$

Putting these together, we always have

$$f(\mathbf{x}^+) \leq f(\mathbf{x}) - \min\{\alpha, (\beta\alpha/M)\}\|\nabla f(\mathbf{x})\|_2^2.$$

Now we can proceed exactly as in the case of exact line search. We subtract p^* from both sides to get

$$f(\mathbf{x}^+) - p^* \leq f(\mathbf{x}) - p^* - \min\{\alpha, (\beta\alpha/M)\}\|\nabla f(\mathbf{x})\|_2^2,$$

and combine this with $\|\nabla f(\mathbf{x})\|_2^2 \geq 2m(f(\mathbf{x}) - p^*)$ to obtain

$$f(\mathbf{x}^+) - p^* \leq (1 - \min\{2m\alpha, 2\beta\alpha m/M\})(f(\mathbf{x}) - p^*).$$

From this we conclude

$$f(\mathbf{x}^{(k)}) - p^* \leq c^k(f(\mathbf{x}^{(0)}) - p^*),$$

where

$$c = 1 - \min\{2m\alpha, 2\beta\alpha m/M\} < 1.$$

In particular, $f(\mathbf{x}^{(k)})$ converges to p^* at least as fast as a geometric series with an exponent that depends (at least in part) on the condition number bound M/m . In the terminology of iterative methods, the convergence is at least linear.

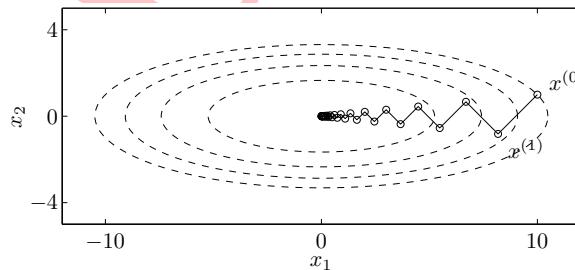


图 5.2: Some contour lines of the function $f(\mathbf{x}) = (1/2)(\mathbf{x}_1^2 + 10\mathbf{x}_2^2)$. The condition number of the sublevel sets, which are ellipsoids, is exactly 10. The figure shows the iterates of the gradient method with exact line search, started at $\mathbf{x}^{(0)} = (10, 1)^T$.

5.2.1.2 Examples

A quadratic problem in \mathbb{R}^2 Our first example is very simple. We consider the quadratic objective function on \mathbb{R}^2

$$f(\mathbf{x}) = \frac{1}{2}(\mathbf{x}_1^2 + \gamma\mathbf{x}_2^2),$$

where $\gamma > 0$. Clearly, the optimal point is $\mathbf{x}^* = \mathbf{0}$, and the optimal value is 0. The Hessian of f is constant, and has eigenvalues 1 and γ , so the condition numbers of the sublevel sets of f are all exactly

$$\frac{\max\{1, \gamma\}}{\min\{1, \gamma\}} = \max\{\gamma, 1/\gamma\}.$$

The tightest choices for the strong convexity constants m and M are $m = \min\{1, \gamma\}$, $M = \max\{1, \gamma\}$. We apply the gradient descent method with exact line search, starting at the point $\mathbf{x}^{(0)} = (\gamma, 1)^T$. In this case we can derive the following closed-form expressions for the iterates $\mathbf{x}^{(k)}$ and their function values (as an exercise):

$$x_1^{(k)} = \gamma \left(\frac{\gamma - 1}{\gamma + 1} \right)^k, \quad x_2^{(k)} = \gamma \left(-\frac{\gamma - 1}{\gamma + 1} \right)^k,$$

and

$$f(\mathbf{x}^{(k)}) = \frac{\gamma(\gamma + 1)}{2} \gamma \left(\frac{\gamma - 1}{\gamma + 1} \right)^{2k} = \gamma \left(\frac{\gamma - 1}{\gamma + 1} \right)^{2k} f(\mathbf{x}^{(0)}).$$

This is illustrated in Figure 5.2, for $\gamma = 10$. For this simple example, convergence is exactly linear, *i.e.*, the error is exactly a geometric series, reduced by the factor $|(\gamma - 1)/(\gamma + 1)|^2$ at each iteration. For $\gamma = 1$, the exact solution is found in one iteration; for γ not far from one (say, between $1/3$ and 3) convergence is rapid. The convergence is very slow for $\gamma \gg 1$ or $\gamma \ll 1$. We can compare the convergence with the bound derived above in Section 5.2.1.1. Using the least conservative values $m = \min\{1, \gamma\}$ and $M = \max\{1, \gamma\}$, the bound (5.22) guarantees that the error in each iteration is reduced at least by the factor $c = 1 - m/M$. We have seen that the error is in fact reduced exactly by the factor

$$\left(\frac{1 - m/M}{1 + m/M} \right)^2$$

in each iteration. For small m/M , which corresponds to large condition number, the upper bound (5.23) implies that the number of iterations required to obtain a given level of accuracy grows at most like M/m . For this example, the exact number of iterations required grows approximately like $(M/m)/4$, *i.e.*, one quarter of the value of the bound. This shows that for this simple example, the bound on the number of iterations derived in our simple analysis is only about a factor of four conservative (using the least conservative

values for m and M). In particular, the convergence rate (as well as its upper bound) is very dependent on the condition number of the sublevel sets.

A nonquadratic problem in \mathbb{R}^2

We now consider a nonquadratic example in \mathbb{R}^2 , with

$$f(x_1, x_2) = e^{x_1+3x_2-0.1} + e^{x_1-3x_2-0.1} + e^{-x_1-0.1}. \quad (5.24)$$

We apply the gradient method with a backtracking line search, with $\alpha = 0.1$, $\beta = 0.7$. Figure 5.3 shows some level curves of f , and the iterates $\mathbf{x}^{(k)}$ generated by the gradient method (shown as small circles). The lines connecting successive iterates show the scaled steps,

$$\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} = -t^{(k)} \nabla f(\mathbf{x}^{(k)}).$$

Figure 5.4 shows the error $f(\mathbf{x}^{(k)}) - p^*$ versus iteration k . The plot reveals that the error converges to zero approximately as a geometric series, *i.e.*, the convergence is approximately linear. In this example, the error is reduced from about 10 to about 10^7 in 20 iterations, so the error is reduced by a factor of approximately $10^{8/20} \approx 0.4$ each iteration. This reasonably rapid convergence is predicted by our convergence analysis, since the sublevel sets of f are not too badly conditioned, which in turn means that M/m can be chosen as not too large.

To compare backtracking line search with an exact line search, we use the gradient method with an exact line search, on the same problem, and with the same starting point. The results are given in Figures 5.5 and 5.4. Here too the convergence is approximately linear, about twice as fast as the gradient method with backtracking line search. With exact line search, the error is reduced by about 10^{11} in 15 iterations, *i.e.*, a reduction by a factor of about $10^{11/15} \approx 0.2$ per iteration.

A problem in \mathbb{R}^{100}

We next consider a larger example, of the form

$$f(\mathbf{x}) = \mathbf{c}^T \mathbf{x} - \sum_{i=1}^m \log(b_i - \mathbf{a}_i^T \mathbf{x}), \quad (5.25)$$

with $m = 500$ terms and $n = 100$ variables.

The progress of the gradient method with backtracking line search, with parameters $\alpha = 0.1$, $\beta = 0.5$, is shown in Figure 5.6. In this example we see an initial approximately linear and fairly rapid convergence for about 20 iterations, followed by a slower linear convergence. Overall, the error is reduced by a factor of around 106 in around 175 iterations, which gives an average error reduction by a factor of around $10^{-6/175} \approx 0.92$

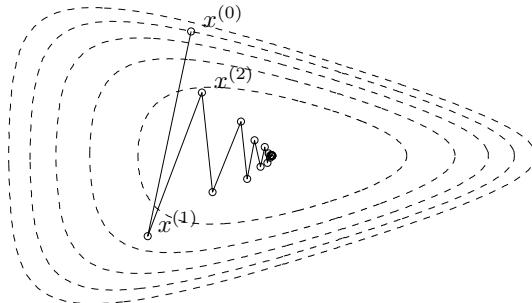


图 5.3: Iterates of the gradient method with backtracking line search, for the problem in \mathbb{R}^2 with objective f given in (5.24). The dashed curves are level curves of f , and the small circles are the iterates of the gradient method. The solid lines, which connect successive iterates, show the scaled steps $t^{(k)}\nabla f(\mathbf{x}^{(k)})$.

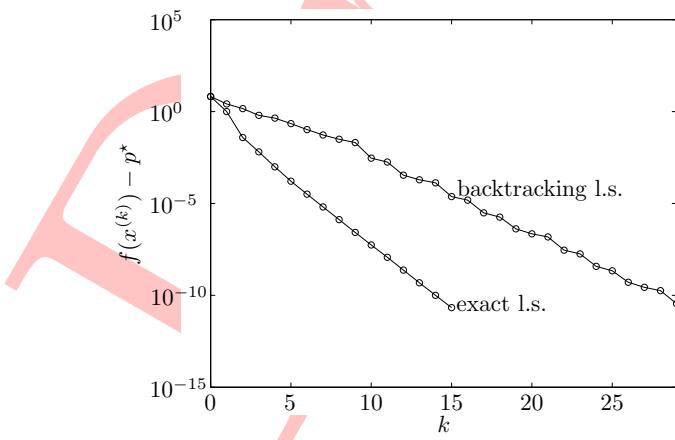


图 5.4: Error $f(\mathbf{x}^{(k)}) - p^*$ versus iteration k of the gradient method with backtracking and exact line search, for the problem in \mathbb{R}^2 with objective f given in (5.24). The plot shows nearly linear convergence, with the error reduced approximately by the factor 0.4 in each iteration of the gradient method with backtracking line search, and by the factor 0.2 in each iteration of the gradient method with exact line search.

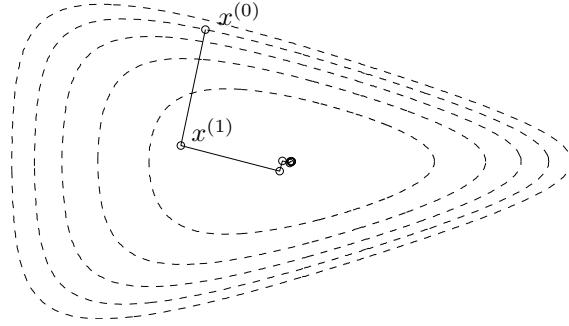


图 5.5: Iterates of the gradient method with exact line search for the problem in \mathbb{R}^2 with objective f given in (5.24).

per iteration. The initial convergence rate, for the first 20 iterations, is around a factor of 0.8 per iteration; the slower final convergence rate, after the first 20 iterations, is around a factor of 0.94 per iteration.

Figure 5.6 shows the convergence of the gradient method with exact line search. The convergence is again approximately linear, with an overall error reduction by approximately a factor $10^{-6/140} \approx 0.91$ per iteration. This is only a bit faster than the gradient method with backtracking line search. Finally, we examine the influence of the backtracking line search parameters α and β on the convergence rate, by determining the number of iterations required to obtain $f(x^{(k)}) - p^* \leq 10^5$. In the first experiment, we fix $\beta = 0.5$, and vary α from 0.05 to 0.5. The number of iterations required varies from about 80, for larger values of α , in the range 0.2 – 0.5, to about 170 for smaller values of α . This, and other experiments, suggest that the gradient method works better with fairly large α , in the range 0.2 – 0.5.

Similarly, we can study the effect of the choice of β by fixing $\alpha = 0.1$ and varying β from 0.05 to 0.95. Again the variation in the total number of iterations is not large, ranging from around 80 (when $\beta \approx 0.5$) to around 200 (for β small, or near 1). This experiment, and others, suggest that $\beta \approx 0.5$ is a good choice.

These experiments suggest that the effect of the backtracking parameters on the convergence is not large, no more than a factor of two or so.

Gradient method and condition number

Our last experiment will illustrate the importance of the condition number of $\nabla^2 f(\mathbf{x})$ (or the sublevel sets) on the rate of convergence of the gradient method. We start with the function given by (5.25), but replace the variable \mathbf{x} by $\mathbf{x} = \mathbf{T}\bar{\mathbf{x}}$, where

$$\mathbf{T} = \text{diag}(1, \gamma^{1/n}, \gamma^{2/n}, \dots, \gamma^{(n-1)/n}),$$

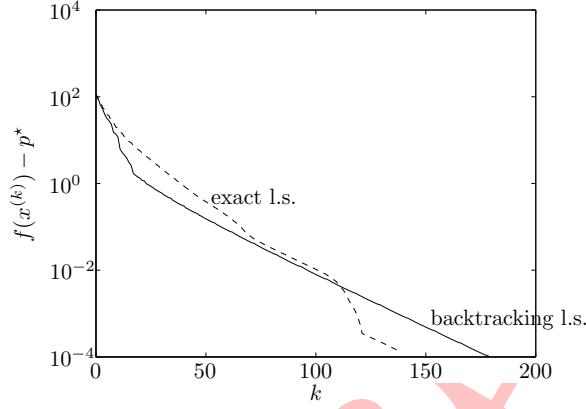


图 5.6: Error $f(\mathbf{x}^{(k)}) - p^*$ versus iteration k for the gradient method with backtracking and exact line search, for a problem in \mathbb{R}^{100} .

i.e., we minimize

$$\bar{f}(\bar{\mathbf{x}}) = \mathbf{c}^T \mathbf{T} \bar{\mathbf{x}} - \sum_{i=1}^m \log(b_i - \mathbf{a}_i^T \mathbf{T} \bar{\mathbf{x}}). \quad (5.26)$$

This gives us a family of optimization problems, indexed by γ , which affects the problem condition number. Figure 5.7 shows the number of iterations required to achieve $\bar{f}(\bar{\mathbf{x}}^{(k)}) - \bar{p}^* < 10^{-5}$ as a function of γ , using a backtracking line search with $\alpha = 0.3$ and $\beta = 0.7$. This plot shows that for diagonal scaling as small as 10 : 1 (i.e., $\gamma = 10$), the number of iterations grows to more than a thousand; for a diagonal scaling of 20 or more, the gradient method slows to essentially useless. The condition number of the Hessian $\nabla^2 \bar{f}(\bar{\mathbf{x}}^*)$ at the optimum is shown in Figure 5.8. For large and small γ , the condition number increases roughly as $\max\{\gamma^2, 1/\gamma^2\}$, in a very similar way as the number of iterations depends on γ . This shows again that the relation between conditioning and convergence speed is a real phenomenon, and not just an artifact of our analysis.

Conclusions

From the numerical examples shown, and others, we can make the conclusions summarized below.

The gradient method often exhibits approximately linear convergence, i.e., the error $f(\mathbf{x}^{(k)}) - p^*$ converges to zero approximately as a geometric series.

The choice of backtracking parameters α, β has a noticeable but not dramatic effect on the convergence. An exact line search sometimes improves the convergence of the gradient method, but the effect is not large (and probably not worth the trouble of implementing the exact line search).

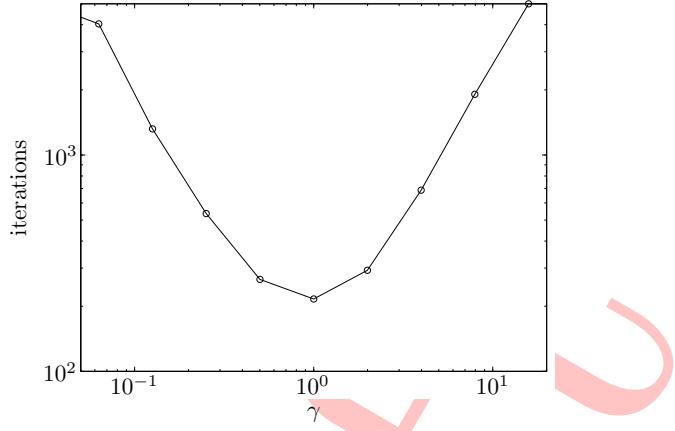


图 5.7: Number of iterations of the gradient method applied to problem (5.26). The vertical axis shows the number of iterations required to obtain $\bar{f}(\bar{\mathbf{x}}^{(k)}) - \bar{p}^* < 10^{-5}$. The horizontal axis shows γ , which is a parameter that controls the amount of diagonal scaling. We use a backtracking line search with $\alpha = 0.3, \beta = 0.7$.

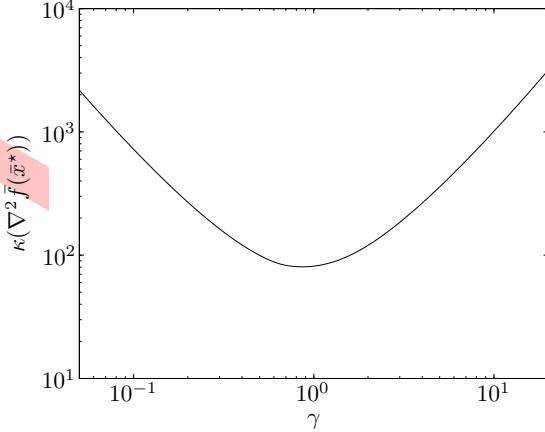


图 5.8: Condition number of the Hessian of the function at its minimum, as a function of γ . By comparing this plot with the one in Figure 5.7, we see that the condition number has a very strong influence on convergence rate.

The convergence rate depends greatly on the condition number of the Hessian, or the sublevel sets. Convergence can be very slow, even for problems that are moderately well conditioned (say, with condition number in the 100s). When the condition number is larger (say, 1000 or more) the gradient method is so slow that it is useless in practice.

The main advantage of the gradient method is its simplicity. Its main disadvantage is that its convergence rate depends so critically on the condition number of the Hessian or sublevel sets.

5.2.2 Steepest descent method

(Taken from Chapter 9.4 of [18])

The first-order Taylor approximation of $f(x + v)$ around x is $f(x + v) \approx \hat{f}(x + v) = f(x) + \nabla f(x)^T v$. The second term on the righthand side, $\nabla f(x)^T v$, is the directional derivative of f at x in the direction v . It gives the approximate change in f for a small step v . The step v is a descent direction if the directional derivative is negative.

We now address the question of how to choose v to make the directional derivative as negative as possible. Since the directional derivative $\nabla f(x)^T v$ is linear in v , it can be made as negative as we like by taking v large (provided v is a descent direction, *i.e.*, $\nabla f(x)^T v < 0$). To make the question sensible we have to limit the size of v , or normalize by the length of v . Let $\|\cdot\|$ be any norm on \mathbb{R}^n . We define a normalized steepest descent direction (with respect to the norm $\|\cdot\|$) as

$$\Delta x_{nsd} = \operatorname{argmin}\{\nabla f(x)^T v | \|v\| = 1\}. \quad (5.27)$$

(We say ‘a’ steepest descent direction because there can be multiple minimizers.) A normalized steepest descent direction Δx_{nsd} is a step of unit norm that gives the largest decrease in the linear approximation of f . A normalized steepest descent direction can be interpreted geometrically as follows. We can just as well define Δx_{nsd} as

$$\Delta x_{nsd} = \operatorname{argmin}\{\nabla f(x)^T v | \|v\| \leq 1\},$$

i.e., as the direction in the unit ball of $\|\cdot\|$ that extends farthest in the direction $-\nabla f(x)$.

It is also convenient to consider a steepest descent step Δx_{sd} that is unnormalized, by scaling the normalized steepest descent direction in a particular way:

$$\Delta x_{sd} = \|\nabla f(x)\|_* \Delta x_{nsd}, \quad (5.28)$$

where $\|\cdot\|_*$ denotes the dual norm. Note that for the steepest descent step, we have

$$\nabla f(x)^T \Delta x_{sd} = \|\nabla f(x)\|_* \nabla f(x)^T \Delta x_{nsd} = -\|\nabla f(x)\|_*^2$$

(as an exercise).

The steepest descent method uses the steepest descent direction as search direction.

Algorithm 9.4 Steepest descent method.

given a starting point $x \in \text{dom } f$.

repeat

1. Compute steepest descent direction Δx_{sd} .
2. Line search. Choose t via backtracking or exact line search.
3. Update. $x := x + t\Delta x_{sd}$.

until stopping criterion is satisfied.

When exact line search is used, scale factors in the descent direction have no effect, so the normalized or unnormalized direction can be used.

5.2.2.1 Steepest descent for Euclidean and quadratic norms

Steepest descent for Euclidean norm

If we take the norm $\|\cdot\|$ to be the Euclidean norm we find that the steepest descent direction is simply the negative gradient, i.e., $\Delta x_{sd} = -\nabla f(x)$. The steepest descent method for the Euclidean norm coincides with the gradient descent method.

Steepest descent for quadratic norm

We consider the quadratic norm $\|z\|_P = (z^T P z)^{1/2} = \|P^{1/2}z\|_2$, where $P \in \mathbb{S}_{++}^n$. The normalized steepest descent direction is given by

$$\Delta x_{nsd} = -(\nabla f(x)^T P^{-1} \nabla f(x))^{-1/2} P^{-1} \nabla f(x).$$

The dual norm is given by $\|z\|_* = \|P^{-1/2}z\|_2$, so the steepest descent step with respect to $\|\cdot\|_P$ is given by

$$\Delta x_{nsd} = -P^{-1} \nabla f(x). \quad (5.29)$$

The normalized steepest descent direction for a quadratic norm is illustrated in Figure 5.9.

Interpretation via change of coordinates

We can give an interesting alternative interpretation of the steepest descent direction ∇x_{nsd} as the gradient search direction after a change of coordinates is applied to the problem. Define $\bar{u} = P^{1/2}u$, so we have $\|u\|_P = \|\bar{u}\|_2$. Using this change of coordinates, we can solve the original problem of minimizing f by solving the equivalent problem of minimizing the function $\bar{f} : \mathbb{R}^n \rightarrow \mathbb{R}$, given by

$$\bar{f}(\bar{u}) = f(P^{-1/2}\bar{u}) = f(u).$$

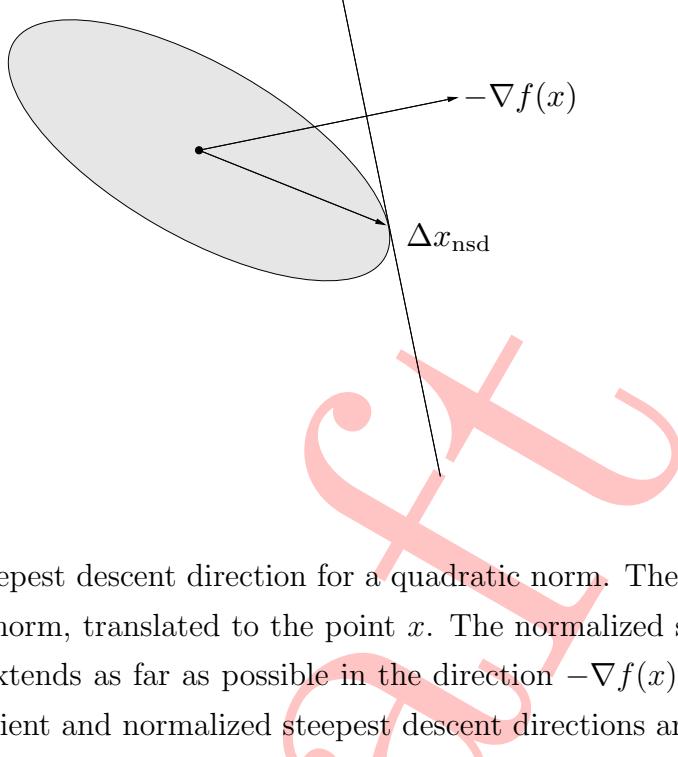


图 5.9: Normalized steepest descent direction for a quadratic norm. The ellipsoid shown is the unit ball of the norm, translated to the point x . The normalized steepest descent direction ∇x_{nsd} at x extends as far as possible in the direction $-\nabla f(x)$ while staying in the ellipsoid. The gradient and normalized steepest descent directions are shown.

If we apply the gradient method to \bar{f} , the search direction at a point \bar{x} (which corresponds to the point $x = P^{-1/2}\bar{x}$ for the original problem) is

$$\Delta\bar{x} = -\nabla\bar{f}(\bar{x}) = -P^{-1/2}\nabla f(P^{-1/2}\bar{x}) = -P^{-1/2}\nabla f(x).$$

This gradient search direction corresponds to the direction

$$\Delta x = P^{-1/2}(-P^{-1/2}\nabla f(x)) = -P^{-1}\nabla f(x),$$

for the original variable x . In other words, the steepest descent method in the quadratic norm $\|\cdot\|_P$ can be thought of as the gradient method applied to the problem after the change of coordinates $\bar{x} = P^{1/2}x$.

5.2.2.2 Steepest descent for l_1 -norm

As another example, we consider the steepest descent method for the l_1 -norm. A normalized steepest descent direction,

$$\Delta x_{nsd} = \operatorname{argmin}\{\nabla f(x)^T v | \|v\|_1 \leq 1\}$$

is easily characterized. Let i be any index for which $\|\nabla f(x)\|_\infty = |(\nabla f(x))_i|$. Then a normalized steepest descent direction Δx_{nsd} for the l_1 -norm is given by

$$\Delta x_{nsd} = -\operatorname{sign}\left(\frac{\partial f(x)}{\partial x_i}\right) e_i,$$

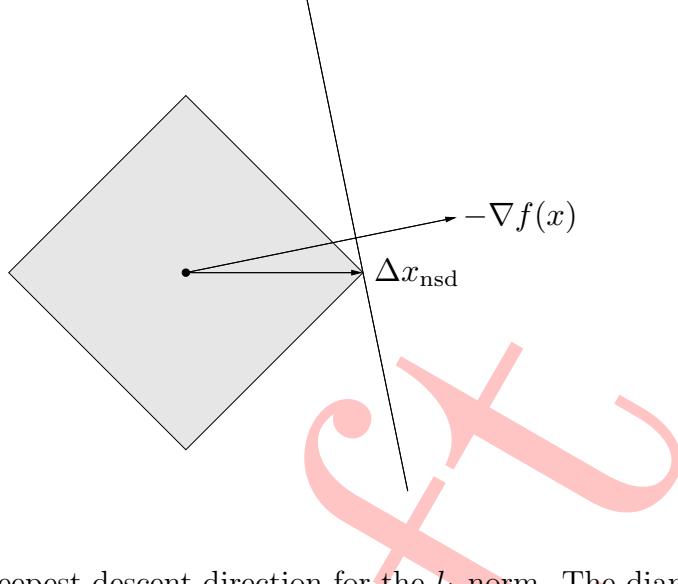


图 5.10: Normalized steepest descent direction for the l_1 -norm. The diamond is the unit ball of the l_1 -norm, translated to the point x . The normalized steepest descent direction can always be chosen in the direction of a standard basis vector; in this example we have $\Delta x_{nsd} = e_1$.

where e_i is the i th standard basis vector. An unnormalized steepest descent direction is then

$$\Delta x_{sd} = \Delta x_{nsd} \|\nabla f(x)\|_\infty = -\frac{\partial f(x)}{\partial x_i} e_i.$$

Thus, the normalized steepest descent direction in l_1 -norm can always be chosen to be a standard basis vector (or a negative standard basis vector). It is the coordinate axis direction along which the approximate decrease in f is greatest. This is illustrated in Figure 5.10.

The steepest descent algorithm in the l_1 -norm has a very natural interpretation: At each iteration we select a component of $\nabla f(x)$ with maximum absolute value, and then decrease or increase the corresponding component of x , according to the sign of $(\nabla f(x))_i$. The algorithm is sometimes called a coordinate-descent algorithm, since only one component of the variable x is updated at each iteration. This can greatly simplify, or even trivialize, the line search.

Example 9.2 *Frobenius norm scaling.* Consider the unconstrained geometric program

$$\text{minimize } \sum_{i,j=1}^n M_{ij}^2 d_i^2 / d_j^2,$$

where $\mathbf{M} \in \mathbb{R}^{n \times n}$ is given, and the variable is $\mathbf{d} \in \mathbb{R}^n$. Using the change of variables

$x_i = 2 \log d_i$ we can express this geometric program in convex form as

$$\text{minimize } f(x) = \sum_{i,j=1}^n M_{ij}^2 e^{x_i - x_j}.$$

It is easy to minimize f one component at a time. Keeping all components except the k th fixed, we can write $f(x) = \alpha_k + \beta_k e^{-x_k} + \gamma_k e^{x_k}$, where

$$\alpha_k = M_{kk}^2 + \sum_{i \neq k} M_{ij}^2 e^{x_i - x_j}, \beta_k = \sum_{i \neq k} M_{ij}^2 e^{x_i}, \gamma_k = \sum_{j \neq k} M_{ij}^2 e^{-x_j}.$$

The minimum of $f(x)$, as a function of x_k , is obtained for $x_k = \log(\beta_k/\gamma_k)/2$. So for this problem an exact line search can be carried out using a simple analytical formula.

The l_1 -steepest descent algorithm with exact line search consists of repeating the following steps.

1. Compute the gradient

$$(\nabla f(x))_i = \beta_i e^{-x_i} + \gamma_i e^{x_i}, i = 1, \dots, n.$$

2. Select a largest (in absolute value) component of $\nabla f(x)$: $|\nabla f(x)|_k = \|\nabla f(x)\|_\infty$.

3. Minimize f over the scalar variable x_k , by setting $x_k = \log(\beta_k/\gamma_k)/2$.

5.2.2.3 Convergence analysis

In this section we extend the convergence analysis for the gradient method with backtracking line search to the steepest descent method for an arbitrary norm. We will use the fact that any norm can be bounded in terms of the Euclidean norm, *i.e.*, there exists a constant $\gamma \in (0, 1]$ such that

$$\|x\|_* \geq \gamma \|x\|_2.$$

(see Section 2.11.1.1) Again we assume f is strongly convex on the initial sublevel set S . The upper bound $\nabla^2 f(x) \preceq M\mathbf{I}$ implies an upper bound on the function $f(x + t\Delta x_{sd})$ as a function of t :

$$\begin{aligned} f(x + t\Delta x_{sd}) &\leq f(x) + t\nabla f(x)^T \Delta x_{sd} + \frac{M\|\Delta x_{sd}\|_2^2}{2}t^2, \\ &\leq f(x) + t\nabla f(x)^T \Delta x_{sd} + \frac{M\|\Delta x_{sd}\|_*^2}{2\gamma^2}t^2 \\ &= f(x) - t\|\nabla f(x)\|_*^2 + \frac{M}{2\gamma^2}t^2\|\nabla f(x)\|_*^2. \end{aligned} \tag{5.30}$$

The step size $\hat{t} = \gamma^2/M$ (which minimizes the quadratic upper bound (5.30)) satisfies the exit condition for the backtracking line search:

$$f(x + \hat{t}\Delta x_{sd}) \leq f(x) - \frac{\gamma^2}{2M} \|\nabla f(x)\|_*^2 \leq f(x) + \frac{\alpha\gamma^2}{M} \nabla f(x)^T \Delta x_{sd}. \quad (5.31)$$

since $\alpha < 1/2$ and $\nabla f(x)^T \Delta x_{sd} = -\|\nabla f(x)\|_*^2$. The line search therefore returns a step size $t \geq \min\{1, \beta\gamma^2/M\}$, and we have

$$\begin{aligned} f(x^+) &= f(x + t\Delta x_{sd}) \leq f(x) - \alpha \min\{1, \beta\gamma^2/M\} \|\nabla f(x)\|_*^2, \\ &\leq f(x) - \alpha\gamma^2 \min\{1, \beta\gamma^2/M\} \|\nabla f(x)\|_2^2. \end{aligned}$$

Subtracting p^* from both sides and using (5.9), we obtain

$$f(x^+) - p^* \leq c(f(x) - p^*),$$

where

$$c = 1 - 2m\alpha\gamma^2 \min\{1, \beta\gamma^2/M\} < 1.$$

Therefore we have

$$f(x^{(k)}) - p^* \leq c^k (f(x^{(0)}) - p^*),$$

i.e., linear convergence exactly as in the gradient method.

5.2.2.4 Discussion and examples

Choice of norm for steepest descent

The choice of norm used to define the steepest descent direction can have a dramatic effect on the convergence rate. For simplicity, we consider the case of steepest descent with quadratic P -norm. In Section 5.2.2.1, we showed that the steepest descent method with quadratic P -norm is the same as the gradient method applied to the problem after the change of coordinates $\bar{x} = P^{1/2}x$. We know that the gradient method works well when the condition numbers of the sublevel sets (or the Hessian near the optimal point) are moderate, and works poorly when the condition numbers are large. It follows that when the sublevel sets, after the change of coordinates $\bar{x} = P^{1/2}x$, are moderately conditioned, the steepest descent method will work well.

This observation provides a prescription for choosing P : It should be chosen so that the sublevel sets of f , transformed by $P^{-1/2}$, are well conditioned. For example if an approximation \hat{H} of the Hessian at the optimal point $H(x^*)$ were known, a very good choice of P would be $P = \hat{H}$, since the Hessian of \tilde{f} at the optimum is then

$$\hat{H}^{-1/2} \nabla^2 f(x^*) \hat{H}^{-1/2} \approx I,$$

and so is likely to have a low condition number. This same idea can be described without a change of coordinates. Saying that a sublevel set has low condition number after the change of coordinates $\hat{x} = P^{1/2}x$ is the same as saying that the ellipsoid

$$\varepsilon = \{x | x^T P x \leq 1\}$$

approximates the shape of the sublevel set. (In other words, it gives a good approximation after appropriate scaling and translation.)

This dependence of the convergence rate on the choice of P can be viewed from two sides. The optimist's viewpoint is that for any problem, there is always a choice of P for which the steepest descent methods works very well. The challenge, of course, is to find such a P . The pessimist's viewpoint is that for any problem, there are a huge number of choices of P for which steepest descent works very poorly. In summary, we can say that the steepest descent method works well in cases where we can identify a matrix P for which the transformed problem has moderate condition number.

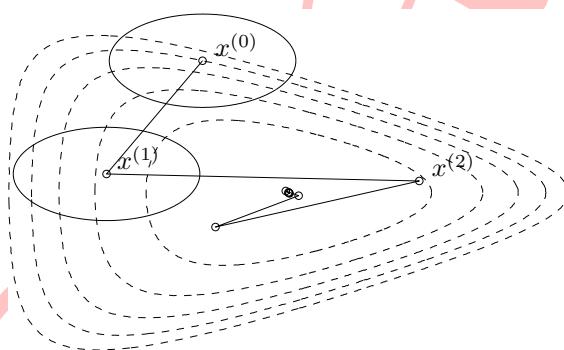


图 5.11: Steepest descent method with a quadratic norm $\|\cdot\|_{P_1}$. The ellipses are the boundaries of the norm balls $\{x | \|x - x^{(k)}\|_{P_1} \leq 1\}$ at $x^{(0)}$ and $x^{(1)}$.

Examples

In this section we illustrate some of these ideas using the nonquadratic problem in \mathbb{R}^2 with objective function (5.24). We apply the steepest descent method to the problem, using the two quadratic norms defined by

$$P_1 = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix}, P_2 = \begin{bmatrix} 8 & 0 \\ 0 & 2 \end{bmatrix}.$$

In both cases we use a backtracking line search with $\alpha = 0.1$ and $\beta = 0.7$.

Figures 5.11 and 5.12 show the iterates for steepest descent with norm $\|\cdot\|_{P_1}$ and norm $\|\cdot\|_{P_2}$. Figure 5.13 show the error versus iteration number for both norms. Figure

5.13 shows that the choice of norm strongly influences the convergence. With the norm $\|\cdot\|_{P_1}$, convergence is a bit more rapid than the gradient method, whereas with the norm $\|\cdot\|_{P_2}$, convergence is far slower.

This can be explained by examining the problems after the changes of coordinates $\bar{x} = P_1^{1/2}x$ and $\bar{x} = P_2^{1/2}x$, respectively. Figures 5.14 and 5.15 show the problems in the transformed coordinates. The change of variables associated with P_1 yields sublevel sets with modest condition number, so convergence is fast. The change of variables associated with P_2 yields sublevel sets that are more poorly conditioned, which explains the slower convergence.

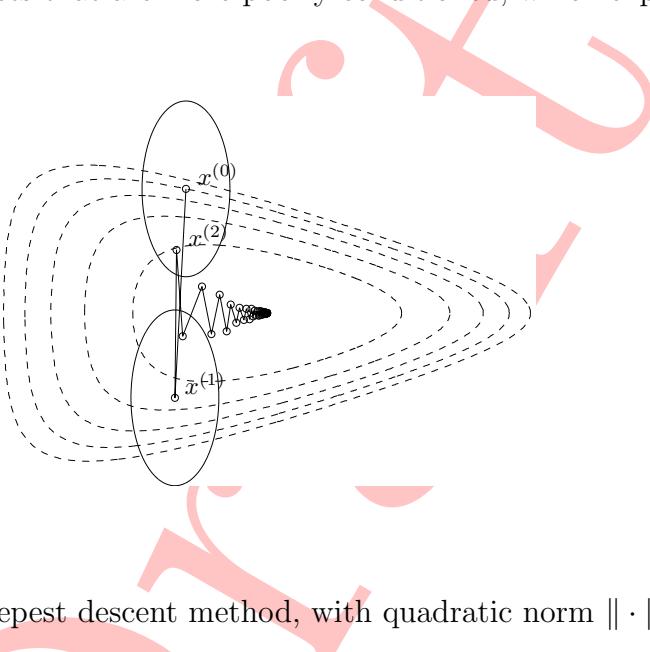


图 5.12: Steepest descent method, with quadratic norm $\|\cdot\|_{P_2}$.

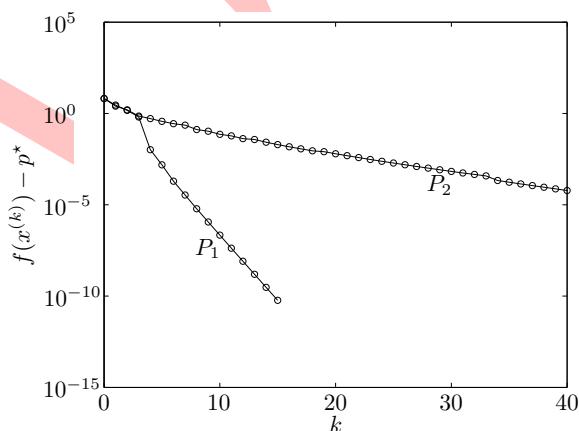


图 5.13: Error $f(x^{(k)}) - p^*$ versus iteration k , for the steepest descent method with the quadratic norm $\|\cdot\|_{P_1}$ and the quadratic norm $\|\cdot\|_{P_2}$. Convergence is rapid for the norm $\|\cdot\|_{P_1}$ and very slow for $\|\cdot\|_{P_2}$.

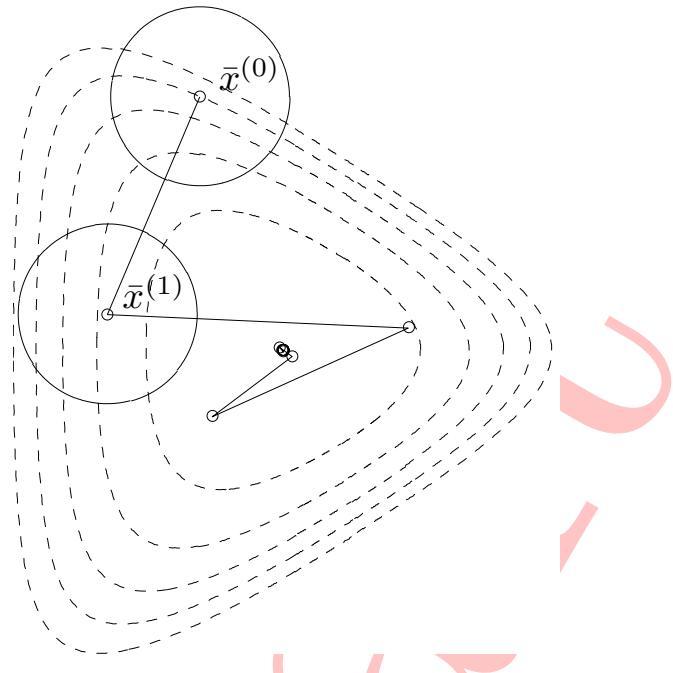


图 5.14: The iterates of steepest descent with norm $\|\cdot\|_{P_1}$, after the change of coordinates. This change of coordinates reduces the condition number of the sublevel sets, and so speeds up convergence.

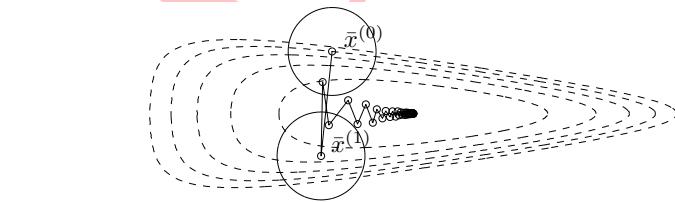


图 5.15: The iterates of steepest descent with norm $\|\cdot\|_{P_2}$, after the change of coordinates. This change of coordinates increases the condition number of the sublevel sets, and so slows down convergence.

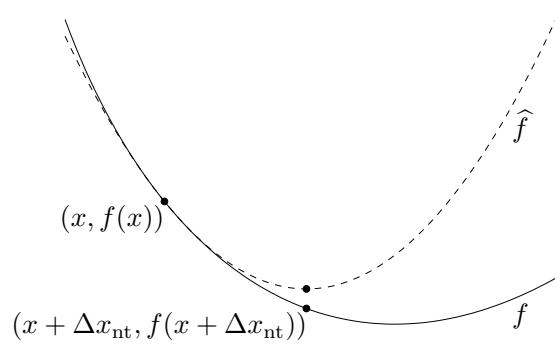


图 5.16: The function f (shown solid) and its second-order approximation \hat{f} at x (dashed). The Newton step Δx_{nt} is what must be added to x to give the minimizer of \hat{f} .

5.2.3 Exercises

(Taken from Chapter 9 of [18])

Unconstrained minimization

Exercise 342 (Minimizing a quadratic function). Consider the problem of minimizing a quadratic function:

$$\min_{\mathbf{x}} f(\mathbf{x}) = (1/2)\mathbf{x}^T \mathbf{P}\mathbf{x} + \mathbf{q}^T \mathbf{x} + r,$$

where $\mathbf{P} \in \mathbb{S}^n$ (but we do not assume $\mathbf{P} \succeq 0$).

- (a) Show that if $\mathbf{P} \not\succeq 0$, i.e., the objective function f is not convex, then the problem is unbounded below.
- (b) Now suppose that $\mathbf{P} \succeq 0$ (so the objective function is convex), but the optimality condition $\mathbf{P}\mathbf{x}^* = -\mathbf{q}$ does not have a solution. Show that the problem is unbounded below.

Exercise 343 (Minimizing a quadratic-over-linear fractional function). Consider the problem of minimizing the function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, defined as

$$f(\mathbf{x}) = \frac{\|\mathbf{Ax} - \mathbf{b}\|_2^2}{\mathbf{c}^T \mathbf{x} + d}, \quad \text{dom } f = \{\mathbf{x} | \mathbf{c}^T \mathbf{x} + d > 0\}.$$

We assume $\text{rank } \mathbf{A} = n$ and $\mathbf{b} \notin \mathcal{R}(\mathbf{A})$.

- (a) Show that f is closed.
- (b) Show that the minimizer \mathbf{x}^* of f is given by

$$\mathbf{x}^* = \mathbf{x}_1 + t\mathbf{x}_2$$

where $\mathbf{x}_1 = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$, $\mathbf{x}_2 = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{c}$, and $t \in \mathbb{R}$ can be calculated by solving a quadratic equation.

Exercise 344 (Initial point and sublevel set condition). Consider the function $f(\mathbf{x}) = x_1^2 + x_2^2$ with domain $\text{dom } f = \{(x_1, x_2) | x_1 > 1\}$.

(a) What is p^* ?

(b) Draw the sublevel set $\mathcal{S} = \{\mathbf{x} | f(\mathbf{x}) \leq f(\mathbf{x}^{(0)})\}$ for $\mathbf{x}^{(0)} = (2, 2)^T$. Is the sublevel set \mathcal{S} closed? Is f strongly convex on \mathcal{S} ?

(c) What happens if we apply the gradient method with backtracking line search, starting at $\mathbf{x}^{(0)}$? Does $f(\mathbf{x}^{(k)})$ converge to p^* ?

Exercise 345. Do you agree with the following argument? The ℓ_1 -norm of a vector $\mathbf{x} \in \mathbb{R}^m$ can be expressed as

$$\|\mathbf{x}\|_1 = (1/2) \inf_{\mathbf{y} > \mathbf{0}} \left(\sum_{i=1}^m x_i^2 / y_i + \mathbf{1}^T \mathbf{y} \right).$$

Therefore the ℓ_1 -norm approximation problem

$$\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_1$$

is equivalent to the minimization problem

$$\min_{\mathbf{x}, \mathbf{y}} f(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^m (\mathbf{a}_i^T \mathbf{x} - b_i)^2 / y_i + \mathbf{1}^T \mathbf{y} \quad (5.32)$$

with $\text{dom } f = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^m | \mathbf{y} > \mathbf{0}\}$, where \mathbf{a}_i^T is the i -th row of \mathbf{A} . Since f is twice differentiable and convex, we can solve the ℓ_1 -norm approximation problem by applying Newton's method to (5.32).

Exercise 346 (Backtracking line search). Suppose f is strongly convex with $m\mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq M\mathbf{I}$. Let $\Delta \mathbf{x}$ be a descent direction at \mathbf{x} . Show that the backtracking stopping condition holds for

$$0 < t \leq -\frac{\nabla f(\mathbf{x})^T \Delta \mathbf{x}}{M \|\Delta \mathbf{x}\|_2^2}.$$

Use this to give an upper bound on the number of backtracking iterations.

Gradient and steepest descent methods

Exercise 347 (Quadratic problem in \mathbb{R}^2). Verify the expressions for the iterates $\mathbf{x}^{(k)}$ in the first example of Section 5.2.1.2.

Exercise 348. Let $\Delta\mathbf{x}_{nsd}$ and $\Delta\mathbf{x}_{sd}$ be the normalized and unnormalized steepest descent directions at \mathbf{x} , for the norm $\|\cdot\|$. Prove the following identities.

- (a) $\nabla f(\mathbf{x})^T \Delta\mathbf{x}_{nsd} = -\|\nabla f(\mathbf{x})\|_*$.
- (b) $\nabla f(\mathbf{x})^T \Delta\mathbf{x}_{sd} = -\|\nabla f(\mathbf{x})\|_*^2$.
- (c) $\Delta\mathbf{x}_{sd} = \operatorname{argmin}_{\mathbf{v}} (\nabla f(\mathbf{x})^T \mathbf{v} + (1/2)\|\mathbf{v}\|^2)$.

Exercise 349 (Steepest descent method in ℓ_∞ -norm). Explain how to find a steepest descent direction in the ℓ_∞ -norm, and give a simple interpretation.

Exercise 350. Consider the unconstrained problem

$$\min_{\mathbf{x}} f(\mathbf{x}) = -\sum_{i=1}^m \log(1 - \mathbf{a}_i^T \mathbf{x}) - \sum_{i=1}^n \log(1 - x_i^2),$$

with variable $\mathbf{x} \in \mathbb{R}^n$, and $\operatorname{dom} f = \{\mathbf{x} | \mathbf{a}_i^T \mathbf{x} < 1, i = 1, \dots, m, |x_i| < 1, i = 1, \dots, n\}$. This is the problem of computing the analytic center of the set of linear inequalities

$$\mathbf{a}_i^T \mathbf{x} \leq 1, \quad i = 1, \dots, m, \quad |x_i| \leq 1, \quad i = 1, \dots, n.$$

Note that we can choose $\mathbf{x}^{(0)} = \mathbf{0}$ as our initial point. You can generate instances of this problem by choosing \mathbf{a}^i from some distribution on \mathbb{R}^n . Use the gradient method to solve the problem, using reasonable choices for the backtracking parameters, and a stopping criterion of the form $\|\nabla f(\mathbf{x})\|_2 \leq \eta$. Plot the objective function and step length versus iteration number. (Once you have determined p^* to high accuracy, you can also plot $f - p^*$ versus iteration.) Experiment with the backtracking parameters α and β to see their effect on the total number of iterations required. Carry these experiments out for several instances of the problem, of different sizes.

Exercise 351. Test the Gauss-Newton method on some problem instances of the form

$$f(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^m [f_i(\mathbf{x})]^2, \quad \text{where } f_i(\mathbf{x}) = (1/2)\mathbf{x}^T \mathbf{A}_i \mathbf{x} + \mathbf{b}_i^T \mathbf{x} + 1,$$

with $\mathbf{A}_i \in \mathbf{S}_{++}^n$ and $\mathbf{b}_i^T \mathbf{A}_i^{-1} \mathbf{b}_i \leq 2$ (which ensures that f is convex).

(Taken from Chapter 8 of [30])

Exercise 352. Perform two iterations leading to the minimization of

$$f(x_1, x_2) = x_1 + \frac{1}{2}x_2 + \frac{1}{2}x_1^2 + x_2^2 + 3$$

using the steepest descent method with the starting point $\mathbf{x}^{(0)} = \mathbf{0}$. Also determine an optimal solution analytically.

Exercise 353. Suppose that we wish to minimize a function $f : \mathbb{R} \rightarrow \mathbb{R}$ that has a derivative f' . A simple line search method, called derivative descent search (DDS), is described as follows: given that we are at a point $x^{(k)}$ we move in the direction of the negative derivative with step size α ; that is, $x^{(k+1)} = x^{(k)} - \alpha f'(x^{(k)})$ where $\alpha > 0$ is a constant.

In the following parts, assume that f is quadratic: $f(x) = \frac{1}{2}ax^2 - bx + c$ (where a , b , and c are constants, and $a > 0$).

- a. Write down the value of x^* (in terms of a , b , and c) that minimizes f .
- b. Write down the recursive equation for the DDS algorithm explicitly for this quadratic f .
- c. Assuming that the DDS algorithm converges, show that it converges to the optimal value x^* (found in part a).
- d. Find the order of convergence of the algorithm, assuming that it does converge.
- e. Find the range of values of α for which the algorithm converges (for this particular f) for all starting points $x^{(0)}$.

Exercise 354. Consider the function $f(\mathbf{x}) = 3(x_1^2 + x_2^2) + 4x_1x_2 + 5x_1 + 6x_2 + 7$, where $\mathbf{x} = [x_1, x_2]^T \in \mathbf{R}^2$. Suppose that we use a fixed-step-size gradient algorithm to find the minimizer of f :

$$x^{(k+1)} = x^{(k)} - \alpha \nabla f(x^{(k)}).$$

Find the largest range of values of α for which the algorithm is globally convergent.

Exercise 355. Consider the function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ given by

$$f(\mathbf{x}) = -\frac{3}{2}(x_1^2 + x_2^2) + (1+a)x_1x_2 - (x_1 + x_2) + b,$$

where a and b are some unknown real-valued parameters.

- a. Write the function f in the usual multivariable quadratic form.
- b. Find the largest set of values of a and b such that the unique global minimizer of f exists, and write down the minimizer (in terms of the parameters a and b).
- c. Consider the following algorithm:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \frac{2}{5} \nabla f(\mathbf{x}^{(k)}).$$

Find the largest set of values of a and b for which this algorithm converges to the global minimizer of f for any initial point $\mathbf{x}^{(0)}$.

Exercise 356. Consider the function $f : \mathbb{R} \rightarrow \mathbb{R}$ given by $f(x) = \frac{1}{2}(x - c)^2$, $c \in \mathbb{R}$. We are interested in computing the minimizer of f using the iterative algorithm

$$x^{(k+1)} = x^{(k)} - \alpha_k f'(x^{(k)}),$$

where f' is the derivative of f and α_k is a step size satisfying $0 < \alpha_k < 1$.

- a. Derive a formula relating $f(x^{(k+1)})$ with $f(x^{(k)})$, involving α_k .
- b. Show that the algorithm is globally convergent if and only if

$$\sum_{k=0}^{\infty} \alpha_k = \infty.$$

Exercise 357. Consider the function $f : \mathbb{R} \rightarrow \mathbb{R}$ given by $f(x) = x^3 - x$. Suppose that we use a fixed-step-size algorithm $x^{(k+1)} = x^{(k)} - \alpha f'(x^{(k)})$ to find a local minimizer of f . Find the largest range of values of α such that the algorithm is locally convergent.

Exercise 358. Consider the function f given by $f(x) = (x - 1)^2$, $x \in \mathbb{R}$. We are interested in computing the minimizer of f using the iterative algorithm $x^{(k+1)} = x^{(k)} - 2^{-k} \alpha f'(x^{(k)})$, where f' is the derivative of f and $0 < \alpha < 1$. Does the algorithm have the descent property? Is the algorithm globally convergent?

Exercise 359. Let $f : \mathbb{R} \rightarrow \mathbb{R}$, $f \in C^3$, with first derivative f' , second derivative f'' , and unique minimizer x^* . Consider a fixed-step-size gradient algorithm

$$x^{(k+1)} = x^{(k)} - \alpha f'(x^{(k)}).$$

Suppose that $f''(x^*) \neq 0$ and $\alpha = l/f''(x^*)$. Assuming that the algorithm converges to x^* , show that the order of convergence is at least 2.

Exercise 360. Consider the problem of minimizing $f(x) = \|\mathbf{a}x - \mathbf{b}\|^2$, where \mathbf{a} and \mathbf{b} are vectors in \mathbb{R}^n , and $\mathbf{a} \neq \mathbf{0}$.

- a. Derive an expression (in terms of \mathbf{a} and \mathbf{b}) for the solution to this problem.
- b. To solve the problem, suppose that we use an iterative algorithm of the form

$$x^{(k+1)} = x^{(k)} - \alpha f'(x^{(k)}),$$

where f' is the derivative of f . Find the largest range of values of α (in terms of \mathbf{a} and \mathbf{b}) for which the algorithm converges globally.

Exercise 361. Consider the optimization problem

$$\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|^2,$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m \geq n$, and $\mathbf{b} \in \mathbb{R}^m$.

- a. Show that the objective function for this problem is a quadratic function, and write down the gradient and Hessian of this quadratic.
- b. Write down the fixed-step-size gradient algorithm for solving this optimization problem.
- c. Suppose that

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}.$$

Find the largest range of values for α such that the algorithm in part b converges to the solution of the problem.

Exercise 362. Consider a function $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ given by $f(\mathbf{x}) = \mathbf{Ax} + \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{n \times n}$ and $\mathbf{b} \in \mathbb{R}^n$. Suppose that \mathbf{A} is invertible and \mathbf{x}^* is the zero of f (i.e., $f(\mathbf{x}^*) = \mathbf{0}$). We wish to compute \mathbf{x}^* using the iterative algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha f(\mathbf{x}^{(k)})$$

where $\alpha \in \mathbb{R}$, $\alpha > 0$. We say that the algorithm is globally monotone if for any $\mathbf{x}^{(0)}$, $\|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| \leq \|\mathbf{x}^{(k)} - \mathbf{x}^*\|$ for all k .

- a. Assume that all the eigenvalues of \mathbf{A} are real. Show that a necessary condition for the algorithm above to be globally monotone is that all the eigenvalues of \mathbf{A} are nonnegative. Hint: Use contraposition.
- b. Suppose that

$$\mathbf{A} = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 3 \\ -1 \end{bmatrix}.$$

Find the largest range of values of α for which the algorithm is globally convergent.

Exercise 363. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be given by $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b}$, where $\mathbf{b} \in \mathbb{R}^n$ and \mathbf{Q} is a real symmetric positive definite $n \times n$ matrix. Suppose that we apply the steepest descent method to this function, with $\mathbf{x}^{(0)} \neq \mathbf{Q}^{-1}\mathbf{b}$. Show that the method converges in one step, that is, $\mathbf{x}^{(1)} = \mathbf{Q}^{-1}\mathbf{b}$, if and only if $\mathbf{x}^{(0)}$ is chosen such that $\mathbf{g}^{(0)} = \mathbf{Q}\mathbf{x}^{(0)} - \mathbf{b}$ is an eigenvector of \mathbf{Q} .

Exercise 364. Suppose that we apply the steepest descent algorithm $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{g}^{(k)}$ to a quadratic function f with Hessian $\mathbf{Q} \succ \mathbf{0}$. Let λ_{\max} and λ_{\min} be the largest and the smallest eigenvalue of \mathbf{Q} , respectively. Which of the following two inequalities are possibly true? (When we say here that an inequality is “possibly” true, we mean that there exists a choice of f and $\mathbf{x}^{(0)}$ such that the inequality holds.)

a. $\alpha_0 \geq 2/\lambda_{\max}$.

b. $\alpha_0 \geq 1/\lambda_{\min}$.

Exercise 365. Suppose that we apply a fixed-step-size gradient algorithm to minimize

$$f(\mathbf{x}) = \mathbf{x}^T \begin{bmatrix} 3/2 & 2 \\ 0 & 3/2 \end{bmatrix} \mathbf{x} + \mathbf{x}^T \begin{bmatrix} 3 \\ -1 \end{bmatrix} - 22.$$

- a. Find the range of values of the step size for which the algorithm converges to the minimizer.
- b. Suppose that we use a step size of 1000 (which is too large). Find an initial condition that will cause the algorithm to diverge.

Exercise 366. Consider a fixed-step-size gradient algorithm applied to each of the functions $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ in parts a and b below. In each case, find the largest range of values of the step size a for which the algorithm is globally convergent.

a. $f(\mathbf{x}) = 1 + 2x_1 + 3(x_1^2 + x_2^2) + 4x_1x_2$.

b. $f(\mathbf{x}) = \mathbf{x}^T \begin{bmatrix} 3 & 3 \\ 1 & 3 \end{bmatrix} \mathbf{x} + [16, 23]\mathbf{x} + \pi^2$.

Exercise 367. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be given by $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b}$, where $\mathbf{b} \in \mathbb{R}^n$ and \mathbf{Q} is a real symmetric positive definite $n \times n$ matrix. Consider the algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \beta \alpha_k \mathbf{g}^{(k)},$$

where $\mathbf{g}^{(k)} = \mathbf{Q} \mathbf{x}^{(k)} - \mathbf{b}$, $\mathbf{a}_k = \mathbf{g}^{(k)T} \mathbf{g}^{(k)} / \mathbf{g}^{(k)T} \mathbf{Q} \mathbf{g}^{(k)}$ and $\beta \in \mathbb{R}$ is a given constant. (Note that the above reduces to the steepest descent algorithm if $\beta = 1$.) Show that $\{\mathbf{x}^{(k)}\}$ converges to $\mathbf{x}^* = \mathbf{Q}^{-1} \mathbf{b}$ for any initial condition $\mathbf{x}^{(0)}$ if and only if $0 < \beta < 2$.

Exercise 368. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be given by $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b}$, where $\mathbf{b} \in \mathbb{R}^n$ and \mathbf{Q} is a real symmetric positive definite $n \times n$ matrix. Consider a gradient algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{g}^{(k)},$$

where $\mathbf{g}^{(k)} = \mathbf{Q}\mathbf{x}^{(k)} - \mathbf{b}$ is the gradient of f at $\mathbf{x}^{(k)}$ and α_k is some step size. Show that the algorithm has the descent property (i.e., $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$ whenever $\mathbf{g}^{(k)} \neq \mathbf{0}$) if and only if $\gamma_k > 0$ for all k , where

$$\gamma_k = \alpha_k \frac{\mathbf{g}^{(k)T} \mathbf{Q} \mathbf{g}^{(k)}}{\mathbf{g}^{(k)T} \mathbf{Q}^{-1} \mathbf{g}^{(k)}} \left(2 \frac{\mathbf{g}^{(k)T} \mathbf{g}^{(k)}}{\mathbf{g}^{(k)T} \mathbf{Q} \mathbf{g}^{(k)}} - \alpha_k \right).$$

Exercise 369. Given $f : \mathbb{R}^n \rightarrow \mathbb{R}$, consider the general iterative algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)},$$

where $\mathbf{d}^{(1)}, \mathbf{d}^{(2)}, \dots$ are given vectors in \mathbb{R}^n and α_k is chosen to minimize $f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)})$; that is,

$$\alpha_k = \underset{\alpha}{\operatorname{argmin}} f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}).$$

Show that for each k , the vector $\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$ is orthogonal to $\nabla f(\mathbf{x}^{(k+1)})$ (assuming that the gradient exists).

Exercise 370. Write a simple MATLAB program for implementing the steepest descent algorithm using the secant method for the line search. For the stopping criterion, use the condition $\|\mathbf{g}^{(k)}\| \leq \varepsilon$ where $\varepsilon = 10^{-6}$. Test your program using

$$f(x_1, x_2, x_3) = (x_1 - 4)^4 + (x_2 - 3)^2 + 4(x_3 + 5)^4.$$

and an initial condition of $[-4, 5, 1]^T$, and determine the number of iterations required to satisfy the stopping criterion. Evaluate the objective function at the final point to see how close it is to 0.

Exercise 371. Apply the MATLAB program from Exercise 370 to Rosenbrock's function:

$$f(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2.$$

Use an initial condition of $\mathbf{x}^{(0)} = [-2, 2]^T$. Terminate the algorithm when the norm of the gradient of f is less than 10^{-4} .

Exercise 372. Consider the quadratic function $f(\mathbf{y}) = \mathbf{c}^T \mathbf{y} + (1/2) \mathbf{y}^T \mathbf{H} \mathbf{y}$, where \mathbf{H} is an $n \times n$ symmetric, positive definite matrix. Suppose that we use some algorithm for which the iterate $\mathbf{y}_{j+1} = \mathbf{y}_j - \lambda_j \mathbf{D}_j \nabla f(\mathbf{y}_j)$ is generated by an exact line search along the direction $-\mathbf{D}_j \nabla f(\mathbf{y}_j)$ from the previous iterate \mathbf{y}_j , where \mathbf{D}_j is some positive definite matrix. Then, if \mathbf{y}^* is the minimizing solution for f and if $e(\mathbf{y}) = (1/2)(\mathbf{y} - \mathbf{y}^*)^T \mathbf{H}(\mathbf{y} - \mathbf{y}^*)$ is an error function, show that at every step j , we have

$$e(\mathbf{y}_{j+1}) \leq \frac{(\alpha_j - 1)^2}{(\alpha_j + 1)^2} e(\mathbf{y}_j),$$

where α_j is the ratio of the largest to the smallest eigenvalue of $\mathbf{D}_j \mathbf{H}$.

5.3 Newton's Method

(Taken from Chapter 9 of [30])

5.3.1 Introduction

Recall that the method of steepest descent uses only first derivatives (gradients) in selecting a suitable search direction. This strategy is not always the most effective. If higher derivatives are used, the resulting iterative algorithm may perform better than the steepest descent method. Newton's method (sometimes called the Newton-Raphson method) uses first and second derivatives and indeed does perform better than the steepest descent method if the initial point is close to the minimizer. The idea behind this method is as follows. Given a starting point, we construct a quadratic approximation to the objective function that matches the first and second derivative values at that point. We then minimize the approximate (quadratic) function instead of the original objective function. We use the minimizer of the approximate function as the starting point in the next step and repeat the procedure iteratively. If the objective function is quadratic, then the approximation is exact, and the method yields the true minimizer in one step. If, on the other hand, the objective function is not quadratic, then the approximation will provide only an estimate of the position of the true minimizer. Figure 5.17 illustrates this idea.

We can obtain a quadratic approximation to the given twice continuously differentiable objection function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ using the Taylor series expansion of f about the current point , neglecting terms of order three and higher. We obtain

$$f(\mathbf{x}) \approx f(\mathbf{x}^{(k)}) + \mathbf{g}^{(k)T}(\mathbf{x} - \mathbf{x}^{(k)}) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^{(k)})^T F(\mathbf{x}^{(k)})(\mathbf{x} - \mathbf{x}^{(k)}) \triangleq q(\mathbf{x}), \quad (5.33)$$

where, for simplicity, we use the notation $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)})$. Applying the First-Order Necessary Condition (FONC) to q yields

$$\mathbf{0} = \nabla q(\mathbf{x}) = \mathbf{g}^{(k)} + \mathbf{F}(\mathbf{x}^{(k)})(\mathbf{x} - \mathbf{x}^{(k)}). \quad (5.34)$$

If $\mathbf{F}(\mathbf{x}^{(k)}) \succ 0$, then q achieves a minimum at

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mathbf{F}(\mathbf{x}^{(k)})^{-1}\mathbf{g}^{(k)}. \quad (5.35)$$

This recursive formula represents Newton's method.

Example 373. Use Newton's method to minimize the Powell function:

$$f(x_1, x_2, x_3, x_4) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4.$$

Use as the starting point $\mathbf{x}^{(0)} = [3, -1, 0, 1]^T$. Perform three iterations.

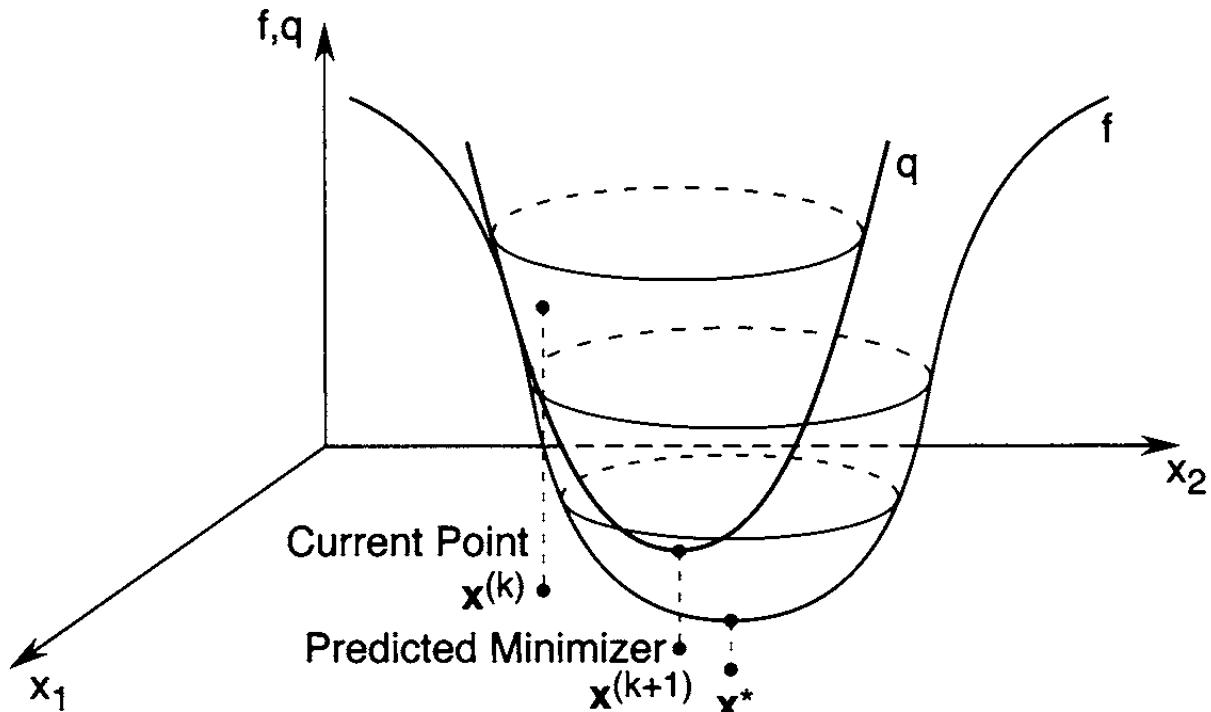


图 5.17: Quadratic approximation to the objective function using first and second derivatives.

Note that $f(\mathbf{x}^{(0)}) = 215$. We have

$$\nabla f(\mathbf{x}) = \begin{bmatrix} 2(x_1 + 10x_2) + 40(x_1 - x_4)^3 \\ 20(x_1 + 10x_2) + 4(x_2 - 2x_3)^3 \\ 10(x_3 - x_4) - 8(x_2 - 2x_3)^3 \\ -10(x_3 - x_4) - 40(x_1 - x_4)^3 \end{bmatrix}$$

and $\mathbf{F}(\mathbf{x})$ is given by

$$\begin{bmatrix} 2 + 120(x_1 - x_4)^2 & 20 & 0 & -120(x_1 - x_4)^2 \\ 20 & 200 + 12(x_2 - 2x_3)^2 & -24(x_2 - 2x_3)^2 & 0 \\ 0 & -24(x_2 - 2x_3)^2 & 10 + 48(x_2 - 2x_3)^2 & -10 \\ -120(x_1 - x_4)^2 & 0 & -10 & 10 + 120(x_1 - x_4)^2 \end{bmatrix}.$$

Iteration 1.

$$\mathbf{g}^{(0)} = [306, -144, -2, -310]^T,$$

$$\mathbf{F}(\mathbf{x}^{(0)}) = \begin{bmatrix} 482 & 20 & 0 & -480 \\ 20 & 212 & -24 & 0 \\ 0 & -24 & 58 & -10 \\ -480 & 0 & -10 & 490 \end{bmatrix},$$

$$\mathbf{F}(\mathbf{x}^{(0)})^{-1} = \begin{bmatrix} .1126 & -.0089 & .0154 & .1106 \\ -.0089 & .0057 & .0008 & -.0087 \\ .0154 & .0008 & .0203 & .0155 \\ .1106 & -.0087 & .0155 & .1107 \end{bmatrix},$$

$$\mathbf{F}(\mathbf{x}^{(0)})^{-1}\mathbf{g}^{(0)} = [1.4127, -0.8413, -0.2540, 0.7460]^T.$$

Hence,

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} - \mathbf{F}(\mathbf{x}^{(0)})^{-1}\mathbf{g}^{(0)} = [1.5873, -0.1587, 0.2540, 0.2540]^T,$$

$$f(\mathbf{x}^{(1)}) = 31.8.$$

Iteration 2.

$$\mathbf{g}^{(1)} = [94.81, -1.179, 2.371, -94.81]^T,$$

$$\mathbf{F}(\mathbf{x}^{(1)}) = \begin{bmatrix} 215.3 & 20 & 0 & -213.3 \\ 20 & 205.3 & -10.67 & 0 \\ 0 & -10.67 & 31.34 & -10 \\ -213.3 & 0 & -10 & 223.3 \end{bmatrix},$$

$$\mathbf{F}(\mathbf{x}^{(1)})^{-1}\mathbf{g}^{(1)} = [0.5291, -0.0529, -0.0846, 0.0846]^T.$$

Hence,

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} - \mathbf{F}(\mathbf{x}^{(1)})^{-1}\mathbf{g}^{(1)} = [1.0582, -0.1058, 0.1694, 0.1694]^T,$$

$$f(\mathbf{x}^{(2)}) = 6.28.$$

Iteration 3.

$$\mathbf{g}^{(2)} = [28.09, -0.3475, 0.7031, -28.08]^T,$$

$$\mathbf{F}(\mathbf{x}^{(2)}) = \begin{bmatrix} 96.80 & 20 & 0 & -94.80 \\ 20 & 202.4 & -4.744 & 0 \\ 0 & -4.744 & 19.49 & -10 \\ -94.80 & 0 & -10 & 104.80 \end{bmatrix},$$

$$\mathbf{x}^{(3)} = [0.7037, -0.0704, 0.1121, 0.1111]^T,$$

$$f(\mathbf{x}^{(3)}) = 1.24.$$

Observe that the k th iteration of Newton's method can be written in two steps as

1. Solve $\mathbf{F}(\mathbf{x}^{(k)})\mathbf{d}^{(k)} = -\mathbf{g}^{(k)}$ for $\mathbf{d}^{(k)}$;
2. Set $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{d}^{(k)}$.

Step 1 requires the solution of an $n \times n$ system of linear equations. Thus, an efficient method for solving systems of linear equations is essential when using Newton's method. As in the one-variable case, Newton's method can also be viewed as a technique for iteratively solving the equation

$$\mathbf{g}(\mathbf{x}) = \mathbf{0},$$

where $\mathbf{x} \in \mathbb{R}^n$, and $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^n$. In this case, $\mathbf{F}(\mathbf{x})$ is the Jacobian matrix of \mathbf{g} at \mathbf{x} , that is, $\mathbf{F}(\mathbf{x})$ is the $n \times n$ matrix whose (i, j) entry is $(\partial g_i / \partial x_j)(\mathbf{x})$, $i, j = 1, 2, \dots, n$.

5.3.2 Analysis of Newton's Method

As in the one-variable case, there is no guarantee that Newton's algorithm heads in the direction of decreasing values of the objective function if $\mathbf{F}(\mathbf{x}^{(k)})$ is not positive definite. (recall Figure 7.7 of [18] illustrating Newton's method for functions of one variable when $f'' < 0$). Moreover, even if $\mathbf{F}(\mathbf{x}^{(k)}) \succ 0$, Newton's method may not be a descent method; that is, it is possible that $f(\mathbf{x}^{(k+1)}) \geq f(\mathbf{x}^{(k)})$. For example, this may occur if our starting point \mathbf{x}^0 is far away from the solution. See the end of this section for a possible remedy to this problem. Despite the above drawbacks, Newton's method has superior convergence properties when the starting point is near the solution, as we shall see in the remainder of this section. The convergence analysis of Newton's method when f is a quadratic function is straightforward. In fact, Newton's method reaches the points: \mathbf{x}^* such that $\nabla f(\mathbf{x}^*) = \mathbf{0}$ in just one step starting from any initial point $\mathbf{x}^{(0)}$. To see this, suppose that $\mathbf{Q} = \mathbf{Q}^T$ is invertible, and

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b}.$$

Then,

$$\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x}) = \mathbf{Q} \mathbf{x} - \mathbf{b},$$

and

$$\mathbf{F}(\mathbf{x}) = \mathbf{Q}.$$

Hence, given any initial point $x^{(0)}$, by Newton's algorithm

$$\begin{aligned} \mathbf{x}^{(1)} &= \mathbf{x}^{(0)} - \mathbf{F}(\mathbf{x}^{(0)})^{-1} \mathbf{g}^{(0)} \\ &= \mathbf{x}^{(0)} - \mathbf{Q}^{-1}[\mathbf{Q} \mathbf{x}^{(0)} - \mathbf{b}] \\ &= \mathbf{Q}^{-1} \mathbf{b} \\ &= \mathbf{x}^*. \end{aligned}$$

Therefore, for the quadratic case, the order of convergence of Newton's algorithm is ∞ for any initial point $\mathbf{x}^{(0)}$ (compare the above with Exercise 8.13 of [18], which deals with the steepest descent algorithm). To analyze the convergence of Newton's method in the general case, we use results from Section 6.6.1. Let $\mathbf{x}^{(k)}$ be the Newton's method sequence for minimizing a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$. We show that $\mathbf{x}^{(k)}$ converges to the minimizer \mathbf{x}^* with order of convergence at least 2.

Theorem 374. Suppose that $f \in \mathcal{C}^3$, and $\mathbf{x}^* \in \mathbb{R}^n$ is a point such that $\nabla f(\mathbf{x}^*) = \mathbf{0}$ and $\mathbf{F}(\mathbf{x}^*)$ is invertible. Then, for all $\mathbf{x}^{(0)}$ sufficiently close to \mathbf{x}^* , Newton's method is well defined for all k , and converges to \mathbf{x}^* with order of convergence at least 2.

Proof. The Taylor series expansion of ∇f about $x^{(0)}$ yields

$$\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^{(0)}) - \mathbf{F}(\mathbf{x}^{(0)})(\mathbf{x} - \mathbf{x}^{(0)}) = \mathcal{O}(\|\mathbf{x} - \mathbf{x}^{(0)}\|^2).$$

Because by assumption $f \in \mathcal{C}^3$ and $\mathbf{F}(\mathbf{x}^*)$ is invertible, there exist constants $\epsilon > 0$, $c_1 > 0$ and $c_2 > 0$ such that if $\mathbf{x}^{(0)}, \mathbf{x} \in \{\mathbf{x} : \|\mathbf{x} - \mathbf{x}^*\| \leq \epsilon\}$, we have

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^{(0)}) - \mathbf{F}(\mathbf{x}^{(0)})(\mathbf{x} - \mathbf{x}^{(0)})\| \leq c_1 \|\mathbf{x} - \mathbf{x}^{(0)}\|^2$$

and by Lemma 18, $\mathbf{F}(\mathbf{x})^{-1}$ exists and satisfies

$$\|\mathbf{F}(\mathbf{x})^{-1}\| \leq c_2.$$

The first inequality above holds because the remainder term in the Taylor series expansion contains third derivatives of f that are continuous and hence bounded on $\{\mathbf{x} : \|\mathbf{x} - \mathbf{x}^*\| \leq \epsilon\}$.

Suppose that $\mathbf{x}^{(0)} \in \{\mathbf{x} : \|\mathbf{x} - \mathbf{x}^*\| \leq \epsilon\}$. Then, substituting $\mathbf{x} = \mathbf{x}^*$ in the above inequality and using the assumption that $\nabla f(\mathbf{x}^*) = \mathbf{0}$, we get

$$\|\mathbf{F}(\mathbf{x}^{(0)})(\mathbf{x}^{(0)} - \mathbf{x}^*) - \nabla f(\mathbf{x}^{(0)})\| \leq c_1 \|\mathbf{x}^{(0)} - \mathbf{x}^*\|^2.$$

Now, subtracting \mathbf{x}^* from both sides of Newton's algorithm and taking norms yields

$$\begin{aligned} \|\mathbf{x}^{(1)} - \mathbf{x}^*\| &= \|\mathbf{x}^{(0)} - \mathbf{x}^* - \mathbf{F}(\mathbf{x}^{(0)})^{-1} \nabla f(\mathbf{x}^{(0)})\| \\ &= \|\mathbf{F}(\mathbf{x}^{(0)})^{-1} (\mathbf{F}(\mathbf{x}^{(0)})(\mathbf{x}^{(0)} - \mathbf{x}^*) - \nabla f(\mathbf{x}^{(0)}))\| \\ &\leq \|\mathbf{F}(\mathbf{x}^{(0)})^{-1}\| \|\mathbf{F}(\mathbf{x}^{(0)})(\mathbf{x}^{(0)} - \mathbf{x}^*) - \nabla f(\mathbf{x}^{(0)})\|. \end{aligned}$$

Applying the above inequalities involving the constants c_1 and c_2 gives

$$\|\mathbf{x}^{(1)} - \mathbf{x}^*\| \leq c_1 c_2 \|\mathbf{x}^{(0)} - \mathbf{x}^*\|^2.$$

Suppose that $\mathbf{x}^{(0)}$ is such that

$$\|\mathbf{x}^{(0)} - \mathbf{x}^*\| \leq \frac{\alpha}{c_1 c_2},$$

where $\alpha \in (0, 1)$. Then,

$$\|\mathbf{x}^{(1)} - \mathbf{x}^*\| \leq \alpha \|\mathbf{x}^{(0)} - \mathbf{x}^*\|.$$

By induction, we obtain

$$\begin{aligned}\|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| &\leq c_1 c_2 \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2, \\ \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| &\leq \alpha \|\mathbf{x}^{(k)} - \mathbf{x}^*\|.\end{aligned}$$

Hence,

$$\lim_{k \rightarrow \infty} \|\mathbf{x}^{(k)} - \mathbf{x}^*\| = 0,$$

and therefore the sequence $\{\mathbf{x}^{(k)}\}$ converges to \mathbf{x}^* . The order of convergence is at least 2 because $\|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| \leq c_1 c_2 \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2$, that is, $\|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| = \mathcal{O}(\|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2)$. \square

Warning: In Theorem 374, we did not state that \mathbf{x}^* is a local minimizer. For example, if \mathbf{x}^* is a local maximizer then provided that $f \in \mathbb{C}^3$ and $\mathbf{F}(\mathbf{x}^*)$ is invertible, Newton's method would converge to \mathbf{x}^* if we start close enough to it.

As stated in the above theorem, Newton's method has superior convergence properties if the starting point is near the solution. However, the method is not guaranteed to converge to the solution if we start far away from it (in fact, it may not even be well defined because the Hessian may be singular). In particular, the method may not be a descent method; that is, it is possible that $f(\mathbf{x}^{(k+1)}) \geq f(\mathbf{x}^{(k)})$. Fortunately, it is possible to modify the algorithm such that the descent property holds. To see this, we need the following result.

Theorem 375. Let $\{\mathbf{x}^{(k)}\}$ be the sequence generated by Newton's method for minimizing a given objective function $f(\mathbf{x})$. If the Hessian $\mathbf{F}(\mathbf{x}^{(k)}) \succ \mathbf{0}$ and $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)}) \neq \mathbf{0}$, then the direction

$$\mathbf{d}^{(k)} = -\mathbf{F}(\mathbf{x}^{(k)})^{-1} \mathbf{g}^{(k)} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$$

from $\mathbf{x}^{(k)}$ to $\mathbf{x}^{(k+1)}$ is a descent direction for f in the sense that there exists an $\hat{\alpha} > 0$ such that for all $\alpha \in (0, \hat{\alpha})$,

$$f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}) < f(\mathbf{x}^{(k)}).$$

Proof. Let

$$\phi(\alpha) = f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}).$$

Then, using the chain rule, we obtain

$$\phi'(\alpha) = \nabla f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)})^T \mathbf{d}^{(k)}.$$

Hence,

$$\phi'(0) = \nabla f(\mathbf{x}^{(k)})^T \mathbf{d}^{(k)} = -\mathbf{g}^{(k)} \mathbf{F}(\mathbf{x}^{(k)})^{-1} \mathbf{g}^{(k)} < 0,$$

because $\mathbf{F}(\mathbf{x}^{(k)})^{-1} \succ \mathbf{0}$ and $\mathbf{g}^{(k)} \neq \mathbf{0}$. Thus, there exists an $\hat{\alpha} > 0$ so that for all $\alpha \in (0, \hat{\alpha})$, $\phi(\alpha) < \phi(0)$. This implies that for all $\alpha \in (0, \hat{\alpha})$,

$$f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}) < f(\mathbf{x}^{(k)})$$

and the proof is completed. \square

The above theorem motivates the following modification of Newton's method:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{F}(\mathbf{x}^{(k)})^{-1} \mathbf{g}^{(k)},$$

where

$$\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} - \alpha \mathbf{F}(\mathbf{x}^{(k)})^{-1} \mathbf{g}^{(k)}),$$

that is, at each iteration, we perform a line search in the direction $-\mathbf{F}(\mathbf{x}^{(k)})^{-1} \mathbf{g}^{(k)}$. By Theorem 375, we conclude that the above modified Newton's method has the descent property; that is

$$f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$$

whenever $\mathbf{g}^{(k)} \neq \mathbf{0}$.

A drawback of Newton's method is that evaluation of $\mathbf{F}(\mathbf{x}^{(k)})$ for large n can be computationally expensive. Furthermore, we have to solve the set of n linear equations $\mathbf{F}(\mathbf{x}^{(k)}) \mathbf{d}^{(k)} = -\mathbf{g}^{(k)}$. In Sections 5.4 and 5.5 (Chapters 10 and 11 of [18]), we discuss methods that alleviate this difficulty. Another source of potential problems in Newton's method arises from the Hessian matrix not being positive definite. In the next section, we describe a simple modification to Newton's method to overcome this problem.

5.3.3 Levenberg-Marquardt modification

If the Hessian matrix $\mathbf{F}(\mathbf{x}^{(k)})$ is not positive definite, then the search direction $\mathbf{d}^{(k)} = -\mathbf{F}(\mathbf{x}^{(k)})^{-1} \mathbf{g}^{(k)}$ may not point in a descent direction. A simple technique to ensure that the

search direction is a descent direction is to introduce the so-called *Levenberg-Marquardt* modification to Newton's algorithm:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - (\mathbf{F}(\mathbf{x}^{(k)}) + \mu_k \mathbf{I})^{-1} \mathbf{g}^{(k)},$$

where $\mu_k \geq 0$.

The idea underlying the Levenberg-Marquardt modification is as follows. Consider a symmetric matrix \mathbf{F} , which may not be positive definite. Let $\lambda_1, \dots, \lambda_n$ be the eigenvalues of \mathbf{F} with corresponding eigenvectors $\mathbf{v}_1, \dots, \mathbf{v}_n$. The eigenvalues $\lambda_1, \dots, \lambda_n$ are real, but may not all be positive. Next, consider the matrix $\mathbf{G} = \mathbf{F} + \mu \mathbf{I}$, where $\mu \geq 0$. Note that the eigenvalues of \mathbf{G} are $\lambda_1 + \mu, \dots, \lambda_n + \mu$. Indeed,

$$\begin{aligned}\mathbf{G}\mathbf{v}_i &= (\mathbf{F} + \mu \mathbf{I})\mathbf{v}_i \\ &= \mathbf{F}\mathbf{v}_i + \mu \mathbf{I}\mathbf{v}_i \\ &= \lambda_i \mathbf{v}_i + \mu \mathbf{v}_i \\ &= (\lambda_i + \mu) \mathbf{v}_i\end{aligned}$$

which shows that for all $i = 1, \dots, n$, \mathbf{v}_i is also an eigenvector of \mathbf{G} with eigenvalue $\lambda_i + \mu$. Therefore, if μ is sufficiently large, then all the eigenvalues of \mathbf{G} are positive, and \mathbf{G} is positive definite. Accordingly, if the parameter μ_k in the Levenberg-Marquardt modification of Newton's algorithm is sufficiently large, then the search direction $\mathbf{d}^{(k)} = -(\mathbf{F}(\mathbf{x}^{(k)}) + \mu_k \mathbf{I})^{-1} \mathbf{g}^{(k)}$ always points in a descent direction (in the sense of Theorem 375). In this case, if we further introduce a step size α_k as described in the previous section,

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k (\mathbf{F}(\mathbf{x}^{(k)}) + \mu_k \mathbf{I})^{-1} \mathbf{g}^{(k)},$$

then we are guaranteed that the descent property holds.

The Levenberg-Marquardt modification of Newton's algorithm can be made to approach the behavior of the pure Newton's method by letting $\mu_k \rightarrow 0$. On the other hand, by letting $\mu_k \rightarrow \infty$, the algorithm approaches a pure gradient method with small step size. In practice, we may start with a small value of μ_k , and then slowly increase it until we find that the iteration is descent, that is $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$.

5.3.4 Newton's method for nonlinear least-squares

We now examine a particular class of optimization problems and the use of Newton's method for solving them. Consider the following problem:

$$\min_{\mathbf{x}} \sum_{i=1}^m (r_i(\mathbf{x}))^2,$$

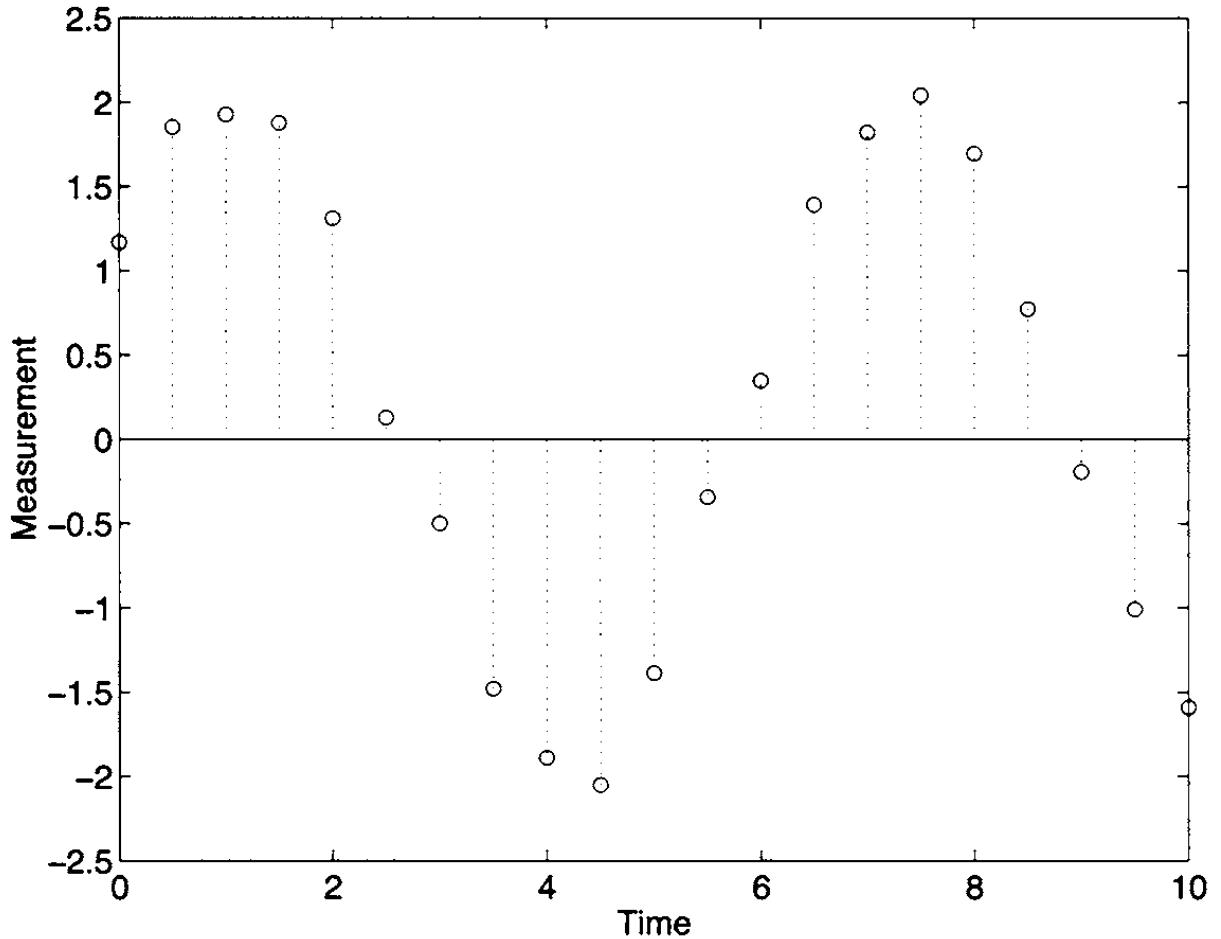


图 5.18: Measurement data for Example 376.

where $r_i : \mathbb{R} \rightarrow \mathbb{R}, i = 1, \dots, m$, are given functions. This particular problem is called a *nonlinear least-squares problem*.

Defining $\mathbf{r} = [r_1, \dots, r_m]^T$, we write the objective function as $f(\mathbf{x}) = \mathbf{r}(\mathbf{x})^T \mathbf{r}(\mathbf{x})$. To apply Newton's method, we need to compute the gradient and the Hessian of f . The j th component of $\nabla f(\mathbf{x})$ is

$$\begin{aligned} (\nabla f(\mathbf{x}))_j &= \frac{\partial f}{\partial x_j}(\mathbf{x}) \\ &= 2 \sum_{i=1}^m r_i(\mathbf{x}) \frac{\partial r_i}{\partial x_j}(\mathbf{x}). \end{aligned}$$

Denote the Jacobian matrix of \mathbf{r} by

$$\mathbf{J}(\mathbf{x}) = \begin{bmatrix} \frac{\partial r_1}{\partial x_1}(\mathbf{x}) & \cdots & \frac{\partial r_1}{\partial x_n}(\mathbf{x}) \\ \vdots & & \vdots \\ \frac{\partial r_m}{\partial x_1}(\mathbf{x}) & \cdots & \frac{\partial r_m}{\partial x_n}(\mathbf{x}) \end{bmatrix}.$$

Then, the gradient of f can be represented as

$$\nabla f(\mathbf{x}) = 2\mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x}).$$

Next, we compute the Hessian matrix of f . The (k, j) th component of the Hessian is given by

$$\begin{aligned}\frac{\partial^2 f}{\partial x_k \partial x_j}(\mathbf{x}) &= \frac{\partial}{\partial x_k} \left(\frac{\partial f}{\partial x_j}(\mathbf{x}) \right) \\ &= \frac{\partial}{\partial x_k} \left(2 \sum_{i=1}^m r_i(\mathbf{x}) \frac{\partial r_i}{\partial x_j}(\mathbf{x}) \right) \\ &= 2 \sum_{i=1}^m \left(\frac{\partial r_i}{\partial x_k}(\mathbf{x}) \frac{\partial r_i}{\partial x_j}(\mathbf{x}) + r_i(\mathbf{x}) \frac{\partial^2 r_i}{\partial x_k \partial x_j}(\mathbf{x}) \right).\end{aligned}$$

Letting $\mathbf{S}(\mathbf{x})$ be the matrix whose (k, j) th component is

$$\sum_{i=1}^m r_i(\mathbf{x}) \frac{\partial^2 r_i}{\partial x_k \partial x_j}(\mathbf{x}),$$

we write the Hessian matrix as

$$\mathbf{F}(\mathbf{x}) = 2(\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x}) + \mathbf{S}(\mathbf{x})).$$

Therefore, Newton's method applied to the nonlinear least-squares problem is given by

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - (\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x}) + \mathbf{S}(\mathbf{x}))^{-1} \mathbf{J}(\mathbf{x})^T \mathbf{r}(\mathbf{x}).$$

Note that the Gauss-Newton method does not require calculation of the second derivatives of \mathbf{r} .

Example 376. Suppose we are given m measurements of a process at m points in time, as depicted in Figure 5.18 (here $m = 21$). Let t_1, \dots, t_m denote the measurement times, and y_1, \dots, y_m the measurement values. Note that $t_1 = 0$ while $t_{21} = 10$. We wish to fit a sinusoid to the measurement data. The equation of the sinusoid is

$$y = A \sin(\omega t + \phi)$$

with appropriate choices of the parameters A, ω , and ϕ . To formulate the data-fitting problem, we construct the objective function

$$\sum_{i=1}^m (y_i - A \sin(\omega t_i + \phi))^2,$$

representing the sum of the squared errors between the measurement values and the function values at the corresponding points in time. Let $\mathbf{x} = [A, \omega, \phi]^T$ represent the vector of decision variables. We therefore obtain a nonlinear least-squares problem with

$$r_i(\mathbf{x}) = y_i - A \sin(\omega t_i + \phi).$$

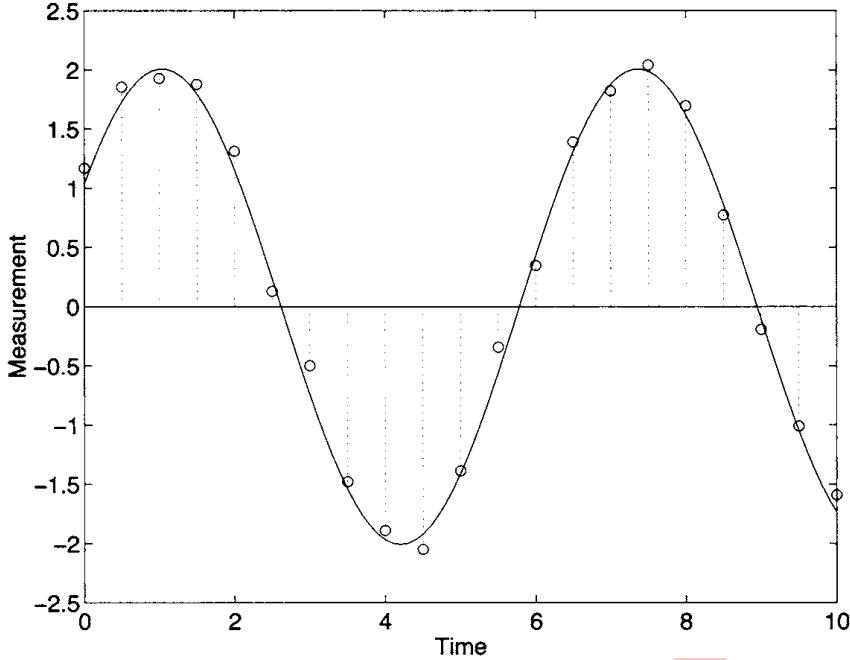


图 5.19: Sinusoid of best fit in Example 377.

Example 377. Recall the data fitting problem in Example 376, with

$$r_i(\mathbf{x}) = y_i - A \sin(\omega t_i + \phi), \quad i = 1, \dots, 21.$$

The Jacobian matrix $\mathbf{J}(\mathbf{x})$ in this problem is a 21×3 matrix with elements given by:

$$\begin{aligned} (\mathbf{J}(\mathbf{x}))_{(i,1)} &= -\sin(\omega t_i + \phi) \\ (\mathbf{J}(\mathbf{x}))_{(i,2)} &= -t_i A \sin(\omega t_i + \phi) \\ (\mathbf{J}(\mathbf{x}))_{(i,3)} &= -A \cos(\omega t_i + \phi), \quad i = 1, \dots, 21. \end{aligned}$$

Using the above expressions, we apply the Gauss-Newton algorithm to find the sinusoid of best fit, given the data pairs $(t_1, y_1), \dots, (t_m, y_m)$. Figure 5.19 shows a plot of the sinusoid of best fit obtained from the Gauss-Newton algorithm. The parameters of this sinusoid are: $A = 2.01$, $\omega = 0.992$, and $\phi = 0.541$.

A potential problem with the Gauss-Newton method is that the matrix $\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x})$ may not be positive definite. As described before, this problem can be overcome using a Levenberg-Marquardt modification:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - (\mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x}) + \mu_k \mathbf{I})^{-1} \mathbf{J}(\mathbf{x})^T r(\mathbf{x}).$$

The above is referred to in the literature as the Levenberg-Marquardt algorithm, because the original Levenberg-Marquardt modification was developed specifically for the non-linear least-squares problem. An alternative interpretation of the Levenberg-Marquardt algorithm is to view the term $\mu_k \mathbf{I}$ as an approximation to $S(\mathbf{x})$ in Newton's algorithm.

5.3.5 Exercises

Exercise 378. Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be given by $f(x) = (x - x_0)^4$, where $x_0 \in \mathbb{R}$ is a constant. Suppose that we apply Newton's method to the problem of minimizing f .

- a. Write down the update equation for Newton's method applied to the problem.
- b. Let $y^{(k)} = |x^{(k)} - x_0|$, where $x^{(k)}$ is the k th iterate in Newton's method. Show that the sequence $\{y^{(k)}\}$ satisfies $y^{(k+1)} = \frac{2}{3}y^{(k)}$.
- c. Show that $x^{(k)} \rightarrow x_0$ for any initial guess $x^{(0)}$.
- d. Show that the order of convergence of the sequence $\{x^{(k)}\}$ in part b is 1.
- e. Theorem 374 states that under certain conditions, the order of convergence of Newton's method is at least 2. Why does that theorem not hold in this particular problem?

Exercise 379. Consider the problem of minimizing $f(x) = x^{\frac{4}{3}} = (\sqrt[3]{x})^4$, $x \in \mathbb{R}$. Note that 0 is the global minimizer of f .

- a. Write down the algorithm for Newton's method applied to this problem.
- b. Show that as long as the starting point is not 0, the algorithm in part a does not converge to 0 (no matter how close to 0 we start).

Exercise 380. Consider Rosenbrock's Function: $f(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$, where $\mathbf{x} = [x_1, x_2]^T$ (known to be a “nasty” function, often used as a benchmark for testing algorithms). This function is also known as the banana function because of the shape of its level sets.

- a. Prove that $[1, 1]^T$ is the unique global minimizer of f over \mathbb{R}^2 .
- b. With a starting point of $[0, 0]^T$, apply two iterations of Newton's method. Hint:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \quad (5.36)$$

- c. Repeat part b using a gradient algorithm with a fixed step size of $\alpha_k = 0.05$ at each iteration.

Exercise 381. Consider the modified Newton's algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{F}(\mathbf{x}^{(k)})^{-1} \mathbf{g}^{(k)}, \quad (5.37)$$

where $\alpha_k = \arg \min_{\alpha > 0} f(\mathbf{x}^{(k)} - \alpha_k \mathbf{F}(\mathbf{x}^{(k)})^{-1} \mathbf{g}^{(k)})$. Suppose that we apply the algorithm to a quadratic function $f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b}$, where $\mathbf{Q} = \mathbf{Q}^T \succ 0$. Recall that the standard Newton's method reaches the point \mathbf{x}^* such that $\nabla f(\mathbf{x}^*) = 0$ in just one step starting from any initial point $\mathbf{x}^{(0)}$. Does the above modified Newton's algorithm possess the same property? Justify your answer.

(Taken from Chapter 9 of [18])

Exercise 382 (Newton decrement). Show that the Newton decrement $\lambda(\mathbf{x})$ satisfies

$$\lambda(\mathbf{x}) = \sup_{\mathbf{v}^T \nabla^2 f(\mathbf{x}) \mathbf{v} = 1} (-\mathbf{v}^T \nabla f(\mathbf{x})) = \sup_{\mathbf{v} \neq 0} \frac{-\mathbf{v}^T \nabla f(\mathbf{x})}{(\mathbf{v}^T \nabla^2 f(\mathbf{x}) \mathbf{v})^{1/2}}.$$

Exercise 383 (The pure Newton method). Newton's method with fixed step size $t = 1$ can diverge if the initial point is not close to \mathbf{x}^* . In this problem we consider two examples.

- (a) $f(x) = \log(e^x + e^{-x})$ has a unique minimizer $x^* = 0$. Run Newton's method with fixed step size $t = 1$, starting at $x^{(0)} = 1$ and at $x^{(0)} = 1.1$.
- (b) $f(x) = -\log x + x$ has a unique minimizer $x^* = 1$. Run Newton's method with fixed step size $t = 1$, starting at $x^{(0)} = 3$.

Plot f and f' , and show the first few iterates.

Exercise 384 (Gradient and Newton methods for composition functions). Suppose $\phi : \mathbb{R} \rightarrow \mathbb{R}$ is increasing and convex, and $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex, so $g(x) = \phi(f(x))$ is convex. (We assume that f and g are twice differentiable.) The problems of minimizing f and minimizing g are clearly equivalent.

Compare the gradient method and Newton's method, applied to f and g . How are the search directions related? How are the methods related if an exact line search is used? Hint. Use the matrix inversion lemma (2.57).

Exercise 385. Trust region Newton method. If $\nabla^2 f(\mathbf{x})$ is singular (or very ill-conditioned), the Newton step $\Delta \mathbf{x}_{nt} = -\nabla^2 f(\mathbf{x})^{-1} \nabla f(\mathbf{x})$ is not well defined. Instead we can define a search direction $\Delta \mathbf{x}_{tr}$ as the solution of

$$\begin{aligned} \min_{\mathbf{v}} \quad & (1/2) \mathbf{v}^T \mathbf{H} \mathbf{v} + \mathbf{g}^T \mathbf{v} \\ \text{s.t.} \quad & \|\mathbf{v}\|_2 \leq \gamma, \end{aligned}$$

where $\mathbf{H} = \nabla^2 f(\mathbf{x})$, $\mathbf{g} = \nabla f(\mathbf{x})$, and γ is a positive constant. The point $\mathbf{x} + \Delta \mathbf{x}_{tr}$ minimizes the second-order approximation of f at \mathbf{x} , subject to the constraint that $\|(\mathbf{x} +$

$\Delta\mathbf{x}_{tr}) - \mathbf{x}\|_2 \leq \gamma$. The set $\{\mathbf{v} | \|\mathbf{v}\|_2 \leq \gamma\}$ is called the trust region. The parameter γ , the size of the trust region, reflects our confidence in the second-order model.

Show that $\Delta\mathbf{x}_{tr}$ minimizes

$$(1/2)\mathbf{v}^T \mathbf{H}\mathbf{v} + \mathbf{g}^T \mathbf{v} + \hat{\beta}\|\mathbf{v}\|_2^2,$$

for some $\hat{\beta}$. This quadratic function can be interpreted as a regularized quadratic model for f around \mathbf{x} .

(Taken from Chapter 6 of [109])

Exercise 386. The displacements of a system of carts can be found by minimizing the potential energy of the system (f):

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{K}\mathbf{x} - \mathbf{x}^T \mathbf{p},$$

where $\mathbf{K} = \begin{bmatrix} k_1 + k_4 + k_5 & -k_4 & -k_5 \\ -k_4 & k_2 + k_4 + k_6 & -k_6 \\ -k_5 & -k_6 & k_3 + k_5 + k_6 + k_7 + k_8 \end{bmatrix}$, $\mathbf{p} = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}$ and $\mathbf{x} =$

$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$. Derive the function $f(x_1, x_2, x_3)$ for the following data: $k_1 = 5000$, $k_2 = 1500$, $k_3 = 2000$, $k_4 = 1000$, $k_5 = 2500$, $k_6 = 500$, $k_7 = 3000$, $k_8 = 3500$, $p_1 = 1000$, $p_2 = 2000$ and $p_3 = 3000$. Complete one iteration of Newton's method and find the equilibrium configuration of the carts. Use $\mathbf{x}^{(0)} = [0, 0, 0]^T$.

Exercise 387. Perform two iterations of the Marquardt's method to minimize the function given in Exercise 411 from the stated starting point.

Exercise 388. Solve the equations $x_1 + 2x_2 + 3x_3 = 14$, $x_1 - x_2 + x_3 = 1$, and $3x_1 - 2x_2 + x_3 = 2$ using Marquardt's method of unconstrained minimization. Use the starting point $\mathbf{x}^{(0)} = [0, 0, 0]^T$.

Exercise 389. By the no-free-lunch (NFL) theorem, give two problems A and B, in which steepest descent is faster than Newton's method on problem A but slower on problem B.

5.4 Conjugate Direction Methods

(Taken from Chapter 10 of [30])

5.4.1 Introduction

The class of *conjugate direction methods* can be viewed as being intermediate between the method of steepest descent and Newton's method. The conjugate direction methods have the following properties:

1. Solve quadratics of n variables in n steps;
2. The usual implementation, the conjugate gradient algorithm, requires no Hessian matrix evaluations;
3. No matrix inversion and no storage of an $n \times n$ matrix required.

The conjugate direction methods typically perform better than the method of steepest descent, but not as well as Newton's method. As we saw from the method of steepest descent and Newton's method, the crucial factor in the efficiency of an iterative search method is the direction of search at each iteration. For a quadratic function of n variables $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b}$, $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{Q} = \mathbf{Q}^T \succ 0$, the best direction of search, as we shall see, is in the so-called \mathbf{Q} -conjugate direction. Basically, two directions $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$ in \mathbb{R}^n are said to be \mathbf{Q} -conjugate if $\mathbf{d}^{(1)T} \mathbf{Q} \mathbf{d}^{(2)} = 0$. In general, we have the following definition.

Definition 390. Let \mathbf{Q} be a real symmetric $n \times n$ matrix. The directions $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(m)}$ are \mathbf{Q} -conjugate if, for all $i \neq j$, we have $\mathbf{d}^{(i)T} \mathbf{Q} \mathbf{d}^{(j)} = 0$.

Lemma 391. Let \mathbf{Q} be a symmetric positive definite $n \times n$ matrix. If the directions $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(k)} \in \mathbb{R}^n$, $k \leq n - 1$, are nonzero and \mathbf{Q} -conjugate, then they are linearly independent.

Proof. Let $\alpha_0, \dots, \alpha_k$ be scalars such that

$$\alpha_0 \mathbf{d}^{(0)} + \alpha_1 \mathbf{d}^{(1)} + \dots + \alpha_k \mathbf{d}^{(k)} = 0.$$

Premultiplying the above equality by $\mathbf{d}^{(j)T} \mathbf{Q}$, $0 \leq j \leq k$, yields

$$\alpha_j \mathbf{d}^{(j)T} \mathbf{Q} \mathbf{d}^{(j)} = 0,$$

because all other terms $\mathbf{d}^{(i)T} \mathbf{Q} \mathbf{d}^{(j)}$, $i \neq j$, by \mathbf{Q} -conjugacy. But $\mathbf{Q} = \mathbf{Q}^T \succ 0$ and $\mathbf{d}^{(j)} \neq 0$; hence $\alpha_j = 0$, $j = 0, 1, \dots, k$. Therefore, $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \mathbf{d}^{(2)}, \dots, \mathbf{d}^{(k)}$, $k \leq n - 1$, are linearly independent. \square

Example 392. Let

$$Q = \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}.$$

Note that $\mathbf{Q} = \mathbf{Q}^T > 0$. The matrix \mathbf{Q} is positive definite because all its leading principal minors are positive:

$$\Delta_1 = 3 > 0, \quad \Delta_2 = \det \begin{bmatrix} 3 & 0 \\ 0 & 4 \end{bmatrix} = 12 > 0, \quad \Delta_3 = \det \mathbf{Q} = 20 > 0.$$

Our goal is to construct a set of \mathbf{Q} -conjugate vectors $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \mathbf{d}^{(2)}$. Let $\mathbf{d}^{(0)} = [1, 0, 0]^T$, $\mathbf{d}^{(1)} = [d_1^{(1)}, d_2^{(1)}, d_3^{(1)}]^T$, $\mathbf{d}^{(2)} = [d_1^{(2)}, d_2^{(2)}, d_3^{(2)}]^T$. We require $\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(1)} = 0$. We have

$$\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(1)} = [1, 0, 0] \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} d_1^{(1)} \\ d_2^{(1)} \\ d_3^{(1)} \end{bmatrix} = 3d_1^{(1)} + d_3^{(1)}.$$

Let $d_1^{(1)} = 1, d_2^{(1)} = 0, d_3^{(1)} = -3$. Then $\mathbf{d}^{(1)} = [1, 0, -3]^T$, and thus $\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(1)} = 0$. To find the third vector $\mathbf{d}^{(2)}$, which would be \mathbf{Q} -conjugate with $\mathbf{d}^{(0)}$ and $\mathbf{d}^{(1)}$, we require $\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(2)} = 0$ and $\mathbf{d}^{(1)T} \mathbf{Q} \mathbf{d}^{(2)} = 0$. We have

$$\begin{aligned} \mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(2)} &= 3d_1^{(2)} + d_3^{(2)} = 0, \\ \mathbf{d}^{(1)T} \mathbf{Q} \mathbf{d}^{(2)} &= -6d_2^{(2)} - 8d_3^{(2)} = 0. \end{aligned}$$

If we take $\mathbf{d}^{(2)} = [1, 4, -3]^T$, then the resulting set of vectors is mutually conjugate.

The above method of finding \mathbf{Q} -conjugate vectors is inefficient. A systematic procedure for finding \mathbf{Q} -conjugate vectors can be devised using the idea underlying the Gram-Schmidt process of transforming a given basis of \mathbb{R}^n into an orthonormal basis of \mathbb{R}^n (see Exercise).

5.4.2 The Conjugate Direction Algorithm

We now present the conjugate direction algorithm for minimizing the quadratic function of n variables

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b},$$

where $\mathbf{Q} = \mathbf{Q}^T \succ 0, \mathbf{x} \in \mathbb{R}^n$. Note that because $\mathbf{Q} \succ 0$, the function f has a global minimizer that can be found by solving $\mathbf{Q}\mathbf{x} = \mathbf{b}$.

Basic Conjugate Direction Algorithm. Given a starting point \mathbf{x}^0 , and \mathbf{Q} -conjugate directions, $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(n-1)}$; for $k \geq 0$,

$$\begin{aligned}\mathbf{g}^{(k)} &= \nabla f(\mathbf{x}^{(k)}) = \mathbf{Q}\mathbf{x}^{(k)} - \mathbf{b}, \\ \alpha_k &= -\frac{\mathbf{g}^{(k)T}\mathbf{d}^{(k)}}{\mathbf{d}^{(k)T}\mathbf{Q}\mathbf{d}^{(k)}}, \\ \mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}.\end{aligned}$$

Theorem 393. For any starting point \mathbf{x}^0 , the basic conjugate direction algorithm converges to the unique \mathbf{x}^* (that solves $\mathbf{Q}\mathbf{x} = \mathbf{b}$) in n steps; that is $\mathbf{x}^n = \mathbf{x}^*$.

Proof. Consider $\mathbf{x}^* - \mathbf{x}^0 \in \mathbb{R}^n$. Because the $\mathbf{d}^{(i)}$ are linearly independent, there exist constants $\beta_i, i = 0, \dots, n-1$, such that

$$\mathbf{x}^* - \mathbf{x}^0 = \beta_0 \mathbf{d}^{(0)} + \dots + \beta_{n-1} \mathbf{d}^{(n-1)}.$$

Now premultiply both sides of the above equation by $\mathbf{d}^{(k)T}\mathbf{Q}$, $0 \leq k \leq n$, to obtain

$$\mathbf{d}^{(k)T}\mathbf{Q}(\mathbf{x}^* - \mathbf{x}^0) = \beta_k \mathbf{d}^{(k)T}\mathbf{Q}\mathbf{d}^{(k)},$$

where the terms $\mathbf{d}^{(k)T}\mathbf{Q} = 0, k \neq i$, by the \mathbf{Q} -conjugate property. Hence,

$$\beta_k = \frac{\mathbf{d}^{(k)T}\mathbf{Q}(\mathbf{x}^* - \mathbf{x}^0)}{\mathbf{d}^{(k)T}\mathbf{Q}\mathbf{d}^{(k)}}.$$

Now, we can write

$$\mathbf{x}^{(k)} = \mathbf{x}^{(0)} + \alpha_0 \mathbf{d}^{(0)} + \dots + \alpha_{k-1} \mathbf{d}^{(k-1)}.$$

Therefore,

$$\mathbf{x}^{(k)} - \mathbf{x}^{(0)} = \alpha_0 \mathbf{d}^{(0)} + \dots + \alpha_{k-1} \mathbf{d}^{(k-1)}.$$

So writing

$$\mathbf{x}^* - \mathbf{x}^{(0)} = \mathbf{x}^* - \mathbf{x}^{(k)} + \mathbf{x}^{(k)} - \mathbf{x}^{(0)}$$

and premultiplying the above by $\mathbf{d}^{(k)T}\mathbf{Q}$ we obtain

$$\mathbf{d}^{(k)T}\mathbf{Q}(\mathbf{x}^* - \mathbf{x}^0) = \mathbf{d}^{(k)T}\mathbf{Q}(\mathbf{x}^* - \mathbf{x}^{(k)}) = -\mathbf{d}^{(k)T}\mathbf{g}^{(k)},$$

Because $\mathbf{g}^{(k)} = \mathbf{Q}\mathbf{x}^{(k)} - \mathbf{b}$ and $\mathbf{Q}\mathbf{x}^* = \mathbf{b}$. Thus,

$$\beta_k = \frac{\mathbf{d}^{(k)T}\mathbf{g}^{(k)}}{\mathbf{d}^{(k)T}\mathbf{Q}\mathbf{d}^{(k)}} = \alpha_k$$

and $\mathbf{x}^* = \mathbf{x}^{(n)}$, which completes the proof. \square

Example 394. Find the minimizer of

$$f(x_1, x_2) = \frac{1}{2} \mathbf{x}^T \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \mathbf{x} - \mathbf{x}^T \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \mathbf{x} \in \mathbb{R}^2, \quad (5.38)$$

using the conjugate direction method with the initial point $\mathbf{x}^{(0)} = [0, 0]^T$, and \mathbf{Q} -conjugate directions $\mathbf{d}^{(0)} = [1, 0]^T$ and $\mathbf{d}^{(1)} = [-\frac{3}{8}, \frac{3}{4}]^T$.

We have

$$\mathbf{g}^{(0)} = -\mathbf{b} = [1, -1]^T, \quad (5.39)$$

and hence

$$\alpha_0 = -\frac{\mathbf{g}^{(0)T} \mathbf{d}^{(0)}}{\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(0)}} = -\frac{[1, -1] \begin{bmatrix} 1 \\ 0 \end{bmatrix}}{[1, 0] \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}} = -\frac{1}{4}.$$

Thus,

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \alpha_0 \mathbf{d}^{(0)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \frac{1}{4} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -\frac{1}{4} \\ 0 \end{bmatrix}.$$

To find $\mathbf{x}^{(2)}$, we compute

$$\mathbf{g}^{(1)} = \mathbf{Q} \mathbf{x}^{(1)} - \mathbf{b} = \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} -\frac{1}{4} \\ 0 \end{bmatrix} - \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -\frac{3}{2} \end{bmatrix},$$

and

$$\alpha_1 = -\frac{\mathbf{g}^{(1)T} \mathbf{d}^{(1)}}{\mathbf{d}^{(1)T} \mathbf{Q} \mathbf{d}^{(1)}} = -\frac{[0, -\frac{3}{2}] \begin{bmatrix} -\frac{3}{8} \\ \frac{3}{4} \end{bmatrix}}{[-\frac{3}{8}, \frac{3}{4}] \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} -\frac{3}{8} \\ \frac{3}{4} \end{bmatrix}} = 2.$$

Therefore,

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \alpha_1 \mathbf{d}^{(1)} = \begin{bmatrix} -\frac{1}{4} \\ 0 \end{bmatrix} + 2 \begin{bmatrix} -\frac{3}{8} \\ \frac{3}{4} \end{bmatrix} = \begin{bmatrix} -1 \\ \frac{3}{2} \end{bmatrix}.$$

Because f is a quadratic function in two variables, $\mathbf{x}^{(2)} = \mathbf{x}^*$

For a quadratic function of n variables, the conjugate direction method reaches the solution after n steps. As we shall see below, the method also possesses a certain desirable property in the intermediate steps. To see this, suppose that we start at $\mathbf{x}^{(0)}$ and search in the direction $\mathbf{d}^{(0)}$ to obtain

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} - \frac{\mathbf{g}^{(0)T} \mathbf{d}^{(0)}}{\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(0)}} \mathbf{d}^{(0)}.$$

We claim that

$$\mathbf{g}^{(1)T} \mathbf{d}^{(0)} = 0.$$

To see this,

$$\begin{aligned}\mathbf{g}^{(1)T} \mathbf{d}^{(0)} &= (\mathbf{Qx}^{(1)} - \mathbf{b})^T \mathbf{d}^{(0)} \\ &= \mathbf{x}^{(0)T} \mathbf{Qd}^{(0)} - \left(\frac{\mathbf{g}^{(0)T} \mathbf{d}^{(0)}}{\mathbf{d}^{(0)T} \mathbf{Qd}^{(0)}} \right) \mathbf{d}^{(0)T} \mathbf{Qd}^{(0)} - \mathbf{b}^T \mathbf{d}^{(0)} \\ &= \mathbf{g}^{(0)T} \mathbf{d}^{(0)} - \mathbf{g}^{(0)T} \mathbf{d}^{(0)} = 0.\end{aligned}$$

The equation $\mathbf{g}^{(0)T} \mathbf{d}^{(0)} = 0$ implies that α_0 has the property that $\alpha_0 = \arg \min \phi_0(\alpha)$, where $\phi_0(\alpha) = f(\mathbf{x}^{(0)} + \alpha \mathbf{d}^{(0)})$. To see this, apply the chain rule to get

$$\frac{d\phi_0}{d\alpha}(\alpha) = \nabla f(\mathbf{x}^{(0)} + \alpha \mathbf{d}^{(0)})^T \mathbf{d}^{(0)}.$$

Evaluating the above at $\alpha = \alpha_0$, we get

$$\frac{d\phi_0}{d\alpha}(\alpha_0) = \mathbf{g}^{(1)T} \mathbf{d}^{(0)} = 0.$$

Because ϕ_0 is a quadratic function of α , and the coefficient of the α term in ϕ_0 is $\mathbf{d}^{(0)T} \mathbf{Qd}^{(0)} > 0$, the above implies that $\alpha_0 = \arg \min_{\alpha \in \mathbb{R}} \phi_0(\alpha)$.

Using a similar argument, we can show that for all k ,

$$\mathbf{g}^{(k+1)T} \mathbf{d}^{(k)} = 0$$

and hence

$$\alpha_k = \arg \min f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}).$$

In fact, an even stronger condition holds, as given by the following lemma.

Lemma 395. *In the conjugate direction algorithm,*

$$\mathbf{g}^{(k+1)T} \mathbf{d}^{(i)} = 0$$

for all $k, 0 \leq k \leq n - 1$, and $0 \leq i \leq k$.

Proof. Note that

$$\mathbf{Q}(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}) = \mathbf{Qx}^{(k+1)} - \mathbf{b} - (\mathbf{Qx}^{(k)} - \mathbf{b}) = \mathbf{g}^{(k+1)} - \mathbf{g}^{(k)},$$

because $\mathbf{g}^{(k)} = \mathbf{Qx}^{(k)} - \mathbf{b}$. Thus,

$$\mathbf{g}^{(k+1)} = \mathbf{g}^{(k)} + \alpha_k \mathbf{Qd}^{(k)}.$$

We prove the lemma by induction. The result is true for $k = 0$ because $\mathbf{g}^{(1)T}\mathbf{d}^{(0)} = 0$, as shown before. We now show that if the result is true for $k - 1$ (i.e., $\mathbf{g}^{(k)T}\mathbf{d}^{(i)} = 0, i \leq k - 1$) then it is true for k (i.e., $\mathbf{g}^{(k+1)T}\mathbf{d}^{(i)} = 0, i \leq k$). Fix $k > 0$ and $0 \leq i < k$. By the induction hypothesis, $\mathbf{g}^{(k)T}\mathbf{d}^{(i)} = 0$. Because

$$\mathbf{g}^{(k+1)} = \mathbf{g}^{(k)} + \alpha_k \mathbf{Q}\mathbf{d}^{(k)},$$

and $\mathbf{d}^{(k)T}\mathbf{Q}\mathbf{d}^{(i)} = 0$ by \mathbf{Q} -conjugacy, we have

$$\mathbf{g}^{(k+1)T}\mathbf{d}^{(i)} = \mathbf{g}^{(k)T}\mathbf{d}^{(i)} + \alpha_k \mathbf{d}^{(k)T}\mathbf{Q}\mathbf{d}^{(i)} = 0.$$

It remains to be shown that

$$\mathbf{g}^{(k+1)T}\mathbf{d}^{(k)} = 0.$$

Indeed,

$$\begin{aligned} \mathbf{g}^{(k+1)T}\mathbf{d}^{(k)} &= (\mathbf{Q}\mathbf{x}^{(k+1)} - \mathbf{b})^T\mathbf{d}^{(k)} \\ &= \left(\mathbf{x}^{(k)} - \left(\frac{\mathbf{g}^{(k)T}\mathbf{d}^{(k)}}{\mathbf{d}^{(k)T}\mathbf{Q}\mathbf{d}^{(k)}} \right) \mathbf{d}^{(k)} \right)^T \mathbf{Q}\mathbf{d}^{(k)} - \mathbf{b}^T\mathbf{d}^{(k)} \\ &= (\mathbf{Q}\mathbf{x}^{(k)} - \mathbf{b})^T\mathbf{d}^{(k)} - \mathbf{g}^{(k)T}\mathbf{d}^{(k)} \\ &= 0, \end{aligned}$$

because $\mathbf{Q}\mathbf{x}^{(k)} - \mathbf{b} = \mathbf{g}^{(k)}$.

Therefore, by induction, for all $0 \leq k \leq n - 1$ and $0 \leq i \leq k$,

$$\mathbf{g}^{(k+1)T}\mathbf{d}^{(i)} = 0.$$

□

By the above lemma, we see that $\mathbf{g}^{(k+1)}$ is orthogonal to any vector from the subspace spanned by $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(k-1)}$. Figure 5.20 illustrates this statement. The above lemma can be used to show an interesting optimal property of the conjugate direction algorithm. Specifically, we now show that not only does $f(\mathbf{x}^{(k+1)})$ satisfy $f(\mathbf{x}^{(k+1)}) = \min_{\alpha} f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)})$, as indicated before, but also,

$$f(\mathbf{x}^{(k+1)}) = \min_{\alpha_0, \dots, \alpha_k} f\left(\mathbf{x}^{(0)} + \sum_{i=1}^k \alpha_i \mathbf{d}^{(i)}\right).$$

In other words, $f(\mathbf{x}^{(k+1)}) = \min_{x \in \mathcal{V}_k} f(x)$, where $\mathcal{V}_k = \mathbf{x}^{(0)} + \text{span}[\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(k)}]$. As k increases, the subspace $\text{span}[\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(k)}]$ “expands,” and will eventually fill the

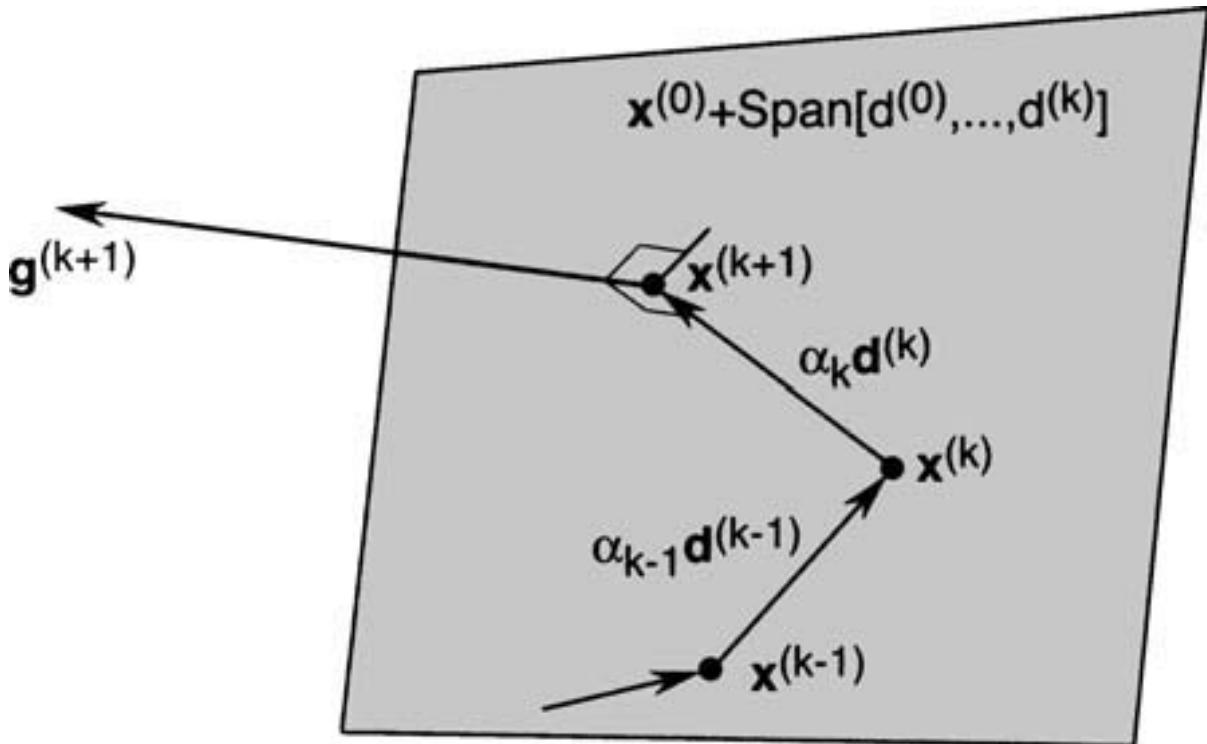


图 5.20: Illustration of Lemma 391.

whole of \mathbb{R}^n (provided the vectors $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots$, are linearly independent). Therefore, for some sufficiently large k , \mathbf{x}^* will lie in \mathcal{V}_k . For this reason, the above result is sometimes called the “expanding subspace” theorem.

To prove the expanding subspace theorem, define the matrix $\mathbf{D}^{(k)}$ by

$$\mathbf{D}^{(k)} = [\mathbf{d}^{(0)}, \dots, \mathbf{d}^{(k)}],$$

that is, $\mathbf{d}^{(i)}$ is the i th column of $\mathbf{D}^{(k)}$. Note that $\mathbf{x}^{(0)} + \mathcal{R}(\mathbf{D}^{(k)}) = \mathcal{V}_k$. Also,

$$\begin{aligned}\mathbf{x}^{(k+1)} &= \mathbf{x}^{(0)} + \sum_{i=0}^k \alpha_i \mathbf{d}^{(i)} \\ &= \mathbf{x}^{(0)} + \mathbf{D}^{(k)} \boldsymbol{\alpha},\end{aligned}$$

where $\boldsymbol{\alpha} = [\alpha_0, \dots, \alpha_k]^T$. Hence,

$$\mathbf{x}^{(k+1)} \in \mathbf{x}^{(0)} + \mathcal{R}(\mathbf{D}^{(k)}) = \mathcal{V}_k.$$

Now, consider any vector $\mathbf{x} \in \mathcal{V}_k$. There exists a vector \mathbf{a} such that $\mathbf{x} = \mathbf{x}^{(0)} + \mathbf{D}^{(k)} \boldsymbol{\alpha}$. Let $\phi_k(\boldsymbol{\alpha}) = f(\mathbf{x}^{(0)} + \mathbf{D}^{(k)} \boldsymbol{\alpha})$. Note that ϕ_k is a quadratic function and has a unique minimizer that satisfies the FONC. By the chain rule,

$$D\phi_k(\boldsymbol{\alpha}) = \nabla f(\mathbf{x}^{(0)} + \mathbf{D}^{(k)} \boldsymbol{\alpha})^T \mathbf{D}^{(k)}.$$

Therefore,

$$\begin{aligned} D\phi_k(\boldsymbol{\alpha}) &= \nabla f(\mathbf{x}^{(0)} + \mathbf{D}^{(k)}\boldsymbol{\alpha})^T \mathbf{D}^{(k)} \\ &= \nabla f(\mathbf{x}^{(k+1)})^T \mathbf{D}^{(k)} \\ &= \mathbf{g}^{(k+1)T} \mathbf{D}^{(k)}. \end{aligned}$$

By Lemma 395, $\mathbf{g}^{(k+1)T} \mathbf{D}^{(k)} = \mathbf{0}$. Therefore, α satisfies the FONC for the quadratic function ϕ_k , and hence \mathbf{a} is the minimizer of ϕ_k ; that is,

$$f(\mathbf{x}^{(k)}) = \min_{\boldsymbol{\alpha}} f(\mathbf{x}^{(0)} + \mathbf{D}^{(k)}\boldsymbol{\alpha}) = \min_{x \in \mathcal{V}_k} f(\mathbf{x}),$$

and the proof of our result is completed.

The conjugate direction algorithm is very effective. However, to use the algorithm, we need to specify the \mathbf{Q} -conjugate directions. Fortunately there is a way to generate \mathbf{Q} -conjugate directions as we perform iterations. In the next section, we discuss an algorithm that incorporates the generation of \mathbf{Q} -conjugate directions.

5.4.3 The Conjugate Gradient Algorithm

The conjugate gradient algorithm does not use prespecified conjugate directions, but instead computes the directions as the algorithm progresses. At each stage of the algorithm, the direction is calculated as a linear combination of the previous direction and the current gradient, in such a way that all the directions are mutually \mathbf{Q} conjugate – hence the name conjugate gradient algorithm. This calculation exploits the fact that for a quadratic function of n variables, we can locate the function minimizer by performing n searches along mutually conjugate directions. As before, we consider the quadratic function

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b}, \mathbf{x} \in \mathbb{R}^n,$$

where $\mathbf{Q} = \mathbf{Q}^T \succ \mathbf{0}$. Our first search direction from an initial point $\mathbf{x}^{(0)}$ is in the direction of steepest descent; that is

$$\mathbf{d}^{(0)} = -\mathbf{g}^{(0)}.$$

Thus,

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \alpha_0 \mathbf{d}^{(0)},$$

where

$$\alpha_0 = \arg \min_{\alpha > 0} f(\mathbf{x}^{(0)} + \alpha \mathbf{d}^{(0)}) = -\frac{\mathbf{g}^{(0)T} \mathbf{d}^{(0)}}{\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(0)}}.$$

In the next stage, we search in a direction $\mathbf{d}^{(1)}$ that is \mathbf{Q} -conjugate to $\mathbf{d}^{(0)}$. In general, at the $(k + 1)$ st step, we choose $\mathbf{d}^{(k+1)}$ to be a linear combination of $\mathbf{g}^{(k+1)}$ and $\mathbf{d}^{(k)}$. Specifically, we choose

$$\mathbf{d}^{(k+1)} = -\mathbf{g}^{(k+1)} + \beta_k \mathbf{d}^{(k)}, k = 0, 1, 2, \dots$$

The coefficients $\beta_k, k = 1, 2, \dots$ are chosen in such a way that $\mathbf{d}^{(k+1)}$ is \mathbf{Q} -conjugate to $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(k)}$. This is accomplished by choosing β_k to be

$$\beta_k = \frac{\mathbf{g}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(k)}}{\mathbf{d}^{(k)T} \mathbf{Q} \mathbf{d}^{(k)}}.$$

The conjugate gradient algorithm is summarized below.

1. Set $k := 0$; select the initial point $\mathbf{x}^{(0)}$.
2. $\mathbf{g}^{(0)} = \nabla f(\mathbf{x}^{(0)})$. If $\mathbf{g}^{(0)} = \mathbf{0}$, stop, else set $\mathbf{d}^{(0)} = -\mathbf{g}^{(0)}$.
3. $\alpha_k = -\frac{\mathbf{g}^{(k)T} \mathbf{d}^{(k)}}{\mathbf{d}^{(k)T} \mathbf{Q} \mathbf{d}^{(k)}}$.
4. $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$.
5. $\mathbf{g}^{(k+1)} = \nabla f(\mathbf{x}^{(k+1)})$. If $\mathbf{g}^{(k+1)} = \mathbf{0}$, stop.
6. $\beta_k = \frac{\mathbf{g}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(k)}}{\mathbf{d}^{(k)T} \mathbf{Q} \mathbf{d}^{(k)}}$.
7. $\mathbf{d}^{(k+1)} = -\mathbf{g}^{(k+1)} + \beta_k \mathbf{d}^{(k)}$.
8. Set $k := k + 1$; go to step 3.

Proposition 396. *In the conjugate gradient algorithm, the directions $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(n-1)}$ are \mathbf{Q} -conjugate.*

Proof. We use induction. We first show $\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(1)} = 0$. To this end, we write

$$\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(1)} = \mathbf{d}^{(0)T} \mathbf{Q} (-\mathbf{g}^{(1)} + \beta_0 \mathbf{d}^{(0)}).$$

Substituting for

$$\beta_0 = \frac{\mathbf{g}^{(1)T} \mathbf{Q} \mathbf{d}^{(0)}}{\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(0)}}$$

in the above equation, we see that $\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(1)} = 0$. We now assume that $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(k)}, k < n - 1$, are \mathbf{Q} -conjugate directions. From Lemma 395, we have $\mathbf{g}^{(k+1)T} \mathbf{d}^{(j)} = 0, j = 0, 1, \dots, k$. Thus, $\mathbf{g}^{(k+1)}$ is orthogonal to each of the directions $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(k)}$. We now show that

$$\mathbf{g}^{(k+1)T} \mathbf{g}^{(j)} = 0, j = 0, 1, \dots, k.$$

Fix $j \in \{0, \dots, k\}$. We have

$$\mathbf{d}^{(j)} = -\mathbf{g}^{(j)} + \beta_{j-1} \mathbf{d}^{(j-1)}.$$

Substituting this equation into the previous one yields

$$\mathbf{g}^{(k+1)T} \mathbf{d}^{(j)} = 0 = -\mathbf{g}^{(k+1)T} \mathbf{g}^{(j)} + \beta_{j-1} \mathbf{g}^{(k+1)T} \mathbf{d}^{(j-1)}.$$

Because $\mathbf{g}^{(k+1)T} \mathbf{d}^{(j-1)} = 0$, it follows that $\mathbf{g}^{(k+1)T} \mathbf{g}^{(j)} = 0$.

We are now ready to show that $\mathbf{d}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(j)} = 0, j = 0, \dots, k$. We have

$$\mathbf{d}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(j)} = (-\mathbf{g}^{(k+1)T} + \beta_k \mathbf{d}^{(k)})^T \mathbf{Q} \mathbf{d}^{(j)}.$$

If $j < k$, then $\mathbf{d}^{(k)T} \mathbf{Q} \mathbf{d}^{(j)} = 0$, by virtue of the induction hypothesis. Hence, we have

$$\mathbf{d}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(j)} = -\mathbf{g}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(j)}.$$

But $\mathbf{g}^{(j+1)} = \mathbf{g}^{(j)} + \alpha_j \mathbf{Q} \mathbf{d}^{(j)}$. Because $\mathbf{g}^{(k+1)T} \mathbf{g}^{(i)} = 0, i = 0, \dots, k$,

$$\mathbf{d}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(j)} = -\mathbf{g}^{(k+1)T} \frac{(\mathbf{g}^{(j+1)} - \mathbf{g}^{(j)})}{\alpha_j} = 0.$$

Thus,

$$\mathbf{d}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(j)} = 0, \quad j = 0, \dots, k-1.$$

It remains to be shown that $\mathbf{d}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(k)} = 0$. We have

$$\mathbf{d}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(k)} = (-\mathbf{g}^{(k+1)T} + \beta_k \mathbf{d}^{(k)})^T \mathbf{Q} \mathbf{d}^{(k)}.$$

Using the expression for β_k , we get $\mathbf{d}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(k)} = 0$, which completes the proof. \square

Example 397. Consider the quadratic function

$$f(x_1, x_2, x_3) = \frac{3}{2}x_1^2 + 2x_2^2 + \frac{3}{2}x_3^2 + x_1x_3 + 2x_2x_3 - 3x_1 - x_3.$$

We find the minimizer using the conjugate gradient algorithm, using the starting point $\mathbf{x}^{(0)} = [0, 0, 0]^T$. We can represent f as

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b},$$

where

$$\mathbf{Q} = \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix}.$$

We have

$$\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x}) = \mathbf{Q}\mathbf{x} - \mathbf{b} = [3x_1 + x_3 - 3, 4x_2 + 2x_3, x_1 + 2x_2 + 3x_3 - 1]^T.$$

Hence,

$$\begin{aligned}\mathbf{g}^{(0)} &= [-3, 0, -1]^T, \\ \mathbf{d}^{(0)} &= -\mathbf{g}^{(0)}, \\ \alpha_0 &= -\frac{\mathbf{g}^{(0)T}\mathbf{d}^{(0)}}{\mathbf{d}^{(0)T}\mathbf{Q}\mathbf{d}^{(0)}} = \frac{10}{36} = 0.2778,\end{aligned}$$

and

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \alpha_0 \mathbf{d}^{(0)} = [0.8333, 0, 0.2778]^T.$$

The next stage yields

$$\begin{aligned}\mathbf{g}^{(1)} &= \nabla f(\mathbf{x}^{(1)}) = [-0.2222, 0.5556, 0.6667]^T, \\ \beta_0 &= \frac{\mathbf{g}^{(1)T}\mathbf{Q}\mathbf{d}^{(0)}}{\mathbf{d}^{(0)T}\mathbf{Q}\mathbf{d}^{(0)}} = 0.08025.\end{aligned}$$

We can now compute

$$\mathbf{d}^{(1)} = -\mathbf{g}^{(1)} + \beta_0 \mathbf{d}^{(0)} = [0.4630, -0.5556, -0.5864]^T.$$

Hence,

$$\alpha_1 = -\frac{\mathbf{g}^{(1)T}\mathbf{d}^{(1)}}{\mathbf{d}^{(1)T}\mathbf{Q}\mathbf{d}^{(1)}} = 0.2187,$$

and

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \alpha_1 \mathbf{d}^{(1)} = [0.9346, -0.1215, 0.1495]^T.$$

To perform the third iteration, we compute

$$\begin{aligned}\mathbf{g}^{(2)} &= \nabla f(\mathbf{x}^{(2)}) = [-0.04673, -0.1869, 0.1402]^T, \\ \beta_1 &= \frac{\mathbf{g}^{(2)T}\mathbf{Q}\mathbf{d}^{(1)}}{\mathbf{d}^{(1)T}\mathbf{Q}\mathbf{d}^{(1)}} = 0.07075, \\ \mathbf{d}^{(2)} &= -\mathbf{g}^{(2)} + \beta_1 \mathbf{d}^{(1)} = [0.07948, 0.1476, -0.1817]^T.\end{aligned}$$

Hence,

$$\alpha_2 = -\frac{\mathbf{g}^{(2)T}\mathbf{d}^{(2)}}{\mathbf{d}^{(2)T}\mathbf{Q}\mathbf{d}^{(2)}} = 0.8231,$$

and

$$\mathbf{x}^{(3)} = \mathbf{x}^{(2)} + \alpha_2 \mathbf{d}^{(2)} = [1.000, 0.000, 0.000]^T.$$

Note that

$$\mathbf{g}^{(3)} = \nabla f(\mathbf{x}^{(3)}) = \mathbf{0},$$

as expected, because f is a quadratic function of three variables. Hence $\mathbf{x}^* = \mathbf{x}^{(3)}$.

5.4.4 The Conjugate Gradient Algorithm for Non-Quadratic Problems

In Section 5.4.3, we showed that the conjugate gradient algorithm is a conjugate direction method, and therefore minimizes a positive definite quadratic function of n variables in n steps. The algorithm can be extended to general nonlinear functions by interpreting $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q}\mathbf{x} - \mathbf{x}^T \mathbf{b}$ as a second-order Taylor series approximation of the objective function. Near the solution such functions behave approximately as quadratics, as suggested by the Taylor series expansion. For a quadratic, the matrix \mathbf{Q} , the Hessian of the quadratic, is constant. However, for a general nonlinear function the Hessian is a matrix that has to be reevaluated at each iteration of the algorithm. This can be computationally very expensive. Thus, an efficient implementation of the conjugate gradient algorithm that eliminates the Hessian evaluation at each step is desirable.

Observe that \mathbf{Q} appears only in the computation of the scalars α_k and β_k . Because

$$\alpha_k = \arg \min_{\alpha > 0} f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}),$$

the closed form formula for α_k in the algorithm can be replaced by a numerical line search procedure. Therefore, we only need to concern ourselves with the formula for β_k . Fortunately, elimination of \mathbf{Q} from the formula is possible and results in algorithms that depend only on the function and gradient values at each iteration. We now discuss modifications of the conjugate gradient algorithm for a quadratic function for the case in which the Hessian is unknown but in which objective function values and gradients are available. The modifications are all based on algebraically manipulating the formula β_k in such a way that \mathbf{Q} is eliminated. We discuss three well-known modifications.

The *Hestenes-Stiefel* formula. Recall that

$$\beta_k = \frac{\mathbf{g}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(k)}}{\mathbf{d}^{(k)T} \mathbf{Q} \mathbf{d}^{(k)}}. \quad (5.40)$$

The Hestenes-Stiefel formula is based on replacing the term $\mathbf{Q}\mathbf{d}^{(k)}$ by the term $(\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)})/\alpha_k$. The two terms are equal in the quadratic case, as we now show. Now, $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$. Premultiplying both sides by \mathbf{Q} , and recognizing that $\mathbf{g}^{(k)} = \mathbf{Q}\mathbf{x}^{(k)} - \mathbf{b}$, we get $\mathbf{g}^{(k+1)} = \mathbf{g}^{(k)} + \alpha_k \mathbf{Q}\mathbf{d}^{(k)}$, which we can rewrite as $\mathbf{Q}\mathbf{d}^{(k)} = (\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)})/\alpha_k$. Substituting this into the original equation for β_k gives

$$\beta_k = \frac{\mathbf{g}^{(k+1)T} [\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}]}{\mathbf{d}^{(k)T} [\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}]},$$

which is called the Hestenes-Stiefel formula.

The Polak-Ribiere formula. Starting from the Hestenes-Stiefel formula, we multiply out the denominator to get

$$\beta_k = \frac{\mathbf{g}^{(k+1)T}[\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}]}{\mathbf{d}^{(k)T}\mathbf{g}^{(k+1)} - \mathbf{d}^{(k)T}\mathbf{g}^{(k)}}. \quad (5.41)$$

By Lemma 395, $\mathbf{d}^{(k)T}\mathbf{g}^{(k+1)} = 0$. Also, since $\mathbf{d}^{(k)} = -\mathbf{g}^{(k)} + \beta_{k-1}\mathbf{d}^{(k-1)}$, and premultiplying this by $\mathbf{g}^{(k)T}$, we get

$$\mathbf{g}^{(k)T}\mathbf{d}^{(k)} = -\mathbf{g}^{(k)T}\mathbf{g}^{(k)} + \beta_{k-1}\mathbf{g}^{(k)T}\mathbf{d}^{(k-1)} = -\mathbf{g}^{(k)T}\mathbf{g}^{(k)},$$

where once again we used Lemma 395. Hence, we get

$$\beta_k = \frac{\mathbf{g}^{(k+1)T}[\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}]}{\mathbf{g}^{(k)T}\mathbf{g}^{(k)}}.$$

This expression for β_k is known as the Polak-Ribiere formula.

The Fletcher-Reeves formula. Starting with the Polak-Ribiere formula, we multiply out the numerator to get

$$\beta_k = \frac{\mathbf{g}^{(k+1)T}\mathbf{g}^{(k+1)} - \mathbf{g}^{(k+1)T}\mathbf{g}^{(k)}}{\mathbf{g}^{(k)T}\mathbf{g}^{(k)}}.$$

We now use the fact that $\mathbf{g}^{(k+1)T}\mathbf{g}^{(k)} = 0$, which we get by using the equation

$$\mathbf{g}^{(k+1)T}\mathbf{d}^{(k)} = -\mathbf{g}^{(k+1)T}\mathbf{g}^{(k)} + \beta_{k-1}\mathbf{g}^{(k+1)T}\mathbf{d}^{(k-1)}$$

and applying Lemma 395. This leads to

$$\beta_k = \frac{\mathbf{g}^{(k+1)T}\mathbf{g}^{(k+1)}}{\mathbf{g}^{(k)T}\mathbf{g}^{(k)}},$$

which is called the Fletcher-Reeves formula.

The above formulas give us conjugate gradient algorithms that do not require explicit knowledge of the Hessian matrix \mathbf{Q} . All we need are the objective function and gradient values at each iteration. For the quadratic case, the three expressions for β_k are exactly equal. However, this is not the case for a general nonlinear objective function.

We need a few more slight modifications to apply the algorithm to general nonlinear functions in practice. First, as mentioned in our discussion of the steepest descent algorithm (Section 5.2.2), the termination criterion $\nabla f(\mathbf{x}^{(k+1)}) = \mathbf{0}$ is not practical. A suitable practical stopping criterion, such as those discussed in Section 8.2, needs to be used.

For nonquadratic problems, the algorithm will not usually converge in n steps, and as the algorithm progresses, the “ \mathbf{Q} -conjugacy” of the direction vectors will tend to

deteriorate. Thus, a common practice is to reinitialize the direction vector to the negative gradient after every few iterations (e.g., n or $n + 1$), and continue until the algorithm satisfies the stopping criterion.

A very important issue in minimization problems of nonquadratic functions is the line search. The purpose of the line search is to minimize $\phi_k(\alpha) = f(\mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)})$ with respect to $\alpha \geq 0$. A typical approach is to bracket or box in the minimizer and then estimate it. The accuracy of the line search is a critical factor in the performance of the conjugate gradient algorithm. If the line search is known to be inaccurate, the Hestenes-Stiefel formula for β_k is recommended.

In general, the choice of which formula for β_k to use depends on the objective function. For example, the Polak-Ribiere formula is known to perform far better than the Fletcher-Reeves formula in some cases but not in others. In fact, there are cases in which the $\mathbf{g}^{(k)}$, $k = 1, 2, \dots$, are bounded away from zero when the Polak-Ribiere formula is used. In the study by Powell in [107], a global convergence analysis suggests that the Fletcher-Reeves formula for β_k is superior. Powell further suggests another formula for β_k .

$$\beta_k = \max \left(0, \frac{\mathbf{g}^{(k+1)T}[\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}]}{\mathbf{g}^{(k)T}\mathbf{g}^{(k)}} \right).$$

For general results on the convergence of conjugate gradient methods, we refer the reader to [139].

5.4.5 Exercises

Exercise 398. Let \mathbf{Q} be a real symmetric positive definite $n \times n$ matrix. Given an arbitrary set of linearly independent vectors $\{\mathbf{p}^{(0)}, \dots, \mathbf{p}^{(n-1)}\}$ in \mathbb{R}^n , the Gram-Schmidt procedure generates a set of vectors $\{\mathbf{d}^{(0)}, \dots, \mathbf{d}^{(n-1)}\}$ as follows:

$$\begin{aligned} \mathbf{d}^{(0)} &= \mathbf{p}^{(0)} \\ \mathbf{d}^{(k+1)} &= \mathbf{p}^{(k+1)} - \sum_{i=0}^k \frac{\mathbf{p}^{(k+1)T}\mathbf{Q}\mathbf{d}^{(i)}}{\mathbf{d}^{(i)T}\mathbf{Q}\mathbf{d}^{(i)}}\mathbf{d}^{(i)}. \end{aligned} \tag{5.42}$$

Show that the vectors $\mathbf{d}^{(0)}, \dots, \mathbf{d}^{(n-1)}$ are \mathbf{Q} -conjugate.

Exercise 399. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be the quadratic function

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} - \mathbf{x}^T\mathbf{b},$$

where $\mathbf{Q} = \mathbf{Q}^T \succ \mathbf{0}$. Given a set of directions $\{\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots\} \subset \mathbb{R}^n$, consider the algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)},$$

where α_k is the step size. Suppose that $\mathbf{g}^{(k+1)T}\mathbf{d}^{(i)} = 0$ for all $k = 0, \dots, n-1$ and $i = 0, \dots, k$, where $\mathbf{g}^{(k+1)} = \nabla f(\mathbf{x}^{(k+1)})$. Show that if $\mathbf{g}^{(k)T}\mathbf{d}^{(k)} \neq 0$ for all $k = 0, \dots, n-1$, then $\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(n-1)}$ are \mathbf{Q} -conjugate.

Exercise 400. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be given by $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} - \mathbf{x}^T\mathbf{b}$, where $\mathbf{b} \in \mathbb{R}^n$, and \mathbf{Q} is a real symmetric positive definite $n \times n$ matrix. Show that in the conjugate gradient method for this f , $\mathbf{d}^{(k)T}\mathbf{Q}\mathbf{d}^{(k)} = -\mathbf{d}^{(k)T}\mathbf{Q}\mathbf{g}^{(k)}$.

Exercise 401. Let \mathbf{Q} be a real $n \times n$ symmetric matrix.

1. Show that there exists a \mathbf{Q} -conjugate set $\{\mathbf{d}^{(1)}, \dots, \mathbf{d}^{(n)}\}$ such that each $\mathbf{d}^{(i)} (i = 1, 2, \dots, n)$ is an eigenvector of \mathbf{Q} . Hint: Use the fact that for any real symmetric $n \times n$ matrix, there exists a set $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ of its eigenvectors such that $\mathbf{v}_i^T\mathbf{v}_j = 0$ for all $i, j = 1, \dots, n, i \neq j$.
2. Suppose that \mathbf{Q} is positive definite. Show that if $\{\mathbf{d}^{(1)}, \dots, \mathbf{d}^{(n)}\}$ is a \mathbf{Q} -conjugate set that is also orthogonal (i.e., $\mathbf{d}^{(i)T}\mathbf{d}^{(j)} = 0$ for all $i, j = 1, \dots, n, i \neq j$), and $\mathbf{d}^{(i)} \neq 0, i = 1, 2, \dots, n$, then each $\mathbf{d}^{(i)}, i = 1, \dots, n$, is an eigenvector of \mathbf{Q} .

Exercise 402. Consider the following algorithm for minimizing a function f :

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)},$$

where $\alpha_k = \arg \min_{\alpha} f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)})$. Let $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)})$ (as usual).

Suppose f is quadratic with Hessian \mathbf{Q} . We choose $\mathbf{d}^{(k+1)} = \gamma_k \mathbf{g}^{(k+1)} + \mathbf{d}^{(k)}$, and we wish the directions $\mathbf{d}^{(k)}$ and $\mathbf{d}^{(k+1)}$ to be \mathbf{Q} -conjugate. Find a formula for γ_k in terms of $\mathbf{d}^{(k)}, \mathbf{g}^{(k+1)}$, and \mathbf{Q} .

Exercise 403. Consider the algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)},$$

with $\alpha_k \in \mathbb{R}$ scalar and $\mathbf{x}^{(0)} = \mathbf{0}$, applied to the quadratic function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ given by

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} - \mathbf{x}^T\mathbf{b},$$

where $\mathbf{Q} \succ 0$. As usual, write $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)})$. Suppose that the search directions are generated according to

$$\mathbf{d}^{(k+1)} = a_k \mathbf{g}^{(k+1)} + b_k \mathbf{d}^{(k)},$$

where a_k and b_k are real constants, and by convention we take $\mathbf{d}^{(-1)} = \mathbf{0}$.

1. Define the subspace $\mathcal{V}_k = \text{span}[\mathbf{b}, \mathbf{Q}\mathbf{b}, \dots, \mathbf{Q}^{(k-1)}\mathbf{b}]$ (called the Krylov subspace of order k). Show that $\mathbf{d}^{(k)} \in \mathcal{V}_{k+1}$ and $\mathbf{x}^{(k)} \in \mathcal{V}_k$. Hint: Use induction. Note that $\mathcal{V}_0 = \{\mathbf{0}\}$ and $\mathcal{V}_0 = \text{span}[\mathbf{b}]$.
2. In light of part 1, what can you say about the “optimality” of the conjugate gradient algorithm with respect to the Krylov subspace?

Exercise 404. Consider the quadratic function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ given by

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b},$$

where $\mathbf{Q} = \mathbf{Q}^T \succ 0$. Let $\mathbf{D} \in \mathbb{R}^{n \times r}$ be of rank r , and $x_0 \in \mathbb{R}^n$. Define the function $\phi : \mathbb{R}^r \rightarrow \mathbb{R}$ by

$$\phi(\mathbf{a}) = f(\mathbf{x}_0 + \mathbf{D}\mathbf{a}).$$

Show that ϕ is a quadratic function with a positive definite quadratic term.

Exercise 405. Consider a conjugate gradient algorithm applied to a quadratic function.

1. Show that the gradients associated with the algorithm are mutually orthogonal. Specifically show that $\mathbf{g}^{(k+1)T} \mathbf{g}^{(i)} = 0$ for all $0 \leq k \leq n-1$ and $0 \leq i \leq k$. Hint: Write $\mathbf{g}^{(i)}$ in terms of $\mathbf{d}^{(i)}$ and $\mathbf{d}^{(i-1)}$.
2. Show that the gradients associated with the algorithm are \mathbf{Q} -conjugate if separated by at least two iterations. Specifically, show that $\mathbf{g}^{(k+1)T} \mathbf{Q} \mathbf{g}^{(i)} = 0$ for all $0 \leq k \leq n-1$ and $0 \leq i \leq k-1$.

Exercise 406. Represent the function

$$f(x_1, x_2) = \frac{5}{2}x_1^2 + x_2^2 - 3x_1x_2 - x_2 - 7$$

in the form $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b} + c$. Then use the conjugate gradient algorithm to construct a vector $\mathbf{d}^{(1)}$ that is \mathbf{Q} -conjugate with $\mathbf{d}^0 = \nabla f(\mathbf{x}^0)$, where $\mathbf{x}^0 = \mathbf{0}$.

Exercise 407. Let $f(\mathbf{x}), \mathbf{x} = [x_1, x_2]^T \in \mathbb{R}^2$, be given by

$$f(\mathbf{x}) = \frac{5}{2}x_1^2 + \frac{1}{2}x_2^2 + 2x_1x_2 - 3x_1 - x_2.$$

1. Express $f(\mathbf{x})$ in the form of $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b}$.
2. Find the minimizer of f using the conjugate gradient algorithm. Use a starting point of $\mathbf{x}^0 = [0, 0]^T$.

3. Calculate the minimizer of f analytically from \mathbf{Q} and \mathbf{b} , and check it with your answer in part 2.

Exercise 408. Write a MATLAB program to implement the conjugate gradient algorithm for general functions. Use the secant method for the line search. Test the different formulas for β_k on Rosenbrock's function (2.2) with an initial condition $\mathbf{x}^{(0)} = [-2, 2]^T$. For this exercise, reinitialize the update direction to the negative gradient every six iterations.

(Taken from Chapter 6 of [109]).

Exercise 409. Determine whether the following vectors serve as conjugate directions for minimizing the function $f(\mathbf{x}) = 2x_1^2 + 16x_2^2 - 2x_1x_2 - x_1 - 6x_2 - 5$.

$$(a) \mathbf{s}_1 = [15, -1]^T, \mathbf{s}_2 = [1, 1]^T.$$

$$(b) \mathbf{s}_1 = [-1, 15]^T, \mathbf{s}_2 = [1, 1]^T.$$

Exercise 410. Find the condition number of each matrix.

$$(a) \mathbf{A} = \begin{bmatrix} 1 & 2 \\ 1.0001 & 2 \end{bmatrix}.$$

$$(b) \mathbf{B} = \begin{bmatrix} 3.9 & 1.6 \\ 6.8 & 2.9 \end{bmatrix}.$$

Exercise 411. Perform two iterations of the Newton's method to minimize the function

$$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

from the starting point $\mathbf{x}^{(0)} = [-1.2, 1.0]^T$.

Exercise 412. Perform two iterations of the BFGS method to minimize the function given in Exercise 411 from the indicated starting point.

Exercise 413. Show that the property of quadratic convergence of conjugate directions is independent of the order in which the one-dimensional minimizations are performed by considering the minimization of

$$f(\mathbf{x}) = 6x_1^2 + 2x_2^2 - 6x_1x_2 - x_1 - 2x_2$$

using the conjugate directions $\mathbf{s}_1 = [1, 2]^T$ and $\mathbf{s}_2 = [1, 0]^T$ and the starting point $\mathbf{x}^{(0)} = [0, 0]^T$.

Exercise 414. Verify whether the following search directions are \mathbf{A} -conjugate while minimizing the function

$$f(\mathbf{x}) = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2.$$

$$(a) \mathbf{s}_1 = [-1, 1]^T, \mathbf{s}_2 = [1, 0]^T.$$

$$(b) \mathbf{s}_1 = [-1, 1]^T, \mathbf{s}_2 = [0, 1]^T.$$

5.5 Quasi-Newton Methods

(Taken from Chapter 11 of [30])

5.5.1 Introduction

Newton's method is one of the more successful algorithms for optimization. If it converges, it has a quadratic order of convergence. However, as pointed out before, for a general nonlinear objective function, convergence to a solution cannot be guaranteed from an arbitrary initial point $\mathbf{x}^{(0)}$. In general, if the initial point is not sufficiently close to the solution, then the algorithm may not possess the descent property (i.e. $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$ for some k).

Recall that the idea behind Newton's method is to locally approximate the function f being minimized, at every iteration, by a quadratic function. The minimizer for the quadratic approximation is used as the starting point for the next iteration. This leads to Newton's recursive algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mathbf{F}(\mathbf{x}^{(k)})^{-1}\mathbf{g}^{(k)}.$$

We may try to guarantee that the algorithm has the descent property by modifying the original algorithm as follows:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{F}(\mathbf{x}^{(k)})^{-1}\mathbf{g}^{(k)},$$

where α_k is chosen to ensure that

$$f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)}).$$

For example, we may choose $\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} - \alpha \mathbf{F}(\mathbf{x}^{(k)})^{-1}\mathbf{g}^{(k)})$ (see Theorem 375). We can then determine an appropriate value of α_k by performing a line search in the direction $-\mathbf{F}(\mathbf{x}^{(k)})^{-1}\mathbf{g}^{(k)}$. Note that although the line search is simply the minimization of the real variable function $\phi_k(\alpha) = f(\mathbf{x}^{(k)} - \alpha \mathbf{F}(\mathbf{x}^{(k)})^{-1}\mathbf{g}^{(k)})$, it is not a trivial problem to solve.

A computational drawback of Newton's method is the need to evaluate $\mathbf{F}(\mathbf{x}^{(k)})$ and solve the equation $\mathbf{F}(\mathbf{x}^{(k)})\mathbf{d}^{(k)} = -\mathbf{g}^{(k)}$ (i.e., compute $\mathbf{d}^{(k)} = -\mathbf{F}(\mathbf{x}^{(k)})^{-1}\mathbf{g}^{(k)}$). To avoid the computation of $\mathbf{F}(\mathbf{x}^{(k)})^{-1}$, the quasi-Newton methods use an approximation to $\mathbf{F}(\mathbf{x}^{(k)})^{-1}$ in place of the true inverse. This approximation is updated at every stage so that it exhibits at least some properties of $\mathbf{F}(\mathbf{x}^{(k)})^{-1}$. To get some idea about the properties that an approximation to $\mathbf{F}(\mathbf{x}^{(k)})^{-1}$ should satisfy, consider the formula

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha \mathbf{H}_k \mathbf{g}^{(k)},$$

where \mathbf{H}_k is an $n \times n$ real matrix, and α is a positive search parameter. Expanding f about $\mathbf{x}^{(k)}$ yields

$$\begin{aligned} f(\mathbf{x}^{(k+1)}) &= f(\mathbf{x}^{(k)}) + \mathbf{g}^{(k)T}(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}) + o(\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|) \\ &= f(\mathbf{x}^{(k)}) - \alpha \mathbf{g}^{(k)T} \mathbf{H}_k \mathbf{g}^{(k)} + o(\|\mathbf{H}_k \mathbf{g}^{(k)}\|). \end{aligned}$$

As α tends to zero, the second term on the right-hand side of the above equation dominates the third. Thus, to guarantee a decrease in f for small α , we have to have

$$\mathbf{g}^{(k)T} \mathbf{H}_k \mathbf{g}^{(k)} > 0.$$

A simple way to ensure this is to require that \mathbf{H}_k be positive definite. We have proved the following result.

Proposition 415. *Let $f \in \mathcal{C}^1$, $\mathbf{x}^{(k)} \in \mathbb{R}^n$, $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)}) \neq \mathbf{0}$, and \mathbf{H}_k an $n \times n$ real symmetric positive definite matrix. If we set $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{H}_k \mathbf{g}^{(k)}$, where $\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} - \alpha \mathbf{H}_k \mathbf{g}^{(k)})$, then $\alpha_k > 0$, and $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$.*

In constructing an approximation to the inverse of the Hessian matrix, we should use only the objective function and gradient values. Thus, if we can find a suitable method of choosing \mathbf{H}_k , the iteration may be carried out without any evaluation of the Hessian and without the solution of any set of linear equations.

5.5.2 Approximating the Inverse Hessian

Let $\mathbf{H}_0, \mathbf{H}_1, \mathbf{H}_2, \dots$ be successive approximations of the inverse $\mathbf{F}(\mathbf{x}^{(k)})^{-1}$ of the Hessian. We now derive a condition that the approximations should satisfy, which forms the starting point for our subsequent discussion of quasi-Newton algorithms. To begin, suppose first that the Hessian matrix $\mathbf{F}(\mathbf{x})$ of the objective function f is constant and independent of \mathbf{x} . In other words, the objective function is quadratic, with Hessian $\mathbf{F}(\mathbf{x}) = \mathbf{Q}$ for all \mathbf{x} , where $\mathbf{Q} = \mathbf{Q}^T$. Then,

$$\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)} = \mathbf{Q}(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}).$$

Let

$$\Delta \mathbf{g}^{(k)} \triangleq \mathbf{g}^{(k+1)} - \mathbf{g}^{(k)},$$

and

$$\Delta \mathbf{x}^{(k)} \triangleq \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}.$$

Then, we may write

$$\Delta \mathbf{g}^{(k)} = \mathbf{Q} \Delta \mathbf{x}^{(k)}.$$

We start with a real symmetric positive definite matrix \mathbf{H}_0 . Note that given k , the matrix \mathbf{Q}^{-1} satisfies

$$\mathbf{Q}^{-1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, \quad 0 \leq i \leq k.$$

Therefore, we also impose the requirement that the approximation \mathbf{H}_{k+1} of the Hessian satisfy

$$\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, \quad 0 \leq i \leq k.$$

If n steps are involved, then moving in n directions $\Delta \mathbf{x}^{(0)}, \Delta \mathbf{x}^{(1)}, \dots, \Delta \mathbf{x}^{(n-1)}$ yields

$$\begin{aligned} \mathbf{H}_n \Delta \mathbf{g}^{(0)} &= \Delta \mathbf{x}^{(0)} \\ \mathbf{H}_n \Delta \mathbf{g}^{(1)} &= \Delta \mathbf{x}^{(1)} \\ &\vdots \\ \mathbf{H}_n \Delta \mathbf{g}^{(n-1)} &= \Delta \mathbf{x}^{(n-1)}. \end{aligned}$$

The above set of equations can be represented as

$$\mathbf{H}_n [\Delta \mathbf{g}^{(0)}, \Delta \mathbf{g}^{(1)}, \dots, \Delta \mathbf{g}^{(n-1)}] = [\Delta \mathbf{x}^{(0)}, \Delta \mathbf{x}^{(1)}, \dots, \Delta \mathbf{x}^{(n-1)}].$$

Note that \mathbf{Q} satisfies

$$\mathbf{Q} [\Delta \mathbf{x}^{(0)}, \Delta \mathbf{x}^{(1)}, \dots, \Delta \mathbf{x}^{(n-1)}] = [\Delta \mathbf{g}^{(0)}, \Delta \mathbf{g}^{(1)}, \dots, \Delta \mathbf{g}^{(n-1)}]$$

and

$$\mathbf{Q}^{-1} [\Delta \mathbf{g}^{(0)}, \Delta \mathbf{g}^{(1)}, \dots, \Delta \mathbf{g}^{(n-1)}] = [\Delta \mathbf{x}^{(0)}, \Delta \mathbf{x}^{(1)}, \dots, \Delta \mathbf{x}^{(n-1)}].$$

Therefore, if $[\Delta \mathbf{g}^{(0)}, \Delta \mathbf{g}^{(1)}, \dots, \Delta \mathbf{g}^{(n-1)}]$ is nonsingular, then \mathbf{Q}^{-1} is determined uniquely after n steps, via

$$\mathbf{Q}^{-1} = \mathbf{H}_n = [\Delta \mathbf{x}^{(0)}, \Delta \mathbf{x}^{(1)}, \dots, \Delta \mathbf{x}^{(n-1)}] [\Delta \mathbf{g}^{(0)}, \Delta \mathbf{g}^{(1)}, \dots, \Delta \mathbf{g}^{(n-1)}]^{-1}.$$

As a consequence, we conclude that if \mathbf{H}_n satisfies the equations $\mathbf{H}_n \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, 0 \leq i \leq n-1$, then the algorithm $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{H}_k \mathbf{g}^{(k)}$, $\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} - \alpha \mathbf{H}_k \mathbf{g}^{(k)})$, is guaranteed to solve problems with quadratic objective functions in $n+1$ steps, because

the update $\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - \alpha_n \mathbf{H}_n \mathbf{g}^{(n)}$ is equivalent to Newton's algorithm. In fact, as we shall see below (Theorem 416), such algorithms solve quadratic problems of n variables in at most n steps.

The above considerations illustrate the basic idea behind the quasi-Newton methods. Specifically, quasi-Newton algorithms have the form

$$\begin{aligned}\mathbf{d}^{(k)} &= -\mathbf{H}_k \mathbf{g}^{(k)} \\ \alpha_k &= \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}) \\ \mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)},\end{aligned}$$

where the matrices $\mathbf{H}_0, \mathbf{H}_1, \dots$ are symmetric. In the quadratic case, the above matrices are required to satisfy

$$\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, \quad 0 \leq i \leq k,$$

where $\Delta \mathbf{x}^{(i)} = \mathbf{x}^{(i+1)} - \mathbf{x}^{(i)} = \alpha_i \mathbf{d}^{(i)}$, and $\Delta \mathbf{g}^{(i)} = \mathbf{g}^{(i+1)} - \mathbf{g}^{(i)} = \mathbf{Q} \Delta \mathbf{x}^{(i)}$. It turns out that quasi-Newton methods are also conjugate direction methods, as stated in the following.

Theorem 416. Consider a quasi-Newton algorithm applied to a quadratic function with Hessian $\mathbf{Q} = \mathbf{Q}^T$, such that for $0 \leq k \leq n-1$,

$$\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, \quad 0 \leq i \leq k,$$

where $\mathbf{H}_{k+1} = \mathbf{H}_{k+1}^T$. if $\alpha_i \neq 0, 0 \leq i \leq k+1$, then $\mathbf{d}_0, \dots, \mathbf{d}_{k+1}$ are \mathbf{Q} -conjugate.

Proof. We proceed by induction. We begin with the $k=0$ case: that $\mathbf{d}^{(0)}$ and $\mathbf{d}^{(1)}$ are \mathbf{Q} -conjugate. Because $\alpha_0 \neq 0$, we can write $\mathbf{d}^{(0)} = \Delta \mathbf{x}^{(0)} / \alpha_0$. Hence,

$$\begin{aligned}\mathbf{d}^{(1)T} \mathbf{Q} \mathbf{d}^{(0)} &= -\mathbf{g}^{(1)T} \mathbf{H}_1 \mathbf{Q} \mathbf{d}^{(0)} \\ &= -\mathbf{g}^{(1)T} \mathbf{H}_1 \frac{\mathbf{Q} \Delta \mathbf{x}^{(0)}}{\alpha_0} \\ &= -\mathbf{g}^{(1)T} \frac{\mathbf{H}_1 \Delta \mathbf{g}^{(0)}}{\alpha_0} \\ &= -\mathbf{g}^{(1)T} \frac{\Delta \mathbf{x}^{(0)}}{\alpha_0} \\ &= -\mathbf{g}^{(1)T} \mathbf{d}^{(0)}.\end{aligned}$$

But $\mathbf{g}^{(1)T} \mathbf{d}^{(0)} = 0$ as a consequence of $\alpha_0 > 0$ being the minimizer of $\phi(\alpha) = f(\mathbf{x}^{(0)} + \alpha \mathbf{d}^{(0)})$ (see Exercise 425). Hence, $\mathbf{d}^{(1)T} \mathbf{Q} \mathbf{d}^{(0)} = 0$.

Assume that the result is true for $k-1$ (where $k < n-1$). We now prove the result for k , that is, that $\mathbf{d}_0, \dots, \mathbf{d}_{k+1}$ are \mathbf{Q} -conjugate. It suffices to show that $\mathbf{d}^{(k+1)T} \mathbf{Q} \mathbf{d}^{(i)} =$

$0, 0 \leq i \leq k$. Given $i, 0 \leq i \leq k$, using the same algebraic steps as in the $k = 0$ case, and using the assumption that $\alpha_i \neq 0$, we obtain

$$\begin{aligned}\mathbf{d}^{(k+1)} \mathbf{Q} \mathbf{d}^{(i)} &= -\mathbf{g}^{(k+1)T} \mathbf{H}_{k+1} \mathbf{Q} \mathbf{d}^{(i)} \\ &\vdots \\ &= -\mathbf{g}^{(k+1)T} \mathbf{d}^{(i)}.\end{aligned}$$

Because $\mathbf{d}_0, \dots, \mathbf{d}_k$ are \mathbf{Q} -conjugate by assumption, we conclude from Lemma 395 that $\mathbf{g}^{(k+1)T} \mathbf{d}^{(i)} = 0$. Hence, $\mathbf{d}^{(k+1)} \mathbf{Q} \mathbf{d}^{(i)} = 0$, which completes the proof. \square

By Theorem 416 we conclude that a quasi-Newton algorithm solves a quadratic of n variables in at most n steps. Note that the equations that the matrices \mathbf{H}_k are required to satisfy do not determine those matrices uniquely. Thus, we have some freedom in the way we compute the \mathbf{H}_k . In the methods we describe, we compute \mathbf{H}_{k+1} by adding a correction to \mathbf{H}_k . In the following sections, we consider three specific updating formulas.

5.5.3 The Rank One Correction Formula

In the rank one correction formula, the correction term is symmetric, and has the form $a_k \mathbf{z}^{(k)} \mathbf{z}^{(k)T}$, where $a_k \in \mathbb{R}$ and $\mathbf{z}^{(k)}$ in \mathbb{R}^n . Therefore, the update equation is

$$\mathbf{H}_{k+1} = \mathbf{H}_k + a_k \mathbf{z}^{(k)} \mathbf{z}^{(k)T}.$$

Note that

$$\text{rank}(\mathbf{z}^{(k)} \mathbf{z}^{(k)T}) = \text{rank} \left(\begin{bmatrix} z_1^{(k)} \\ \vdots \\ z_n^{(k)} \end{bmatrix} [z_1^{(k)} \dots z_n^{(k)}] \right) = 1$$

and hence the name “rank one” correction (it is also called the *single-rank symmetric* (SRS) algorithm). The product $\mathbf{z}^{(k)} \mathbf{z}^{(k)T}$ is sometimes referred to as the *dyadic product or outer product*. Observe that if \mathbf{H}_k is symmetric, then so is \mathbf{H}_{k+1} . Our goal now is to determine a_k and $\mathbf{z}^{(k)}$, given $\mathbf{H}_k, \Delta \mathbf{g}^{(k)}, \Delta \mathbf{x}^{(k)}$, so that the required relationship discussed in the previous section is satisfied, namely, $\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, i = 1, \dots, k$. To begin, let us first consider the condition $\mathbf{H}_{k+1} \Delta \mathbf{g}^{(k)} = \Delta \mathbf{x}^{(k)}$. In other words, given $\mathbf{H}_k, \Delta \mathbf{g}^{(k)}, \Delta \mathbf{x}^{(k)}$, we wish to find a_k and $\mathbf{z}^{(k)}$ to ensure that

$$\mathbf{H}_{k+1} \Delta \mathbf{g}^{(k)} = (\mathbf{H}_k + a_k \mathbf{z}^{(k)} \mathbf{z}^{(k)T}) \Delta \mathbf{g}^{(k)} = \Delta \mathbf{x}^{(k)}.$$

First note that $\mathbf{z}^{(k)T} \Delta \mathbf{g}^{(k)}$ is a scalar. Thus

$$\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)} = (a_k \mathbf{z}^{(k)T} \Delta \mathbf{g}^{(k)}) \mathbf{z}^{(k)},$$

and hence

$$\mathbf{z}^{(k)} = \frac{\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)}}{a_k \mathbf{z}^{(k)T} \Delta \mathbf{g}^{(k)}}.$$

We can now determine

$$a_k \mathbf{z}^{(k)T} \mathbf{z}^{(k)T} = \frac{(\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)}) (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)})^T}{a_k (\mathbf{z}^{(k)T} \Delta \mathbf{g}^{(k)})^2}.$$

Hence,

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{(\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)}) (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)})^T}{a_k (\mathbf{z}^{(k)T} \Delta \mathbf{g}^{(k)})^2}.$$

The next step is to express the denominator of the second term on the right-hand side of the above equation as a function of the given quantities \mathbf{H}_k , $\Delta \mathbf{g}^{(k)}$, and $\Delta \mathbf{x}^{(k)}$. To accomplish this, premultiply $\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)} = (a_k \mathbf{z}^{(k)T} \Delta \mathbf{g}^{(k)}) \mathbf{z}^{(k)}$ by $\Delta \mathbf{g}^{(k)T}$ to obtain

$$\Delta \mathbf{g}^{(k)T} \Delta \mathbf{x}^{(k)} - \Delta \mathbf{g}^{(k)T} \mathbf{H}_k \Delta \mathbf{g}^{(k)} = \Delta \mathbf{g}^{(k)T} a_k \mathbf{z}^{(k)T} \mathbf{z}^{(k)T} \Delta \mathbf{g}^{(k)}.$$

Observe that a_k is a scalar and so is $\Delta \mathbf{g}^{(k)T} \mathbf{z}^{(k)} = \mathbf{z}^{(k)T} \Delta \mathbf{g}^{(k)}$. Thus,

$$\Delta \mathbf{g}^{(k)T} \Delta \mathbf{x}^{(k)} - \Delta \mathbf{g}^{(k)T} \mathbf{H}_k \Delta \mathbf{g}^{(k)} = a_k (\mathbf{z}^{(k)T} \Delta \mathbf{g}^{(k)})^2.$$

Taking the above relation into account yields

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{(\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)}) (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)})^T}{\Delta \mathbf{g}^{(k)T} (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)})}.$$

We summarize the above development in the following algorithm.

Rank One Algorithm

1. Set $k := 0$; select $\mathbf{x}^{(0)}$, and a real symmetric positive definite \mathbf{H}_0 .
2. If $\mathbf{g}^{(k)} = \mathbf{0}$, stop; else $\mathbf{d}^{(k)} = -\mathbf{H}_k \mathbf{g}^{(k)}$.
3. Compute

$$\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}),$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}.$$

4. Compute

$$\Delta \mathbf{x}^{(k)} = \alpha_k \mathbf{d}^{(k)},$$

$$\Delta \mathbf{g}^{(k)} = \mathbf{g}^{(k+1)} - \mathbf{g}^{(k)},$$

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{(\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)}) (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)})^T}{\Delta \mathbf{g}^{(k)T} (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)})}.$$

5. Set $k := k + 1$; go to step 2.

The rank one algorithm is based on satisfying the equation

$$\mathbf{H}_{k+1}\Delta\mathbf{g}^{(k)} = \Delta\mathbf{x}^{(k)}.$$

However, what we want is

$$\mathbf{H}_{k+1}\Delta\mathbf{g}^{(i)} = \Delta\mathbf{x}^{(i)}, i = 0, 1, \dots, k.$$

It turns out that the above is, in fact, automatically true, as stated in the following theorem.

Theorem 417. *For the rank one algorithm applied to the quadratic with Hessian $\mathbf{Q} = \mathbf{Q}^T$, we have $\mathbf{H}_{k+1}\Delta\mathbf{g}^{(i)} = \Delta\mathbf{x}^{(i)}$, $0 \leq i \leq k$.*

Proof. We prove the result by induction. From the discussion before the theorem it is clear that the claim is true for $k = 0$. Suppose now that the theorem is true for $k - 1 \geq 0$; that is, $\mathbf{H}_k\Delta\mathbf{g}^{(i)} = \Delta\mathbf{x}^{(i)}$, $i < k$. We now show that the theorem is true for k . Our construction of the correction term ensures that

$$\mathbf{H}_{k+1}\Delta\mathbf{g}^{(k)} = \Delta\mathbf{x}^{(k)}.$$

So we only have to show

$$\mathbf{H}_{k+1}\Delta\mathbf{g}^{(i)} = \Delta\mathbf{x}^{(i)}, i < k.$$

To this end, fix $i < k$. We have

$$\mathbf{H}_{k+1}\Delta\mathbf{g}^{(i)} = \mathbf{H}_k\Delta\mathbf{g}^{(i)} + \frac{(\Delta\mathbf{x}^{(k)} - \mathbf{H}_k\Delta\mathbf{g}^{(k)})(\Delta\mathbf{x}^{(k)} - \mathbf{H}_k\Delta\mathbf{g}^{(k)})^T}{\Delta\mathbf{g}^{(k)T}(\Delta\mathbf{x}^{(k)} - \mathbf{H}_k\Delta\mathbf{g}^{(k)})}\Delta\mathbf{g}^{(i)}.$$

By the induction hypothesis, $\mathbf{H}_k\Delta\mathbf{g}^{(i)} = \Delta\mathbf{x}^{(i)}$. To complete the proof it is enough to show that the second term on the right-hand side of the above equation is equal to zero. For this to be true it is enough that

$$(\Delta\mathbf{x}^{(k)} - \mathbf{H}_k\Delta\mathbf{g}^{(k)})^T\Delta\mathbf{g}^{(i)} = \Delta\mathbf{x}^{(k)T}\Delta\mathbf{g}^{(i)} - \Delta\mathbf{g}^{(k)T}\mathbf{H}_k\Delta\mathbf{g}^{(i)} = 0.$$

Indeed, since

$$\Delta\mathbf{g}^{(k)T}\mathbf{H}_k\Delta\mathbf{g}^{(i)} = \Delta\mathbf{g}^{(k)T}(\mathbf{H}_k\Delta\mathbf{g}^{(i)}) = \Delta\mathbf{g}^{(k)T}\Delta\mathbf{x}^{(i)}.$$

by the induction hypothesis, and because $\Delta\mathbf{g}^{(k)} = \mathbf{Q}\Delta\mathbf{x}^{(k)}$, we have

$$\Delta\mathbf{g}^{(k)T}\mathbf{H}_k\Delta\mathbf{g}^{(i)} = \Delta\mathbf{g}^{(k)T}\Delta\mathbf{x}^{(i)} = \Delta\mathbf{x}^{(k)T}\mathbf{Q}\Delta\mathbf{x}^{(i)} = \Delta\mathbf{x}^{(k)T}\Delta\mathbf{g}^{(i)}.$$

Hence,

$$(\Delta\mathbf{x}^{(k)} - \mathbf{H}_k\Delta\mathbf{g}^{(k)})^T\Delta\mathbf{g}^{(i)} = \Delta\mathbf{x}^{(k)T}\Delta\mathbf{g}^{(i)} - \Delta\mathbf{x}^{(k)T}\Delta\mathbf{g}^{(i)} = 0,$$

which completes the proof. \square

Example 418. Let

$$f(x_1, x_2) = x_1^2 + \frac{1}{2}x_2^2 + 3.$$

Apply the rank one correction algorithm to minimize f . Use $\mathbf{x}^{(0)} = [1, 2]^T$ and $\mathbf{H}_0 = \mathbf{I}_2$ (2×2 identity matrix). We can represent f as

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x} + 3.$$

Thus,

$$\mathbf{g}^{(k)} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \mathbf{x}^{(k)}.$$

Because $\mathbf{H}_0 = \mathbf{I}_2$,

$$\mathbf{d}^{(0)} = -\mathbf{g}^{(0)} = [-2, -2]^T.$$

The objective function is quadratic, and hence

$$\begin{aligned} \alpha_0 &= \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(0)} + \alpha \mathbf{d}^{(0)}) = -\frac{\mathbf{g}^{(0)T} \mathbf{d}^{(0)}}{\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(0)}} \\ &= -\frac{[2, 2] \begin{bmatrix} 2 \\ 2 \end{bmatrix}}{[2, 2] \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \end{bmatrix}} = \frac{2}{3}, \end{aligned} \quad (5.43)$$

and thus

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \alpha_0 \mathbf{d}^{(0)} = \left[-\frac{1}{3}, \frac{2}{3}\right]^T.$$

We then compute

$$\begin{aligned} \Delta \mathbf{x}^{(0)} &= \alpha_0 \mathbf{d}^{(0)} = \left[-\frac{4}{3}, -\frac{4}{3}\right]^T, \\ \mathbf{g}^{(1)} &= \mathbf{Q} \mathbf{x}^{(1)} = \left[-\frac{2}{3}, -\frac{2}{3}\right]^T, \\ \Delta \mathbf{g}^{(0)} &= \mathbf{g}^{(1)} - \mathbf{g}^{(0)} = \left[-\frac{8}{3}, -\frac{4}{3}\right]^T. \end{aligned}$$

Because

$$\Delta \mathbf{g}^{(0)T} (\Delta \mathbf{x}^{(0)} - \mathbf{H}_0 \Delta \mathbf{g}^{(0)}) = \left[-\frac{8}{3}, -\frac{4}{3}\right] \begin{bmatrix} \frac{4}{3} \\ 0 \end{bmatrix} = -\frac{32}{9},$$

we obtain

$$\mathbf{H}_1 = \mathbf{H}_0 + \frac{(\Delta \mathbf{x}^{(0)} - \mathbf{H}_0 \Delta \mathbf{g}^{(0)})(\Delta \mathbf{x}^{(0)} - \mathbf{H}_0 \Delta \mathbf{g}^{(0)})^T}{\Delta \mathbf{g}^{(0)T} (\Delta \mathbf{x}^{(0)} - \mathbf{H}_0 \Delta \mathbf{g}^{(0)})} = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{bmatrix}.$$

Therefore,

$$\mathbf{d}^{(1)} = -\mathbf{H}_1 \mathbf{g}^{(1)} = \left[\frac{1}{3}, -\frac{2}{3} \right]^T,$$

and

$$\alpha_1 = -\frac{\mathbf{g}^{(1)T} \mathbf{d}^{(1)}}{\mathbf{d}^{(1)T} \mathbf{Q} \mathbf{d}^{(1)}} = 1.$$

We now compute

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \alpha_1 \mathbf{d}^{(1)} = [0, 0]^T.$$

Note that $\mathbf{g}^{(2)} = \mathbf{0}$, and therefore $\mathbf{x}^{(2)} = \mathbf{x}^*$. As expected, the algorithm solves the problem in two steps. Note that the directions $\mathbf{d}^{(0)}$ and $\mathbf{d}^{(1)}$ are \mathbf{Q} -conjugate, in accordance with Theorem 4.16.

The rank one correction algorithm works well for the case of constant Hessian matrix; that is, the quadratic case. Our analysis was, in fact, done for this case. However, ultimately we wish to apply the algorithm to general functions, not just quadratics. Unfortunately, for the nonquadratic case, the rank one correction algorithm is not very satisfactory for several reasons. For a nonquadratic objective function, \mathbf{H}_{k+1} may not be positive definite (see Example 4.19 below) and thus $\mathbf{d}^{(k+1)}$ may not be a descent direction. Furthermore, if

$$\Delta \mathbf{g}^{(k)T} (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)})$$

is close to zero, then there may be numerical problems in evaluating \mathbf{H}_{k+1} .

Example 4.19. Assume that $\mathbf{H}_k \succ \mathbf{0}$. It turns out that if $\Delta \mathbf{g}^{(k)T} (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)}) > 0$, then $\mathbf{H}_{k+1} \succ \mathbf{0}$ (see Exercise 4.27). However, if $\Delta \mathbf{g}^{(k)T} (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)}) < 0$, then \mathbf{H}_{k+1} may not be positive definite. As an example of what might happen if $\Delta \mathbf{g}^{(k)T} (\Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)}) < 0$, consider applying the rank one algorithm to the function

$$f(\mathbf{x}) = \frac{x_1^4}{4} + \frac{x_2^2}{2} - x_1 x_2 + x_1 - x_2$$

with an initial point

$$\mathbf{x}^{(0)} = [0.59607, 0.59607]^T,$$

and initial matrix

$$\mathbf{H}_0 = \begin{bmatrix} 0.94913 & 0.14318 \\ 0.14318 & 0.59702 \end{bmatrix}.$$

Note that $\mathbf{H}_0 > 0$. We have

$$\Delta \mathbf{g}^{(0)T} (\Delta \mathbf{x}^{(0)}) - \mathbf{H}_0 \Delta \mathbf{g}^{(0)} = -0.03276$$

and

$$\mathbf{H}_1 = \begin{bmatrix} 0.94481 & 0.23324 \\ 0.23324 & -1.2788 \end{bmatrix},$$

It is easy to check that \mathbf{H}_1 is not positive definite (it is indefinite, with eigenvalues 0.96901 and -1.3030).

Fortunately, alternative algorithms have been developed for updating \mathbf{H}_k . In particular, if we use a “rank two” update, then \mathbf{H}_k is guaranteed to be positive definite for all k , provided the line search is exact. We discuss this in the next section.

5.5.4 The DFP Algorithm

The rank two update was originally developed by Davidon in 1959 and was subsequently modified by Fletcher and Powell in 1963; hence the name DFP algorithm. The DFP algorithm is also known as the variable metric algorithm. We summarize the algorithm below.

DFP Algorithm

1. Set $k := 0$; select $\mathbf{x}^{(0)}$, and a real symmetric positive definite \mathbf{H}_0 .
2. If $\mathbf{g}^{(k)} = \mathbf{0}$, stop; else $\mathbf{d}^{(k)} = -\mathbf{H}_k \mathbf{g}^{(k)}$.
3. Compute

$$\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)})$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}.$$

4. Compute

$$\Delta \mathbf{x}^{(k)} = \alpha_k \mathbf{d}^{(k)}$$

$$\Delta \mathbf{g}^{(k)} = \mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}$$

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{\Delta \mathbf{x}^{(k)} \Delta \mathbf{x}^{(k)T}}{\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(k)}} - \frac{[\mathbf{H}_k \Delta \mathbf{g}^{(k)}][\mathbf{H}_k \Delta \mathbf{g}^{(k)}]^T}{\Delta \mathbf{g}^{(k)T} \mathbf{H}_k \Delta \mathbf{g}^{(k)}}.$$

5. Set $k := k + 1$; go to step 2.

We now show that the DFP algorithm is a quasi-Newton method, in the sense that when applied to quadratic problems, we have $\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, 0 \leq i \leq k$.

Theorem 420. *In the DFP algorithm applied to the quadratic with Hessian $\mathbf{Q} = \mathbf{Q}^T$, we have $\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, 0 \leq i \leq k$.*

Proof. We use induction. For $k = 0$, we have

$$\begin{aligned}\mathbf{H}_1 \Delta \mathbf{g}^{(0)} &= \mathbf{H}_0 \Delta \mathbf{g}^{(0)} + \frac{\Delta \mathbf{x}^{(0)} \Delta \mathbf{x}^{(0)T}}{\Delta \mathbf{x}^{(0)T} \Delta \mathbf{g}^{(0)}} \Delta \mathbf{g}^{(0)} - \frac{\mathbf{H}_0 \Delta \mathbf{g}^{(0)} \Delta \mathbf{g}^{(0)T} \mathbf{H}_0}{\Delta \mathbf{g}^{(0)T} \mathbf{H}_0 \Delta \mathbf{g}^{(0)}} \Delta \mathbf{g}^{(0)} \\ &= \Delta \mathbf{x}^{(0)}.\end{aligned}$$

Assume the result is true for $k - 1$; that is, $\mathbf{H}_k \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, 0 \leq i \leq k - 1$. We now show that $\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, 0 \leq i \leq k$. First, consider $i = k$. We have

$$\begin{aligned}\mathbf{H}_{k+1} \Delta \mathbf{g}^{(k)} &= \mathbf{H}_k \Delta \mathbf{g}^{(k)} + \frac{\Delta \mathbf{x}^{(k)} \Delta \mathbf{x}^{(k)T}}{\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(k)}} \Delta \mathbf{g}^{(k)} - \frac{\mathbf{H}_k \Delta \mathbf{g}^{(k)} \Delta \mathbf{g}^{(k)T} \mathbf{H}_k}{\Delta \mathbf{g}^{(k)T} \mathbf{H}_k \Delta \mathbf{g}^{(k)}} \Delta \mathbf{g}^{(k)} \\ &= \Delta \mathbf{x}^{(k)}.\end{aligned}$$

It remains to consider the case $i < k$. To this end,

$$\begin{aligned}\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} &= \mathbf{H}_k \Delta \mathbf{g}^{(i)} + \frac{\Delta \mathbf{x}^{(k)} \Delta \mathbf{x}^{(k)T}}{\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(k)}} \Delta \mathbf{g}^{(i)} - \frac{\mathbf{H}_k \Delta \mathbf{g}^{(k)} \Delta \mathbf{g}^{(k)T} \mathbf{H}_k}{\Delta \mathbf{g}^{(k)T} \mathbf{H}_k \Delta \mathbf{g}^{(k)}} \Delta \mathbf{g}^{(i)} \\ &= \Delta \mathbf{x}^{(i)} + \frac{\Delta \mathbf{x}^{(k)}}{\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(k)}} (\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(i)}) - \frac{\mathbf{H}_k \Delta \mathbf{g}^{(k)}}{\Delta \mathbf{g}^{(k)T} \mathbf{H}_k \Delta \mathbf{g}^{(k)}} (\Delta \mathbf{g}^{(k)T} \Delta \mathbf{x}^{(i)}).\end{aligned}$$

Now,

$$\begin{aligned}\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(i)} &= \Delta \mathbf{x}^{(k)T} \mathbf{Q} \Delta \mathbf{x}^{(i)} \\ &= \alpha_k \alpha_i \mathbf{d}^{(k)T} \mathbf{Q} \mathbf{d}^{(i)} \\ &= 0,\end{aligned}$$

by the induction hypothesis and Theorem 416. The same arguments yield $\Delta \mathbf{g}^{(k)T} \Delta \mathbf{x}^{(i)} = 0$. Hence,

$$\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)},$$

which completes the proof. \square

By Theorems 416 and 420 we conclude that the DFP algorithm is a conjugate direction algorithm.

Example 421. Locate the minimizer of

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \mathbf{x} - \mathbf{x}^T \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \mathbf{x} \in \mathbb{R}^2.$$

Use the initial point $\mathbf{x}^{(0)} = [0, 0]^T$ and $\mathbf{H}_0 = \mathbf{I}_2$. Note that in this case,

$$\mathbf{g}^{(k)} = \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \mathbf{x}^{(k)} - \begin{bmatrix} -1 \\ 1 \end{bmatrix}.$$

Hence,

$$\mathbf{g}^{(0)} = [1, -1]^T,$$

$$\mathbf{d}^{(0)} = -\mathbf{H}_0 \mathbf{g}^{(0)} = -\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}.$$

Because f is a quadratic function,

$$\begin{aligned} \alpha_0 &= \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(0)} + \alpha \mathbf{d}^{(0)}) = -\frac{\mathbf{g}^{(0)T} \mathbf{d}^{(0)}}{\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(0)}} \\ &= \frac{[1, -1] \begin{bmatrix} -1 \\ 1 \end{bmatrix}}{[-1, 1] \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix}} = 1. \end{aligned}$$

Therefore,

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \alpha_0 \mathbf{d}^{(0)} = [-1, 1]^T.$$

We then compute

$$\begin{aligned} \Delta \mathbf{x}^{(0)} &= \mathbf{x}^{(1)} - \mathbf{x}^{(0)} = [-1, 1]^T, \\ \mathbf{g}^{(1)} &= \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} - \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \end{aligned}$$

and

$$\Delta \mathbf{g}^{(0)} = \mathbf{g}^{(1)} - \mathbf{g}^{(0)} = [-2, 0]^T.$$

Observe that

$$\begin{aligned} \Delta \mathbf{x}^{(0)} \Delta \mathbf{x}^{(0)T} &= \begin{bmatrix} -1 \\ 1 \end{bmatrix} [-1, 1] = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \\ \Delta \mathbf{x}^{(0)T} \Delta \mathbf{g}^{(0)} &= [-1, 1] \begin{bmatrix} -2 \\ 0 \end{bmatrix} = 2, \\ \mathbf{H}_0 \Delta \mathbf{g}^{(0)} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ 0 \end{bmatrix} = \begin{bmatrix} -2 \\ 0 \end{bmatrix}. \end{aligned}$$

Thus,

$$(\mathbf{H}_0 \Delta \mathbf{g}^{(0)}) (\mathbf{H}_0 \Delta \mathbf{g}^{(0)})^T = \begin{bmatrix} -2 \\ 0 \end{bmatrix} [-2, 0] = \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix},$$

and

$$\Delta \mathbf{g}^{(0)T} \mathbf{H}_0 \Delta \mathbf{g}^{(0)} = [-2, 0] \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ 0 \end{bmatrix} = 4.$$

Using the above, we now compute \mathbf{H}_1 :

$$\begin{aligned}\mathbf{H}_1 &= \mathbf{H}_0 + \frac{\Delta\mathbf{x}^{(0)}\Delta\mathbf{x}^{(0)T}}{\Delta\mathbf{x}^{(0)T}\Delta\mathbf{g}^{(0)}} - \frac{(\mathbf{H}_0\Delta\mathbf{g}^{(0)})(\mathbf{H}_0\Delta\mathbf{g}^{(0)})^T}{\Delta\mathbf{g}^{(0)T}\mathbf{H}_0\Delta\mathbf{g}^{(0)}} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} - \frac{1}{4} \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{3}{2} \end{bmatrix}.\end{aligned}$$

We now compute $\mathbf{d}^{(1)} = -\mathbf{H}_1\mathbf{g}^{(1)} = [0, 1]^T$, and

$$\alpha_1 = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(1)} + \alpha\mathbf{d}^{(1)}) = -\frac{\mathbf{g}^{(1)T}\mathbf{d}^{(1)}}{\mathbf{d}^{(1)T}\mathbf{Q}\mathbf{d}^{(1)}} = \frac{1}{2}.$$

Hence,

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \alpha_1\mathbf{d}^{(1)} = [-1, 3/2]^T = \mathbf{x}^*,$$

because f is a quadratic function of two variables.

Note that we have $\mathbf{d}^{(0)T}\mathbf{Q}\mathbf{d}^{(1)} = \mathbf{d}^{(1)T}\mathbf{Q}\mathbf{d}^{(0)} = 0$; that is, $\mathbf{d}^{(0)}$ and $\mathbf{d}^{(1)}$ are \mathbf{Q} -conjugate directions.

We now show that in the DFP algorithm, \mathbf{H}_{k+1} inherits positive definiteness from \mathbf{H}_k .

Theorem 4.22. Suppose that $\mathbf{g}^{(k)} \neq \mathbf{0}$. In the DFP algorithm, if \mathbf{H}_k is positive definite, then so is \mathbf{H}_{k+1} .

Proof. We first write the following quadratic form

$$\begin{aligned}\mathbf{x}^T\mathbf{H}_{k+1}\mathbf{x} &= \mathbf{x}^T\mathbf{H}_k\mathbf{x} + \frac{\mathbf{x}^T\Delta\mathbf{x}^{(k)}\Delta\mathbf{x}^{(k)T}\mathbf{x}}{\Delta\mathbf{x}^{(k)T}\Delta\mathbf{g}^{(k)}} - \frac{\mathbf{x}^T(\mathbf{H}_k\Delta\mathbf{g}^{(k)})(\mathbf{H}_k\Delta\mathbf{g}^{(k)})^T\mathbf{x}}{\Delta\mathbf{g}^{(k)T}\mathbf{H}_k\Delta\mathbf{g}^{(k)}} \\ &= \mathbf{x}^T\mathbf{H}_k\mathbf{x} + \frac{(\mathbf{x}^T\Delta\mathbf{x}^{(k)})^2}{\Delta\mathbf{x}^{(k)T}\Delta\mathbf{g}^{(k)}} - \frac{(\mathbf{x}^T\mathbf{H}_k\Delta\mathbf{g}^{(k)})^2}{\Delta\mathbf{g}^{(k)T}\mathbf{H}_k\Delta\mathbf{g}^{(k)}}\end{aligned}$$

Define

$$\begin{aligned}\mathbf{a} &\triangleq \mathbf{H}_k^{1/2}\mathbf{x}, \\ \mathbf{b} &\triangleq \mathbf{H}_k^{1/2}\Delta\mathbf{g}^{(k)},\end{aligned}$$

where

$$\mathbf{H}_k = \mathbf{H}_k^{1/2}\mathbf{H}_k^{1/2}.$$

Note that because $\mathbf{H}_k \succ \mathbf{0}$, its square root is well defined; see Section 3.4 for more information on this property of positive definite matrices. Using the definitions of \mathbf{a} and \mathbf{b} , we obtain

$$\begin{aligned}\mathbf{x}^T \mathbf{H}_k \mathbf{x} &= \mathbf{x}^T \mathbf{H}_k^{1/2} \mathbf{H}_k^{1/2} \mathbf{x} = \mathbf{a}^T \mathbf{a}, \\ \mathbf{x}^T \mathbf{H}_k \Delta \mathbf{g}^{(k)} &= \mathbf{x}^T \mathbf{H}_k^{1/2} \mathbf{H}_k^{1/2} \Delta \mathbf{g}^{(k)} = \mathbf{a}^T \mathbf{b},\end{aligned}$$

and

$$\Delta \mathbf{g}^{(k)T} \mathbf{H}_k \Delta \mathbf{g}^{(k)} = \Delta \mathbf{g}^{(k)T} \mathbf{H}_k^{1/2} \mathbf{H}_k^{1/2} \Delta \mathbf{g}^{(k)} = \mathbf{b}^T \mathbf{b}.$$

Hence,

$$\begin{aligned}\mathbf{x}^T \mathbf{H}_{k+1} \mathbf{x} &= \mathbf{a}^T \mathbf{a} + \frac{(\mathbf{x}^T \Delta \mathbf{x}^{(k)})^2}{\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(k)}} - \frac{(\mathbf{a}^T \mathbf{b})^2}{\mathbf{b}^T \mathbf{b}} \\ &= \frac{\|\mathbf{a}\|^2 \|\mathbf{b}\|^2 - (\langle \mathbf{a}, \mathbf{b} \rangle)^2}{\|\mathbf{b}\|^2} + \frac{(\mathbf{x}^T \Delta \mathbf{x}^{(k)})^2}{\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(k)}}.\end{aligned}$$

We also have

$$\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(k)} = \Delta \mathbf{x}^{(k)T} (\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}) = -\Delta \mathbf{x}^{(k)T} \mathbf{g}^{(k)},$$

since $\Delta \mathbf{x}^{(k)T} \mathbf{g}^{(k+1)} = \alpha_k \mathbf{d}^{(k)T} \mathbf{g}^{(k+1)} = 0$ by Lemma 395. Because

$$\Delta \mathbf{x}^{(k)} = \alpha_k \mathbf{d}^{(k)} = -\alpha_k \mathbf{H}_k \mathbf{g}^{(k)},$$

we have

$$\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(k)} = -\Delta \mathbf{x}^{(k)T} \mathbf{g}^{(k)} = \alpha_k \mathbf{g}^{(k)T} \mathbf{H}_k \mathbf{g}^{(k)}.$$

The above yields

$$\mathbf{x}^T \mathbf{H}_{k+1} \mathbf{x} = \frac{\|\mathbf{a}\|^2 \|\mathbf{b}\|^2 - (\langle \mathbf{a}, \mathbf{b} \rangle)^2}{\|\mathbf{b}\|^2} + \frac{(\mathbf{x}^T \Delta \mathbf{x}^{(k)})^2}{\alpha_k \mathbf{g}^{(k)T} \mathbf{H}_k \mathbf{g}^{(k)}}.$$

Both terms on the right-hand side of the above equation are nonnegative – the first term is nonnegative because of the Cauchy-Schwarz inequality, and the second term is nonnegative because $\mathbf{H}_k \succ \mathbf{0}$ and $\alpha_k > 0$ (by Proposition 415). Therefore, to show that $\mathbf{x}^T \mathbf{H}_{k+1} \mathbf{x} > 0$ for $\mathbf{x} \neq \mathbf{0}$, we only need to demonstrate that these terms do not both vanish simultaneously. The first term vanishes only if \mathbf{a} and \mathbf{b} are proportional; that is, if $\mathbf{a} = \beta \mathbf{b}$ for some scalar β . Thus, to complete the proof it is enough to show that if $\mathbf{a} = \beta \mathbf{b}$, then $(\mathbf{x}^T \Delta \mathbf{x}^{(k)})^2 / (\alpha_k \mathbf{g}^{(k)T} \mathbf{H}_k \mathbf{g}^{(k)}) > 0$. Indeed, first observe that

$$\mathbf{H}_k^{1/2} \mathbf{x} = \mathbf{a} = \beta \mathbf{b} = \beta \mathbf{H}_k^{1/2} \Delta \mathbf{g}^{(k)} = \mathbf{H}_k^{1/2} (\beta \mathbf{g}^{(k)}).$$

Hence,

$$\mathbf{x} = \beta \Delta \mathbf{g}^{(k)}.$$

Using the above expression for \mathbf{x} and the expression $\Delta\mathbf{x}^{(k)T}\Delta\mathbf{g}^{(k)} = -\alpha\mathbf{g}^{(k)T}\mathbf{H}_k\mathbf{g}^{(k)}$, we obtain

$$\begin{aligned}\frac{(\mathbf{x}^T\Delta\mathbf{x}^{(k)})^2}{\alpha_k\mathbf{g}^{(k)T}\mathbf{H}_k\mathbf{g}^{(k)}} &= \frac{\beta^2(\Delta\mathbf{g}^{(k)T}\Delta\mathbf{x}^{(k)})^2}{\alpha_k\mathbf{g}^{(k)T}\mathbf{H}_k\mathbf{g}^{(k)}} = \frac{\beta^2(\alpha\mathbf{g}^{(k)T}\mathbf{H}_k\mathbf{g}^{(k)})^2}{\alpha_k\mathbf{g}^{(k)T}\mathbf{H}_k\mathbf{g}^{(k)}} \\ &= \beta^2\alpha\mathbf{g}^{(k)T}\mathbf{H}_k\mathbf{g}^{(k)} > 0.\end{aligned}$$

Thus, for all $\mathbf{x} \neq 0$,

$$\mathbf{x}^T\mathbf{H}_{k+1}\mathbf{x} > 0,$$

and the proof is completed. \square

The DFP algorithm is superior to the rank one algorithm in that it preserves the positive definiteness of \mathbf{H}_k . However, it turns out that in the case of larger nonquadratic problems the algorithm has the tendency of sometimes getting “stuck.” This phenomenon is attributed to \mathbf{H}_k becoming nearly singular [19]. In the next section, we discuss an algorithm that alleviates this problem.

5.5.5 The BFGS Algorithm

In 1970, an alternative update formula was suggested independently by Broyden, Fletcher, Goldfarb, and Shanno. The method is now called the BFGS algorithm, which we discuss in this section.

To derive the BFGS update, we use the concept of duality, or complementarity. To discuss this concept, recall that the updating formulas for the approximation of the inverse of the Hessian matrix were based on satisfying the equations

$$\mathbf{H}_{k+1}\Delta\mathbf{g}^{(i)} = \Delta\mathbf{x}^{(i)}, 0 \leq i \leq k,$$

which were derived from $\Delta\mathbf{g}^{(i)} = \mathbf{Q}\Delta\mathbf{x}^{(i)}, 0 \leq i \leq k$. We then formulate update formulas for the approximations to the inverse of the Hessian matrix \mathbf{Q}^{-1} . An alternative to approximating \mathbf{Q}^{-1} is to approximate \mathbf{Q} itself. To do this let \mathbf{B}_k be our estimate of \mathbf{Q} at the k th step. We require \mathbf{B}_{k+1} to satisfy

$$\Delta\mathbf{g}^{(i)} = \mathbf{B}_{k+1}\Delta\mathbf{x}^{(i)}, 0 \leq i \leq k.$$

Notice that the above set of equations is similar to the previous set of equations for \mathbf{H}_{k+1} , the only difference being that the roles of $\Delta\mathbf{x}^{(i)}$ and $\Delta\mathbf{g}^{(i)}$ are interchanged. Thus, given any update formula for \mathbf{H}_k , a corresponding update formula for \mathbf{B}_k can be found by interchanging the roles of \mathbf{B}_k and \mathbf{H}_k , and of $\Delta\mathbf{g}^{(k)}$ and $\Delta\mathbf{x}^{(k)}$. In particular, the BFGS

update for \mathbf{B}_k corresponds to the DFP update for \mathbf{H}_k . Formulas related in this way are said to be *dual* or *complementary* [46].

Recall that the DFP update for the approximation \mathbf{H}_k of the inverse Hessian is

$$\mathbf{H}_{k+1}^{DFP} = \mathbf{H}_k + \frac{\Delta \mathbf{x}^{(k)} \Delta \mathbf{x}^{(k)T}}{\Delta \mathbf{g}^{(k)T} \Delta \mathbf{g}^{(k)}} - \frac{\mathbf{H}_k \Delta \mathbf{g}^{(k)} \Delta \mathbf{g}^{(k)T} \mathbf{H}_k}{\Delta \mathbf{g}^{(k)T} \mathbf{H}_k \Delta \mathbf{g}^{(k)}}.$$

Using the complementarity concept, we can easily obtain an update equation for the approximation \mathbf{B}_k of the Hessian:

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\Delta \mathbf{g}^{(k)} \Delta \mathbf{g}^{(k)T}}{\Delta \mathbf{g}^{(k)T} \Delta \mathbf{x}^{(k)}} - \frac{\mathbf{B}_k \Delta \mathbf{x}^{(k)} \Delta \mathbf{x}^{(k)T} \mathbf{B}_k}{\Delta \mathbf{x}^{(k)T} \mathbf{B}_k \Delta \mathbf{x}^{(k)}}.$$

This is the BFGS update of \mathbf{B}_k .

Now, to obtain the BFGS update for the approximation of the inverse Hessian, we take the inverse of \mathbf{B}_{k+1} to obtain

$$\begin{aligned} \mathbf{H}_{k+1}^{BFGS} &= (\mathbf{B}_{k+1})^{-1} \\ &= \left(\mathbf{B}_k + \frac{\Delta \mathbf{g}^{(k)} \Delta \mathbf{g}^{(k)T}}{\Delta \mathbf{g}^{(k)T} \Delta \mathbf{x}^{(k)}} - \frac{\mathbf{B}_k \Delta \mathbf{x}^{(k)} \Delta \mathbf{x}^{(k)T} \mathbf{B}_k}{\Delta \mathbf{x}^{(k)T} \mathbf{B}_k \Delta \mathbf{x}^{(k)}} \right)^{-1}. \end{aligned}$$

To compute \mathbf{H}_{k+1}^{BFGS} by inverting the right-hand side of the above equation, we apply the following formula for a matrix inverse, known as the Sherman-Morrison formula.

Lemma 423.

$$(\mathbf{A} + \mathbf{U}\mathbf{C}\mathbf{V}^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{C}^{-1} + \mathbf{V}^T\mathbf{A}^{-1}\mathbf{U})^{-1}\mathbf{V}^T\mathbf{A}^{-1},$$

where \mathbf{A} and \mathbf{C} are invertible and the sizes of \mathbf{U} , \mathbf{C} , and \mathbf{V} are compatible.

Proof. We can prove the result easily by verification. \square

From the above lemma it follows that if \mathbf{A}^{-1} is known, then the inverse of the matrix \mathbf{A} augmented by a rank one matrix can be obtained by a modification of the matrix \mathbf{A}^{-1} .

Applying the above lemma to \mathbf{B}_{k+1} yields

$$\mathbf{H}_{k+1}^{BFGS} = \mathbf{H}_k + \left(1 + \frac{\Delta \mathbf{g}^{(k)T} \mathbf{H}_k \Delta \mathbf{g}^{(k)}}{\Delta \mathbf{g}^{(k)T} \Delta \mathbf{x}^{(k)}} \right) \frac{\Delta \mathbf{x}^{(k)} \Delta \mathbf{x}^{(k)T}}{\Delta \mathbf{x}^{(k)T} \Delta \mathbf{g}^{(k)}} - \frac{\mathbf{H}_k \Delta \mathbf{g}^{(k)} \Delta \mathbf{x}^{(k)T} + (\mathbf{H}_k \Delta \mathbf{g}^{(k)} \Delta \mathbf{x}^{(k)T})^T}{\Delta \mathbf{g}^{(k)T} \Delta \mathbf{x}^{(k)}}.$$

The above represents the BFGS formula for updating \mathbf{H}_k .

Recall that for the quadratic case, the DFP algorithm satisfies $\mathbf{H}_{k+1}^{DFP} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, 0 \leq i \leq k$. Therefore, the BFGS update for \mathbf{B}_k satisfies $\mathbf{B}_{k+1} \Delta \mathbf{x}^{(i)} = \Delta \mathbf{g}^{(i)}, 0 \leq i \leq k$. By construction of the BFGS formula for \mathbf{H}_{k+1}^{BFGS} , we conclude that $\mathbf{H}_{k+1}^{BFGS} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, 0 \leq$

$i \leq k$. Hence, the BFGS algorithm enjoys all the properties of quasi-Newton methods, including the conjugate directions property. Moreover, the BFGS algorithm also inherits the positive definiteness property of the DFP algorithm; that is, if $\mathbf{g}^{(k)} \neq \mathbf{0}$ and $\mathbf{H}_k \succ \mathbf{0}$, then $\mathbf{H}_{k+1}^{BFGS} \succ \mathbf{0}$.

The BFGS update is reasonably robust when the line searches are sloppy (see [14]). This property allows us to save time in the line search part of the algorithm. The BFGS formula is often far more efficient than the DFP formula (see [77] for further discussion).

We conclude our discussion of the BFGS algorithm with the following numerical example.

Example 424. Use the BFGS method to minimize

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q}\mathbf{x} - \mathbf{x}^T \mathbf{b} + \log(\pi),$$

where

$$\mathbf{Q} = \begin{bmatrix} 5 & -3 \\ -3 & 2 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Take $\mathbf{H}_0 = \mathbf{I}_2$ and $\mathbf{x}^{(0)} = [0, 0]^T$. Verify that $\mathbf{H}_2 = \mathbf{Q}^{-1}$. We have

$$\mathbf{d}^{(0)} = -\mathbf{g}^{(0)} = -(\mathbf{Q}\mathbf{x}^{(0)} - \mathbf{b}) = \mathbf{b} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

The objective function is a quadratic, and hence we can use the following formula to compute α_0 :

$$\alpha_0 = -\frac{\mathbf{g}^{(0)T} \mathbf{d}^{(0)}}{\mathbf{d}^{(0)T} \mathbf{Q} \mathbf{d}^{(0)}} = \frac{1}{2}.$$

Therefore,

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + \alpha_0 \mathbf{d}^{(0)} = \begin{bmatrix} 0 \\ 1/2 \end{bmatrix}.$$

To compute $\mathbf{H}_1 = \mathbf{H}_1^{BFGS}$, We need the following quantities:

$$\Delta \mathbf{x}^{(0)} = \mathbf{x}^{(1)} - \mathbf{x}^{(0)} = \begin{bmatrix} 0 \\ 1/2 \end{bmatrix},$$

$$\mathbf{g}^{(1)} = \mathbf{Q}\mathbf{x}^{(1)} - \mathbf{b} = \begin{bmatrix} -3/2 \\ 0 \end{bmatrix},$$

$$\Delta \mathbf{g}^{(0)} = \mathbf{g}^{(1)} - \mathbf{g}^{(0)} = \begin{bmatrix} -3/2 \\ 1 \end{bmatrix}.$$

Therefore,

$$\begin{aligned}\mathbf{H}_1 &= \mathbf{H}_0 + \left(1 + \frac{\Delta \mathbf{g}^{(0)T} \mathbf{H}_0 \Delta \mathbf{g}^{(0)}}{\Delta \mathbf{g}^{(0)T} \Delta \mathbf{x}^{(0)}}\right) \frac{\Delta \mathbf{x}^{(0)} \Delta \mathbf{x}^{(0)T}}{\Delta \mathbf{x}^{(0)T} \Delta \mathbf{g}^{(0)}} - \frac{\mathbf{H}_0 \Delta \mathbf{g}^{(0)} \Delta \mathbf{x}^{(0)T} + (\mathbf{H}_0 \Delta \mathbf{g}^{(0)} \Delta \mathbf{x}^{(0)T})^T}{\Delta \mathbf{g}^{(0)T} \Delta \mathbf{x}^{(0)}} \\ &= \begin{bmatrix} 1 & 3/2 \\ 3/2 & 11/4 \end{bmatrix}.\end{aligned}$$

Hence, we have

$$\begin{aligned}\mathbf{d}^{(1)} &= -\mathbf{H}_1 \mathbf{g}^{(1)} = \begin{bmatrix} 3/2 \\ 9/4 \end{bmatrix}, \\ \alpha_1 &= -\frac{\mathbf{g}^{(1)T} \mathbf{d}^{(1)}}{\mathbf{d}^{(1)T} \mathbf{Q} \mathbf{d}^{(1)}} = 2.\end{aligned}$$

Therefore,

$$\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \alpha_1 \mathbf{d}^{(1)} = \begin{bmatrix} 3 \\ 5 \end{bmatrix}.$$

Because our objective function is a quadratic on \mathbb{R}^2 , $\mathbf{x}^{(2)}$ is the minimizer. Notice that the gradient at $\mathbf{x}^{(2)}$ is $\mathbf{0}$; that is, $\mathbf{g}^{(2)} = \mathbf{0}$.

To verify that $\mathbf{H}_2 = \mathbf{Q}^{-1}$, we compute:

$$\begin{aligned}\Delta \mathbf{x}^{(1)} &= \mathbf{x}^{(2)} - \mathbf{x}^{(1)} = \begin{bmatrix} 3 \\ 9/2 \end{bmatrix}, \\ \Delta \mathbf{g}^{(1)} &= \mathbf{g}^{(2)} - \mathbf{g}^{(1)} = \begin{bmatrix} 3/2 \\ 0 \end{bmatrix}.\end{aligned}$$

Hence,

$$\begin{aligned}\mathbf{H}_2 &= \mathbf{H}_1 + \left(1 + \frac{\Delta \mathbf{g}^{(1)T} \mathbf{H}_1 \Delta \mathbf{g}^{(1)}}{\Delta \mathbf{g}^{(1)T} \Delta \mathbf{x}^{(1)}}\right) \frac{\Delta \mathbf{x}^{(1)} \Delta \mathbf{x}^{(1)T}}{\Delta \mathbf{x}^{(1)T} \Delta \mathbf{g}^{(1)}} - \frac{\mathbf{H}_1 \Delta \mathbf{g}^{(1)} \Delta \mathbf{x}^{(1)T} + (\mathbf{H}_1 \Delta \mathbf{g}^{(1)} \Delta \mathbf{x}^{(1)T})^T}{\Delta \mathbf{g}^{(1)T} \Delta \mathbf{x}^{(1)}} \\ &= \begin{bmatrix} 2 & 3 \\ 3 & 5 \end{bmatrix}.\end{aligned}$$

Note that indeed $\mathbf{H}_2 \mathbf{Q} = \mathbf{Q} \mathbf{H}_2 = \mathbf{I}_2$, and hence $\mathbf{H}_2 = \mathbf{Q}^{-1}$.

For nonquadratic problems, quasi-Newton algorithms will not usually converge in n steps. As in the case of the conjugate gradient methods, here too some modifications may be necessary to deal with nonquadratic problems. For example, we may reinitialize the direction vector to the negative gradient after every few iterations (e.g., n or $n + 1$), and continue until the algorithm satisfies the stopping criterion.

5.5.6 Exercises

Exercise 425. Given $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $f \in \mathcal{C}^1$, consider the algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)},$$

where $\mathbf{d}^{(1)}, \mathbf{d}^{(2)}, \dots$ are vectors in \mathbb{R}^n , and $\alpha_k \geq 0$ is chosen to minimize $f(\mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)})$; that is

$$\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}).$$

Note that the above general algorithm encompasses almost all algorithms that we discussed in this part, including the steepest descent, Newton, conjugate gradient, and quasi-Newton algorithms.

Let $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)})$, and assume that $\mathbf{d}^{(k)T} \mathbf{g}^{(k)} < 0$.

1. Show that $\mathbf{d}^{(k)}$ is a descent direction for f , in the sense that there exists $\bar{\alpha} > 0$ such that for all $\alpha \in (0, \bar{\alpha}]$,

$$f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}) < f(\mathbf{x}^{(k)}).$$

2. Show that $\alpha_k > 0$.

3. Show that $\mathbf{d}^{(k)T} \mathbf{g}^{(k+1)} = 0$.

4. Show that the following algorithms all satisfy the condition $\mathbf{d}^{(k)T} \mathbf{g}^{(k)} < 0$, if $\mathbf{g}^{(k)} \neq \mathbf{0}$:

- (a) Steepest descent algorithm;
- (b) Newton's method, assuming the Hessian is positive definite;
- (c) Conjugate gradient algorithm;
- (d) Quasi-Newton algorithm, assuming $\mathbf{H}_k \succ 0$.

5. For the case where $f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b}$, with $\mathbf{Q} = \mathbf{Q}^T \succ 0$, derive an expression for α_k in terms of \mathbf{Q} , $\mathbf{d}^{(k)}$, and $\mathbf{g}^{(k+1)}$.

Exercise 426. Consider Newton's algorithm applied to a function $f \in \mathcal{C}^2$:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{F}(\mathbf{x}^{(k)})^{-1} \Delta f(\mathbf{x}^{(k)}),$$

where α_k is chosen according to a line search. Is this algorithm a member of the quasi-Newton family?

Exercise 427. In some optimization methods, when minimizing a given function $f(\mathbf{x})$, we select an initial guess $\mathbf{x}^{(0)}$ and a real symmetric positive definite matrix \mathbf{H}_0 . Then we iteratively compute \mathbf{H}_k , $\mathbf{d}^{(k)} = -\mathbf{H}_k \mathbf{g}^{(k)}$ (where $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)})$), and $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$, where

$$\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} + \alpha \mathbf{d}^{(k)}).$$

Suppose that the function we wish to minimize is a standard quadratic of the form

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{b} + c, \quad \mathbf{Q} = \mathbf{Q}^T \succ 0.$$

1. Find an expression for α_k in terms of \mathbf{Q} , \mathbf{H}_k , $\mathbf{g}^{(k)}$, and $\mathbf{d}^{(k)}$;
2. Give a sufficient condition on \mathbf{H}_k for α_k to be positive.

Exercise 428. Consider the algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mathbf{H} \mathbf{g}^{(k)},$$

where, as usual, $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)})$ and \mathbf{H} is a fixed symmetric matrix.

1. Suppose that $f \in \mathcal{C}^3$ and there is a point \mathbf{x}^* such that $\Delta f(\mathbf{x}^*) = \mathbf{0}$ and $\mathbf{F}(\mathbf{x}^*)^{-1}$ converges to \mathbf{x}^* with order of convergence of at least 2.
2. With the setting of \mathbf{H} in part 1, is the given algorithm a quasi-Newton method?

Exercise 429. Minimize the function

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \mathbf{x} - \mathbf{x}^T \begin{bmatrix} 1 \\ -1 \end{bmatrix} + 7$$

using the rank one correction method with the starting point $\mathbf{x}^{(0)} = \mathbf{0}$.

Exercise 430. Consider the algorithm

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \mathbf{M}_k \nabla f(\mathbf{x}^k),$$

where $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, $f \in \mathcal{C}^1$, $\mathbf{M}_k \in \mathbb{R}^{2 \times 2}$ is given by

$$\mathbf{M}_k = \begin{bmatrix} 1 & 0 \\ 0 & a \end{bmatrix}$$

with $a \in \mathbb{R}$, and

$$\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}^{(k)} - \alpha \mathbf{M}_k \nabla f(\mathbf{x}^{(k)})).$$

Suppose at some iteration k we have $\nabla f(\mathbf{x}^{(k)}) = [1, 1]^T$. Find the largest range of values of a that guarantees that $\alpha_k > 0$ for any f .

Exercise 431. Consider the rank one algorithm. Assume that $\mathbf{H}_k > 0$. Show that if $\Delta\mathbf{g}^{(k)T}(\Delta\mathbf{x}^{(k)} - \mathbf{H}_k\Delta\mathbf{g}^{(k)}) > 0$, then $\mathbf{H}_{k+1} > 0$.

Exercise 432. Based on the rank one update equation, derive an update formula using complementarity and the matrix inverse formula.

Exercise 433. Let

$$\begin{aligned} f(\mathbf{x}) &= \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} - \mathbf{x}^T\mathbf{b} + c \\ &= \frac{1}{2}\mathbf{x}^T \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \mathbf{x} - \mathbf{x}^T \begin{bmatrix} 1 \\ -1 \end{bmatrix} + 7 \end{aligned}$$

and $\mathbf{x}^{(0)} = \mathbf{0}$. Use the rank one correction method to generate two \mathbf{Q} -conjugate directions.

Exercise 434. Apply the rank one algorithm to the problem in Example 421.

Exercise 435. Consider the DFP algorithm applied to the quadratic function

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} - \mathbf{x}^T\mathbf{b},$$

where $\mathbf{Q} = \mathbf{Q}^T > 0$.

1. Write down a formula for α_k in terms of \mathbf{Q} , $\mathbf{g}^{(k)}$ and $\mathbf{d}^{(k)}$.
2. Show that if $\mathbf{g}^{(k)} \neq 0$, then $\alpha_k > 0$.

Exercise 436. Use Lemma 423 to derive the BFGS update formula based on the DFP formula, using complementarity.

Hint. Define

$$\begin{aligned} \mathbf{A}_0 &= \mathbf{B}_k, \\ \mathbf{u}_0 &= \frac{\Delta\mathbf{g}^{(k)}}{\Delta\mathbf{g}^{(k)T}\Delta\mathbf{x}^{(k)}}, \\ \mathbf{v}_0^T &= \Delta\mathbf{g}^{(k)T}, \\ \mathbf{u}_1 &= -\frac{\mathbf{B}_k\Delta\mathbf{x}^{(k)}}{\mathbf{x}^{(k)T}\mathbf{B}_k\mathbf{x}^{(k)}}, \\ \mathbf{v}_1^T &= \Delta\mathbf{x}^{(k)T}\mathbf{B}_k, \\ \mathbf{A}_1 &= \mathbf{B}_k + \frac{\Delta\mathbf{g}^{(k)}\Delta\mathbf{g}^{(k)T}}{\Delta\mathbf{g}^{(k)T}\Delta\mathbf{x}^{(k)}} = \mathbf{A}_0 + \mathbf{u}_0\mathbf{v}_0^T. \end{aligned}$$

Using the notation above, represent \mathbf{B}_{k+1} as

$$\begin{aligned} \mathbf{B}_{k+1} &= \mathbf{A}_0 + \mathbf{u}_0\mathbf{v}_0^T + \mathbf{u}_1\mathbf{v}_1^T \\ &= \mathbf{A}_1 + \mathbf{u}_1\mathbf{v}_1^T. \end{aligned}$$

Apply Lemma 423 to the above.

Exercise 437. Assuming exact line search, show that if $\mathbf{H}_0 = \mathbf{I}_n$ ($n \times n$ identity matrix), then the first two steps of the BFGS algorithm yield the same points $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ as conjugate gradient algorithms with the Hestenes-Stiefel, the Polak-Ribiere, and the Fletcher-Reeves formulas.

Exercise 438. Let $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ be such that $f \in \mathcal{C}^1$. Consider an optimization algorithm applied to this f , of the usual form $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$, where $\alpha_k \geq 0$ is chosen according to line search. Suppose that $\mathbf{d}^{(k)} = -\mathbf{H}_k \mathbf{g}^{(k)}$, where $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)})$ and \mathbf{H}_k is symmetric.

1. Show that if \mathbf{H}_k satisfies the following conditions whenever the algorithm is applied to a quadratic, then the algorithm is quasi-Newton:
 - (a) $\mathbf{H}_{k+1} = \mathbf{H}_k + \mathbf{U}_k$.
 - (b) $\mathbf{U}_k \Delta \mathbf{g}^{(k)} = \Delta \mathbf{x}^{(k)} - \mathbf{H}_k \Delta \mathbf{g}^{(k)}$.
 - (c) $\mathbf{U}_k = \mathbf{a}^{(k)} \Delta \mathbf{x}^{(k)T} + \mathbf{b}^{(k)} \Delta \mathbf{g}^{(k)T} \mathbf{H}_k$, where $\mathbf{a}^{(k)}$ and $\mathbf{b}^{(k)}$ are in \mathbb{R}^n .
2. Which (if any) among the rank-one, DFP, and BFGS algorithms satisfy the three conditions in part 1 (whenever the algorithm is applied to a quadratic)? For those that do, specify the vectors $\mathbf{a}^{(k)}$ and $\mathbf{b}^{(k)}$.

Exercise 439. Given a function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, consider an algorithm $\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \mathbf{H}_k \mathbf{g}^{(k)}$ for finding the minimizer of f , where $\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)})$ and $\mathbf{H}_k \in \mathbb{R}^{n \times n}$ is symmetric. Suppose that $\mathbf{H}_k = \phi \mathbf{H}_k^{DFP} + (1 - \phi) \mathbf{H}_k^{BFGS}$, where $\phi \in \mathbb{R}$, and \mathbf{H}_k^{DFP} and \mathbf{H}_k^{BFGS} are matrices generated by the DFP and BFGS algorithms, respectively.

1. Show that the above algorithm is a quasi-Newton algorithm. Is the above algorithm a conjugate direction algorithm?
2. Suppose $0 \leq \phi \leq 1$. Show that if $\mathbf{H}_0^{DFP} \succ 0$ and $\mathbf{H}_0^{BFGS} \succ 0$, then $\mathbf{H}_k \succ 0$ for all k . What can you conclude from this about whether or not the algorithm has the descent property?

Exercise 440. Consider the following simple modification to the quasi-Newton family of algorithms. In the quadratic case, instead of the usual quasi-Newton condition $\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \Delta \mathbf{x}^{(i)}, 0 \leq i \leq k$, suppose that we have $\mathbf{H}_{k+1} \Delta \mathbf{g}^{(i)} = \rho_i \Delta \mathbf{x}^{(i)}, 0 \leq i \leq k$, where $\rho_i > 0$. We refer to the set of algorithms that satisfy the above condition as the symmetric Huang family. Show that the symmetric Huang family algorithms are conjugate direction algorithms.

Exercise 441. Write a MATLAB routine to implement the quasi-Newton algorithm for general functions. Use the secant method for the line search. Test the different update formulas for \mathbf{H}_k on Rosenbrock's function (see Exercise 380), with an initial condition $\mathbf{x}^{(0)} = [-2, 2]^T$. For this exercise, reinitialize the update direction to the negative gradient every 6 iterations.

Exercise 442. Consider the function

$$f(\mathbf{x}) = \frac{x_1^4}{4} + \frac{x_2^2}{2} - x_1x_2 + x_1 - x_2.$$

1. Use MATLAB to plot the level sets of f at levels $-0.72, -0.6, -0.2, 0.5, 2$. Locate the minimizers of f from the plots of the level sets.
2. Apply the DFP algorithm to minimize the above function with the following starting initial conditions: (i) $[0, 0]^T$; (ii) $[1.5, 1]^T$. Use $\mathbf{H}_0 = \mathbf{I}_2$. Does the algorithm converge to the same point for the two initial conditions? If not, explain.

(Taken from Chapter 6 of [109])

Exercise 443. Solve the equations $5x_1 + 3x_2 = 1$ and $4x_1 - 7x_2 = 76$ using the BFGS method with the starting point $(0, 0)^T$.

5.5.7 Limited-Memory Quasi-Newton Methods

(Taken from Chapter 7.2 of [104])

Limited-memory quasi-Newton methods are useful for solving large problems whose Hessian matrices cannot be computed at a reasonable cost or are not sparse. These methods maintain simple and compact approximations of Hessian matrices: Instead of storing fully dense $n \times n$ approximations, they save only a few vectors of length n that represent the approximations implicitly. Despite these modest storage requirements, they often yield an acceptable (albeit linear) rate of convergence. Various limited-memory methods have been proposed; we focus mainly on an algorithm known as L-BFGS, which, as its name suggests, is based on the BFGS updating formula. The main idea of this method is to use curvature information from only the most recent iterations to construct the Hessian approximation. Curvature information from earlier iterations, which is less likely to be relevant to the actual behavior of the Hessian at the current iteration, is discarded in the interest of saving storage.

Following our discussion of L-BFGS and its convergence behavior, we discuss its relationship to the nonlinear conjugate gradient methods of Section 5.4.4 (Chapter 5 of

[104]). We then discuss implementations of limited-memory schemes that make use of a compact representation of approximate Hessian information. These techniques can be applied not only to L-BFGS but also to limited-memory versions of other quasi-Newton procedures such as SR1. Finally, we discuss quasi-Newton updating schemes that impose a particular sparsity pattern on the approximate Hessian.

5.5.8 Limited-Memory BFGS

We begin our description of the L-BFGS method by recalling its parent, the BFGS method, which was described in Section 5.5.5. Each step of the BFGS method has the form

$$x_{k+1} = x_k - \alpha_k H_k \nabla f_k, \quad (5.44)$$

where α_k is the step length and H_k is updated at every iteration by means of the formula

$$H_{k+1} = V_k^T H_k V_k + \rho_k s_k s_k^T, \quad (5.45)$$

where

$$\rho_k = \frac{1}{y_k^T s_k}, \quad V_k = I - \rho_k y_k s_k^T, \quad (5.46)$$

and

$$s_k = x_{k+1} - x_k, \quad y_k = \nabla f_{k+1} - \nabla f_k. \quad (5.47)$$

Since the inverse Hessian approximation H_k will generally be dense, the cost of storing and manipulating it is prohibitive when the number of variables is large. To circumvent this problem, we store a modified version of H_k implicitly, by storing a certain number (say, m) of the vector pairs $\{s_i, y_i\}$ used in the formulas (5.45)-(5.47). The product $H_k \nabla f_k$ can be obtained by performing a sequence of inner products and vector summations involving ∇f_k and the pairs $\{s_i, y_i\}$. After the new iterate is computed, the oldest vector pair in the set of pairs $\{s_i, y_i\}$ is replaced by the new pair $\{s_k, y_k\}$ obtained from the current step (5.47). In this way, the set of vector pairs includes curvature information from the m most recent iterations. Practical experience has shown that modest values of m (between 3 and 20, say) often produce satisfactory results.

We now describe the updating process in a little more detail. At iteration k , the current iterate is x_k and the set of vector pairs is given by $\{s_i, y_i\}$ for $i = k-m, \dots, k-1$. We first choose some initial Hessian approximation H_k^0 (in contrast to the standard BFGS iteration, this initial approximation is allowed to vary from iteration to iteration) and find by repeated application of the formula (5.45) that the L-BFGS approximation H_k satisfies

the following formula:

$$\begin{aligned}
 H_k = & (V_{k-1}^T \cdots V_{k-m}^T) H_k^0 (V_{k-m} \cdots V_{k-1}) \\
 & + \rho_{k-m} (V_{k-1}^T \cdots V_{k-m+1}^T) s_{k-m} s_{k-m}^T (V_{k-m+1} \cdots V_{k-1}) \\
 & + \rho_{k-m+1} (V_{k-1}^T \cdots V_{k-m+2}^T) s_{k-m+1} s_{k-m+1}^T (V_{k-m+2} \cdots V_{k-1}) \\
 & + \cdots \\
 & + \rho_{k-1} s_{k-1} s_{k-1}^T.
 \end{aligned} \tag{5.48}$$

From this expression we can derive a recursive procedure to compute the product $H_k \nabla f_k$ efficiently.

Algorithm 1 L-BFGS two-loop recursion

```

 $q \leftarrow \nabla f_k;$ 
for  $i = k - 1, k - 2, \dots, k - m$  do
   $\alpha_i \leftarrow \rho_i s_i^T q;$ 
   $q \leftarrow q - \alpha_i y_i;$ 
end for
 $p \leftarrow H_k^0 q;$ 
for  $i = k - m, k - m + 1, \dots, k - 1$  do
   $\beta \leftarrow \rho_i y_i^T p;$ 
   $p \leftarrow p + (\alpha_i - \beta) s_i;$ 
end for
stop with result  $H_k \nabla f_k = p$ .

```

Without considering the multiplication $H_k^0 q$, the two-loop recursion scheme requires $4mn$ multiplications; if H_k^0 is diagonal, then n additional multiplications are needed. Apart from being inexpensive, this recursion has the advantage that the multiplication by the initial matrix H_k^0 is isolated from the rest of the computations, allowing this matrix to be chosen freely and to vary between iterations. We may even use an implicit choice of H_k^0 by defining some initial approximation B_k^0 to the Hessian (not its inverse) and obtaining p by solving the system $B_k^0 p = q$.

A method for choosing H_k^0 that has proved effective in practice is to set $H_k^0 = \gamma_k I$, where

$$\gamma_k = \frac{s_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}}. \tag{5.49}$$

As discussed in Section 5.5 (Chapter 6 of [104]), γ_k is the scaling factor that attempts to estimate the size of the true Hessian matrix along the most recent search direction (see

(6.21) of [104]). This choice helps to ensure that the search direction p_k is well scaled, and as a result the step length $\alpha_k = 1$ is accepted in most iterations. As discussed in Chapter 6 of [104], it is important that the line search be based on the Wolfe conditions:

$$\begin{aligned} f(x_k + \alpha_k p_k) &\leq f(x_k) + c_1 \alpha_k \nabla f_k^T p_k, \\ \nabla f(x_k + \alpha_k p_k)^T p_k &\geq c_2 \nabla f_k^T p_k, \end{aligned} \quad (5.50)$$

with $0 < c_1 < c_2 < 1$, or strong Wolfe conditions

$$\begin{aligned} f(x_k + \alpha_k p_k) &\leq f(x_k) + c_1 \alpha_k \nabla f_k^T p_k, \\ |\nabla f(x_k + \alpha_k p_k)^T p_k| &\leq c_2 |\nabla f_k^T p_k|, \end{aligned} \quad (5.51)$$

so that BFGS updating is stable.

The limited-memory BFGS algorithm can be stated formally as follows.

Algorithm 2 L-BFGS

Choose starting point x_0 , integer $m > 0$;

$k \leftarrow 0$;

repeat

 Choose H_k^0 (for example, by using (5.49));

 Compute $p_k \leftarrow -H_k \nabla f_k$ from Algorithm 1;

 Compute $x_{k+1} \leftarrow x_k + \alpha_k p_k$, where α_k is chosen to satisfy the Wolfe conditions (5.50);

if $k > m$ **then**

 Discard the vector pair $\{s_{k-m}, y_{k-m}\}$ from storage;

end if

 Compute and save $s_k \leftarrow x_{k+1} - x_k$, $y_k \leftarrow \nabla f_{k+1} - \nabla f_k$;

$k \leftarrow k + 1$;

until convergence.

The strategy of keeping the m most recent correction pairs $\{s_i, y_i\}$ works well in practice; indeed no other strategy has yet proved to be consistently better. During its first $m - 1$ iterations, Algorithm 2 is equivalent to the BFGS algorithm of Section 5.5.5 (Chapter 6 of [104]) if the initial matrix H_0 is the same in both methods, and if L-BFGS chooses $H_k^0 = H_0$ at each iteration.

Table 5.5.8 presents results illustrating the behavior of Algorithm 2 for various levels of memory m . It gives the number of function and gradient evaluations (nfg) and the total CPU time. The test problems are taken from the CUTE collection [35], the number

of variables is indicated by n , and the termination criterion $\|\nabla f_k\| \neq 10^{-5}$ is used. The table shows that the algorithm tends to be less robust when m is small. As the amount of storage increases, the number of function evaluations tends to decrease; but since the cost of each iteration increases with the amount of storage, the best CPU time is often obtained for small values of m . Clearly, the optimal choice of m is problem dependent.

表 5.1: Performance of Algorithm 2

Problem	n	L-BFGS		L-BFGS		L-BFGS		L-BFGS	
		$m = 3$	$m = 5$	$m = 17$	$m = 29$	nfg	time	nfg	time
DIXMAANL	1500	146	16.5	134	17.4	120	28.2	125	44.4
EIGENALS	110	821	21.5	569	15.7	363	16.2	168	12.5
FREUROTH	1000	> 999	—	> 999	—	69	8.1	38	6.3
TRIDIA	1000	876	46.6	611	41.4	531	84.6	462	127.1

Because some rival algorithms are inefficient, Algorithm 2 is often the approach of choice for large problems in which the true Hessian is not sparse. In particular, a Newton method in which the exact Hessian is computed and factorized is not practical in such circumstances. The L-BFGS approach may also outperform Hessian-free Newton methods such as Newton-CG approaches, in which Hessian-vector products are calculated by finite differences or automatic differentiation. The main weakness of the L-BFGS method is that it converges slowly on ill-conditioned problems — specifically, on problems where the Hessian matrix contains a wide distribution of eigenvalues. On certain applications, the nonlinear conjugate gradient methods discussed in Section 5.4.4 (Chapter 5 of [104]) are competitive with limited-memory quasi-Newton methods.

5.5.9 Relationship with Conjugate Gradient Methods

Limited-memory methods evolved as an attempt to improve nonlinear conjugate gradient methods, and early implementations resembled conjugate gradient methods more than quasi-Newton methods. The relationship between the two classes is the basis of a memoryless BFGS iteration, which we now outline.

We start by considering the Hestenes-Stiefel form of the nonlinear conjugate gradient method (5.40). Recalling that $s_k = \alpha_k p_k$, we have that the search direction for this method is given by

$$p_{k+1} = -\nabla f_{k+1} + \frac{\nabla f_{k+1}^T y_k}{y_k^T p_k} p_k = -\left(I - \frac{s_k y_k^T}{y_k^T s_k}\right) \nabla f_{k+1} \equiv -\hat{H}_{k+1} \nabla f_{k+1}. \quad (5.52)$$

This formula resembles a quasi-Newton iteration, but the matrix \hat{H}_{k+1} is neither symmetric nor positive definite. We could symmetrize it as $\hat{H}_{k+1}^T \hat{H}_{k+1}$, but this matrix does not satisfy the secant equation $\hat{H}_{k+1} y_k = s_k$ and is, in any case, singular. An iteration matrix that is symmetric, positive definite, and satisfies the secant equation is given by

$$H_{k+1} = \left(I - \frac{s_k y_k^T}{y_k^T s_k} \right) \left(I - \frac{y_k s_k^T}{y_k^T s_k} \right) + \frac{s_k s_k^T}{y_k^T s_k}. \quad (5.53)$$

This matrix is exactly the one obtained by applying a single BFGS update (5.45) to the identity matrix. Hence, an algorithm whose search direction is given by $p_{k+1} = -H_{k+1} \nabla f_{k+1}$, with H_{k+1} defined by (5.53), can be thought of as a “memoryless” BFGS method, in which the previous Hessian approximation is always reset to the identity matrix before updating it and where only the most recent correction pair (s_k, y_k) is kept at every iteration. Alternatively, we can view the method as a variant of Algorithm 2 in which $m = 1$ and $H_k^0 = I$ at each iteration.

A more direct connection with conjugate gradient methods can be seen if we consider the memoryless BFGS formula (5.53) in conjunction with an exact line search, for which $\nabla f_{k+1}^T p_k = 0$ for all k . We then obtain

$$p_{k+1} = -H_{k+1} \nabla f_{k+1} = -\nabla f_{k+1} + \frac{\nabla f_{k+1}^T y_k}{y_k^T p_k} p_k, \quad (5.54)$$

which is none other than the Hestenes-Stiefel conjugate gradient method. Moreover, it is easy to verify that when $\nabla f_{k+1}^T p_k = 0$, the Hestenes-Stiefel formula reduces to the Polak-Ribiere formula (5.41). Even though the assumption of exact line searches is unrealistic, it is intriguing that the BFGS formula is related in this way to the Polak-Ribiere and Hestenes-Stiefel methods.

5.5.10 Exercises

(Taken from Chapter 8 of [8])

Exercise 444. Consider the problem to minimize $(3 - x_1)^2 + 7(x_2 - x_1^2)^2$. Starting from the point $(0, 0)^T$, solve the problem by the following procedures:

- a. The cyclic coordinate method.
- b. The method of Hooke and Jeeves.
- c. The method of Rosenbrock.
- d. The method of Davidon-Fletcher-Powell.

e. The method of Broyden-Fletcher-Goldfarb-Shanno (BFGS).

Exercise 445. Consider the problem to maximize $-2x_1^2 - 3x_2^2 + 3x_1x_2 - 2x_1 + 4x_2$. Starting from the origin, solve the problem by the Davidon-Fletcher-Powell method, with \mathbf{D}_l as the identity. Also solve the problem by the Fletcher and Reeves conjugate gradient method. Note that the two procedures generate identical sets of directions. Show that, in general, if $\mathbf{D}_l = \mathbf{I}$, then the two methods are identical for quadratic functions.

Exercise 446. Solve the problem to minimize $2x_1 + 3x_2^2 + \exp(2x_1^2 + x_2^2)$, starting with the point $(1, 0)^T$ and using both the Fletcher and Reeves conjugate gradient method and the BFGS quasi-Newton method.

(Taken from Chapter 7 of [104])

Exercise 447. Use L-BFGS to solve extended Rosenbrock function

$$f(\mathbf{x}) = \sum_{i=1}^{n/2} [\alpha(x_{2i} - x_{2i-1}^2)^2 + (1 - x_{2i-1})^2],$$

where α is a parameter that you can vary (for example, 1 or 100). The solution is $\mathbf{x}^* = (1, 1, \dots, 1)^T$, $f^* = 0$. Choose the starting point as $(-1, -1, \dots, -1)^T$. Observe the behavior of your program for various values of the memory parameter m .

(Taken from Chapter 11 of [73])

Exercise 448. Let \mathbf{M} be an $n \times n$ matrix and \mathbf{d} and \mathbf{g} be $n \times 1$ vectors. Show that the matrix

$$\mathbf{N}_{opt} = \mathbf{M} + \|\mathbf{d}\|^2(\mathbf{g} - \mathbf{Md})\mathbf{d}^T$$

minimizes the distance $\|\mathbf{N} - \mathbf{M}\|$ between \mathbf{M} and an arbitrary $n \times n$ matrix \mathbf{N} subject to the secant condition $\mathbf{Nd} = \mathbf{g}$. Unfortunately, the rank-one update \mathbf{N}_{opt} is not symmetric when \mathbf{M} is symmetric.

Exercise 449. Let \mathbf{M} be an $n \times n$ symmetric matrix and \mathbf{d} and \mathbf{g} be $n \times 1$ vectors. Powell proposed the rank-two update

$$\mathbf{N}_{opt} = \mathbf{M} + \frac{(\mathbf{g} - \mathbf{Md})\mathbf{d}^T + \mathbf{d}(\mathbf{g} - \mathbf{Md})^T}{\|\mathbf{d}\|^2} - \frac{(\mathbf{g} - \mathbf{Md})^T \mathbf{d} \mathbf{d}^T}{\|\mathbf{d}\|^4}$$

to \mathbf{M} . Show that \mathbf{N}_{opt} is symmetric, has rank two, and satisfies the secant condition $\mathbf{N}_{opt}\mathbf{d} = \mathbf{g}$.

Algorithm 3 Basic majorization-minimization (MM) scheme.

- 1: **Input:** $\boldsymbol{\theta}_0 \in \Theta$ (initial estimate); N (number of iterations).
 - 2: for $n = 1, \dots, N$ do
 - 3: Compute a surrogate function g_n of f near $\boldsymbol{\theta}_{n-1}$;
 - 4: Minimize the surrogate and update the solution: $\boldsymbol{\theta}_n \in \operatorname{argmin}_{\boldsymbol{\theta} \in \Theta} g_n(\boldsymbol{\theta})$.
 - 5: end for
 - 6: **Output:** $\boldsymbol{\theta}_N$ (final estimate).
-

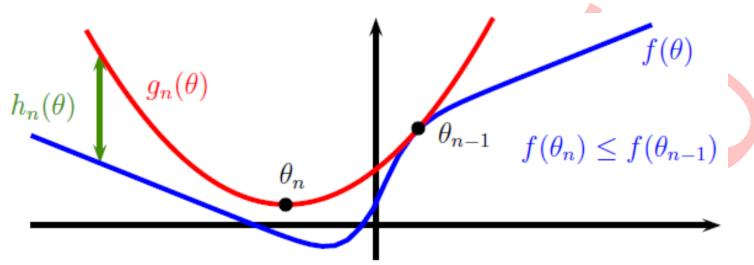


图 5.21: Illustration of the basic majorization-minimization principle. We compute a surrogate g_n of f near the current estimate $\boldsymbol{\theta}_{n-1}$. The new estimate $\boldsymbol{\theta}_n$ is a minimizer of g_n . The function $h_n = g_n - f$ is the approximation error that is made when replacing f by g_n .

Exercise 450. Continuing Exercise 449, show that the matrix \mathbf{N}_{opt} minimizes the distance $\|\mathbf{N} - \mathbf{M}\|_F$ between \mathbf{M} and an arbitrary $n \times n$ symmetric matrix \mathbf{N} subject to the secant condition $\mathbf{N}\mathbf{d} = \mathbf{g}$. Here $\|\mathbf{A}\|_F$ denotes the Frobenius norm of the matrix \mathbf{A} viewed as a vector. Unfortunately, Powell's update does not preserve positive definiteness.

Exercise 451. Consider the quadratic function

$$Q(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} \mathbf{x} + (1, 1)\mathbf{x}$$

defined on \mathbb{R}^2 . Compute by hand the iterates of the conjugate gradient and BFGS algorithms starting from $\mathbf{x}_1 = \mathbf{0}$. For the BFGS algorithm take $\mathbf{H}_1 = \mathbf{I}$ and use an exact line search. You should find that the two sequences of iterates coincide. This phenomenon holds more generally for any strictly convex quadratic function in the BFGS algorithm given $\mathbf{H}_1 = \mathbf{I}$.

5.6 Majorization minimization with first-order surrogate functions

(Taken from [97])

We present the generic majorization-minimization (MM) scheme for minimizing a function f . We describe the procedure in Algorithm 3 and illustrate its principle in Figure 5.21. At iteration n , the estimate $\boldsymbol{\theta}_n$ is obtained by minimizing a surrogate function g_n of f . When g_n uniformly majorizes f and when $g_n(\boldsymbol{\theta}_{n-1}) = f(\boldsymbol{\theta}_{n-1})$, it is clear that the objective function value monotonically decreases:

$$f(\boldsymbol{\theta}_n) \leq g_n(\boldsymbol{\theta}_n) \leq g_n(\boldsymbol{\theta}_{n-1}) = f(\boldsymbol{\theta}_{n-1}). \quad (5.55)$$

For this approach to be effective, intuition tells us that we need functions g_n that are easy to minimize and that approximate well the objective f . Therefore, we measure the quality of the approximation through the smoothness of the error $h_n = g_n - f$, which is a key quantity arising in the convergence analysis. Specifically, we require h_n to be L -smooth for some constant $L > 0$ in the following sense:

Definition 452 (L -smooth functions). A function $f : \mathbb{R}^p \rightarrow \mathbb{R}$ is called L -smooth when it is differentiable and when its gradient ∇f is L -Lipschitz continuous.

With this definition in hand, we now introduce the class of “first-order surrogate functions”, which will be shown to have good enough properties for analyzing the convergence of Algorithm 3 and other variants.

Definition 453 (First-order surrogate functions). A function $g : \mathbb{R}^p \rightarrow \mathbb{R}$ is a first-order surrogate function of f near $\boldsymbol{\kappa}$ in Θ when

- (i) $g(\boldsymbol{\theta}') \geq f(\boldsymbol{\theta}')$ for all minimizers $\boldsymbol{\theta}'$ of g over Θ . When the more general condition $g \geq f$ holds, we say that g is a majorizing surrogate;
- (ii) the approximation error $h \triangleq g - f$ is L -smooth, $h(\boldsymbol{\kappa}) = 0$, and $\nabla h(\boldsymbol{\kappa}) = 0$. We denote by $\mathcal{S}_L(f, \boldsymbol{\kappa})$ the set of first-order surrogate functions and by $\mathcal{S}_{L,\rho}(f, \boldsymbol{\kappa}) \subset \mathcal{S}_L(f, \boldsymbol{\kappa})$ the subset of ρ -strongly convex surrogates.

First-order surrogates are interesting because their approximation error – the difference between the surrogate and the objective – can be easily controlled. This is formally stated in the next lemma, which is a building block of our analysis:

Lemma 454 (Basic properties of first-order surrogate functions). Let g be a surrogate function in $\mathcal{S}_L(f, \boldsymbol{\kappa})$ for some $\boldsymbol{\kappa}$ in Θ . Define the approximation error $h \triangleq g - f$, and let $\boldsymbol{\theta}'$ be a minimizer of g over Θ . Then, for all $\boldsymbol{\theta} \in \Theta$,

$$(1) |h(\boldsymbol{\theta})| \leq \frac{L}{2} \|\boldsymbol{\theta} - \boldsymbol{\kappa}\|_2^2;$$

$$(2) \quad f(\boldsymbol{\theta}') \leq f(\boldsymbol{\theta}) + \frac{L}{2} \|\boldsymbol{\theta} - \boldsymbol{\kappa}\|_2^2.$$

(3) Assume that g is ρ -strongly convex, i.e., g is in $\mathcal{S}_{L,\rho}(f, \boldsymbol{\kappa})$. Then, for all $\boldsymbol{\theta} \in \Theta$,

$$f(\boldsymbol{\theta}') + \frac{\rho}{2} \|\boldsymbol{\theta}' - \boldsymbol{\theta}\|_2^2 \leq f(\boldsymbol{\theta}) + \frac{L}{2} \|\boldsymbol{\theta} - \boldsymbol{\kappa}\|_2^2.$$

Proof. The first inequality is a direct application of a classical result on quadratic upper bounds for L -smooth functions, when noticing that $h(\boldsymbol{\kappa}) = 0$ and $\nabla h(\boldsymbol{\kappa}) = 0$. Then, for all $\boldsymbol{\theta} \in \Theta$, we have $f(\boldsymbol{\theta}') \leq g(\boldsymbol{\theta}') \leq g(\boldsymbol{\theta}) = f(\boldsymbol{\theta}) + h(\boldsymbol{\theta})$, and we obtain the second inequality from the first one.

When g is ρ -strongly convex, we use the following classical lower bound:

$$g(\boldsymbol{\theta}') + \frac{\rho}{2} \|\boldsymbol{\theta}' - \boldsymbol{\theta}\|_2^2 \leq g(\boldsymbol{\theta}).$$

Since $f(\boldsymbol{\theta}') \leq g(\boldsymbol{\theta}')$ by Definition 453 and $g(\boldsymbol{\theta}) = f(\boldsymbol{\theta}) + h(\boldsymbol{\theta})$, the third inequality follows from the first one. \square

5.6.1 Non-convex convergence analysis

For general non-convex problems, proving convergence to a global (or local) minimum is impossible in general, and classical analysis studies instead asymptotic stationary point conditions. To do so, we make the following mild assumption when f is non-convex:

- (A) f is bounded below and for all $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta$, the directional derivative $\nabla f(\boldsymbol{\theta}; \boldsymbol{\theta}' - \boldsymbol{\theta})$ of f at $\boldsymbol{\theta}$ in the direction $\boldsymbol{\theta}' - \boldsymbol{\theta}$ exists.

The definitions of stationary points is provided below.

Definition 455 (Stationary point). Consider a function $f : \Theta \subseteq \mathbb{R}^p \rightarrow \mathbb{R}$, where Θ is a convex set, such that f admits a directional derivative $\nabla f(\boldsymbol{\theta}; \boldsymbol{\theta}' - \boldsymbol{\theta})$ for all $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta$. We say that $\boldsymbol{\theta} \in \Theta$ is a stationary point if for all $\boldsymbol{\theta}' \in \Theta$, $\nabla f(\boldsymbol{\theta}; \boldsymbol{\theta}' - \boldsymbol{\theta}) \geq 0$.

A necessary first-order condition for $\boldsymbol{\theta}$ to be a local minimum of f is that it is a stationary point defined above. Thus, we consider the following condition for assessing the quality of a sequence $\{\boldsymbol{\theta}_n\}_{n \geq 0}$ for non-convex problems:

Definition 456 (Asymptotic stationary point). Under assumption (A), a sequence $\{\boldsymbol{\theta}_n\}_{n \geq 0}$ satisfies the asymptotic stationary point condition if

$$\liminf_{n \rightarrow +\infty} \inf_{\boldsymbol{\theta} \in \Theta} \frac{\nabla f(\boldsymbol{\theta}_n; \boldsymbol{\theta} - \boldsymbol{\theta}_n)}{\|\boldsymbol{\theta} - \boldsymbol{\theta}_n\|_2} \geq 0. \quad (5.56)$$

Note that if f is differentiable on \mathbb{R}^p and $\Theta = \mathbb{R}^p$, $\nabla f(\boldsymbol{\theta}_n; \boldsymbol{\theta} - \boldsymbol{\theta}_n) = \nabla f(\boldsymbol{\theta}_n)^T (\boldsymbol{\theta} - \boldsymbol{\theta}_n)$, and the condition (5.56) implies that the sequence $\{\nabla f(\boldsymbol{\theta}_n)\}_{n \geq 0}$ converges to $\mathbf{0}$. As noted, we recover the classical definition of critical points for the smooth unconstrained case. We now give a first convergence result about Algorithm 3.

Proposition 457 (Non-convex analysis for Algorithm 3). *Assume that (A) holds and that the surrogates g_n from Algorithm 3 are in $\mathcal{S}_L(f, \boldsymbol{\theta}_{n-1})$ and are either majorizing f or strongly convex. Then, $\{f(\boldsymbol{\theta}_n)\}_{n \geq 0}$ monotonically decreases, and $\{\boldsymbol{\theta}_n\}_{n \geq 0}$ satisfies the asymptotic stationary point condition.*

Proof. The fact that $\{f(\boldsymbol{\theta}_n)\}_{n \geq 0}$ is non-increasing and convergent has been shown in (5.55).

Let us now denote by f^* the limit of the sequence $\{f(\boldsymbol{\theta}_n)\}_{n \geq 0}$ and by $h_n \triangleq g_n - f$ the approximation error function at iteration n , which is L -smooth by Definition 453 and such that $h_n(\boldsymbol{\theta}_n) \geq 0$. Then, $h_n(\boldsymbol{\theta}_n) = g_n(\boldsymbol{\theta}_n) - f(\boldsymbol{\theta}_n) \leq f(\boldsymbol{\theta}_{n-1}) - f(\boldsymbol{\theta}_n)$, and

$$\sum_{n=1}^{\infty} h_n(\boldsymbol{\theta}_n) \leq f(\boldsymbol{\theta}_0) - f^*.$$

Thus, the non-negative sequence $\{h_n(\boldsymbol{\theta}_n)\}_{n \geq 0}$ necessarily converges to zero. Then, we have two possibilities (according to the assumptions made in the proposition).

- (1) If the functions g_n are majorizing f , we define $\boldsymbol{\theta}' = \boldsymbol{\theta}_n - \frac{1}{L} \nabla h_n(\boldsymbol{\theta}_n)$, and we use the following classical inequality for L -smooth functions:

$$h_n(\boldsymbol{\theta}') \leq h_n(\boldsymbol{\theta}_n) - \frac{1}{2L} \|\nabla h_n(\boldsymbol{\theta}_n)\|_2^2.$$

Therefore, we may use the fact that $h_n(\boldsymbol{\theta}') \geq 0$ because $g_n \geq f$, and

$$\|\nabla h_n(\boldsymbol{\theta}_n)\|_2^2 \leq 2L(h_n(\boldsymbol{\theta}_n) - h_n(\boldsymbol{\theta}')) \leq 2Lh_n(\boldsymbol{\theta}_n) \rightarrow 0.$$

- (2) If instead the functions g_n are ρ -strongly convex, the last inequality of Lemma 454 with $\boldsymbol{\kappa} = \boldsymbol{\theta} = \boldsymbol{\theta}_{n-1}$ and $\boldsymbol{\theta}' = \boldsymbol{\theta}_n$ gives us

$$\frac{\rho}{2} \|\boldsymbol{\theta}_n - \boldsymbol{\theta}_{n-1}\|_2^2 \leq f(\boldsymbol{\theta}_{n-1}) - f(\boldsymbol{\theta}_n).$$

By summing over n , we obtain that $\|\boldsymbol{\theta}_n - \boldsymbol{\theta}_{n-1}\|_2^2$ converges to zero, and

$$\|\nabla h_n(\boldsymbol{\theta}_n)\|_2 = \|\nabla h_n(\boldsymbol{\theta}_n) - \nabla h_n(\boldsymbol{\theta}_{n-1})\|_2 \leq L \|\boldsymbol{\theta}_n - \boldsymbol{\theta}_{n-1}\|_2 \rightarrow 0,$$

since $\nabla h_n(\boldsymbol{\theta}_{n-1}) = \mathbf{0}$ according to Definition 453.

We now consider the directional derivative of f at $\boldsymbol{\theta}_n$ and a direction $\boldsymbol{\theta} - \boldsymbol{\theta}_n$, where $n \geq 1$ and $\boldsymbol{\theta}$ is in Θ ,

$$\nabla f(\boldsymbol{\theta}_n; \boldsymbol{\theta} - \boldsymbol{\theta}_n) = \nabla g_n(\boldsymbol{\theta}_n; \boldsymbol{\theta} - \boldsymbol{\theta}_n) - \nabla h_n(\boldsymbol{\theta}_n)^T(\boldsymbol{\theta} - \boldsymbol{\theta}_n).$$

Note that $\boldsymbol{\theta}_n$ minimizes g_n on Θ and therefore $\nabla g_n(\boldsymbol{\theta}_n; \boldsymbol{\theta} - \boldsymbol{\theta}_n) \geq 0$. Therefore,

$$\nabla f(\boldsymbol{\theta}_n; \boldsymbol{\theta} - \boldsymbol{\theta}_n) \geq -\|\nabla h_n(\boldsymbol{\theta}_n)\|_2 \|\boldsymbol{\theta} - \boldsymbol{\theta}_n\|_2,$$

by Cauchy-Schwarz's inequality. By minimizing over $\boldsymbol{\theta}$ and taking the infimum limit, we finally obtain

$$\liminf_{n \rightarrow +\infty} \inf_{\boldsymbol{\theta} \in \Theta} \frac{\nabla f(\boldsymbol{\theta}_n; \boldsymbol{\theta} - \boldsymbol{\theta}_n)}{\|\boldsymbol{\theta} - \boldsymbol{\theta}_n\|_2} \geq -\lim_{n \rightarrow +\infty} \|\nabla h_n(\boldsymbol{\theta}_n)\|_2 = 0.$$

□

This proposition provides convergence guarantees for a large class of existing algorithms, including cases where f is non-smooth.

5.6.2 Examples of first-order surrogate functions

We now present practical first-order surrogate functions and links between Algorithm 3 and existing approaches. Even though our generic analysis does not always bring new results for each specific case, its main asset is to provide a unique theoretical treatment to all of them.

5.6.2.1 Lipschitz gradient surrogates

When f is L -smooth, it is natural to consider the following surrogate:

$$g : \boldsymbol{\theta} \mapsto f(\boldsymbol{\kappa}) + \nabla f(\boldsymbol{\kappa})^T(\boldsymbol{\theta} - \boldsymbol{\kappa}) + \frac{L}{2} \|\boldsymbol{\theta} - \boldsymbol{\kappa}\|_2^2.$$

The function g is an upper bound of f , which is a classical result. It is then easy to see that g is L -strongly convex and L -smooth. As a consequence, the difference $g - f$ is $2L$ -smooth (as a sum of two L -smooth functions), and thus g is in $\mathcal{S}_{2L,L}(f, \boldsymbol{\kappa})$.

Lemma 458 (Regularity of residual functions). *Let $f, g : \mathbb{R}^p \rightarrow \mathbb{R}$ be two functions. Define $h \triangleq g - f$. Then, if g is ρ -strongly convex and f is L -smooth, with $\rho \geq L$, h is $(\rho - L)$ -strongly convex; if g and f are convex and L -smooth, h is also L -smooth; if g and f are μ -strongly convex and L -smooth, h is $(L - \mu)$ -smooth.*

When f is convex, it is also possible to show by using Lemma 458 that g is in fact in $\mathcal{S}_{L,L}(f, \boldsymbol{\kappa})$, and when f is μ -strongly convex, g is in $\mathcal{S}_{L-\mu,L}(f, \boldsymbol{\kappa})$. We remark that minimizing g amounts to performing a gradient descent step: $\boldsymbol{\theta}' \leftarrow \boldsymbol{\kappa} - \frac{1}{L} \nabla f(\boldsymbol{\kappa})$.

5.6.2.2 Proximal gradient surrogates

Let us now consider a composite optimization problem, meaning that f splits into two parts $f = f_1 + f_2$, where f_1 is L -smooth. Then, a natural surrogate of f is the following function:

$$g : \boldsymbol{\theta} \mapsto f_1(\boldsymbol{\kappa}) + \nabla f_1(\boldsymbol{\kappa})^T (\boldsymbol{\theta} - \boldsymbol{\kappa}) + \frac{L}{2} \|\boldsymbol{\theta} - \boldsymbol{\kappa}\|_2^2 + f_2(\boldsymbol{\theta}).$$

The function g majorizes f and the approximation error $g - f$ is the same as in Section 5.6.2.1. Thus, g is in $\mathcal{S}_{2L}(f, \boldsymbol{\kappa})$ or in $\mathcal{S}_{2L,L}(f, \boldsymbol{\kappa})$ when f_2 is convex. Moreover,

- (1) when f_1 is convex, g is in $\mathcal{S}_L(f, \boldsymbol{\kappa})$. If f_2 is also convex, g is in $\mathcal{S}_{L,L}(f, \boldsymbol{\kappa})$;
- (2) when f_1 is μ -strongly convex, g is in $\mathcal{S}_{L-\mu}(f, \boldsymbol{\kappa})$. If f_2 is also convex, g is in $\mathcal{S}_{L-\mu,L}(f, \boldsymbol{\kappa})$.

Minimizing g amounts to performing one step of the proximal gradient algorithm. It is indeed easy to show that the minimum $\boldsymbol{\theta}'$ of g – assuming it is unique – can be equivalently obtained as follows:

$$\boldsymbol{\theta}' = \underset{\boldsymbol{\theta} \in \Theta}{\operatorname{argmin}} \left[\frac{1}{2} \left\| \boldsymbol{\theta} - \left(\boldsymbol{\kappa} - \frac{1}{L} \nabla f_1(\boldsymbol{\kappa}) \right) \right\|_2^2 + \frac{1}{L} f_2(\boldsymbol{\theta}) \right],$$

which is often written under the form $\boldsymbol{\theta}' = \operatorname{Prox}_{1/L} f_2(\boldsymbol{\kappa} - (1/L) \nabla f_1(\boldsymbol{\kappa}))$, where “Prox” is called the “proximal operator”. In some cases, the proximal operator can be computed efficiently in closed form, for example when f_2 is the ℓ_1 -norm; it yields the iterative soft-thresholding algorithm for sparse estimation.

5.6.2.3 Linearizing concave functions and DC programming

Assume that $f = f_1 + f_2$, where f_2 is concave and L -smooth. Then, the following function g is a majorizing surrogate in $\mathcal{S}_L(f, \boldsymbol{\kappa})$:

$$g : \boldsymbol{\theta} \mapsto f_1(\boldsymbol{\theta}) + f_2(\boldsymbol{\kappa}) + \nabla f_2(\boldsymbol{\kappa})^T (\boldsymbol{\theta} - \boldsymbol{\kappa}).$$

Such a surrogate appears in DC (difference of convex) programming. When f_1 is convex, f is indeed the difference of two convex functions. It is also used in sparse estimation for dealing with some non-convex penalties. For example, consider a cost function of the form $\boldsymbol{\theta} \mapsto f_1(\boldsymbol{\theta}) + \lambda \sum_{j=1}^p \log(|\boldsymbol{\theta}(j)| + \varepsilon)$, where $\boldsymbol{\theta}(j)$ is the j -th entry in $\boldsymbol{\theta}$. Even though the functions $\boldsymbol{\theta} \mapsto \log(|\boldsymbol{\theta}(j)| + \varepsilon)$ are not differentiable, they can be written as the composition of a concave smooth function $u \mapsto \log(u + \varepsilon)$ on \mathbb{R}^+ , and a Lipschitz function $\boldsymbol{\theta} \mapsto |\boldsymbol{\theta}(j)|$.

By upper-bounding the logarithm function by its linear approximation, Proposition 2.6 of [97] justifies the following surrogate:

$$g : \boldsymbol{\theta} \mapsto f_1(\boldsymbol{\theta}) + \lambda \sum_{j=1}^p \log(|\kappa(j)| + \varepsilon) + \lambda \sum_{j=1}^p \frac{|\boldsymbol{\theta}(j)| - |\kappa(j)|}{|\kappa(j)| + \varepsilon}, \quad (5.57)$$

and minimizing g amounts to performing one step of a reweighted- ℓ_1 algorithm. Similarly, other penalty functions are adapted to this framework. For instance, the logarithm can be replaced by any smooth concave nondecreasing function, or group-sparsity penalties can be used, such as $\boldsymbol{\theta} \mapsto \sum_{g \in \mathcal{G}} \log(\|\boldsymbol{\theta}_g\|_2 + \varepsilon)$, where \mathcal{G} is a partition of $\{1, \dots, p\}$ and $\boldsymbol{\theta}_g$ records the entries of $\boldsymbol{\theta}$ corresponding to the set g .

5.6.2.4 Variational surrogates

Let us now consider a real-valued function f defined on $\mathbb{R}^{p_1} \times \mathbb{R}^{p_2}$. Let $\Theta_1 \subseteq \mathbb{R}^{p_1}$ and $\Theta_2 \subseteq \mathbb{R}^{p_2}$ be two convex sets. Minimizing f over $\Theta_1 \times \Theta_2$ is equivalent to minimizing the function \tilde{f} over Θ_1 defined as $\tilde{f}(\boldsymbol{\theta}_1) \triangleq \min_{\boldsymbol{\theta}_2 \in \Theta_2} f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$. Assume now that

- (1) $\boldsymbol{\theta} \mapsto f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is μ -strongly convex for all $\boldsymbol{\theta}_1$ in \mathbb{R}^{p_1} ;
- (2) $\boldsymbol{\theta}_1 \mapsto f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is differentiable for all $\boldsymbol{\theta}_2$;
- (3) $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \mapsto \nabla_1 f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is L' -Lipschitz with respect to $\boldsymbol{\theta}_1$ and L -Lipschitz with respect to $\boldsymbol{\theta}_2$.

Let us fix $\boldsymbol{\kappa}_1$ in Θ_1 . Then, the following function is a majorizing surrogate in $\mathcal{S}_{L''}(\tilde{f}, \boldsymbol{\kappa})$:

$$g : \boldsymbol{\theta}_1 \mapsto f(\boldsymbol{\theta}_1, \boldsymbol{\kappa}_2^*) \text{ with } \boldsymbol{\kappa}_2^* \triangleq \operatorname{argmin}_{\boldsymbol{\theta}_2 \in \Theta_2} f(\boldsymbol{\kappa}_1, \boldsymbol{\theta}_2),$$

with $L'' = 2L' + L^2/\mu$. We can indeed apply Lemma 459, which ensures that \tilde{f} is differentiable with $\nabla \tilde{f}(\boldsymbol{\theta}_1) = \nabla_1 f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2^*)$ and $\boldsymbol{\theta}_2^* \triangleq \operatorname{argmin}_{\boldsymbol{\theta}_2 \in \Theta_2} f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ for all $\boldsymbol{\theta}_1$. Moreover, g is L' -smooth and \tilde{f} is $(L' + L^2/\mu)$ -smooth according to Lemma 459, and thus $h \triangleq g - \tilde{f}$ is L'' -smooth. Note that a better constant $L'' = L'$ can be obtained when f is convex.

Lemma 459 (Regularity of optimal value functions). *Let $f : \mathbb{R}^{p_1} \times \Theta_2 \rightarrow \mathbb{R}$ be a function of two variables where $\Theta_2 \subseteq \mathbb{R}^{p_2}$ is a convex set. Assume that*

1. $\boldsymbol{\theta}_1 \mapsto f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is differentiable for all $\boldsymbol{\theta}_2 \in \Theta_2$;
2. $\boldsymbol{\theta}_2 \mapsto \nabla f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is L -Lipschitz continuous for all $\boldsymbol{\theta}_1 \in \mathbb{R}^{p_1}$;
3. $\boldsymbol{\theta}_2 \mapsto f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is μ -strongly convex for all $\boldsymbol{\theta}_1 \in \mathbb{R}^{p_1}$.

Also define $\tilde{f}(\boldsymbol{\theta}_1) \triangleq \operatorname{argmin}_{\boldsymbol{\theta}_2 \in \Theta_2} f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$. Then, \tilde{f} is differentiable and $\nabla \tilde{f}(\boldsymbol{\theta}_1) = \nabla_1 f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2^*)$, where $\boldsymbol{\theta}_2^* \triangleq \operatorname{argmin}_{\boldsymbol{\theta}_2 \in \Theta_2} f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$. Moreover, if $\boldsymbol{\theta}_1 \mapsto \nabla_1 f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is L' -Lipschitz continuous for all $\boldsymbol{\theta}_1 \in \mathbb{R}^{p_1}$, the gradient $\nabla \tilde{f}$ is $(L' + L^2/\mu)$ -Lipschitz.

The surrogate g leads to an alternate minimization algorithm. Variational surrogates might also be useful for problems of a single variable $\boldsymbol{\theta}_1$. For instance, consider a regression problem with a Huber loss function H defined for all u in \mathbb{R} as

$$H(u) \triangleq \begin{cases} \frac{u^2}{2\delta} + \frac{\delta}{2}, & \text{if } |u| \leq \delta, \\ |u|, & \text{otherwise,} \end{cases}$$

where δ is a positive constant. The Huber loss can be seen as a smoothed version of the ℓ_1 -norm when δ is small, or simply a robust variant of the squared loss $u \mapsto \frac{1}{2}u^2$ that asymptotically grows linearly. Then, it is easy to show that

$$H(u) = \frac{1}{2} \min_{w \geq \delta} \left(\frac{u^2}{w} + w \right).$$

Consider now a regression problem with m training data points represented by vectors \mathbf{x}_i in \mathbb{R}^p , associated to real numbers y_i , for $i = 1, \dots, m$. The robust regression problem with the Huber loss can be formulated as the minimization over \mathbb{R}^p of

$$\tilde{f} : \boldsymbol{\theta}_1 \mapsto \sum_{i=1}^m H(y_i - \mathbf{x}_i^T \boldsymbol{\theta}_1) = \min_{\boldsymbol{\theta}_2 \in \mathbb{R}^m : \boldsymbol{\theta}_2 \geq \delta} \left[f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \triangleq \frac{1}{2} \sum_{i=1}^m \left(\frac{(y_i - \mathbf{x}_i^T \boldsymbol{\theta}_1)^2}{\boldsymbol{\theta}_2(i)} + \boldsymbol{\theta}_2(i) \right) \right],$$

where $\boldsymbol{\theta}_1$ is the parameter vector of a linear model. The conditions described at the beginning of this section can be shown to be satisfied with a Lipschitz constant proportional to $(1/\delta)$; the resulting algorithm is the iterative reweighted least-square method, which appears both in the literature about robust statistics, and about sparse estimation where the Huber loss is used to approximate the ℓ_1 -norm.

5.6.2.5 Jensen surrogates

Jensen's inequality also provides a natural mechanism to obtain surrogates for convex functions. We consider a convex function $f : \mathbb{R} \rightarrow \mathbb{R}$, a vector $\mathbf{x} \in \mathbb{R}^p$, and define $\tilde{f} : \mathbb{R}^p \rightarrow \mathbb{R}$ as $\tilde{f}(\boldsymbol{\theta}) \triangleq f(\mathbf{x}^T \boldsymbol{\theta})$ for all $\boldsymbol{\theta}$. Let \mathbf{w} be a weight vector in \mathbf{R}_+^p such that $\|\mathbf{w}\|_1 = 1$ and $\mathbf{w}(i) \neq 0$ whenever $\mathbf{x}(i) \neq 0$. Then, we define for any $\boldsymbol{\kappa} \in \mathbb{R}^p$:

$$g : \boldsymbol{\theta} \mapsto \sum_{i=1}^p \mathbf{w}(i) f \left(\frac{\mathbf{x}(i)}{\mathbf{w}(i)} (\boldsymbol{\theta}(i) - \boldsymbol{\kappa}(i)) + \mathbf{x}^T \boldsymbol{\kappa} \right).$$

When f is L -smooth, and when $\mathbf{w}(i) \triangleq |\mathbf{x}(i)|^\nu / \|\mathbf{x}\|_\nu^\nu$, g is in $\mathcal{S}_{L'}(\tilde{f}, \boldsymbol{\kappa})$ with

1. $L' = L\|\mathbf{x}\|_\infty^2\|\mathbf{x}\|_0$ for $\nu = 0$;
2. $L' = L\|\mathbf{x}\|_\infty\|\mathbf{x}\|_1$ for $\nu = 1$;
3. $L' = L\|\mathbf{x}\|_2^2$ for $\nu = 2$.

To the best of our knowledge, non-asymptotic convergence rates have not been studied before for such surrogates, and thus we believe that our analysis may provide new results in the present case. Jensen surrogates are indeed quite uncommon; they appear nevertheless in a few occasions. In addition to the few examples given in [35] of [97], they are used for instance in machine learning by Della Pietra ([19] of [97]) for interpreting boosting procedures through the concept of auxiliary functions.

Jensen's inequality is also used in a different fashion in EM algorithms. Consider T non-negative functions $f^t : \mathbb{R}^p \rightarrow \mathbb{R}_+$, and, for some $\kappa \in \mathbb{R}^p$, define some weights $\mathbf{w}(t) = f^t(\kappa) / \sum_{t'=1}^T f^{t'}(\kappa)$. By exploiting the concavity of the logarithm, and assuming that $\mathbf{w}(t) > 0$ for all t to simplify, Jensen's inequality yields

$$-\log \left(\sum_{t=1}^T f^t(\boldsymbol{\theta}) \right) \leq -\sum_{t=1}^T \mathbf{w}(t) \log \left(\frac{f^t(\boldsymbol{\theta})}{\mathbf{w}(t)} \right). \quad (5.58)$$

The relation (5.58) is key to EM algorithms minimizing a negative log-likelihood. The right side of this equation can be interpreted as a majorizing surrogate of the left side since it is easy to show that both terms are equal for $\boldsymbol{\theta} = \kappa$. Unfortunately the resulting approximation error functions are not L -smooth in general and these surrogates do not follow the assumptions of Definition 453. As a consequence, our analysis may apply to some EM algorithms, but not to all of them.

5.6.2.6 Quadratic surrogates

When f is twice differentiable and admits a matrix \mathbf{H} such that $\mathbf{H} - \nabla^2 f$ is always positive definite, the following function is a first-order majorizing surrogate:

$$g : \boldsymbol{\theta} \mapsto f(\kappa) + \nabla f(\kappa)^T (\boldsymbol{\theta} - \kappa) + \frac{1}{2} (\boldsymbol{\theta} - \kappa)^T \mathbf{H} (\boldsymbol{\theta} - \kappa).$$

The Lipschitz constant of $\nabla(g - f)$ is the largest eigenvalue of $\mathbf{H} - \nabla^2 f(\boldsymbol{\theta})$ over Θ . Such surrogates appear frequently in the statistics and machine learning literature. The goal is to model the global curvature of the objective function during each iteration, without resorting to the Newton method. Even though quadratic surrogates do not necessarily lead to better theoretical convergence rates than simpler Lipschitz gradient surrogates, they can be quite effective in practice.

Examples of using the above convex surrogates in optimization algorithms can be found in [96].

Draft

第六章 Optimality Conditions and Duality

(Taken from Chapter 2 of [48])

6.1 Introduction

In this chapter we will focus on necessary and sufficient optimality conditions for constrained problems.

As an introduction let us remind ourselves of the optimality conditions for *unconstrained* and *equality constrained* problems, which are commonly dealt with in basic mathematics lectures.

We consider a real-valued function $f : D \rightarrow \mathbb{R}$ with domain $D \subset \mathbb{R}^n$ and define, as usual, for a point $\mathbf{x}_0 \in D$:

1. f has a *local minimum* in \mathbf{x}_0

$$\iff \exists U \in \mathcal{U}_{\mathbf{x}_0}, \forall \mathbf{x} \in U \cap D, f(\mathbf{x}) \geq f(\mathbf{x}_0).$$

2. f has a *strict local minimum* in \mathbf{x}_0

$$\iff \exists U \in \mathcal{U}_{\mathbf{x}_0}, \forall \mathbf{x} \in U \cap D \setminus \{\mathbf{x}_0\}, f(\mathbf{x}) > f(\mathbf{x}_0).$$

3. f has a *global minimum* in \mathbf{x}_0

$$\iff \forall \mathbf{x} \in D, f(\mathbf{x}) \geq f(\mathbf{x}_0).$$

4. f has a *strict global minimum* in \mathbf{x}_0

$$\iff \forall \mathbf{x} \in D \setminus \{\mathbf{x}_0\}, f(\mathbf{x}) > f(\mathbf{x}_0).$$

Here, $\mathcal{U}_{\mathbf{x}_0}$ denotes the neighborhood system of \mathbf{x}_0 .

We often say “ \mathbf{x}_0 is a *local minimizer* of f ” or “ \mathbf{x}_0 is a *local minimum point* of f ” instead of “ f has a *local minimum* in \mathbf{x}_0 ” and so on. The *minimizer* is a point $\mathbf{x}_0 \in D$, the *minimum* is the corresponding value $f(\mathbf{x}_0)$.

Necessary Condition

Suppose that the function f has a local minimum in $\mathbf{x}_0 \in D^\circ$, that is, in an interior point of D . Then:

- a) If f is differentiable in \mathbf{x}_0 , then $\nabla f(\mathbf{x}_0) = \mathbf{0}$ holds.
- b) If f is twice continuously differentiable in a neighborhood of \mathbf{x}_0 , then the Hessian $H_f(\mathbf{x}_0) = \nabla^2 f(\mathbf{x}_0) = \left(\frac{\partial^2 f}{\partial x_i \partial x_j}(\mathbf{x}_0) \right)$ is positive semidefinite.

We will use the notation $f'(\mathbf{x}_0)$ (to denote the derivative of f at \mathbf{x}_0 ; as we know, this is a linear map from \mathbb{R}^n to \mathbb{R}^n , read as a *row vector*) as well as the corresponding transposed vector $\nabla f(\mathbf{x}_0)$ (gradient, *column vector*).

Points $\mathbf{x} \in D^\circ$ with $\nabla f(\mathbf{x}) = \mathbf{0}$ are called *stationary points*. At a stationary point there can be a local minimum, a local maximum or a *saddle point*. To determine that there is a local minimum at a stationary point, we use the following:

Sufficient Condition

Suppose that the function f is twice continuously differentiable in a neighborhood of $\mathbf{x}_0 \in D$; also suppose that the necessary optimality condition $\nabla f(\mathbf{x}_0) = \mathbf{0}$ holds and that the Hessian $\nabla^2 f(\mathbf{x}_0)$ is positive definite. Then f has a strict local minimum in \mathbf{x}_0 .

The proof of this proposition is based on the Taylor theorem and we regard it as known from Calculus. Let us recall that a symmetric (n, n) -matrix \mathbf{A} is *positive definite* if and only if all principal subdeterminants

$$\det \begin{pmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} \end{pmatrix}, \quad k = 1, \dots, n$$

are positive.

Now let f be a real-valued function with domain $D \subset \mathbb{R}^n$ which we want to minimize subject to the *equality constraints*

$$h_j(\mathbf{x}) = 0, \quad j = 1, \dots, p,$$

for $p < n$; here, let h_1, \dots, h_p also be defined on D . We are looking for local minimizers of f , that is, points $\mathbf{x}_0 \in D$ which belong to the *feasible region*

$$\mathcal{F} := \{ \mathbf{x} \in D \mid h_j(\mathbf{x}) = 0, j = 1, \dots, p \}$$

and to which a neighborhood U exists with $f(\mathbf{x}) \geq f(\mathbf{x}_0)$ for all $\mathbf{x} \in U \cap \mathcal{F}$.

Intuitively, it seems reasonable to solve the constraints for p of the n variables, and to eliminate these by inserting them into the objective function. For the *reduced objective function* we thereby get a nonrestricted problem for which under suitable assumptions the above necessary optimality condition holds.

After these preliminary remarks, we are now able to formulate the following *necessary optimality condition*.

Theorem 460 (Lagrange Multiplier Rule). *Let $D \subset \mathbb{R}^n$ be open and f, h_1, \dots, h_p continuously differentiable in D . Suppose that f has a local minimum in $\mathbf{x}_0 \in \mathcal{F}$ subject to the constraints*

$$h_j(\mathbf{x}) = 0, \quad j = 1, \dots, p.$$

Let also the Jacobian $\left(\frac{\partial h_j}{\partial \mathbf{x}_k}(\mathbf{x}_0) \right)_{p,n}$ have rank p . Then there exist real numbers μ_1, \dots, μ_p – the so-called Lagrange multipliers – such that

$$\nabla f(\mathbf{x}_0) + \sum_{j=1}^p \mu_j \nabla h_j(\mathbf{x}_0) = \mathbf{0}.$$

Corresponding to our preliminary remarks, a main tool in a proof would be the *Implicit Function Theorem*. We assume that interested readers are familiar with a proof from multidimensional analysis. In addition, the results will be generalized in Theorem 475. Therefore we do not give a proof here, but instead illustrate the matter with the following simple problem:

Example 461.

With $f(\mathbf{x}) := x_1 x_2^2$ and $h(\mathbf{x}) := h_1(\mathbf{x}) := x_1^2 + x_2^2 - 2$ for $\mathbf{x} = (x_1, x_2)^\top \in D := \mathbb{R}^2$ we consider the problem:

$$\min_{\mathbf{x}} f(\mathbf{x}), \quad s.t. \quad h(\mathbf{x}) = 0.$$

We hence have $n = 2$ and $p = 1$.

Before we start, however, note that this problem can of course be solved very easily straight away: One inserts x_2^2 from the constraint $x_1^2 + x_2^2 - 2 = 0$ into $f(\mathbf{x})$ and thus gets a one-dimensional problem.

Points \mathbf{x} meeting the constraint are different from $\mathbf{0}$ and thus also meet the rank condition. With $\mu := \mu_1$ the equation $\nabla f(\mathbf{x}) + \mu \nabla h(\mathbf{x}) = \mathbf{0}$ translates into

$$x_2^2 + 2\mu x_1 = 0 \quad \text{and} \quad 2x_1 x_2 + 2\mu x_2 = 0.$$

Multiplication of the first equation by x_2 and the second by x_1 gives

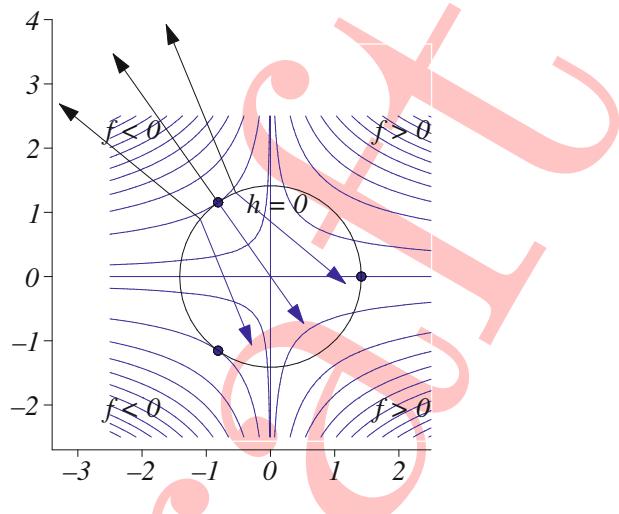
$$x_2^3 + 2\mu x_1 x_2 = 0 \quad \text{and} \quad 2x_1^2 x_2 + 2\mu x_1 x_2 = 0$$

and thus

$$x_2^3 = 2x_1^2 x_2.$$

For $x_2 = 0$ the constraint yields $x_1 = \pm\sqrt{2}$. Of these two evidently only $x_1 = \sqrt{2}$ remains as a potential minimizer. If $x_2 \neq 0$, we have $x_2^2 = 2x_1^2$ and hence with the constraint

$3x_1^2 = 2$, thus $x_1 = \pm\sqrt{2/3}$ and then $x_2 = \pm 2/\sqrt{3}$. In this case the distribution of the zeros and signs of f gives that only $\mathbf{x} = (-\sqrt{2/3}, \pm 2/\sqrt{3})^\top$ remain as potential minimizers. Since f is continuous on the compact set $\{\mathbf{x} \in \mathbb{R}^2 | h(\mathbf{x}) = 0\}$, we know that there exists a global minimizer. Altogether, we get: f attains its global minimum at $(-\sqrt{2/3}, \pm 2/\sqrt{3})^\top$, the point $(\sqrt{2}, 0)^\top$ yields a local minimum. The following picture illustrates the gradient condition very well:



The aim of our further investigations will be to generalize the Lagrange Multiplier Rule to *minimization problems with inequality constraints*:

$$(P) \quad \begin{cases} \min_{\mathbf{x}} f(\mathbf{x}), \\ \text{s.t. } g_i(\mathbf{x}) \leq 0, \text{ for } i \in \mathcal{I} := \{1, \dots, m\}, \\ h_j(\mathbf{x}) = 0, \text{ for } j \in \mathcal{E} := \{1, \dots, p\}. \end{cases} \quad (6.1)$$

With $m, p \in \mathbb{N}_0$ (hence, $\mathcal{E} = \emptyset$ or $\mathcal{I} = \emptyset$ are allowed), the functions $f, g_1, \dots, g_m, h_1, \dots, h_p$ are supposed to be continuously differentiable on an open subset D in \mathbb{R}^n and $p \leq n$. The set

$$\mathcal{F} := \{\mathbf{x} \in D | g_i(\mathbf{x}) \leq 0 \text{ for } i \in \mathcal{I}, h_j(\mathbf{x}) = 0 \text{ for } j \in \mathcal{E}\}$$

– in analogy to the above – is called the *feasible region* or *set of feasible points* of (P) .

In most cases we state the problem in the slightly shortened form

$$(P) \quad \begin{cases} \min_{\mathbf{x}} f(\mathbf{x}), \\ g_i(\mathbf{x}) \leq 0, \text{ for } i \in \mathcal{I}, \\ h_j(\mathbf{x}) = 0, \text{ for } j \in \mathcal{E}. \end{cases}$$

The *optimal value* $v(P)$ to problem (P) is defined as

$$v(P) := \inf\{f(\mathbf{x}) | \mathbf{x} \in \mathcal{F}\}.$$

We allow $v(P)$ to attain the extended values $+\infty$ and $-\infty$. We follow the standard convention that the infimum of the empty set is ∞ . If there are feasible points \mathbf{x}_k with $f(\mathbf{x}_k) \rightarrow -\infty (k \rightarrow \infty)$, then $v(P) = -\infty$ and we say problem (P) – or the function f on \mathcal{F} – is unbounded from below.

We say \mathbf{x}_0 is a *minimal point* or a *minimizer* if \mathbf{x}_0 is feasible and $f(\mathbf{x}_0) = v(P)$.

In order to formulate optimality conditions for (P) , we will need some simple tools from *Convex Analysis*. These will be provided in the following section.

6.2 Local First-Order Optimality Conditions

We want to take up the minimization problem (P) (6.1) and use the notation introduced there. For $\mathbf{x}_0 \in \mathcal{F}$, the index set

$$\mathcal{A}(\mathbf{x}_0) := \{i \in \mathcal{I} | g_i(\mathbf{x}_0) = 0\}$$

describes the *inequality restrictions which are active at \mathbf{x}_0* .

The active constraints have a special significance: They restrict feasible corrections around a feasible point. If a constraint is *inactive* ($g_i(\mathbf{x}_0) < 0$) at the feasible point \mathbf{x}_0 , it is possible to move from \mathbf{x}_0 a bit in any direction without violating this constraint.

Definition 462. Let $\mathbf{d} \in \mathbb{R}^n$ and $\mathbf{x}_0 \in \mathcal{F}$. Then \mathbf{d} is called the *feasible direction* of \mathcal{F} at \mathbf{x}_0 : $\Leftrightarrow \exists \delta > 0, \forall \tau \in [0, \delta], \mathbf{x}_0 + \tau \mathbf{d} \in \mathcal{F}$.

A ‘small’ movement from \mathbf{x}_0 along such a direction gives feasible points. The set of all feasible directions of \mathcal{F} at \mathbf{x}_0 is a *cone*, denoted by

$$\mathcal{C}_{fd}(\mathbf{x}_0).$$

Let \mathbf{d} be a feasible direction of \mathcal{F} at \mathbf{x}_0 . If we choose a δ according to the definition, then we have

$$\underbrace{g_i(\mathbf{x}_0 + \tau \mathbf{d})}_{\leq 0} = \underbrace{g_i(\mathbf{x}_0)}_{=0} + \tau g'_i(\mathbf{x}_0) \mathbf{d} + o(\tau)$$

for $i \in \mathcal{A}(\mathbf{x}_0)$ and $0 < \tau \leq \delta$. Dividing by τ and passing to the limit as $\tau \rightarrow 0$ gives $g'_i(\mathbf{x}_0) \mathbf{d} \leq 0$. In the same way we get $h'_j(\mathbf{x}_0) \mathbf{d} = 0$ for all $j \in \mathcal{E}$.

Definition 463. For any $\mathbf{x}_0 \in \mathcal{F}$,

$$\mathcal{C}_l(P, \mathbf{x}_0) := \{\mathbf{d} \in \mathbb{R}^n | \forall i \in \mathcal{A}(\mathbf{x}_0), g'_i(\mathbf{x}_0)\mathbf{d} \leq 0, \forall j \in \mathcal{E}, h'_j(\mathbf{x}_0)\mathbf{d} = 0\}$$

is called the linearizing cone of (P) at \mathbf{x}_0 .

Hence, $\mathcal{C}_l(\mathbf{x}_0) := \mathcal{C}_l(P, \mathbf{x}_0)$ contains at least all feasible directions of \mathcal{F} at \mathbf{x}_0 :

$$\mathcal{C}_{fd}(\mathbf{x}_0) \subset \mathcal{C}_l(\mathbf{x}_0).$$

The linearizing cone is not only dependent on the set of feasible points \mathcal{F} but also on the representation of \mathcal{F} (compare Example 473). We therefore write more precisely $\mathcal{C}_l(P, \mathbf{x}_0)$.

Definition 464. For any $\mathbf{x}_0 \in D$

$$\mathcal{C}_{dd}(\mathbf{x}_0) := \{\mathbf{d} \in \mathbb{R}^n | f'(\mathbf{x}_0)\mathbf{d} < 0\}$$

is called the cone of descent directions of f at \mathbf{x}_0 .

Note that $\mathbf{0}$ is not in $\mathcal{C}_{dd}(\mathbf{x}_0)$; also, for all $\mathbf{d} \in \mathcal{C}_{dd}(\mathbf{x}_0)$

$$f(\mathbf{x}_0 + \tau\mathbf{d}) = f(\mathbf{x}_0) + \tau \underbrace{f'(\mathbf{x}_0)\mathbf{d}}_{<0} + o(\tau)$$

holds and therefore, $f(\mathbf{x}_0 + \tau\mathbf{d}) < f(\mathbf{x}_0)$ for sufficiently small $\tau > 0$.

Thus, $\mathbf{d} \in \mathcal{C}_{dd}(\mathbf{x}_0)$ guarantees that the objective function f can be reduced along this direction. Hence, for a local minimizer \mathbf{x}_0 of (P) it necessarily holds that $\mathcal{C}_{dd}(\mathbf{x}_0) \cap \mathcal{C}_{fd}(\mathbf{x}_0) = \emptyset$.

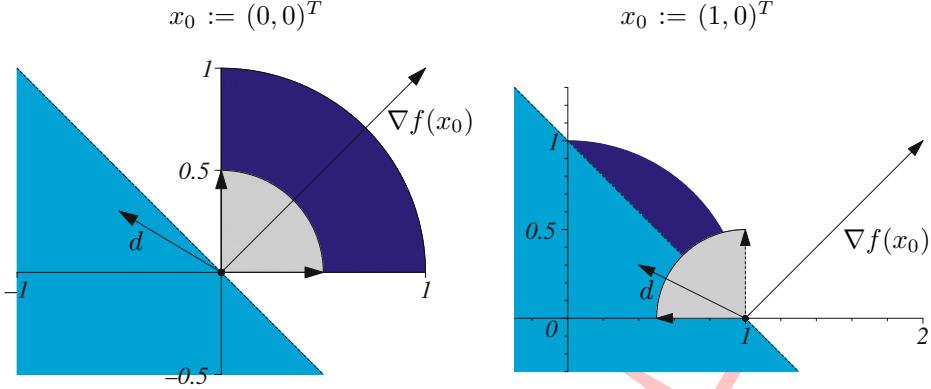
We will illustrate the above definitions with the following

Example 465. Let

$$\mathcal{F} := \{\mathbf{x} = (x_1, x_2)^\top \in \mathbb{R}^2 | x_1^2 + x_2^2 - 1 \leq 0, -x_1 \leq 0, -x_2 \leq 0\},$$

and f be defined by $f(\mathbf{x}) := x_1 + x_2$. Hence, \mathcal{F} is the part of the unit disk which lies in the first quadrant. The objective function f evidently attains a (strict, global) minimum at $(0, 0)^\top$.

In both of the following pictures \mathcal{F} is colored in dark blue.



- a) Let $\mathbf{x}_0 := (0,0)^\top$. $g_1(\mathbf{x}) := x_1^2 + x_2^2 - 1$, $g_2(\mathbf{x}) := -x_1$ and $g_3(\mathbf{x}) := -x_2$ give $\mathcal{A}(\mathbf{x}_0) = \{2, 3\}$. A vector $\mathbf{d} := (d_1, d_2)^\top \in \mathbb{R}^2$ is a feasible direction of \mathcal{F} at \mathbf{x}_0 if and only if $d_1 \geq 0$ and $d_2 \geq 0$ hold. Hence, the set $\mathcal{C}_{fd}(\mathbf{x}_0)$ of feasible directions is a convex cone, namely, the first quadrant, and it is represented in the left picture by the gray angular domain. $g'_2(\mathbf{x}_0) = (-1, 0)$ and $g'_3(\mathbf{x}_0) = (0, -1)$ produce

$$\mathcal{C}_l(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^2 \mid -d_1 \leq 0, -d_2 \leq 0\}.$$

Hence, in this example, the linearizing cone and the cone of feasible directions are the same. Moreover, the cone of descent directions $\mathcal{C}_{dd}(\mathbf{x}_0)$ – colored in light blue in the picture – is, because of $f'(\mathbf{x}_0)\mathbf{d} = (1, 1)^\top \mathbf{d} = d_1 + d_2$, an open half space and disjoint to $\mathcal{C}_l(\mathbf{x}_0)$.

- b) If $\mathbf{x}_0 := (1,0)^\top$, we have $\mathcal{A}(\mathbf{x}_0) = \{1, 3\}$ and $\mathbf{d} := (d_1, d_2)^\top \in \mathbb{R}^2$ is a feasible direction of \mathcal{F} at \mathbf{x}_0 if and only if $\mathbf{d} = (0,0)^\top$ or $d_1 < 0$ and $d_2 \geq 0$ hold. The set of feasible directions is again a convex cone. In the right picture it is depicted by the shifted gray angular domain. Because of $g'_1(\mathbf{x}_0) = (2, 0)$ and $g'_3(\mathbf{x}_0) = (0, -1)$, we get

$$\mathcal{C}_l(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^2 \mid d_1 \leq 0, d_2 \geq 0\}.$$

As we can see, in this case the linearizing cone includes the cone of feasible directions properly as a subset. In the picture the cone of descent directions has also been moved to \mathbf{x}_0 . We can see that it contains feasible directions of \mathcal{F} at \mathbf{x}_0 . Consequently, f does not have a local minimum in \mathbf{x}_0 .

Proposition 466. For $\mathbf{x}_0 \in \mathcal{F}$ it holds that $\mathcal{C}_l(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0) = \emptyset$ if and only if there exist $\lambda \in \mathbb{R}_+^m$ and $\mu \in \mathbb{R}^p$ such that

$$\nabla f(\mathbf{x}_0) + \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}_0) + \sum_{j=1}^p \mu_j \nabla h_j(\mathbf{x}_0) = \mathbf{0} \quad (6.2)$$

and

$$\lambda_i g_i(\mathbf{x}_0) = 0 \text{ for all } i \in \mathcal{I}. \quad (6.3)$$

Together, these conditions – $\mathbf{x}_0 \in \mathcal{F}$, $\boldsymbol{\lambda} \geq \mathbf{0}$, (6.2) and (6.3) – are called Karush-Kuhn-Tucker *conditions*, or KKT *conditions*. (6.3) is called the *complementary slackness condition* or *complementarity condition*. This condition of course means $\lambda_i = 0$ or (in the nonexclusive sense) $g_i(\mathbf{x}_0) = 0$ for all $i \in \mathcal{I}$. A corresponding pair $(\boldsymbol{\lambda}, \boldsymbol{\mu})$ or the scalars $\lambda_1, \dots, \lambda_m, \mu_1, \dots, \mu_p$ are called Lagrange *multipliers*. The function L defined by

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) := f(\mathbf{x}) + \sum_{i=1}^m \lambda_i g_i(\mathbf{x}) + \sum_{j=1}^p \mu_j h_j(\mathbf{x}) = f(\mathbf{x}) + \boldsymbol{\lambda}^\top g(\mathbf{x}) + \boldsymbol{\mu}^\top h(\mathbf{x})$$

for $\mathbf{x} \in D$, $\boldsymbol{\lambda} \in \mathbb{R}_+^m$ and $\boldsymbol{\mu} \in \mathbb{R}^p$ is called the *Lagrange function* or *Lagrangian* of (P) . Here we have combined the m functions g_i to a vector-valued function g and respectively the p functions h_j to a vector-valued function h .

Points $\mathbf{x}_0 \in \mathcal{F}$ fulfilling (6.2) and (6.3) with a suitable $\boldsymbol{\lambda} \in \mathbb{R}_+^m$ and $\boldsymbol{\mu} \in \mathbb{R}^p$ play an important role. They are called Karush-Kuhn-Tucker *points*, or KKT *points*.

Owing to the complementarity condition (6.3), the multipliers λ_i corresponding to *inactive restrictions* at \mathbf{x}_0 must be zero. So we can omit the terms for $i \in \mathcal{I} \setminus \mathcal{A}(\mathbf{x}_0)$ from (6.2) and rewrite this condition as

$$\nabla f(\mathbf{x}_0) + \sum_{i \in \mathcal{A}(\mathbf{x}_0)} \lambda_i \nabla g_i(\mathbf{x}_0) + \sum_{j=1}^p \mu_j \nabla h_j(\mathbf{x}_0) = \mathbf{0}.$$

Proof. By definition of $\mathcal{C}_l(\mathbf{x}_0)$ and $\mathcal{C}_{dd}(\mathbf{x}_0)$ it holds that:

$$\begin{aligned} \mathbf{d} \in \mathcal{C}_l(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0) &\Leftrightarrow \begin{cases} f'(\mathbf{x}_0)\mathbf{d} < 0 \\ \forall i \in \mathcal{A}(\mathbf{x}_0), g'_i(\mathbf{x}_0)\mathbf{d} \leq 0 \\ \forall j \in \mathcal{E}, h'_j(\mathbf{x}_0)\mathbf{d} = 0. \end{cases} \\ &\Leftrightarrow \begin{cases} f'(\mathbf{x}_0)\mathbf{d} < 0 \\ \forall i \in \mathcal{A}(\mathbf{x}_0), -g'_i(\mathbf{x}_0)\mathbf{d} \geq 0 \\ \forall j \in \mathcal{E}, -h'_j(\mathbf{x}_0)\mathbf{d} \geq 0; \forall j \in \mathcal{E}, h'_j(\mathbf{x}_0)\mathbf{d} \geq 0. \end{cases} \end{aligned}$$

With that the Theorem of the Alternative (Theorem 130) directly provides the following equivalence:

$\mathcal{C}_l(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0) = \emptyset$ if and only if there exist $\lambda_i \geq 0$ for $i \in \mathcal{A}(\mathbf{x}_0)$ and $\mu'_j \geq 0, \mu''_j \geq 0$ for $j \in \mathcal{E}$ such that

$$\nabla f(\mathbf{x}_0) = \sum_{i \in \mathcal{A}(\mathbf{x}_0)} \lambda_i (-\nabla g_i(\mathbf{x}_0)) + \sum_{j=1}^p \mu'_j (-\nabla h_j(\mathbf{x}_0)) + \sum_{j=1}^p \mu''_j \nabla h_j(\mathbf{x}_0).$$

If we now set $\lambda_i := 0$ for $i \in \mathcal{I} \setminus \mathcal{A}(\mathbf{x}_0)$ and $\mu_j := \mu'_j - \mu''_j$ for $j \in \mathcal{E}$, the above is equivalent to: There exist $\lambda_i \geq 0$ for $i \in \mathcal{I}$ and $\mu_j \in \mathbb{R}$ for $j \in \mathcal{E}$ with

$$\nabla f(\mathbf{x}_0) + \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}_0) + \sum_{j=1}^p \mu_j \nabla h_j(\mathbf{x}_0) = \mathbf{0}$$

and

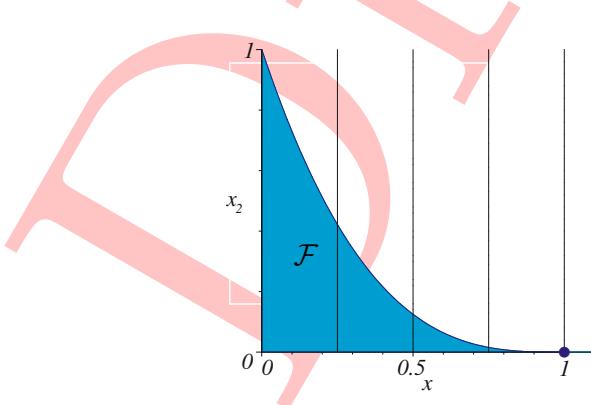
$$\lambda_i g_i(\mathbf{x}_0) = 0 \text{ for all } i \in \mathcal{I}.$$

□

So now the question arises whether not just $\mathcal{C}_{fd}(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0) = \emptyset$, but even $\mathcal{C}_l(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0) = \emptyset$ is true for any local minimizer $\mathbf{x}_0 \in \mathcal{F}$. The following simple example gives a negative answer to this question:

Example 467 (Kuhn-Tucker (1951)). For $n = 2$ and $\mathbf{x} = (x_1, x_2)^\top \in \mathbb{R}^2 =: D$, let $f(\mathbf{x}) := -x_1$, $g_1(\mathbf{x}) := x_2 + (x_1 - 1)^3$, $g_2(\mathbf{x}) := -x_1$ and $g_3(\mathbf{x}) := -x_2$. For $\mathbf{x}_0 := (1, 0)^\top$, $m = 3$ and $p = 0$ we have: $\nabla f(\mathbf{x}_0) = (-1, 0)^\top$, $\nabla g_1(\mathbf{x}_0) = (0, 1)^\top$, $\nabla g_2(\mathbf{x}_0) = (-1, 0)^\top$ and $\nabla g_3(\mathbf{x}_0) = (0, -1)^\top$.

Since $\mathcal{A}(\mathbf{x}_0) = \{1, 3\}$, we get $\mathcal{C}_l(\mathbf{x}_0) = \{(d_1, d_2)^\top \in \mathbb{R}^2 | d_2 = 0\}$, as well as $\mathcal{C}_{dd}(\mathbf{x}_0) = \{(d_1, d_2)^\top \in \mathbb{R}^2 | d_1 > 0\}$; evidently, $\mathcal{C}_l(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0)$ is nonempty. However, the function f has a minimum at \mathbf{x}_0 subject to the given constraints.



Lemma 468.

For $\mathbf{x}_0 \in \mathcal{F}$ it holds that: $\mathcal{C}_l(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0) = \emptyset \Leftrightarrow \nabla f(\mathbf{x}_0) \in \mathcal{C}_l(\mathbf{x}_0)^*$.

Proof.

$$\begin{aligned} \mathcal{C}_l(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0) = \emptyset &\Leftrightarrow \forall \mathbf{d} \in \mathcal{C}_l(\mathbf{x}_0), \langle \nabla f(\mathbf{x}_0), \mathbf{d} \rangle = f'(\mathbf{x}_0) \mathbf{d} \geq 0 \\ &\Leftrightarrow \nabla f(\mathbf{x}_0) \in \mathcal{C}_l(\mathbf{x}_0)^*. \end{aligned}$$

□

The cone $\mathcal{C}_{fd}(\mathbf{x}_0)$ of all feasible directions is too small to ensure general optimality conditions. Difficulties may occur due to the fact that the boundary of \mathcal{F} is curved. Therefore, we have to consider a set which is less intuitive but bigger and with more suitable properties. To attain this goal, it is useful to state the *concept of being tangent to a set* more precisely.

Definition 469. A sequence (\mathbf{x}_k) converges in direction \mathbf{d} to \mathbf{x}_0

$$\Leftrightarrow \mathbf{x}_k = \mathbf{x}_0 + \alpha_k(\mathbf{d} + \mathbf{r}_k) \text{ with } \alpha_k \downarrow 0 \text{ and } \mathbf{r}_k \rightarrow \mathbf{0}.$$

We will use the following notation: $\mathbf{x}_k \xrightarrow{\mathbf{d}} \mathbf{x}_0$.

$\mathbf{x}_k \xrightarrow{\mathbf{d}} \mathbf{x}_0$ simply means: There exists a sequence of positive numbers (α_k) such that $\alpha_k \downarrow 0$ and

$$\frac{1}{\alpha_k}(\mathbf{x}_k - \mathbf{x}_0) \rightarrow \mathbf{d} \text{ for } k \rightarrow \infty.$$

Definition 470. Let M be a nonempty subset of \mathbb{R}^n and $\mathbf{x}_0 \in M$. Then

$$\mathcal{C}_t(M, \mathbf{x}_0) := \left\{ \mathbf{d} \in \mathbb{R}^n \mid \exists \{\mathbf{x}_k\} \in M^{\mathbb{N}}, \mathbf{x}_k \xrightarrow{\mathbf{d}} \mathbf{x}_0 \right\}$$

is called the tangent cone of M at \mathbf{x}_0 . The vectors of $\mathcal{C}_t(M, \mathbf{x}_0)$ are called tangents or tangent directions of M at \mathbf{x}_0 .

Of main interest is the special case

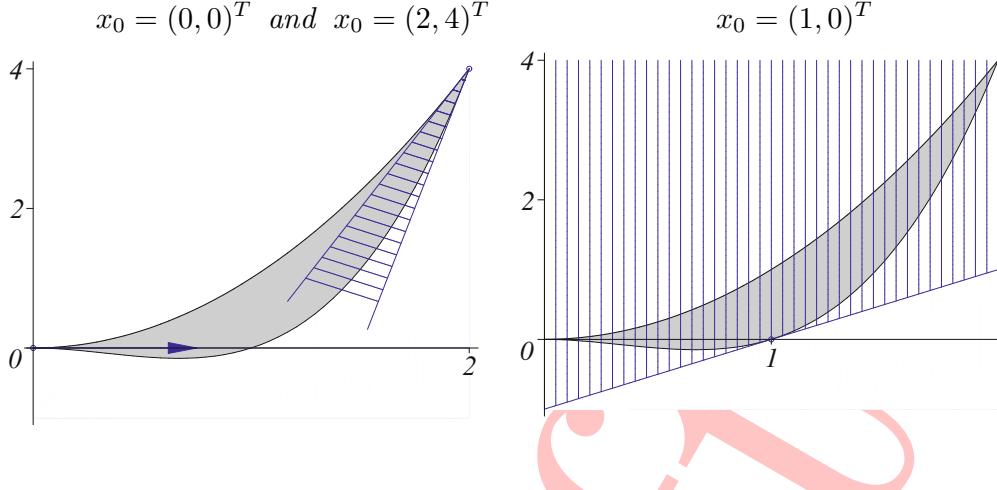
$$\mathcal{C}_t(\mathbf{x}_0) := \mathcal{C}_t(\mathcal{F}, \mathbf{x}_0).$$

Example 471.

a) The following two figures illustrate the cone of tangents for

$$\mathcal{F} := \left\{ \mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^2 \mid x_1 \geq 0, x_1^2 \geq x_2 \geq x_1^2(x_1 - 1) \right\}$$

and the points $\mathbf{x}_0 \in \{(0, 0)^T, (2, 4)^T, (1, 0)^T\}$. For convenience the origin is translated to \mathbf{x}_0 . The reader is invited to verify this:



- b) $\mathcal{F} := \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\|_2 = 1\} : \mathcal{C}_t(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^n \mid \langle \mathbf{d}, \mathbf{x}_0 \rangle = 0\}.$
- c) $\mathcal{F} := \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\|_2 \leq 1\}: \text{Then } \mathcal{C}_t(\mathbf{x}_0) = \mathbb{R}^n \text{ if } \|\mathbf{x}_0\|_2 < 1 \text{ holds, and } \mathcal{C}_t(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^n \mid \langle \mathbf{d}, \mathbf{x}_0 \rangle \leq 0\} \text{ if } \|\mathbf{x}_0\|_2 = 1.$

These assertions can be exercises.

Lemma 472. 1) $\mathcal{C}_t(\mathbf{x}_0)$ is a closed cone, $\mathbf{0} \in \mathcal{C}_t(\mathbf{x}_0)$.

2) $\overline{\mathcal{C}_{fd}(\mathbf{x}_0)} \subset \mathcal{C}_t(\mathbf{x}_0) \subset \mathcal{C}_l(\mathbf{x}_0)$.

Proof. 1) The proof of 1) can be an exercise.

2) First inclusion: As the tangent cone $\mathcal{C}_t(\mathbf{x}_0)$ is closed, it is sufficient to show the inclusion $\mathcal{C}_{fd}(\mathbf{x}_0) \subset \mathcal{C}_t(\mathbf{x}_0)$. For $\mathbf{d} \in \mathcal{C}_{fd}(\mathbf{x}_0)$ and ‘large’ integers k it holds that $\mathbf{x}_0 + \frac{1}{k}\mathbf{d} \in \mathcal{F}$. With $\alpha_k := \frac{1}{k}$ and $\mathbf{r}_k := \mathbf{0}$ this shows $\mathbf{d} \in \mathcal{C}_t(\mathbf{x}_0)$.

Second inclusion: Let $\mathbf{d} \in \mathcal{C}_t(\mathbf{x}_0)$ and $(\mathbf{x}_k) \in \mathcal{F}^\mathbb{N}$ be a sequence with $\mathbf{x}_k = \mathbf{x}_0 + \alpha_k(\mathbf{d} + \mathbf{r}_k)$, $\alpha_k \downarrow 0$ and $\mathbf{r}_k \rightarrow \mathbf{0}$. For $i \in \mathcal{A}(\mathbf{x}_0)$

$$\underbrace{g_i(\mathbf{x}_k)}_{\leq 0} = \underbrace{g_i(\mathbf{x}_0)}_{=0} + \alpha_k g'_i(\mathbf{x}_0)(\mathbf{d} + \mathbf{r}_k) + o(\alpha_k)$$

produces the inequality $g'_i(\mathbf{x}_0)\mathbf{d} \leq 0$. In the same way we get $h'_j(\mathbf{x}_0)\mathbf{d} = 0$ for $j \in \mathcal{E}$.

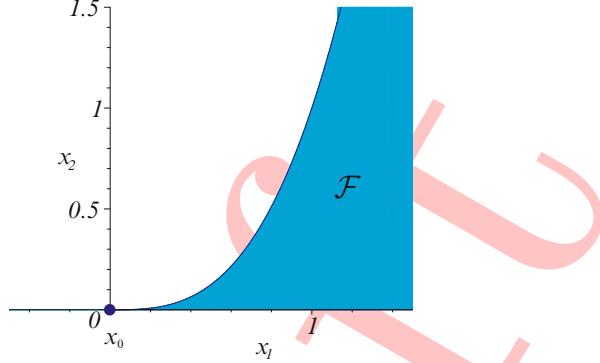
□

Now the question arises whether $\mathcal{C}_t(\mathbf{x}_0) = \mathcal{C}_l(\mathbf{x}_0)$ always holds. The following example gives a negative answer:

Example 473. a) Consider $\mathcal{F} := \{\mathbf{x} \in \mathbb{R}^2 \mid -x_1^3 + x_2 \leq 0, -x_2 \leq 0\}$ and $\mathbf{x}_0 := (0,0)^\top$.

In this case $\mathcal{A}(\mathbf{x}_0) = \{1, 2\}$. This gives $\mathcal{C}_l(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^2 \mid d_2 = 0\}$ and $\mathcal{C}_t(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^2 \mid d_1 \geq 0, d_2 = 0\}$. The last statement can be an exercise.

- b) Now let $\mathcal{F} := \{\mathbf{x} \in \mathbb{R}^2 \mid -x_1^3 + x_2 \leq 0, -x_1 \leq 0, -x_2 \leq 0\}$ and $\mathbf{x}_0 := (0, 0)^\top$. Then $\mathcal{A}(\mathbf{x}_0) = \{1, 2, 3\}$ and therefore $\mathcal{C}_l(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^2 \mid d_1 \geq 0, d_2 = 0\} = \mathcal{C}_t(\mathbf{x}_0)$. Hence, the linearizing cone is dependent on the representation of the set of feasible points \mathcal{F} which is the same in both cases!



Lemma 474.

For a local minimizer \mathbf{x}_0 of (P) it holds that $\nabla f(\mathbf{x}_0) \in \mathcal{C}_t(\mathbf{x}_0)^*$, hence $\mathcal{C}_{dd}(\mathbf{x}_0) \cap \mathcal{C}_t(\mathbf{x}_0) = \emptyset$.

Geometrically this condition states that for a local minimizer \mathbf{x}_0 of (P) the angle between the gradient and any tangent direction, especially any feasible direction, does not exceed 90° .

Proof. Let $\mathbf{d} \in \mathcal{C}_t(\mathbf{x}_0)$. Then there exists a sequence $\{\mathbf{x}_k\} \in \mathcal{F}^\mathbb{N}$ such that $\mathbf{x}_k = \mathbf{x}_0 + \alpha_k(\mathbf{d} + \mathbf{r}_k)$, $\alpha_k \downarrow 0$ and $\mathbf{r}_k \rightarrow \mathbf{0}$.

$$0 \leq f(\mathbf{x}_k) - f(\mathbf{x}_0) = \alpha_k \nabla f(\mathbf{x}_0)(\mathbf{d} + \mathbf{r}_k) + o(\alpha_k)$$

gives the result $\nabla f(\mathbf{x}_0)\mathbf{d} \geq 0$. □

The principal result in this section is the following:

Theorem 475 (Karush-Kuhn-Tucker). Suppose that \mathbf{x}_0 is a local minimizer of (P) , and the constraint qualification $\mathcal{C}_l(\mathbf{x}_0)^* = \mathcal{C}_t(\mathbf{x}_0)^*$ is fulfilled. Then there exist vectors $\boldsymbol{\lambda} \in \mathbb{R}_+^m$ and $\boldsymbol{\mu} \in \mathbb{R}^p$ such that

$$\nabla f(\mathbf{x}_0) + \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}_0) + \sum_{j=1}^p \mu_j \nabla h_j(\mathbf{x}_0) = \mathbf{0} \text{ and}$$

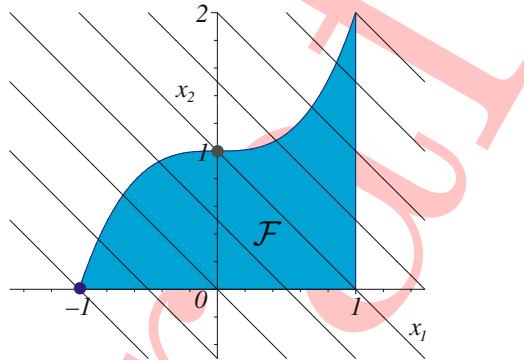
$$\lambda_i g_i(\mathbf{x}_0) = 0 \text{ for } i = 1, \dots, m.$$

Proof. If \mathbf{x}_0 is a local minimizer of (P) , it follows from Lemma 474 with the help of the presupposed constraint qualification that

$$\nabla f(\mathbf{x}_0) \in \mathcal{C}_t(\mathbf{x}_0)^* = \mathcal{C}_l(\mathbf{x}_0)^*;$$

Lemma 468 yields $\mathcal{C}_l(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0) = \emptyset$ and the latter together with Proposition 466 gives the result. \square

In the presence of the presupposed constraint qualification $\mathcal{C}_t(\mathbf{x}_0)^* = \mathcal{C}_l(\mathbf{x}_0)^*$ the condition $\nabla f(\mathbf{x}_0) \in \mathcal{C}_t(\mathbf{x}_0)^*$ of Lemma 474 transforms to $\nabla f(\mathbf{x}_0) \in \mathcal{C}_l(\mathbf{x}_0)^*$. This claim can be confirmed with the aid of a simple linear optimization problem:



Example 476. For $\mathbf{x} = (x_1, x_2)^\top \in \mathbb{R}^2$ we consider the problem

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &:= x_1 + x_2, \\ \text{s.t. } &-x_1^3 + x_2 \leq 1, \\ &x_1 \leq 1, -x_2 \leq 0, \end{aligned}$$

and ask whether the feasible points $\mathbf{x}_0 := (-1, 0)^\top$ and $\tilde{\mathbf{x}}_0 := (0, 1)^\top$ are local minimizers. (The examination of the picture shows immediately that this is not the case for $\tilde{\mathbf{x}}_0$, and that the objective function f attains a (strict, global) minimum at \mathbf{x}_0 . But we try to forget this for a while.) We have $\mathcal{A}(\mathbf{x}_0) = \{1, 3\}$. In order to show that $\nabla f(\mathbf{x}_0) \in \mathcal{C}_l(\mathbf{x}_0)^*$, hence, $f'(\mathbf{x}_0)\mathbf{d} \geq 0$ for all $\mathbf{d} \in \mathcal{C}_l(\mathbf{x}_0)$, we compute $\min_{\mathbf{d} \in \mathcal{C}_l(\mathbf{x}_0)} f'(\mathbf{x}_0)\mathbf{d}$. So we have the following linear problem:

$$\begin{aligned} \min & d_1 + d_2, \\ \text{s.t. } & -3d_1 + d_2 \leq 0 \\ & -d_2 \leq 0. \end{aligned}$$

Evidently it has the minimal value 0; Lemma 468 gives that $\mathcal{C}_l(\mathbf{x}_0) \cap \mathcal{C}_{dd}(\mathbf{x}_0)$ is empty. Following Proposition 466 there exist $\lambda_1, \lambda_3 \geq 0$ for \mathbf{x}_0 satisfying

$$\begin{pmatrix} 1 \\ 1 \end{pmatrix} + \lambda_1 \begin{pmatrix} -3 \\ 1 \end{pmatrix} + \lambda_3 \begin{pmatrix} 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

The above yields $\lambda_1 = 1/3, \lambda_3 = 4/3$.

For $\tilde{\mathbf{x}}_0$ we have $\mathcal{A}(\tilde{\mathbf{x}}_0) = \{1\}$. In the same way as the above this leads to the subproblem

$$\begin{aligned} & \min d_1 + d_2, \\ & \text{s.t. } d_2 \leq 0 \end{aligned}$$

whose objective function is unbounded; therefore $\mathcal{C}_l(\tilde{\mathbf{x}}_0) \cap \mathcal{C}_{dd}(\tilde{\mathbf{x}}_0) \neq \emptyset$. So $\tilde{\mathbf{x}}_0$ is not a local minimizer, but the point \mathbf{x}_0 remains as a candidate.

Convex Functions

Convexity plays a central role in optimization. We already had some simple results from Convex Analysis in Chapters 三 and 四. Convex optimization problems – the functions f and g_i are supposed to be convex and the functions h_j affinely linear – are by far easier to solve than general nonlinear problems. These assumptions ensure that the problems are well-behaved. They have two significant properties: *A local minimizer is always a global one.* The KKT conditions are sufficient for optimality. A special feature of strictly convex functions is that they have at most one minimal point. But convex functions also play an important role in problems that are not convex. Therefore a simple and short treatment of convex functions is given here:

Definition 477. Let $D \subset \mathbb{R}^n$ be nonempty and convex. A real-valued function f defined on at least D is called convex on D if and only if

$$f((1 - \tau)\mathbf{x} + \tau\mathbf{y}) \leq (1 - \tau)f(\mathbf{x}) + \tau f(\mathbf{y})$$

holds for all $\mathbf{x}, \mathbf{y} \in D$ and $\tau \in (0, 1)$. f is called strictly convex on D if and only if

$$f((1 - \tau)\mathbf{x} + \tau\mathbf{y}) < (1 - \tau)f(\mathbf{x}) + \tau f(\mathbf{y})$$

for all $\mathbf{x}, \mathbf{y} \in D$ with $\mathbf{x} \neq \mathbf{y}$ and $\tau \in (0, 1)$. The addition “on D ” will be omitted, if D is the domain of definition. We say f is concave (on D) iff $-f$ is convex, and strictly concave (on D) iff $-f$ is strictly convex.

For a concave function the line segment joining two points on the graph is never above the graph.

Let $D \subset \mathbb{R}^n$ be nonempty and convex and $f : D \rightarrow \mathbb{R}$ a convex function.

Proposition 478. 1) If f attains a local minimum at a point $\mathbf{x}^* \in D$, then $f(\mathbf{x}^*)$ is the global minimum.

2) f is continuous in D° , the interior of D .

3) The function ψ defined by $\psi(\tau) := \frac{f(\mathbf{x} + \tau\mathbf{h}) - f(\mathbf{x})}{\tau}$ for $\mathbf{x} \in D^\circ, \mathbf{h} \in \mathbb{R}^n$ and sufficiently small, positive τ is isotone, that is, order-preserving.

4) For D open and a differentiable f it holds that $f(\mathbf{y}) - f(\mathbf{x}) \geq f'(\mathbf{x})(\mathbf{y} - \mathbf{x})$ for all $\mathbf{x}, \mathbf{y} \in D$.

With the function f defined by $f(x) := 0$ for $x \in [0, 1)$ and $f(1) := 1$ we can see that assertion 2) cannot be extended to the whole of D .

Proof. 1) If there exists an $\mathbf{x} \in D$ such that $f(\bar{\mathbf{x}}) < f(\mathbf{x}^*)$, then we would have

$$f((1 - \tau)\mathbf{x}^* + \tau\bar{\mathbf{x}}) \leq (1 - \tau)f(\mathbf{x}^*) + \tau f(\bar{\mathbf{x}}) < f(\mathbf{x}^*)$$

for $0 < \tau \leq 1$ and consequently a contradiction to the fact that f attains a local minimum at \mathbf{x}^* .

2) For $\mathbf{x}_0 \in D^\circ$ consider the function ψ defined by $\psi(\mathbf{h}) := f(\mathbf{x}_0 + \mathbf{h}) - f(\mathbf{x}_0)$ for $\mathbf{h} \in \mathbb{R}^n$ with a sufficiently small norm $\|\mathbf{h}\|_\infty$: It is clear that the function ψ is convex. Let $\rho > 0$ such that for

$$K := \{\mathbf{h} \in \mathbb{R}^n \mid \|\mathbf{h}\|_\infty \leq \rho\}$$

it holds that $\mathbf{x}_0 + K \subset D^\circ$. Evidently, there exist $m \in \mathbb{N}$ and $\mathbf{a}_1, \dots, \mathbf{a}_m \in \mathbb{R}^n$ with $K = \text{conv}(\mathbf{a}_1, \dots, \mathbf{a}_m)$ (convex hull). Every $\mathbf{h} \in K$ may be represented as $\mathbf{h} = \sum_{i=1}^m \gamma_i \mathbf{a}_i$ with $\gamma_i \geq 0$ satisfying $\sum_{i=1}^m \gamma_i = 1$. With

$$\alpha := \begin{cases} \max\{|\psi(\mathbf{a}_i)| \mid i = 1, \dots, m\}, & \text{if positive} \\ 1, & \text{otherwise.} \end{cases}$$

we have $\psi(\mathbf{h}) \leq \sum_{i=1}^m \gamma_i \psi(\alpha_i) \leq \alpha$. Now let $\epsilon \in (0, \alpha]$. Then firstly for all $\mathbf{h} \in \mathbb{R}^n$ with $\|\mathbf{h}\|_\infty \leq \epsilon\rho/\alpha$ we have

$$\psi(\mathbf{h}) = \psi\left(\left(1 - \frac{\epsilon}{\alpha}\right)\mathbf{0} + \frac{\epsilon}{\alpha}\left(\frac{\alpha}{\epsilon}\mathbf{h}\right)\right) \leq \frac{\epsilon}{\alpha}\psi\left(\frac{\alpha}{\epsilon}\mathbf{h}\right) \leq \epsilon,$$

and therefore with

$$0 = \psi(\mathbf{0}) = \psi\left(\frac{1}{2}\mathbf{h} - \frac{1}{2}\mathbf{h}\right) \leq \frac{1}{2}\psi(\mathbf{h}) + \frac{1}{2}\psi(-\mathbf{h}),$$

$\psi(\mathbf{h}) \geq -\psi(-\mathbf{h}) \geq -\epsilon$, hence, all together $|\psi(\mathbf{h})| \leq \epsilon$.

3) Since f is convex, we have

$$f(\mathbf{x} + \tau_0 \mathbf{h}) = f\left(\left(1 - \frac{\tau_0}{\tau_1}\right)\mathbf{x} + \frac{\tau_0}{\tau_1}(\mathbf{x} + \tau_1 \mathbf{h})\right) \leq \left(1 - \frac{\tau_0}{\tau_1}\right)f(\mathbf{x}) + \frac{\tau_0}{\tau_1}f(\mathbf{x} + \tau_1 \mathbf{h})$$

for $0 < \tau_0 < \tau_1$. Transformation leads to

$$\frac{f(\mathbf{x} + \tau_0 \mathbf{h}) - f(\mathbf{x})}{\tau_0} \leq \frac{f(\mathbf{x} + \tau_1 \mathbf{h}) - f(\mathbf{x})}{\tau_1}$$

4) This follows directly from 3) (with $\mathbf{h} = \mathbf{y} - \mathbf{x}$):

$$f'(\mathbf{x})\mathbf{h} = \lim_{\tau \rightarrow 0^+} \frac{f(\mathbf{x} + \tau \mathbf{h}) - f(\mathbf{x})}{\tau} \leq \frac{f(\mathbf{x} + \mathbf{h}) - f(\mathbf{x})}{1}$$

□

Constraint Qualifications

The condition $\mathcal{C}_l(\mathbf{x}_0)^* = \mathcal{C}_t(\mathbf{x}_0)^*$ is very abstract, extremely general, but not easily verifiable. Therefore, for practical problems, we will try to find regularity assumptions called constraint qualifications (CQ) which are more specific, easily verifiable, but also somewhat restrictive.

For the moment we will consider the case that we only have inequality constraints. Hence, $\mathcal{E} = \emptyset$ and $\mathcal{I} = \{1, \dots, m\}$ with an $m \in \mathbb{N}_0$. Linear constraints pose fewer problems than nonlinear constraints. Therefore, we will assume the partition

$$\mathcal{I} = \mathcal{I}_1 \uplus \mathcal{I}_2.$$

If and only if $i \in \mathcal{I}_2$ let $g_i(\mathbf{x}) = \mathbf{a}_i^\top \mathbf{x} - b_i$ with suitable vectors \mathbf{a}_i and \mathbf{b}_i , that is, g_i is ‘linear’, more precisely affinely linear. Corresponding to this partition, we will also split up the set of active constraints $\mathcal{A}(\mathbf{x}_0)$ for $\mathbf{x}_0 \in \mathcal{F}$ into

$$\mathcal{A}_j(\mathbf{x}_0) := \mathcal{I}_j \cap \mathcal{A}(\mathbf{x}_0) \text{ for } j = 1, 2.$$

We will now focus on the following Constraint Qualifications:

(GCQ) Guignard Constraint Qualification: $\mathcal{C}_l(\mathbf{x}_0)^* = \mathcal{C}_t(\mathbf{x}_0)^*$.

(ACQ) Abadie Constraint Qualification: $\mathcal{C}_l(\mathbf{x}_0) = \mathcal{C}_t(\mathbf{x}_0)$.

(MFCQ) Mangasarian-Fromovitz Constraint Qualification:

$$\exists \mathbf{d} \in \mathbb{R}^n \text{ such that } \begin{cases} g'_i(\mathbf{x}_0)\mathbf{d} < 0, & \text{for } i \in \mathcal{A}_1(\mathbf{x}_0), \\ g'_i(\mathbf{x}_0)\mathbf{d} \leq 0, & \text{for } i \in \mathcal{A}_2(\mathbf{x}_0). \end{cases}$$

(SCQ) Slater Constraint Qualification: The functions g_i are convex for all $i \in \mathcal{I}$ and

$$\exists \tilde{\mathbf{x}} \in \mathcal{F}, g_i(\tilde{\mathbf{x}}) < 0 \text{ for } i \in \mathcal{I}_1.$$

The conditions $g'_i(\mathbf{x}_0)\mathbf{d} < 0$ and $g'_i(\mathbf{x}_0)\mathbf{d} \leq 0$ each define half spaces. (MFCQ) means nothing else but that the intersection of all of these half spaces is nonempty.

We will prove: (SCQ) \Rightarrow (MFCQ) \Rightarrow (ACQ).

Proof. (SCQ) \Rightarrow (MFCQ): From the properties of convex and affinely linear functions and the definition of $\mathcal{A}(\mathbf{x}_0)$ we get:

$$g'_i(\mathbf{x}_0)(\tilde{\mathbf{x}} - \mathbf{x}_0) \leq g_i(\tilde{\mathbf{x}}) - g_i(\mathbf{x}_0) = g_i(\tilde{\mathbf{x}}) < 0 \text{ for } i \in \mathcal{A}_1(\mathbf{x}_0),$$

$$g'_i(\mathbf{x}_0)(\tilde{\mathbf{x}} - \mathbf{x}_0) = g_i(\tilde{\mathbf{x}}) - g_i(\mathbf{x}_0) = g_i(\tilde{\mathbf{x}}) \leq 0 \text{ for } i \in \mathcal{A}_2(\mathbf{x}_0).$$

(MFCQ) \Rightarrow (ACQ): Lemma 472 gives that $\mathcal{C}_t(\mathbf{x}_0) \subset \mathcal{C}_l(\mathbf{x}_0)$ and $\mathbf{0} \in \mathcal{C}_t(\mathbf{x}_0)$ always hold. Therefore it remains to prove that $\mathcal{C}_l(\mathbf{x}_0) \setminus \{\mathbf{0}\} \subset \mathcal{C}_t(\mathbf{x}_0)$. So let $\mathbf{d}_0 \in \mathcal{C}_l(\mathbf{x}_0) \setminus \{\mathbf{0}\}$. Take \mathbf{d} as stated in (MFCQ). Then for a sufficiently small $\lambda > 0$ we have $\mathbf{d}_0 + \lambda\mathbf{d} \neq \mathbf{0}$. Since \mathbf{d}_0 is in $\mathcal{C}_l(\mathbf{x}_0)$, it follows that

$$g'_i(\mathbf{x}_0)(\mathbf{d}_0 + \lambda\mathbf{d}) < 0 \text{ for } i \in \mathcal{A}_1(\mathbf{x}_0) \text{ and}$$

$$g'_i(\mathbf{x}_0)(\mathbf{d}_0 + \lambda\mathbf{d}) \leq 0 \text{ for } i \in \mathcal{A}_2(\mathbf{x}_0).$$

For the moment take a fixed λ . Setting $\mathbf{u} := \frac{\mathbf{d}_0 + \lambda\mathbf{d}}{\|\mathbf{d}_0 + \lambda\mathbf{d}\|_2}$ produces

$$g_i(\mathbf{x}_0 + t\mathbf{u}) = \underbrace{g_i(\mathbf{x}_0)}_{=0} + t\underbrace{g'_i(\mathbf{x}_0)\mathbf{u}}_{<0} + o(t) \text{ for } i \in \mathcal{A}_1(\mathbf{x}_0) \text{ and}$$

$$g_i(\mathbf{x}_0 + t\mathbf{u}) = \underbrace{g_i(\mathbf{x}_0)}_{=0} + t\underbrace{g'_i(\mathbf{x}_0)\mathbf{u}}_{\leq 0} \text{ for } i \in \mathcal{A}_2(\mathbf{x}_0).$$

Thus, we have $g_i(\mathbf{x}_0 + t\mathbf{u}) \leq 0$ for $i \in \mathcal{A}(\mathbf{x}_0)$ and $t > 0$ sufficiently small. For the indices $i \in \mathcal{I} \setminus \mathcal{A}(\mathbf{x}_0)$ this is obviously true. Hence, there exists a $t_0 > 0$ such that $\mathbf{x}_0 + t\mathbf{u} \in \mathcal{F}$ for $0 \leq t \leq t_0$. For the sequence $\{\mathbf{x}_k\}$ defined by $\mathbf{x}_k := \mathbf{x}_0 + \frac{t_0}{k}\mathbf{u}$ it holds that $\mathbf{x}_k \xrightarrow{\mathbf{u}} \mathbf{x}_0$. Therefore, $\mathbf{u} \in \mathcal{C}_t(\mathbf{x}_0)$ and consequently $\mathbf{d}_0 + \lambda\mathbf{d} \in \mathcal{C}_t(\mathbf{x}_0)$. Passing to the limit as $\lambda \rightarrow 0$ yields $\mathbf{d}_0 \in \overline{\mathcal{C}_t(\mathbf{x}_0)}$. Lemma 472 gives that $\mathcal{C}_t(\mathbf{x}_0)$ is closed. Hence, $\mathbf{d}_0 \in \mathcal{C}_t(\mathbf{x}_0)$. \square

Now we will consider the general case, where there may also occur equality constraints. In this context one often finds the following linear independence constraint qualification in the literature:

(LICQ)

The vectors $\{\nabla g_i(\mathbf{x}_0) | i \in \mathcal{A}(\mathbf{x}_0)\}$ and $\{\nabla h_i(\mathbf{x}_0) | j \in \mathcal{E}\}$ are linearly independent.

(LICQ) greatly reduces the number of active inequality constraints. Instead of (LICQ) we will now consider the following weaker constraint qualification which is a variant of (MFCQ), and is often cited as the Arrow-Hurwitz-Uzawa constraint qualification:

(AHUCQ)

There exists a $\mathbf{d} \in \mathbb{R}^n$ such that $\begin{cases} g'_i(\mathbf{x}_0)\mathbf{d} < 0, & \text{for } i \in \mathcal{A}(\mathbf{x}_0) \\ h'_j(\mathbf{x}_0)\mathbf{d} = 0, & \text{for } j \in \mathcal{E} \end{cases}$ and the vectors $\{\nabla h_j(\mathbf{x}_0) | j \in \mathcal{E}\}$ are linearly independent.

We will show: (LICQ) \Rightarrow (AHUCQ) \Rightarrow (ACQ).

Proof. (LICQ) \Rightarrow (AHUCQ): (AHUCQ) follows, for example, directly from the solvability of the system of linear equations

$$\begin{aligned} g'_i(\mathbf{x}_0)\mathbf{d} &= -1, \quad \text{for } i \in \mathcal{A}(\mathbf{x}_0), \\ h'_j(\mathbf{x}_0)\mathbf{d} &= 0, \quad \text{for } j \in \mathcal{E}. \end{aligned}$$

(AHUCQ) \Rightarrow (ACQ): Lemma 472 gives that again we only have to show $\mathbf{d}_0 \in \mathcal{C}_t(\mathbf{x}_0)$ for all $\mathbf{d}_0 \in \mathcal{C}_t(\mathbf{x}_0) \setminus \{\mathbf{0}\}$. Take d as stated in (AHUCQ). Then we have $\mathbf{d}_0 + \lambda\mathbf{d} =: \mathbf{w} \neq \mathbf{0}$ for a sufficiently small $\lambda > 0$ and thus

$$\begin{aligned} g'_i(\mathbf{x}_0)\mathbf{w} &< 0, \quad \text{for } i \in \mathcal{A}(\mathbf{x}_0) \text{ and} \\ h'_j(\mathbf{x}_0)\mathbf{w} &= 0, \quad \text{for } j \in \mathcal{E}. \end{aligned}$$

Denote

$$\mathbf{A} := [\nabla h_1(\mathbf{x}_0), \dots, \nabla h_p(\mathbf{x}_0)] \in \mathbb{R}^{n \times p}.$$

For that $\mathbf{A}^\top \mathbf{A}$ is regular because $\text{rank}(\mathbf{A}) = p$. Now consider the following system of linear equations dependent on $\mathbf{u} \in \mathbb{R}^p$ and $t \in \mathbb{R}$:

$$\phi_j(\mathbf{u}, t) := h_j(\mathbf{x}_0 + \mathbf{A}\mathbf{u} + t\mathbf{w}) = 0, \quad j = 1, \dots, p.$$

For the corresponding vector-valued function ϕ we have $\phi(0, 0) = 0$, and because of

$$\frac{\partial \phi_j}{\partial u_i}(\mathbf{u}, t) = h'_j(\mathbf{x}_0 + \mathbf{A}\mathbf{u} + t\mathbf{w})\nabla h_j(\mathbf{x}_0),$$

we are able to solve $\phi(\mathbf{u}, t) = 0$ locally for \mathbf{u} , that is, there exist a null neighborhood $U_0 \subset \mathbb{R}$ and a continuously differentiable function $u : U_0 \rightarrow \mathbb{R}^p$ satisfying

$$\begin{aligned} u(0) &= \mathbf{0}, \\ h_j(\underbrace{\mathbf{x}_0 + \mathbf{A}u(t) + t\mathbf{w}}_{=: \mathbf{x}(t)}) &= 0, \text{ for } t \in U_0, \quad j = 1, \dots, p. \end{aligned}$$

Differentiation with respect to t at $t = 0$ leads to

$$h'_j(\mathbf{x}_0)(\mathbf{A}u'(0) + \mathbf{w}) = 0, \quad j = 1, \dots, p,$$

and consequently – considering that $h'_j(\mathbf{x}_0)\mathbf{w} = 0$ and $\mathbf{A}^\top \mathbf{A}$ is regular – to $u'(0) = 0$. Then for $i \in \mathcal{A}(\mathbf{x}_0)$ it holds that

$$g_i(\mathbf{x}(t)) = g_i(\mathbf{x}_0) + tg'_i(\mathbf{x}_0)\mathbf{x}'(0) + o(t) = tg'_i(\mathbf{x}_0)(\mathbf{A}u'(0) + \mathbf{w}) + o(t).$$

With $u'(0) = 0$ we obtain

$$g_i(\mathbf{x}(t)) = t \left(g'_i(\mathbf{x}_0)\mathbf{w} + \frac{o(t)}{t} \right)$$

and the latter is negative for $t > 0$ sufficiently small.

Hence, there exists a $t_1 > 0$ with $\mathbf{x}(t) \in \mathcal{F}$ for $0 \leq t \leq t_1$. From

$$\mathbf{x}\left(\frac{t_1}{k}\right) = \mathbf{x}_0 + \frac{t_1}{k} \left(\mathbf{w} + \underbrace{\mathbf{A} \frac{u(t_1/k)}{t_1/k}}_{\rightarrow 0 (k \rightarrow \infty)} \right)$$

for $k \in \mathbb{N}$ we get $\mathbf{x}(\frac{t_1}{k}) \xrightarrow{k \rightarrow \infty} \mathbf{x}_0$; this yields $\mathbf{w} = \mathbf{d}_0 + \lambda \mathbf{d} \in \mathcal{C}_t(\mathbf{x}_0)$ and also by passing to the limit as $\lambda \rightarrow 0$

$$\mathbf{d}_0 \in \overline{\mathcal{C}_t(\mathbf{x}_0)} = \mathcal{C}_t(\mathbf{x}_0).$$

□

Convex Optimization Problems

Firstly suppose that $C \subset \mathbb{R}^n$ is nonempty and the functions $f, g_i : C \rightarrow \mathbb{R}$ are arbitrary for $i \in \mathcal{I}$. We consider the general optimization problem

$$(P) \begin{cases} \min f(\mathbf{x}), \\ g_i(\mathbf{x}) \leq 0, \text{ for } i \in \mathcal{I} := \{1, \dots, m\}. \end{cases}$$

In the following section the Lagrangian L to (P) defined by

$$L(\mathbf{x}, \boldsymbol{\lambda}) := f(\mathbf{x}) + \sum_{i=1}^m \lambda_i g_i(\mathbf{x}) = f(\mathbf{x}) + \langle \boldsymbol{\lambda}, g(\mathbf{x}) \rangle \text{ for } \mathbf{x} \in C \text{ and } \boldsymbol{\lambda} \in \mathbb{R}_+^m$$

will play an important role. As usual we have combined the m functions g_i to a vector-valued function g .

Definition 479. A pair $(\mathbf{x}^*, \boldsymbol{\lambda}^*) \in C \times \mathbb{R}_+^m$ is called a saddle point of L if and only if

$$L(\mathbf{x}^*, \boldsymbol{\lambda}) \leq L(\mathbf{x}^*, \boldsymbol{\lambda}^*) \leq L(\mathbf{x}, \boldsymbol{\lambda}^*)$$

holds for all $\mathbf{x} \in C$ and $\boldsymbol{\lambda} \in \mathbb{R}_+^m$, that is, \mathbf{x}^* minimizes $L(\cdot, \boldsymbol{\lambda}^*)$ and $\boldsymbol{\lambda}^*$ maximizes $L(\mathbf{x}^*, \cdot)$.

Lemma 480. If $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ is a saddle point of L , then it holds that:

- \mathbf{x}^* is a global minimizer of (P) .
- $L(\mathbf{x}^*, \boldsymbol{\lambda}^*) = f(\mathbf{x}^*)$.
- $\lambda_i^* g_i(\mathbf{x}^*) = 0$ for all $i \in \mathcal{I}$.

Proof. Let $\mathbf{x} \in C$ and $\boldsymbol{\lambda} \in \mathbb{R}_+^m$. From

$$0 \geq L(\mathbf{x}^*, \boldsymbol{\lambda}) - L(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \langle \boldsymbol{\lambda} - \boldsymbol{\lambda}^*, g(\mathbf{x}^*) \rangle \quad (6.4)$$

we obtain for $\boldsymbol{\lambda} := \mathbf{0}$,

$$\langle \boldsymbol{\lambda}^*, g(\mathbf{x}^*) \rangle \geq 0. \quad (6.5)$$

With $\boldsymbol{\lambda} := \boldsymbol{\lambda}^* + \mathbf{e}_i$ we get – also from (6.4) –

$$g_i(\mathbf{x}^*) \leq 0 \text{ for all } i \in \mathcal{I}, \text{ that is, } g(\mathbf{x}^*) \leq 0. \quad (6.6)$$

Because of (6.6), it holds that $\langle \boldsymbol{\lambda}^*, g(\mathbf{x}^*) \rangle \leq 0$. Together with (6.5) this produces

$$\langle \boldsymbol{\lambda}^*, g(\mathbf{x}^*) \rangle = 0 \text{ and hence, } \lambda_i^* g_i(\mathbf{x}^*) = 0 \text{ for all } i \in \mathcal{I}.$$

For $\mathbf{x} \in \mathcal{F}$ it follows that

$$f(\mathbf{x}^*) = L(\mathbf{x}^*, \boldsymbol{\lambda}^*) \leq L(\mathbf{x}, \boldsymbol{\lambda}^*) = f(\mathbf{x}) + \underbrace{\langle \boldsymbol{\lambda}^*, g(\mathbf{x}) \rangle}_{\leq 0} \leq f(\mathbf{x}).$$

Therefore \mathbf{x}^* is a global minimizer of (P) . □

We assume now that C is open and convex and the functions $f, g_i : C \rightarrow \mathbb{R}$ are continuously differentiable and convex for $i \in \mathcal{I}$. In this case we write more precisely (CP) instead of (P) .

Theorem 481. If the Slater constraint qualification holds and \mathbf{x}^* is a minimizer of (CP) , then there exists a vector $\boldsymbol{\lambda}^* \in \mathbb{R}_+^m$ such that $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ is a saddle point of L .

Proof. Taking into account our observations from *Constraint Qualifications*, Theorem 475 gives that there exists a $\lambda^* \in \mathbb{R}_+^m$ such that

$$0 = L_x(\mathbf{x}^*, \boldsymbol{\lambda}^*) \text{ and } \langle \boldsymbol{\lambda}^*, g(\mathbf{x}^*) \rangle = 0.$$

With that we get for $\mathbf{x} \in C$,

$$L(\mathbf{x}, \boldsymbol{\lambda}^*) - L(\mathbf{x}^*, \boldsymbol{\lambda}^*) \geq L_x(\mathbf{x}^*, \boldsymbol{\lambda}^*)(\mathbf{x} - \mathbf{x}^*) = 0$$

and

$$L(\mathbf{x}^*, \boldsymbol{\lambda}^*) - L(\mathbf{x}^*, \boldsymbol{\lambda}) = -\langle \underbrace{\boldsymbol{\lambda}}_{\geq 0}, \underbrace{g(\mathbf{x}^*)}_{\leq 0} \rangle \geq 0.$$

Hence, $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ is a saddle point of L . □

The following example shows that the Slater constraint qualification is essential in this theorem:

Example 482. With $n = 1$ and $m = 1$ we regard the convex problem

$$(P) \begin{cases} \min_x f(x) := -x, \\ g(x) := x^2 \leq 0. \end{cases}$$

The only feasible point is $x^* = 0$ with value $f(0) = 0$. So 0 minimizes $f(x)$ subject to $g(x) \leq 0$.

$L(x, \lambda) := -x + \lambda x^2$ for $\lambda \geq 0, x \in \mathbb{R}$. There is no $\lambda^* \in [0, \infty)$ such that (x^*, λ^*) is a saddle point of L .

The following important observation shows that neither constraint qualifications nor second-order optimality conditions, which we will deal with in the next section, are needed for a *sufficient* condition for general *convex optimization problems*:

Suppose that $f, g_i, h_j : \mathbb{R}^n \rightarrow \mathbb{R}$ are continuously differentiable functions with f and g_i convex and h_j (affinely) linear ($i \in \mathcal{I}, j \in \mathcal{E}$), and consider the following convex optimization problem:

$$(CP) \begin{cases} \min_{\mathbf{x}} f(\mathbf{x}), \\ g_i(\mathbf{x}) \leq 0, \text{ for } i \in \mathcal{I} \\ h_j(\mathbf{x}) = 0, \text{ for } j \in \mathcal{E}. \end{cases}$$

We will show that *for this special kind of problem every KKT point already gives a (global) minimum*:

Theorem 483. Suppose $\mathbf{x}_0 \in \mathcal{F}$ and there exist vectors $\boldsymbol{\lambda} \in \mathbb{R}_+^m$ and $\boldsymbol{\mu} \in \mathbb{R}^p$ such that

$$\nabla f(\mathbf{x}_0) + \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}_0) + \sum_{j=1}^p \mu_j \nabla h_j(\mathbf{x}_0) = \mathbf{0} \text{ and}$$

$$\lambda_i g_i(\mathbf{x}_0) = 0 \text{ for } i = 1, \dots, m,$$

then (CP) attains its global minimum at \mathbf{x}_0 .

Proof. The proof of this theorem is surprisingly simple. Taking into account Proposition 478-4), we get for $\mathbf{x} \in \mathcal{F}$:

$$\begin{aligned} f(\mathbf{x}) - f(\mathbf{x}_0) &\stackrel{f \text{ convex}}{\geq} f'(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) \\ &= - \sum_{i=1}^m \lambda_i g'_i(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) - \underbrace{\sum_{j=1}^p \mu_j h'_j(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0)}_{=h_j(\mathbf{x})-h_j(\mathbf{x}_0)=0} \\ &\stackrel{g_i \text{ convex}}{\geq} - \sum_{i=1}^m \lambda_i (g_i(\mathbf{x}) - g_i(\mathbf{x}_0)) = - \sum_{i=1}^m \lambda_i g_i(\mathbf{x}) \geq 0. \end{aligned}$$

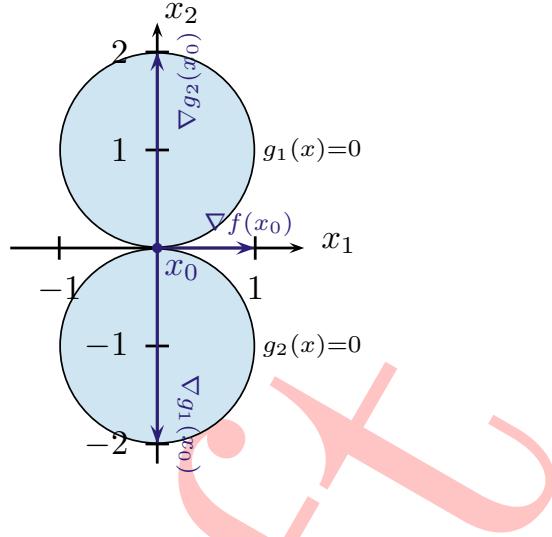
□

The following example shows that even if we have convex problems the KKT conditions are not necessary for minimal points:

Example 484. With $n = 2, m = 2$ and $\mathbf{x} = (x_1, x_2)^\top \in D := \mathbb{R}^2$ we consider:

$$(P) \begin{cases} \min_{\mathbf{x}} f(\mathbf{x}) := x_1, \\ g_1(\mathbf{x}) := x_1^2 + (x_2 - 1)^2 - 1 \leq 0, \\ g_2(\mathbf{x}) := x_1^2 + (x_2 + 1)^2 - 1 \leq 0. \end{cases}$$

Obviously, only the point $\mathbf{x}_0 := (0, 0)^\top$ is feasible. Hence, \mathbf{x}_0 is the (global) minimal point. Since $\nabla f(\mathbf{x}_0) = (1, 0)^\top$, $\nabla g_1(\mathbf{x}_0) = (0, -2)^\top$ and $\nabla g_2(\mathbf{x}_0) = (0, 2)^\top$, the gradient condition of the KKT conditions is not met. f is linear, the functions g_i are convex. Evidently, however, the Slater condition is not fulfilled.



Of course, one could also argue from Proposition 466: The cones

$$\mathcal{C}_{dd}(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^2 | f'(\mathbf{x}_0)\mathbf{d} < 0\} = \{\mathbf{d} \in \mathbb{R}^2 | d_1 < 0\}$$

and

$$\mathcal{C}_l(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^2 | \forall i \in \mathcal{A}(\mathbf{x}_0), g'_i(\mathbf{x}_0)\mathbf{d} \leq 0\} = \{\mathbf{d} \in \mathbb{R}^2 | d_2 = 0\}$$

are clearly not disjoint.

(Taken from Section 5.5.3 from [18])

Example 485. Equality constrained convex quadratic minimization. We consider the problem

$$\begin{aligned} & \text{minimize} && (1/2)x^T Px + q^T x + r \\ & \text{subject to} && Ax = b, \end{aligned} \tag{6.7}$$

where $P \in \mathbf{S}_+^n$. The KKT conditions for this problem are

$$Ax^* = b, \quad Px^* + q + A^T \nu^* = 0,$$

which we can write as

$$\begin{bmatrix} P & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x^* \\ \nu^* \end{bmatrix} = \begin{bmatrix} -q \\ b \end{bmatrix}.$$

Solving this set of $m+n$ equations in the $m+n$ variables x^*, ν^* gives the optimal primal and dual variables for (6.7).

Example 486. Water-filling. We consider the convex optimization problem

$$\begin{aligned} & \text{minimize} && - \sum_{i=1}^n \log(\alpha_i + x_i) \\ & \text{subject to} && x \geq 0, \quad \mathbf{1}^T x = 1, \end{aligned}$$

where $\alpha_i > 0$. This problem arises in information theory, in allocating power to a set of n communication channels. The variable x_i represents the transmitter power allocated to the i th channel, and $\log(\alpha_i + x_i)$ gives the capacity or communication rate of the channel, so the problem is to allocate a total power of one to the channels, in order to maximize the total communication rate.

Introducing Lagrange multipliers $\lambda^* \in \mathbb{R}^n$ for the inequality constraints $x^* \geq 0$, and a multiplier $\nu^* \in \mathbb{R}$ for the equality constraint $\mathbf{1}^T x = 1$, we obtain the KKT conditions

$$\begin{aligned} x^* \geq 0, \quad \mathbf{1}^T x^* = 1, \quad \lambda^* \geq 0, \quad \lambda_i^* x_i^* = 0, \quad i = 1, \dots, n, \\ -1/(\alpha_i + x_i^*) - \lambda_i^* + \nu^* = 0, \quad i = 1, \dots, n. \end{aligned}$$

We can directly solve these equations to find x^* , λ^* , and ν^* . We start by noting that λ^* acts as a slack variable in the last equation, so it can be eliminated, leaving

$$\begin{aligned} x^* \geq 0, \quad \mathbf{1}^T x^* = 1, \quad x_i^* (\nu^* - 1/(\alpha_i + x_i^*)) = 0, \quad i = 1, \dots, n, \\ \nu^* \geq 1/(\alpha_i + x_i^*), \quad i = 1, \dots, n. \end{aligned}$$

If $\nu^* < 1/\alpha_i$, this last condition can only hold if $x_i^* > 0$, which by the third condition implies that $\nu^* = 1/(\alpha_i + x_i^*)$. Solving for x_i^* , we conclude that $x_i^* = 1/\nu^* - \alpha_i$ if $\nu^* < 1/\alpha_i$. If $\nu^* \geq 1/\alpha_i$, then $x_i^* > 0$ is impossible, because it would imply $\nu^* \geq 1/\alpha_i > 1/(\alpha_i + x_i^*)$, which violates the complementary slackness condition. Therefore, $x_i^* = 0$ if $\nu^* \geq 1/\alpha_i$. Thus we have

$$x_i^* = \begin{cases} 1/\nu^* - \alpha_i & \nu^* < 1/\alpha_i \\ 0 & \nu^* \geq 1/\alpha_i, \end{cases}$$

or, put more simply, $x_i^* = \max\{0, 1/\nu^* - \alpha_i\}$. Substituting this expression for x_i^* into the condition $\mathbf{1}^T x^* = 1$ we obtain

$$\sum_{i=1}^n \max\{0, 1/\nu^* - \alpha_i\} = 1.$$

The left hand side is a piecewise-linear increasing function of $1/\nu^*$, with breakpoints at α_i , so the equation has a unique solution which is readily determined.

This solution method is called water-filling for the following reason. We think of α_i as the ground level above patch i , and then flood the region with water to a depth $1/\nu$, as illustrated in Figure 6.1. The total amount of water used is then $\sum_{i=1}^n \max\{0, 1/\nu^* - \alpha_i\}$. We then increase the flood level until we have used a total amount of water equal to one. The depth of water above patch i is then the optimal value x^* .

(Taken from Section 5.5.4 from [18])

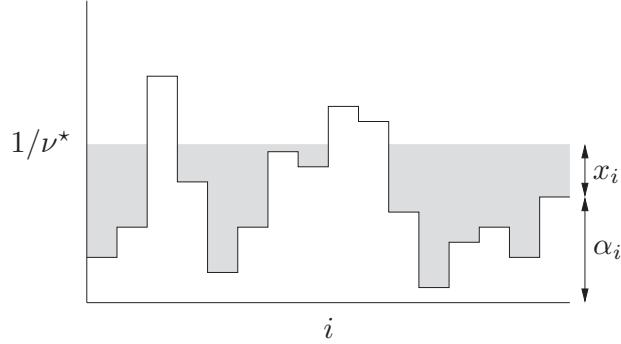


图 6.1: Illustration of water-filling algorithm. The height of each patch is given by α_i . The region is flooded to a level $1/\nu^*$ which uses a total quantity of water equal to one. The height of the water (shown shaded) above each patch is the optimal value of x_i^* .

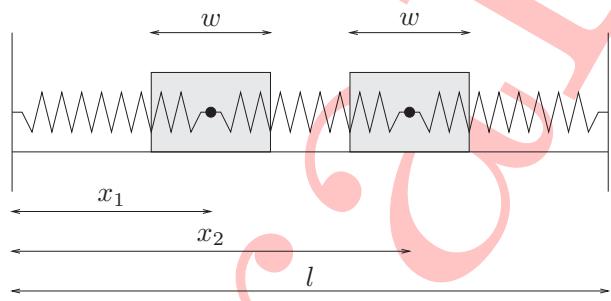


图 6.2: Two blocks connected by springs to each other, and the left and right walls. The blocks have width $w > 0$, and cannot penetrate each other or the walls.

6.2.1 Mechanics interpretation of KKT conditions

The KKT conditions can be given a nice interpretation in mechanics (which indeed, was one of Lagrange's primary motivations). We illustrate the idea with a simple example. The system shown in Figure 6.2 consists of two blocks attached to each other, and to walls at the left and right, by three springs. The position of the blocks are given by $x \in \mathbb{R}^2$ where x_1 is the displacement of the (middle of the) left block, and x_2 is the displacement of the right block. The left wall is at position 0, and the right wall is at position l .

The potential energy in the springs, as a function of the block positions, is given by

$$f_0(x_1, x_2) = \frac{1}{2}k_1x_1^2 + \frac{1}{2}k_2(x_2 - x_1)^2 + \frac{1}{2}k_3(l - x_2)^2,$$

where $k_i > 0$ are the stiffness constants of the three springs. The equilibrium position x^* is the position that minimizes the potential energy subject to the inequalities

$$w/2 - x_1 \leq 0, \quad w + x_1 - x_2 \leq 0, \quad w/2 - l + x_2 \leq 0. \quad (6.8)$$

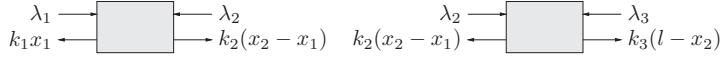


图 6.3: Force analysis of the block-spring system. The total force on each block, due to the springs and also to contact forces, must be zero. The Lagrange multipliers, shown on top, are the contact forces between the walls and blocks. The spring forces are shown at bottom.

These constraints are called *kinematic constraints*, and express the fact that the blocks have width $w > 0$, and cannot penetrate each other or the walls. The equilibrium position is therefore given by the solution of the optimization problem

$$\begin{aligned} \text{minimize} \quad & (1/2)(k_1x_1^2 + k_2(x_2 - x_1)^2 + k_3(l - x_2)^2) \\ \text{subject to} \quad & w/2 - x_1 \leq 0 \\ & w + x_1 - x_2 \leq 0 \\ & w/2 - l + x_2 \leq 0, \end{aligned} \tag{6.9}$$

which is a QP.

With $\lambda_1, \lambda_2, \lambda_3$ as Lagrange multipliers, the KKT conditions for this problem consist of the kinematic constraints (6.8), the nonnegativity constraints $\lambda_i \geq 0$, the complementary slackness conditions

$$\lambda_1(w/2 - x_1) = 0, \quad \lambda_2(w - x_2 + x_1) = 0, \quad \lambda_3(w/2 - l + x_2) = 0, \tag{6.10}$$

and the zero gradient condition

$$\begin{bmatrix} k_1x_1 - k_2(x_2 - x_1) \\ k_2(x_2 - x_1) - k_3(l - x_2) \end{bmatrix} + \lambda_1 \begin{bmatrix} -1 \\ 0 \end{bmatrix} + \lambda_2 \begin{bmatrix} 1 \\ -1 \end{bmatrix} + \lambda_3 \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 0. \tag{6.11}$$

The equation (6.11) can be interpreted as the force balance equations for the two blocks, provided we interpret the Lagrange multipliers as *contact forces* that act between the walls and blocks, as illustrated in Figure 6.3. The first equation states that the sum of the forces on the first block is zero: The term $-k_1x_1$ is the force exerted on the left block by the left spring, the term $k_2(x_2 - x_1)$ is the force exerted by the middle spring, λ_1 is the force exerted by the left wall, and $-\lambda_2$ is the force exerted by the right block. The contact forces must point away from the contact surface (as expressed by the constraints $\lambda_1 \geq 0$ and $-\lambda_2 \leq 0$), and are nonzero only when there is contact (as expressed by the first two complementary slackness conditions (6.10)). In a similar way, the second equation in (6.11) is the force balance for the second block, and the last condition in (6.10) states that λ_3 is zero unless the right block touches the wall.

In this example, the potential energy and kinematic constraint functions are convex, and (the refined form of) Slater's constraint qualification holds provided $2w \leq l$, i.e., there is enough room between the walls to fit the two blocks, so we can conclude that the energy formulation of the equilibrium given by (6.9), gives the same result as the force balance formulation, given by the KKT conditions.

6.3 Local Second-Order Optimality Conditions

To get a finer characterization, it is natural to examine the effects of second-order terms near a given point too. The following second-order results take the ‘curvature’ of the feasible region in a neighborhood of a ‘candidate’ for a minimizer into account. The necessary second-order condition $\mathbf{s}^\top \mathbf{H}\mathbf{s} \geq 0$ and the sufficient second-order condition $\mathbf{s}^\top \mathbf{H}\mathbf{s} > 0$ for the Hessian \mathbf{H} of the Lagrangian with respect to \mathbf{x} regard only certain subsets of vectors \mathbf{s} .

Suppose that the functions f, g_i and h_j are twice continuously differentiable.

Theorem 487 (Necessary second-order condition). *Suppose $\mathbf{x}_0 \in \mathcal{F}$ and there exist $\boldsymbol{\lambda} \in \mathbb{R}_+^m$ and $\boldsymbol{\mu} \in \mathbb{R}^p$ such that*

$$\nabla f(\mathbf{x}_0) + \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}_0) + \sum_{j=1}^p \mu_j \nabla h_j(\mathbf{x}_0) = \mathbf{0} \text{ and}$$

$$\lambda_i g_i(\mathbf{x}_0) = 0 \text{ for all } i \in \mathcal{I}.$$

If (P) has a local minimum at \mathbf{x}_0 , then

$$\mathbf{s}^\top \left(\nabla^2 f(\mathbf{x}_0) + \sum_{i=1}^m \lambda_i \nabla^2 g_i(\mathbf{x}_0) + \sum_{j=1}^p \mu_j \nabla^2 h_j(\mathbf{x}_0) \right) \mathbf{s} \geq 0$$

holds for all $\mathbf{s} \in \mathcal{C}_{t+}(\mathbf{x}_0)$, where

$$\mathcal{F}_+ := \mathcal{F}_+(\mathbf{x}_0) := \{ \mathbf{x} \in \mathcal{F} \mid g_i(\mathbf{x}) = 0 \text{ for all } i \in \mathcal{A}_+(\mathbf{x}_0) \},$$

$$\mathcal{A}_+(\mathbf{x}_0) := \{ i \in \mathcal{A}(\mathbf{x}_0) \mid \lambda_i > 0 \},$$

$$\mathcal{C}_{t+}(\mathbf{x}_0) := \mathcal{C}_t(\mathcal{F}_+, \mathbf{x}_0) = \{ \mathbf{d} \in \mathbb{R}^n \mid \exists \{ \mathbf{x}_k \} \in \mathcal{F}_+^{\mathbb{N}}, \mathbf{x}_k \xrightarrow{\mathbf{d}} \mathbf{x}_0 \}.$$

With the help of the Lagrangian L the second and fifth lines can be written more clearly

$$\nabla_x L(\mathbf{x}_0, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0},$$

respectively

$$\mathbf{s}^\top \nabla_{xx}^2 L(\mathbf{x}_0, \boldsymbol{\lambda}, \boldsymbol{\mu}) \mathbf{s} \geq 0.$$

Proof. It holds that

$$\lambda_i g_i(\mathbf{x}) = 0 \text{ for all } \mathbf{x} \in \mathcal{F}_+$$

because we have $\lambda_i = 0$ for $i \in \mathcal{I} \setminus \mathcal{A}_+(\mathbf{x}_0)$ and $g_i(\mathbf{x}) = 0$ for $i \in \mathcal{A}_+(\mathbf{x}_0)$, respectively.

With the function ϕ defined by

$$\phi(\mathbf{x}) := f(\mathbf{x}) + \sum_{i=1}^m \lambda_i g_i(\mathbf{x}_0) + \sum_{j=1}^p \mu_j h_j(\mathbf{x}_0) = L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$$

for $\mathbf{x} \in D$ this leads to the following relation:

$$\phi(\mathbf{x}) = f(\mathbf{x}) \text{ for } \mathbf{x} \in \mathcal{F}_+.$$

\mathbf{x}_0 gives a local minimum of f on \mathcal{F} , therefore one of ϕ on \mathcal{F}_+ . Now let $\mathbf{s} \in \mathcal{C}_{t+}(\mathbf{x}_0)$. Then by definition of the tangent cone there exists a sequence $\{\mathbf{x}^{(k)}\}$ in \mathcal{F}_+ , such that $\mathbf{x}^{(k)} = \mathbf{x}_0 + \alpha_k(\mathbf{s} + \mathbf{r}_k)$, $\alpha_k \downarrow 0$ and $\mathbf{r}_k \rightarrow \mathbf{0}$. By assumption $\nabla \phi(\mathbf{x}_0) = \mathbf{0}$. With the Taylor theorem we get

$$\begin{aligned} \phi(\mathbf{x}_0) &\leq \phi(\mathbf{x}^{(k)}) = \phi(\mathbf{x}_0) + \underbrace{\alpha_k \phi'(\mathbf{x}_0)(\mathbf{s} + \mathbf{r}_k)}_{=0} \\ &\quad + \frac{1}{2} \alpha_k^2 (\mathbf{s} + \mathbf{r}_k)^\top \nabla^2 \phi(\mathbf{x}_0 + \tau_k(\mathbf{x}^{(k)} - \mathbf{x}_0)) (\mathbf{s} + \mathbf{r}_k) \end{aligned}$$

for all sufficiently large k and a suitable $\tau_k \in (0, 1)$.

Dividing by $\alpha_k^2/2$ and passing to the limit as $k \rightarrow \infty$ gives the result

$$\mathbf{s}^\top \nabla^2 \phi(\mathbf{x}_0) \mathbf{s} \geq 0.$$

□

In the following example we will see that $\mathbf{x}_0 := (0, 0, 0)^\top$ is a stationary point. With the help of Theorem 487 we want to show that the necessary condition for a minimum is not met.

Example 488.

$$\begin{aligned} \min f(\mathbf{x}) &:= x_3 - \frac{1}{2}x_1^2, \\ \text{s.t. } g_1(\mathbf{x}) &:= -x_1^2 - x_2 - x_3 \leq 0, \\ g_2(\mathbf{x}) &:= -x_1^2 + x_2 - x_3 \leq 0, \\ g_3(\mathbf{x}) &:= -x_3 \leq 0. \end{aligned}$$

For the point $\mathbf{x}_0 := (0, 0, 0)^\top$ we have $f'(\mathbf{x}_0) = (0, 0, 1)^\top$, $\mathcal{A}(\mathbf{x}_0) = \{1, 2, 3\}$ and $g'_1(\mathbf{x}_0) = (0, -1, -1)$, $g'_2(\mathbf{x}_0) = (0, 1, -1)$, $g'_3(\mathbf{x}_0) = (0, 0, -1)$. We start with the gradient condition:

$$\nabla_x L(\mathbf{x}_0, \boldsymbol{\lambda}) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \lambda_1 \begin{pmatrix} 0 \\ -1 \\ -1 \end{pmatrix} + \lambda_2 \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix} + \lambda_3 \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{aligned} &\iff \begin{cases} -\lambda_1 + \lambda_2 = 0, \\ -\lambda_1 - \lambda_2 - \lambda_3 = -1. \end{cases} \\ &\iff \lambda_2 = \lambda_1, \quad \lambda_3 = 1 - 2\lambda_1. \end{aligned}$$

For $\lambda_1 := 1/2$ we obtain $\boldsymbol{\lambda} = (1/2, 1/2, 0)^\top \in \mathbb{R}_+^3$ and $\lambda_i g_i(\mathbf{x}_0) = 0$ for $i \in \mathcal{I}$. Hence, we get $\mathcal{A}_+(\mathbf{x}_0) = \{1, 2\}$,

$$\mathcal{F}_+ = \{\mathbf{x} \in \mathbb{R}^3 | g_1(\mathbf{x}) = g_2(\mathbf{x}) = 0, g_3(\mathbf{x}) \leq 0\} = \{(0, 0, 0)^\top\}$$

and therefore $C_{t+}(\mathbf{x}_0) = \{(0, 0, 0)^\top\}$. In this way no decision can be made! Setting $\lambda_1 := 0$ we obtain respectively $\boldsymbol{\lambda} = \mathbf{e}_3$, $\mathcal{A}_+(\mathbf{x}_0) = \{3\}$, $\mathcal{F}_+ = \{\mathbf{x} \in \mathcal{F} | x_3 = 0\}$, $\mathcal{C}_{t+}(\mathbf{x}_0) = \{\alpha \mathbf{e}_1 | \alpha \in \mathbb{R}\}$ and

$$H := \nabla^2 f(\mathbf{x}_0) + \nabla^2 g_3(\mathbf{x}_0) = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

\mathbf{H} is negative definite on $C_{t+}(\mathbf{x}_0)$. Consequently there is no local minimum of (P) at $\mathbf{x}_0 = \mathbf{0}$.

In order to expand the second-order necessary condition to a sufficient condition, we will now have to make stronger assumptions. Before we do that, let us recall that there will remain a ‘gap’ between these two conditions. This fact is well-known (even for real-valued functions of one variable) and is usually demonstrated by the functions f_2 , f_3 and f_4 defined by

$$f_k(x) := x^k \text{ for } x \in \mathbb{R}, k = 2, 3, 4,$$

at the point $x_0 = 0$.

The following **Remark** can be proven in the same way as 2) in Lemma 472:

$$\mathcal{C}_{t+}(\mathbf{x}_0) \subset \mathcal{C}_+(\mathbf{x}_0) := \begin{cases} g'_i(\mathbf{x}_0)\mathbf{s} = 0, \text{ for } i \in \mathcal{A}_+(\mathbf{x}_0), \\ g'_i(\mathbf{x}_0)\mathbf{s} \leq 0, \text{ for } i \in \mathcal{A}(\mathbf{x}_0) \setminus \mathcal{A}_+(\mathbf{x}_0), \quad \mathbf{s} \in \mathbb{R}^n. \\ h'_j(\mathbf{x}_0)\mathbf{s} = 0, \text{ for } j \in \mathcal{E}, \end{cases}$$

Theorem 489 (Sufficient second-order condition). Suppose $\mathbf{x}_0 \in \mathcal{F}$ and there exist vectors $\boldsymbol{\lambda} \in \mathbb{R}_+^m$ and $\boldsymbol{\mu} \in \mathbb{R}^p$ such that

$$\nabla_x L(\mathbf{x}_0, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0} \text{ and } \boldsymbol{\lambda}^\top g(\mathbf{x}_0) = 0.$$

Furthermore, suppose that

$$\mathbf{s}^\top \nabla_{xx}^2 L(\mathbf{x}_0, \boldsymbol{\lambda}, \boldsymbol{\mu}) \mathbf{s} > 0$$

for all $\mathbf{s} \in \mathcal{C}_{l+}(\mathbf{x}_0) \setminus \{\mathbf{0}\}$. Then (P) attains a strict local minimum at \mathbf{x}_0 .

Proof. (indirect):

If f does not have a strict local minimum at \mathbf{x}_0 , then there exists a sequence $\{\mathbf{x}^{(k)}\}$ in $\mathcal{F} \setminus \{\mathbf{x}_0\}$ with $\mathbf{x}^{(k)} \rightarrow \mathbf{x}_0$ and $f(\mathbf{x}^{(k)}) \leq f(\mathbf{x}_0)$. For $\mathbf{s}_k := \frac{\mathbf{x}^{(k)} - \mathbf{x}_0}{\|\mathbf{x}^{(k)} - \mathbf{x}_0\|_2}$ it holds that $\|\mathbf{s}_k\|_2 = 1$. Hence, there exists a convergent subsequence. WLOG suppose $\mathbf{s}_k \rightarrow \mathbf{s}$ for an $\mathbf{s} \in \mathbb{R}^n$. With $\alpha_k := \|\mathbf{x}^{(k)} - \mathbf{x}_0\|_2$ we have $\mathbf{x}^{(k)} = \mathbf{x}_0 + \alpha_k \mathbf{s}_k$ and WLOG $\alpha_k \downarrow 0$. From

$$f(\mathbf{x}_0) \geq f(\mathbf{x}^{(k)}) = f(\mathbf{x}_0) + \alpha_k f'(\mathbf{x}_0) \mathbf{s}_k + o(\alpha_k)$$

it follows that

$$f'(\mathbf{x}_0) \mathbf{s} \leq 0$$

For $i \in \mathcal{A}(\mathbf{x}_0)$ and $j \in \mathcal{E}$ we get in the same way:

$$\underbrace{g_i(\mathbf{x}^{(k)})}_{\leq 0} = \underbrace{g_i(\mathbf{x}_0)}_{=0} + \alpha g'_i(\mathbf{x}_0) \mathbf{s}_k + o(\alpha_k) \Rightarrow g'_i(\mathbf{x}_0) \mathbf{s} \leq 0.$$

$$\underbrace{h_j(\mathbf{x}^{(k)})}_{=0} = \underbrace{h_j(\mathbf{x}_0)}_{=0} + \alpha h'_j(\mathbf{x}_0) \mathbf{s}_k + o(\alpha_k) \Rightarrow h'_j(\mathbf{x}_0) \mathbf{s} = 0.$$

With the assumption $\nabla_x L(\mathbf{x}_0, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0}$ it follows that

$$\underbrace{f'(\mathbf{x}_0) \mathbf{s}}_{\leq 0} + \underbrace{\sum_{i=1}^m \lambda_i g'_i(\mathbf{x}_0) \mathbf{s}}_{= \sum_{i \in \mathcal{A}_+(\mathbf{x}_0)} \lambda_i g'_i(\mathbf{x}_0) \mathbf{s} \leq 0} + \underbrace{\sum_{j=1}^p \mu_j h'_j(\mathbf{x}_0) \mathbf{s}}_{= 0} = 0$$

and from that $g'_i(\mathbf{x}_0) \mathbf{s} = 0$ for all $i \in \mathcal{A}_+(\mathbf{x}_0)$.

Since $\|\mathbf{s}\|_2 = 1$, we get $\mathbf{s} \in \mathcal{C}_{l+}(\mathbf{x}_0) \setminus \{\mathbf{0}\}$. For the function ϕ defined by

$$\phi(\mathbf{x}) := f(\mathbf{x}) + \sum_{i=1}^m \lambda_i g_i(\mathbf{x}) + \sum_{j=1}^p \mu_j h_j(\mathbf{x}) = L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$$

it holds by assumption that $\nabla \phi(\mathbf{x}_0) = \mathbf{0}$.

$$\phi(\mathbf{x}^{(k)}) = \underbrace{f(\mathbf{x}^{(k)})}_{\leq f(\mathbf{x}_0)} + \sum_{i=1}^m \lambda_i \underbrace{g_i(\mathbf{x}^{(k)})}_{\leq 0} + \sum_{j=1}^p \mu_j \underbrace{h_j(\mathbf{x}^{(k)})}_{=0} \leq f(\mathbf{x}_0) = \phi(\mathbf{x}_0).$$

The Taylor theorem yields

$$\phi(\mathbf{x}^{(k)}) = \phi(\mathbf{x}_0) + \alpha_k \underbrace{\phi'(\mathbf{x}_0)}_{=0} \mathbf{s}_k + \frac{1}{2} \alpha_k^2 \mathbf{s}_k^\top \nabla^2 \phi(\mathbf{x}_0 + \tau_k(\mathbf{x}^{(k)} - \mathbf{x}_0)) \mathbf{s}_k$$

with a suitable $\tau_k \in (0, 1)$. From this we deduce, as usual, $\mathbf{s}^\top \nabla^2 \phi(\mathbf{x}_0) \mathbf{s} \leq 0$. With $\mathbf{s} \in \mathcal{C}_{l+}(\mathbf{x}_0) \setminus \{\mathbf{0}\}$ we get a contradiction to our assumption. \square

The following example gives a simple illustration of the necessary and sufficient second-order conditions of Theorems 487 and 489:

Example 490 (Fiacco and McCormick (1968)).

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &:= (x_1 - 1)^2 + x_2^2, \\ \text{s.t. } g_1(\mathbf{x}) &:= x_1 - \rho x_2^2 \leq 0. \end{aligned}$$

We are looking for a $\rho > 0$ such that $\mathbf{x}_0 := (0, 0)^\top$ is a local minimizer of the problem: With $\nabla f(\mathbf{x}_0) = (-2, 0)^\top$, $\nabla g_1(\mathbf{x}_0) = (1, 0)^\top$ the condition $\nabla_x L(\mathbf{x}_0, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0}$ firstly yields $\lambda_1 = 2$.

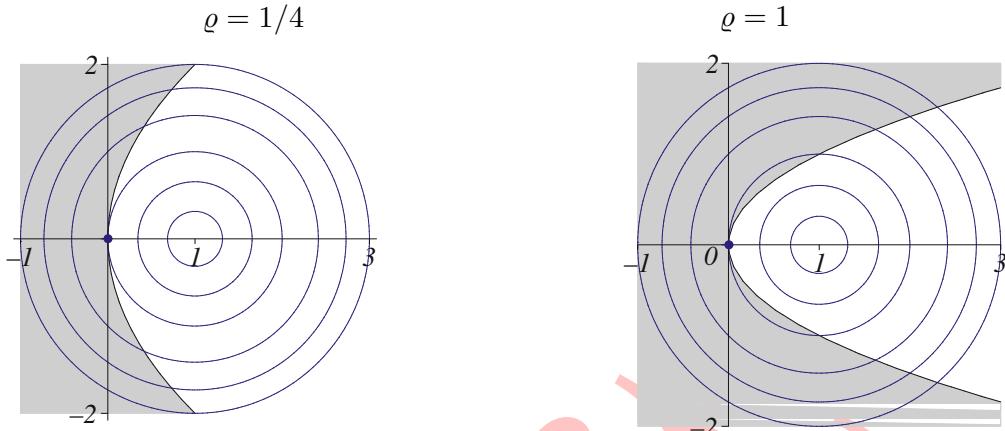
In this case (MFCQ) is fulfilled with $\mathcal{A}_1(\mathbf{x}_0) = \mathcal{A}(\mathbf{x}_0) = \{1\} = \mathcal{A}_+(\mathbf{x}_0)$. We have

$$\mathcal{C}_{l+}(\mathbf{x}_0) = \{\mathbf{d} \in \mathbb{R}^2 \mid d_1 = 0\} = \mathcal{C}_{t+}(\mathbf{x}_0).$$

The matrix

$$\nabla^2 f(\mathbf{x}_0) + 2\nabla^2 g_1(\mathbf{x}_0) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} + 2 \begin{pmatrix} 0 & 0 \\ 0 & -2\rho \end{pmatrix} = 2 \begin{pmatrix} 1 & 0 \\ 0 & 1 - 2\rho \end{pmatrix}$$

is negative definite on $\mathcal{C}_{t+}(\mathbf{x}_0)$ for $\rho > 1/2$. Thus the second-order necessary condition of Theorem 487 is violated and so there is no local minimum at \mathbf{x}_0 . For $\rho < 1/2$ the Hessian is positive definite on $\mathcal{C}_{l+}(\mathbf{x}_0)$. Hence, the sufficient conditions of Theorem 489 are fulfilled and thus there is a strict local minimum at \mathbf{x}_0 . When $\rho = 1/2$, this result is not determined by the second-order conditions; but we can confirm it in the following simple way: $f(\mathbf{x}) = (x_1 - 1)^2 + x_2^2 = x_1^2 + 1 + (x_2^2 - 2x_1)$. Because of $x_2^2 - 2x_1 \geq 0$ this yields $f(\mathbf{x}) \geq 1$ and $f(\mathbf{x}) = 1$ only for $x_1 = 0$ and $x_2^2 - 2x_1 = 0$. Hence, there is a strict local minimum at \mathbf{x}_0 .



6.4 How are Optimality Conditions Used?

The optimality conditions can be used in two ways:

1. Check whether a solution is an optimal solution or a KKT point of an optimization problem. The satisfaction of the optimality conditions can be used as stopping criteria in optimization algorithms.
2. Extremely useful in proving the convergence or convergence rate of an optimization algorithm.

For unconstrained convex programs, another way of checking the optimality of a solution is by the dual gap: $f(\mathbf{x}_k) - g(\boldsymbol{\lambda}_k, \boldsymbol{\nu}_k)$, where $g(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is the objective function of the dual problem.

6.5 Exercises

(Taken from Page 80 of [48].)

Exercise 491 (Orthogonal Distance Line Fitting). Consider the following approximation problem arising from quality control in manufacturing using coordinate measurement techniques. Let

$$M := \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$$

be a set of $m \in \mathbb{N}$ given points in \mathbb{R}^2 . The task is to find a line L

$$L(c, n_1, n_2) := \{(x, y) \in \mathbb{R}^2 \mid c + n_1x + n_2y = 0\}$$

in Hessian normal form with $n_1^2 + n_2^2 = 1$ which best approximates the point set M such that the sum of squares of the distances of the points from the straight line becomes minimal. If we calculate $r_j := c + n_1x_j + n_2y_j$ for a point (x_j, y_j) , then $|r_j|$ is its distance to L .

- a) Formulate the above problem as a constrained optimization problem.
- b) Show the existence of a solution and determine the optimal parameters c , n_1 and n_2 by means of the Lagrange multiplier rule. Explicate when and in which sense these parameters are uniquely defined.
- c) Find a (minimal) example which consists of three points and has infinitely many optimizers.
- d) Solve the optimization problem with Matlab and test your program with the following data:
- | | | | | | | | | | | |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
| x_j | 1.0 | 2.0 | 3.0 | 4.0 | 5.0 | 6.0 | 7.0 | 8.0 | 9.0 | 10.0 |
| y_j | 0.2 | 1.0 | 2.6 | 3.6 | 4.9 | 5.3 | 6.5 | 7.8 | 8.0 | 9.0 |

Exercise 492. a) Solve the optimization problem

$$\max_{\mathbf{x}} f(x_1, x_2) := 2x_1 + 3x_2, \quad \text{s.t.} \quad \sqrt{x_1} + \sqrt{x_2} = 5,$$

using Lagrange multipliers.

- b) Visualize the contour lines of f as well as the set of feasible points, and mark the solution. Explain the result!

Exercise 493. In the “colloquial speech” of mathematicians one can sometimes hear the following statement: “Strictly convex functions always have exactly one minimizer.” However, is it really right to use this term so carelessly?

Consider two typical representatives $f_i : \mathbb{R}^2 \rightarrow \mathbb{R}$, $i \in \{1, 2\}$:

$$f_1(x, y) = x^2 + y^2,$$

$$f_2(x, y) = x^2 - y^2.$$

Visualize these functions and plot their contour lines. Which function is convex? Show this analytically as well. Is the above statement correct?

Let $D_j \subset \mathbb{R}^2$ for $j \in \{1, 2, 3, 4, 5\}$ be a region in \mathbb{R}^2 with

$$D_1 := \{(x, y) \in \mathbb{R}^2 : x_1^2 + x_2^2 \leq 0.04\},$$

$$D_2 := \{(x, y) \in \mathbb{R}^2 : (x_1 - 0.55)^2 + (x_2 - 0.7)^2 \leq 0.04\},$$

$$D_3 := \{(x, y) \in \mathbb{R}^2 : (x_1 - 0.55)^2 + x_2^2 \leq 0.04\}.$$

The outer boundary of the regions D_4 and D_5 is defined by

$$x = 0.5(0.5 + 0.2 \cos(6\theta)) \cos \theta + x_c,$$

$$y = 0.5(0.5 + 0.2 \cos(6\theta)) \sin \theta + y_c,$$

where $\theta \in [0, 2\pi)$, $(x_c, y_c) = (0, 0)$ for D_4 and $(x_c, y_c) = (0, -0.7)$ for D_5 .

If we now restrict the above functions f_i to D_j ($i \in \{1, 2\}$, $j \in \{1, 2, 3, 4, 5\}$), does the statement about the uniqueness of the minimizers still hold? Find all the minimal points, where possible! Where do they lie? Which role does the convexity of the region and the function play?

Exercise 494. Find the tangent cones of the following sets

$$\begin{aligned}\mathcal{F}_1 &:= \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\|_2 = 1\}, \\ \mathcal{F}_2 &:= \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\|_2 \leq 1\}, \\ \mathcal{F}_3 &:= \{\mathbf{x} \in \mathbb{R}^2 \mid -x_1^3 + x_2 \leq 0, -x_2 \leq 0\}.\end{aligned}$$

Exercise 495. With $f(\mathbf{x}) := x_1^2 + x_2^2$ for $\mathbf{x} \in \mathbb{R}^2$ consider

$$(P) \left\{ \begin{array}{l} \min_{\mathbf{x}} f(\mathbf{x}) \\ -x_2 \leq 0 \\ x_1^3 - x_2 \leq 0 \\ x_1^3(x_2 - x_1^3) \leq 0 \end{array} \right.$$

and determine the linearizing cone, the tangent cone and the respective dual cones at the (strict global) minimal point $\mathbf{x}_0 := (0, 0)^T$.

Exercise 496. Let \mathbf{x}_0 be a feasible point of the optimization problem (P). According to “Constraint Qualification” in Section 6.2 it holds that $(LICQ) \Rightarrow (AHUCQ) \Rightarrow (ACQ)$. Show by means of the following examples (with $n = m = 2$ and $p = 0$) that these two implications do not hold in the other direction:

- a) $f(\mathbf{x}) := x_1^2 + (x_2 + 1)^2$, $g_1(\mathbf{x}) := -x_1^3 - x_2$, $g_2(\mathbf{x}) := -x_2$, $\mathbf{x}_0 := (0, 0)^T$;
- b) $f(\mathbf{x}) := x_1^2 + (x_2 + 1)^2$, $g_1(\mathbf{x}) := x_2 - x_1^2$, $g_2(\mathbf{x}) := -x_2$, $\mathbf{x}_0 := (0, 0)^T$.

Exercise 497. Let the following optimization problem be given:

$$\begin{aligned}\min_{\mathbf{x} \in \mathbb{R}^2} & f(\mathbf{x}) \\ g_1(x_1, x_2) &:= 3(x_1 - 1)^3 - 2x_2 + 2 \leq 0 \\ g_2(x_1, x_2) &:= (x_1 - 1)3 + 2x_2 - 2 \leq 0 \\ g_3(x_1, x_2) &:= -x_1 \leq 0 \\ g_4(x_1, x_2) &:= -x_2 \leq 0.\end{aligned}$$

- a) Plot the feasible region.

b) Solve the optimization problem for the following objective functions:

$$(i) f(x_1, x_2) := (x_1 - 1)^2 + (x_2 - 3/2)^2;$$

$$(ii) f(x_1, x_2) := (x_1 - 1)^2 + (x_2 - 4)^2;$$

(iii) Regard the objective function on the ‘upper boundary’ of \mathcal{F} . $f(x_1, x_2) := (x_1 - 5/4)^2 + (x_2 - 5/4)^2$. Do the KKT conditions hold at the optimal point?

Hint: In addition illustrate these problems graphically.

Exercise 498 (Optimal Location of a Rescue Helicopter). a) Formulate the minimax problem

$$d_{\max}(x, y) := \max_{1 \leq j \leq m} \sqrt{(x - x_j)^2 + (y - y_j)^2}$$

as a quadratic optimization problem

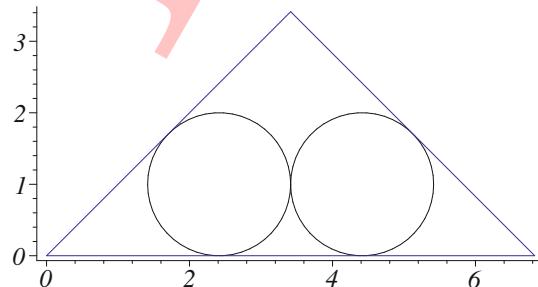
$$\begin{aligned} & \min_{x, y, \rho} f(x, y, \rho) \\ & g_j(x, y, \rho) \leq 0, (j = 1, \dots, m) \end{aligned}$$

(with f quadratic, g_j linear).

b) Visualize the function d_{\max} by plotting its contour lines for the points $(0, 0)$, $(5, -1)$, $(4, 6)$, $(1, 3)$.

c) Give the corresponding Lagrangian. Solve the problem by means of the KKT conditions.

Exercise 499. Determine a triangle with minimal area containing two disjoint disks with radius 1. Wlog let $(0, 0)$, $(x_1, 0)$ and (x_2, x_3) with $x_1, x_3 \geq 0$ be the vertices of the triangle; (x_4, x_5) and (x_6, x_7) denote the centers of the disks.



a) Formulate this problem as a minimization problem in terms of seven variables and nine constraints.

- b) $\mathbf{x}^* = (4 + 2\sqrt{2}, 2 + \sqrt{2}, 2 + \sqrt{2}, 1 + \sqrt{2}, 1, 3 + \sqrt{2}, 1)^T$ is a solution of this problem; calculate the corresponding Lagrange multipliers $\boldsymbol{\lambda}^*$, such that the KKT conditions are fulfilled.
- c) Check the sufficient second-order optimality conditions for $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$.

Exercise 500. Find the point $\mathbf{x} \in \mathbb{R}^2$ that lies closest to the point $\mathbf{p} := (2, 3)^T$ under the constraints $g_1(\mathbf{x}) := x_1 + x_2 \leq 0$ and $g_2(\mathbf{x}) := x_1^2 - 4 \leq 0$.

a) Illustrate the problem graphically.

b) Verify that the problem is convex and fulfills (SCQ).

c) Determine the KKT points by differentiating between three cases: none is active, exactly the first one is active, exactly the second one is active.

d) Now conclude with Theorem 483.

Exercise 501. In a small power network the power r runs through two different channels. Let x_i be the power running through channel i for $i = 1, 2$. The total loss is given by the function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ with

$$f(x_1, x_2) := x_1 + \frac{1}{2} (x_1^2 + x_2^2).$$

Determine the current flow such that the total loss stays minimal. The constraints are given by

$$x_1 + x_2 = r, x_1 \geq 0, x_2 \geq 0.$$

Exercise 502. Consider the optimization problem:

$$\begin{cases} \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) := \sum_{i=1}^n x_i \log(x_i/p_i), \\ \mathbf{A}^T \mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0 \end{cases}$$

where $\mathbf{A} \in \mathbb{R}^{n \times m}$, $\mathbf{b} \in \mathbb{R}^m$ and $p_1, p_2, \dots, p_n \in \mathbb{R}_{++}$ are given. Let further $0 \ln 0$ be defined as 0. Prove:

a) The dual problem is given by

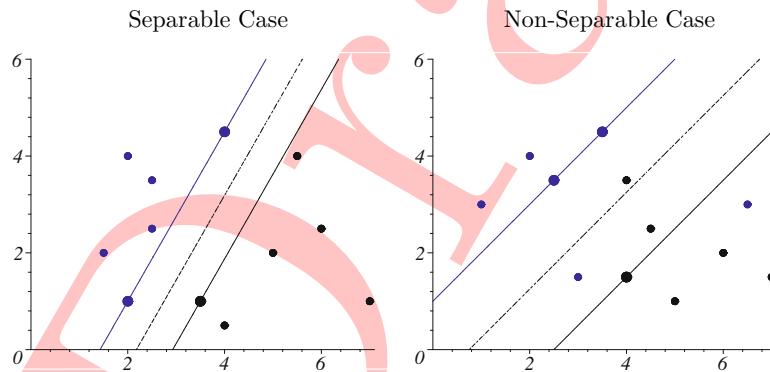
$$\max_{\boldsymbol{\lambda} \in \mathbb{R}^m} \varphi(\boldsymbol{\lambda}) := \mathbf{b}^T \boldsymbol{\lambda} - \sum_{i=1}^n p_i \exp(\mathbf{e}_i^T \mathbf{A} \boldsymbol{\lambda} - 1).$$

b) $\nabla \varphi(\boldsymbol{\lambda}) = \mathbf{b} - \mathbf{A}^T \mathbf{x}$ with $x_i = p_i \exp(\mathbf{e}_i^T \mathbf{A} \boldsymbol{\lambda} - 1)$.

c) $\nabla^2 \varphi(\boldsymbol{\lambda}) = -\mathbf{A}^T \mathbf{X} \mathbf{A}$, where $\mathbf{X} = \text{diag}(\mathbf{x})$ with \mathbf{x} from b).

Exercise 503 (Support Vector Machines). *Support vector machines have been extensively used in machine learning and data mining applications such as classification and regression, text categorization as well as medical applications, for example breast cancer diagnosis. Let two classes of patterns be given, i.e., samples of observable characteristics which are represented by points $\mathbf{x}_i \in \mathbb{R}^n$. The patterns are given in the form (\mathbf{x}_i, y_i) , $i = 1, \dots, m$, with $y_i \in \{1, -1\}$. $y_i = 1$ means that \mathbf{x}_i belongs to class 1; otherwise \mathbf{x}_i belongs to class 2. In the simplest case we are looking for a separating hyperplane described by $\langle \mathbf{w}, \mathbf{x} \rangle + \beta = 0$ with $\langle \mathbf{w}, \mathbf{x}_i \rangle + \beta \geq 1$ if $y_i = 1$ and $\langle \mathbf{w}, \mathbf{x}_i \rangle + \beta \leq -1$ if $y_i = -1$. These conditions can be written as $y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + \beta) \geq 1$ ($i = 1, \dots, m$). We aim to maximize the ‘margin’ (distance) $2/\|\mathbf{w}\|^2$ between the two hyperplanes $\langle \mathbf{w}, \mathbf{x} \rangle + \beta = 1$ and $\langle \mathbf{w}, \mathbf{x} \rangle + \beta = -1$. This gives a linearly constrained convex quadratic minimization problem*

$$\begin{aligned} & \min_{\mathbf{w}} \frac{1}{2} \|\mathbf{w}\|^2, \\ & \text{s.t. } y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + \beta) \geq 1, (i = 1, \dots, m). \end{aligned}$$



In the case that the two classes are not linearly separable (by a hyperplane), we introduce nonnegative penalties ξ_i for the ‘misclassification’ of \mathbf{x}_i and minimize both $\|\mathbf{w}\|^2$ and $\sum_{i=1}^m \xi_i$. We solve this optimization problem in the following way with soft margins

$$(P) \left\{ \begin{array}{l} \min_{\mathbf{w}, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^m \xi_i, \\ y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + \beta) \geq 1 - \xi_i, (i = 1, \dots, m). \end{array} \right.$$

Here, C is a weight parameter of the penalty term.

a) Introducing the dual variables $\boldsymbol{\lambda} \in \mathbb{R}_+^m$, derive the Lagrange dual problem to (P):

$$(D) \left\{ \begin{array}{l} \max_{\boldsymbol{\lambda}} -\frac{1}{2} \sum_{i,j=1}^m y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \lambda_i \lambda_j + \sum_{i=1}^m \lambda_i \\ \sum_{i=1}^m y_i \lambda_i = 0, 0 \leq \lambda_i \leq C, (i = 1, \dots, m). \end{array} \right.$$

Compute the coefficients $\mathbf{w} \in \mathbb{R}^n$ and $\beta \in \mathbb{R}$ of the separating hyperplane by means of the dual solution $\boldsymbol{\lambda}$ and show

$$\mathbf{w} = \sum_{j=1}^m y_j \lambda_j \mathbf{x}_j, \beta = y_j - \langle \mathbf{w}, \mathbf{x}_j \rangle, \text{ if } 0 < \lambda_j < C.$$

Vectors \mathbf{x}_j with $\lambda_j > 0$ are called support vectors.

- b) Calculate a support vector ‘machine’ for breast cancer diagnosis using the file `wisconsin-breast-cancer.data` from the Breast Cancer Wisconsin Data Set (<http://archive.ics.uci.edu/ml/>). The file `wisconsinbreast-cancer.names` gives information on the data set: It contains 699 instances consisting of 11 attributes. The first attribute gives the sample code number. Attributes 2 through 10 describe the medical status and give a 9-dimensional vector \mathbf{x}_i . The last attribute is the class attribute (“2” for benign, “4” for malignant). Sixteen samples have a missing attribute, denoted by “?”. Remove these samples from the data set. Now split the data into two portions: The first 120 instances are used as training data. Take software of your choice to solve the quadratic problem (P), using the penalty parameter $C = 1000$. The remaining instances are used to evaluate the ‘performance’ of the classifier or decision function given by $f(\mathbf{x}) := \text{sgn}\{\langle \mathbf{w}, \mathbf{x} \rangle + \beta\}$.

6.6 Duality

(Taken from Chapter 5 of [18])

6.6.1 The Lagrange dual function

6.6.1.1 The Lagrangian

We consider an optimization problem in the standard form:

$$\begin{aligned} & \text{minimize} && f_0(x) \\ & \text{subject to} && f_i(x) \leq 0, \quad i = 1, \dots, m \\ & && h_i(x) = 0, \quad i = 1, \dots, p, \end{aligned} \tag{6.12}$$

with variable $x \in \mathbb{R}^n$. We assume its domain $\mathcal{D} = \bigcap_{i=0}^m \text{dom } f_i \cap \bigcap_{i=1}^p \text{dom } h_i$ is nonempty, and denote the optimal value of (6.12) by p^* . We do not assume the problem (6.12) is convex.

The basic idea in Lagrangian duality is to take the constraints in (6.12) into account by augmenting the objective function with a weighted sum of the constraint functions.

We define the *Lagrangian* $L : \mathbb{R}^n \times \mathbb{R}_+^m \times \mathbb{R}^p \rightarrow \mathbb{R}$ associated with the problem (6.12) as

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x),$$

with $\text{dom } L = \mathcal{D} \times \mathbb{R}_+^m \times \mathbb{R}^p$. We refer to λ_i as the *Lagrange multiplier* associated with the i th inequality constraint $f_i(x) \leq 0$; similarly we refer to ν_i as the *Lagrange multiplier* associated with the i th equality constraint $h_i(x) = 0$. The vectors λ and ν are called the *dual variables or Lagrange multiplier vectors* associated with the problem (6.12).

6.6.1.2 The Lagrange dual function

We define the *Lagrange dual function* (or just *dual function*) $g : \mathbb{R}_+^m \times \mathbb{R}^p \rightarrow \mathbb{R}$ as the minimum value of the Lagrangian over x : for $\lambda \in \mathbb{R}_+^m$, $\nu \in \mathbb{R}^p$,

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x) \right).$$

When the Lagrangian is unbounded below in x , the dual function takes on the value $-\infty$. Since the dual function is the pointwise infimum of a family of affine functions of (λ, ν) , it is concave, even when the problem (6.12) is not convex.

6.6.1.3 Lower bounds on optimal value

The dual function yields lower bounds on the optimal value p^* of the problem (6.12): For any $\lambda \geq 0$ and any ν we have

$$g(\lambda, \nu) \leq p^*. \quad (6.13)$$

This important property is easily verified. Suppose \tilde{x} is a feasible point for the problem (6.12), i.e., $f_i(\tilde{x}) \leq 0$ and $h_i(\tilde{x}) = 0$, and $\lambda \geq 0$. Then we have

$$\sum_{i=1}^m \lambda_i f_i(\tilde{x}) + \sum_{i=1}^p \nu_i h_i(\tilde{x}) \leq 0,$$

since each term in the first sum is nonpositive, and each term in the second sum is zero, and therefore

$$L(\tilde{x}, \lambda, \nu) = f_0(\tilde{x}) + \sum_{i=1}^m \lambda_i f_i(\tilde{x}) + \sum_{i=1}^p \nu_i h_i(\tilde{x}) \leq f_0(\tilde{x}).$$

Hence

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) \leq L(\tilde{x}, \lambda, \nu) \leq f_0(\tilde{x}).$$

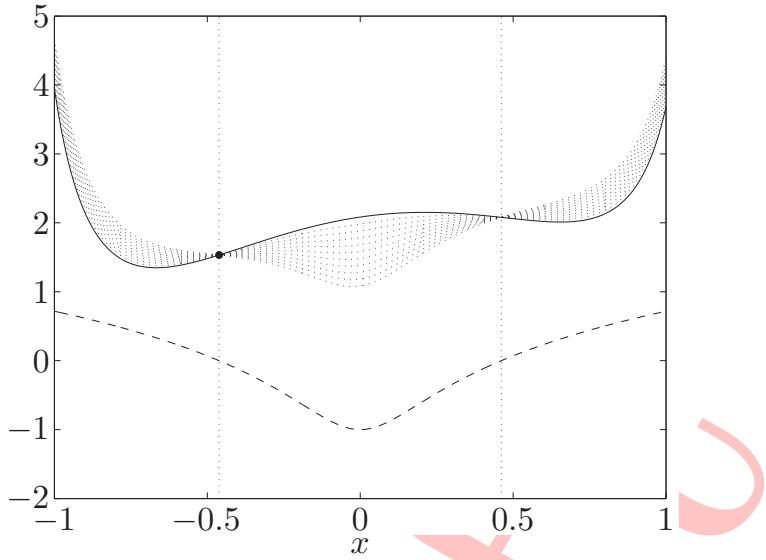


图 6.4: *Lower bound from a dual feasible point.* The solid curve shows the objective function f_0 , and the dashed curve shows the constraint function f_1 . The feasible set is the interval $[-0.46, 0.46]$, which is indicated by the two dotted vertical lines. The optimal point and value are $x^* = 0.46$, $p^* = 1.54$ (shown as a circle). The dotted curves show $L(x, \lambda)$ for $\lambda = 0.1, 0.2, \dots, 1.0$. Each of these has a minimum value smaller than p^* , since on the feasible set (and for $\lambda \geq 0$) we have $L(x, \lambda) \leq f_0(x)$.

Since $g(\lambda, \nu) \leq f_0(\tilde{x})$ holds for every feasible point \tilde{x} , the inequality (6.13) follows. The lower bound (6.13) is illustrated in Figure 6.4, for a simple problem with $x \in \mathbb{R}$ and one inequality constraint.

The inequality (6.13) holds, but is vacuous, when $g(\lambda, \nu) = -\infty$. The dual function gives a nontrivial lower bound on p^* only when $\lambda \geq 0$ and $(\lambda, \nu) \in \text{dom } g$, i.e., $g(\lambda, \nu) > -\infty$. We refer to a pair (λ, ν) with $\lambda \geq 0$ and $(\lambda, \nu) \in \text{dom } g$ as *dual feasible*, for reasons that will become clear later.

6.6.1.4 Linear approximation interpretation

The Lagrangian and lower bound property can be given a simple interpretation, based on a linear approximation of the indicator functions of the sets $\{0\}$ and $-\mathbb{R}_+$.

We first rewrite the original problem (6.12) as an unconstrained problem,

$$\underset{x}{\text{minimize}} \quad f_0(x) + \sum_{i=1}^m I_-(f_i(x)) + \sum_{i=1}^p I_0(h_i(x)), \quad (6.14)$$

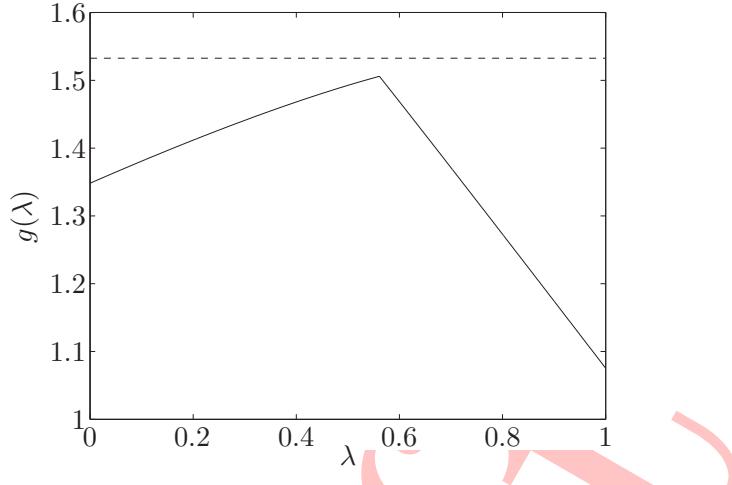


图 6.5: The dual function g for the problem in Figure 6.4. Neither f_0 nor f_1 is convex, but the dual function is concave. The horizontal dashed line shows p^* , the optimal value of the problem.

where $I_- : \mathbb{R} \rightarrow \mathbb{R}$ is the indicator function for the nonpositive reals,

$$I_-(u) = \begin{cases} 0 & u \leq 0 \\ \infty & u > 0, \end{cases}$$

and similarly, I_0 is the indicator function of $\{0\}$. In the formulation (6.14), the function $I_-(u)$ can be interpreted as expressing our irritation or displeasure associated with a constraint function value $u = f_i(x)$: It is zero if $f_i(x) \leq 0$, and infinite if $f_i(x) > 0$. In a similar way, $I_0(u)$ gives our displeasure for an equality constraint value $u = g_i(x)$. We can think of I_- as a “brick wall” or “infinitely hard” displeasure function; our displeasure rises from zero to infinite as $f_i(x)$ transitions from nonpositive to positive.

Now suppose in the formulation (6.14) we replace the function $I_-(u)$ with the linear function $\lambda_i u$, where $\lambda_i \geq 0$, and the function $I_0(u)$ with $\nu_i u$. The objective becomes the Lagrangian function $L(x, \lambda, \nu)$, and the dual function value $g(\lambda, \nu)$ is the optimal value of the problem

$$\text{minimize } L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x). \quad (6.15)$$

In this formulation, we use a linear or “soft” displeasure function in place of I_- and I_0 . For an inequality constraint, our displeasure is zero when $f_i(x) = 0$, and is positive when $f_i(x) > 0$ (assuming $\lambda_i > 0$); our displeasure grows as the constraint becomes “more violated”. Unlike the original formulation, in which any nonpositive value of $f_i(x)$ is

acceptable, in the soft formulation we actually derive pleasure from constraints that have margin, *i.e.*, from $f_i(x) < 0$.

Clearly the approximation of the indicator function $I_-(u)$ with a linear function $\lambda_i u$ is rather poor. But the linear function is at least an *underestimator* of the indicator function. Since $\lambda_i u \leq I_-(u)$ and $\nu_i u \leq I_0(u)$ for all u , we see immediately that the dual function yields a lower bound on the optimal value of the original problem.

The idea of replacing the “hard” constraints with “soft” versions will come up again when we consider interior-point methods (see Section 11.2.1 of [18]).

6.6.1.5 Examples

In this section we give some examples for which we can derive an analytical expression for the Lagrange dual function.

Least-squares solution of linear equations

We consider the problem

$$\begin{aligned} & \text{minimize} && x^T x \\ & \text{subject to} && Ax = b, \end{aligned} \tag{6.16}$$

where $A \in \mathbb{R}^{p \times n}$. This problem has no inequality constraints and p (linear) equality constraints. The Lagrangian is $L(x, \nu) = x^T x + \nu^T (Ax - b)$, with domain $\mathbb{R}^n \times \mathbb{R}^p$. The dual function is given by $g(\nu) = \inf_x L(x, \nu)$. Since $L(x, \nu)$ is a convex quadratic function of x , we can find the minimizing x from the optimality condition

$$\nabla_x L(x, \nu) = 2x + A^T \nu = 0,$$

which yields $x = -(1/2)A^T \nu$. Therefore the dual function is

$$g(\nu) = L\left(-(1/2)A^T \nu, \nu\right) = -(1/4)\nu^T A A^T \nu - b^T \nu,$$

which is a concave quadratic function, with domain \mathbb{R}^p . The lower bound property (6.13) states that for any $\nu \in \mathbb{R}^p$, we have

$$-(1/4)\nu^T A A^T \nu - b^T \nu \leq \inf\{x^T x \mid Ax = b\}.$$

Standard form LP

Consider an LP in standard form,

$$\begin{aligned} & \text{minimize} && c^T x \\ & \text{subject to} && Ax = b \\ & && x \geq 0, \end{aligned} \tag{6.17}$$

which has inequality constraint functions $f_i(x) = -x_i$, $i = 1, \dots, n$. To form the Lagrangian we introduce multipliers λ_i for the n inequality constraints and multipliers ν_i for the equality constraints, and obtain

$$L(x, \lambda, \nu) = c^T x - \sum_{i=1}^n \lambda_i x_i + \nu^T (Ax - b) = -b^T \nu + (c + A^T \nu - \lambda)^T x.$$

The dual function is

$$g(\lambda, \nu) = \inf_x L(x, \lambda, \nu) = -b^T \nu + \inf_x (c + A^T \nu - \lambda)^T x,$$

which is easily determined analytically, since a linear function is bounded below only when it is identically zero. Thus, $g(\lambda, \nu) = -\infty$ except when $c + A^T \nu - \lambda = 0$, in which case it is $-b^T \nu$:

$$g(\lambda, \nu) = \begin{cases} -b^T \nu & A^T \nu - \lambda + c = 0 \\ -\infty & \text{otherwise.} \end{cases}$$

Note that the dual function g is finite only on a proper affine subset of $\mathbb{R}^m \times \mathbb{R}^p$. We will see that this is a common occurrence.

The lower bound property (6.13) is nontrivial only when λ and ν satisfy $\lambda \geq 0$ and $A^T \nu - \lambda + c = 0$. When this occurs, $-b^T \nu$ is a lower bound on the optimal value of the LP (6.17).

Two-way partitioning problem

We consider the (nonconvex) problem

$$\begin{aligned} & \text{minimize} && x^T W x \\ & \text{subject to} && x_i^2 = 1, \quad i = 1, \dots, n, \end{aligned} \tag{6.18}$$

where $W \in \mathbf{S}^n$. The constraints restrict the values of x_i to 1 or -1 , so the problem is equivalent to finding the vector with components ± 1 that minimizes $x^T W x$. The feasible set here is finite (it contains 2^n points) so this problem can in principle be solved by simply checking the objective value of each feasible point. Since the number of feasible points grows exponentially, however, this is possible only for small problems (say, with $n \leq 30$). In general (and for n larger than, say, 50) the problem (6.18) is very difficult to solve.

We can interpret the problem (6.18) as a two-way partitioning problem on a set of n elements, say, $\{1, \dots, n\}$: A feasible x corresponds to the partition

$$\{1, \dots, n\} = \{i \mid x_i = -1\} \cup \{i \mid x_i = 1\}.$$

The matrix coefficient W_{ij} can be interpreted as the cost of having the elements i and j in the same partition, and $-W_{ij}$ is the cost of having i and j in different partitions. The objective in (6.18) is the total cost, over all pairs of elements, and the problem (6.18) is to find the partition with least total cost.

We now derive the dual function for this problem. The Lagrangian is

$$\begin{aligned} L(x, \nu) &= x^T W x + \sum_{i=1}^n \nu_i (x_i^2 - 1) \\ &= x^T (W + \text{diag}(\nu)) x - \mathbf{1}^T \nu. \end{aligned}$$

We obtain the Lagrange dual function by minimizing over x :

$$\begin{aligned} g(\nu) &= \inf_x x^T (W + \text{diag}(\nu)) x - \mathbf{1}^T \nu. \\ &= \begin{cases} -\mathbf{1}^T \nu & W + \text{diag}(\nu) \geq 0, \\ -\infty & \text{otherwise,} \end{cases} \end{aligned}$$

where we use the fact that the infimum of a quadratic form is either zero (if the form is positive semidefinite) or $-\infty$ (if the form is not positive semidefinite).

This dual function provides lower bounds on the optimal value of the difficult problem (6.18). For example, we can take the specific value of the dual variable

$$\nu = -\lambda_{\min}(W)\mathbf{1},$$

which is dual feasible, since

$$W + \text{diag}(\nu) = W - \lambda_{\min}(W)I \succeq 0.$$

This yields the bound on the optimal value p^*

$$p^* \geq -\mathbf{1}^T \nu = n\lambda_{\min}(W). \quad (6.19)$$

Remark 504. This lower bound on p^* can also be obtained without using the Lagrange dual function. First, we replace the constraints $x_1^2 = 1, \dots, x_n^2 = 1$ with $\sum_{i=1}^n x_i^2 = n$, to obtain the modified problem

$$\begin{aligned} &\text{minimize} && x^T W x \\ &\text{subject to} && \sum_{i=1}^n x_i^2 = n, \end{aligned} \quad (6.20)$$

The constraints of the original problem (6.18) imply the constraint here, so the optimal value of the problem (6.20) is a lower bound on p^* , the optimal value of (6.18). But the modified problem (6.20) is easily solved as an eigenvalue problem, with optimal value $n\lambda_{\min}(W)$.

6.6.1.6 The Lagrange dual function and conjugate functions

Recall from §3.3 that the conjugate f^* of a function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ is given by

$$f^*(y) = \sup_{x \in \text{dom } f} (y^T x - f(x)).$$

The conjugate function and Lagrange dual function are closely related. To see one simple connection, consider the problem

$$\begin{aligned} & \text{minimize} && f(x) \\ & \text{subject to} && x = 0 \end{aligned}$$

(which is not very interesting, and solvable by inspection). This problem has Lagrangian $L(x, \nu) = f(x) + \nu^T x$, and dual function

$$g(\nu) = \inf_x (f(x) + \nu^T x) = -\sup_x ((-\nu)^T x - f(x)) = -f^*(-\nu).$$

More generally (and more usefully), consider an optimization problem with linear inequality and equality constraints,

$$\begin{aligned} & \text{minimize} && f_0(x) \\ & \text{subject to} && Ax \leq b \\ & && Cx = d. \end{aligned} \tag{6.21}$$

Using the conjugate of f_0 we can write the dual function for the problem (6.21) as

$$\begin{aligned} g(\lambda, \nu) &= \inf_x (f_0(x) + \lambda^T(Ax - b) + \nu^T(Cx - d)) \\ &= -b^T \lambda - d^T \nu + \inf_x (f_0(x) + (A^T \lambda + C^T \nu)^T x) \\ &= -b^T \lambda - d^T \nu - f_0^*(-A^T \lambda - C^T \nu). \end{aligned} \tag{6.22}$$

The domain of g follows from the domain of f_0^* :

$$\text{dom } g = \{(\lambda, \nu) \mid \lambda \geq 0, -A^T \lambda - C^T \nu \in \text{dom } f_0^*\}.$$

Let us illustrate this with a few examples.

Equality constrained norm minimization

Consider the problem

$$\begin{aligned} & \text{minimize} && \|x\| \\ & \text{subject to} && Ax = b, \end{aligned} \tag{6.23}$$

where $\|\cdot\|$ is any norm. Recall (from Example 224) that the conjugate of $f_0 = \|\cdot\|$ is given by

$$f_0^*(y) = \begin{cases} 0 & \|y\|_* \leq 1 \\ \infty & \text{otherwise,} \end{cases}$$

the indicator function of the dual norm unit ball.

Using the result (6.22) above, the dual function for the problem (6.23) is given by

$$g(\nu) = -b^T \nu - f_0^*(-A^T \nu) = \begin{cases} -b^T \nu & \|A^T \nu\|_* \leq 1 \\ -\infty & \text{otherwise.} \end{cases}$$

Entropy maximization

Consider the entropy maximization problem

$$\begin{aligned} & \text{minimize} \quad f_0(x) = \sum_{i=1}^n x_i \log x_i \\ & \text{subject to} \quad Ax \leq b \\ & \quad \quad \quad \mathbf{1}^T x = 1 \end{aligned} \tag{6.24}$$

where $\text{dom } f_0 = \mathbb{R}_{++}^n$. The conjugate of the negative entropy function $u \log u$, with scalar variable u , is e^{v-1} (see Example 219). Since f_0 is a sum of negative entropy functions of different variables, we conclude that its conjugate is

$$f_0^*(y) = \sum_{i=1}^n e^{y_i - 1},$$

with $\text{dom } f_0^* = \mathbb{R}^n$. Using the result (6.22) above, the dual function of (6.24) is given by

$$g(\lambda, \nu) = -b^T \lambda - \nu - \sum_{i=1}^n e^{-a_i^T \lambda - \nu - 1} = -b^T \lambda - \nu - e^{-\nu - 1} \sum_{i=1}^n e^{-a_i^T \lambda}$$

where a_i is the i th column of A .

Minimum volume covering ellipsoid

Consider the problem with variable $X \in \mathbf{S}^n$

$$\begin{aligned} & \text{minimize} \quad f_0(X) = \log \det X^{-1} \\ & \text{subject to} \quad a_i^T X a_i \leq 1, \quad i = 1, \dots, m, \end{aligned} \tag{6.25}$$

where $\text{dom } f_0 = \mathbf{S}_{++}^n$. The problem (6.25) has a simple geometric interpretation. With each $X \in \mathbf{S}_{++}^n$ we associate the ellipsoid, centered at the origin,

$$\mathcal{E}_X = \{z \mid z^T X z \leq 1\}.$$

The volume of this ellipsoid is proportional to $(\det X^{-1})^{1/2}$, so the objective of (6.25) is, except for a constant and a factor of two, the logarithm of the volume of \mathcal{E}_X . The constraints of the problem (6.25) are that $a_i \in \mathcal{E}_X$. Thus the problem (6.25) is to determine the minimum volume ellipsoid, centered at the origin, that includes the points a_1, \dots, a_m .

The inequality constraints in problem (6.25) are affine; they can be expressed as

$$\text{tr}((a_i a_i^T) X) \leq 1.$$

In Example 221 we found that the conjugate of f_0 is

$$f_0^*(Y) = \log \det(-Y)^{-1} - n,$$

with $\text{dom } f_0^* = -\mathbf{S}_{++}^n$. Applying the result (6.22) above, the dual function for the problem (6.25) is given by

$$g(\lambda) = \begin{cases} \log \det \left(\sum_{i=1}^m \lambda_i a_i a_i^T \right) - \mathbf{1}^T \lambda + n & \sum_{i=1}^m \lambda_i a_i a_i^T \succ 0 \\ -\infty & \text{otherwise.} \end{cases} \quad (6.26)$$

Thus, for any $\lambda \geq 0$ with $\sum_{i=1}^m \lambda_i a_i a_i^T \succ 0$, the number

$$\log \det \left(\sum_{i=1}^m \lambda_i a_i a_i^T \right) - \mathbf{1}^T \lambda + n$$

is a lower bound on the optimal value of the problem (6.25).

6.6.2 The Lagrange Dual Problem

For each pair (λ, ν) with $\lambda \geq 0$, the Lagrange dual function gives us a lower bound on the optimal value p^* of the optimization problem (6.12). Thus we have a lower bound that depends on some parameters λ, ν . A natural question is: What is the *best* lower bound that can be obtained from the Lagrange dual function?

This leads to the optimization problem

$$\begin{aligned} \max_{\lambda, \nu} \quad & g(\lambda, \nu) \\ \text{subject to} \quad & \lambda \geq 0. \end{aligned} \quad (6.27)$$

This problem is called the *Lagrange dual problem* associated with the problem (6.12). In this context the original problem (6.12) is sometimes called the *primal problem*. The term *dual feasible*, to describe a pair (λ, ν) with $\lambda \geq 0$ and $g(\lambda, \nu) > -\infty$, now makes sense. It means, as the name implies, that (λ, ν) is feasible for the dual problem (6.27).

We refer to (λ^*, ν^*) as *dual optimal* or *optimal Lagrange multipliers* if they are optimal for the problem (6.27).

The Lagrange dual problem (6.27) is a convex optimization problem, since the objective to be maximized is concave and the constraint is convex. This is the case whether or not the primal problem (6.12) is convex.

6.6.2.1 Making dual constraints explicit

The examples above show that it is not uncommon for the domain of the dual function,

$$\text{dom } g = \{(\lambda, \nu) \mid \lambda \geq 0, g(\lambda, \nu) > -\infty\},$$

to have dimension smaller than $m + p$. In many cases we can identify the affine hull of $\text{dom } g$, and describe it as a set of linear equality constraints. Roughly speaking, this means we can identify the equality constraints that are ‘hidden’ or ‘implicit’ in the objective g of the dual problem (6.27). In this case we can form an equivalent problem, in which these equality constraints are given explicitly as constraints. The following examples demonstrate this idea.

Lagrange dual of standard form LP

On Section 6.6.1.5 we found that the Lagrange dual function for the standard form LP

$$\begin{aligned} & \text{minimize} && c^T x \\ & \text{subject to} && Ax = b \\ & && x \geq 0 \end{aligned} \tag{6.28}$$

is given by

$$g(\lambda, \nu) = \begin{cases} -b^T \nu & A^T \nu - \lambda + c = 0 \\ -\infty & \text{otherwise.} \end{cases}$$

Strictly speaking, the Lagrange dual problem of the standard form LP is to maximize this dual function g subject to $\lambda \geq 0$, i.e.,

$$\begin{aligned} \max_{\lambda, \nu} & \quad g(\lambda, \nu) = \begin{cases} -b^T \nu & A^T \nu - \lambda + c = 0 \\ -\infty & \text{otherwise} \end{cases} \\ & \text{subject to} \quad \lambda \geq 0. \end{aligned} \tag{6.29}$$

Here g is finite only when $A^T - \lambda + c = 0$. We can form an equivalent problem by making these equality constraints explicit:

$$\begin{aligned} & \text{maximize} && -b^T \nu \\ & \text{subject to} && A^T \nu - \lambda + c = 0 \\ & && \lambda \geq 0. \end{aligned} \tag{6.30}$$

This problem, in turn, can be expressed as

$$\begin{aligned} & \text{maximize} && -b^T \nu \\ & \text{subject to} && A^T \nu + c \geq 0, \end{aligned} \tag{6.31}$$

which is an LP in inequality form.

Note the subtle distinctions between these three problems. The Lagrange dual of the standard form LP (6.28) is the problem (6.29), which is equivalent to (but not the same as) the problems (6.30) and (6.31). With some abuse of terminology, we refer to the problem (6.30) or the problem (6.31) as the Lagrange dual of the standard form LP (6.28).

Lagrange dual of inequality form LP

In a similar way we can find the Lagrange dual problem of a linear program in inequality form

$$\begin{aligned} & \text{minimize} && c^T x \\ & \text{subject to} && Ax \leq b. \end{aligned} \tag{6.32}$$

The Lagrangian is

$$L(x, \lambda) = c^T x + \lambda^T (Ax - b) = -b^T \lambda + (A^T \lambda + c)^T x,$$

so the dual function is

$$g(\lambda) = \inf_x L(x, \lambda) = -b^T \lambda + \inf_x (A^T \lambda + c)^T x.$$

The infimum of a linear function is $-\infty$, except in the special case when it is identically zero, so the dual function is

$$g(\lambda) = \begin{cases} -b^T \lambda & A^T \lambda + c = 0 \\ -\infty & \text{otherwise.} \end{cases}$$

The dual variable λ is dual feasible if $\lambda \geq 0$ and $A^T \lambda + c = 0$.

The Lagrange dual of the LP (6.32) is to maximize g over all $\lambda \geq 0$. Again we can reformulate this by explicitly including the dual feasibility conditions as constraints, as

in

$$\begin{aligned} & \text{maximize}_{\nu} - b^T \nu \\ & \text{subject to } A^T \lambda + c = 0 \\ & \quad \lambda \geq 0, \end{aligned} \tag{6.33}$$

which is an LP in standard form.

Note the interesting symmetry between the standard and inequality form LPs and their duals: The dual of a standard form LP is an LP with only inequality constraints, and vice versa. One can also verify that the Lagrange dual of (6.33) is (equivalent to) the primal problem (6.32).

6.6.2.2 Weak duality

The optimal value of the Lagrange dual problem, which we denote d^* , is, by definition, the best lower bound on p^* that can be obtained from the Lagrange dual function. In particular, we have the simple but important inequality

$$d^* \leq p^*, \tag{6.34}$$

which holds even if the original problem is not convex. This property is called *weak duality*.

The weak duality inequality (6.34) holds when d^* and p^* are infinite. For example, if the primal problem is unbounded below, so that $p^* = -\infty$, we must have $d^* = -\infty$, i.e., the Lagrange dual problem is infeasible. Conversely, if the dual problem is unbounded above, so that $d^* = \infty$, we must have $p^* = \infty$, i.e., the primal problem is infeasible.

We refer to the difference $p^* - d^*$ as the *optimal duality gap* of the original problem, since it gives the gap between the optimal value of the primal problem and the best (i.e., greatest) lower bound on it that can be obtained from the Lagrange dual function. The optimal duality gap is always nonnegative.

The bound (6.34) can sometimes be used to find a lower bound on the optimal value of a problem that is difficult to solve, since the dual problem is always convex, and in many cases can be solved efficiently, to find d^* . As an example, consider the two-way partitioning problem (6.18) described in Section 6.6.1.5. The dual problem is an SDP,

$$\begin{aligned} & \max_{\nu} -\mathbf{1}^T \nu \\ & \text{s.t. } W + \text{diag}(\nu) \succeq 0, \end{aligned}$$

with variable $\nu \in \mathbb{R}^n$. This problem can be solved efficiently, even for relatively large values of n , such as $n = 1000$. Its optimal value is a lower bound on the optimal value

of the two-way partitioning problem, and is always at least as good as the lower bound (6.19) based on $\lambda_{\min}(W)$.

6.6.2.3 Strong duality and Slater's constraint qualification

If the equality

$$d^* = p^* \quad (6.35)$$

holds, *i.e.*, the optimal duality gap is zero, then we say that *strong duality* holds. This means that the best bound that can be obtained from the Lagrange dual function is tight.

Strong duality does not, in general, hold. But if the primal problem (6.12) is convex, *i.e.*, of the form

$$\begin{aligned} & \text{minimize} && f_0(x) \\ & \text{subject to} && f_i(x) \leq 0, \quad i = 1, \dots, m, \\ & && Ax = b, \end{aligned} \quad (6.36)$$

with f_0, \dots, f_m convex, we usually (but not always) have strong duality. There are many results that establish conditions on the problem, beyond convexity, under which strong duality holds. These conditions are called *constraint qualifications*.

One simple constraint qualification is *Slater's condition*: There exists an $x \in \text{relint } \mathcal{D}$ such that

$$f_i(x) < 0 \quad i = 1, \dots, m, \quad Ax = b. \quad (6.37)$$

Such a point is sometimes called *strictly feasible*, since the inequality constraints hold with strict inequalities. Slater's theorem states that strong duality holds, if Slater's condition holds (and the problem is convex).

Slater's condition can be refined when some of the inequality constraint functions f_i are affine. If the first k constraint functions f_1, \dots, f_k are affine, then strong duality holds provided the following weaker condition holds: There exists an $x \in \text{relint } \mathcal{D}$ with

$$f_i(x) \leq 0, \quad i = 1, \dots, k, \quad f_i(x) < 0, \quad i = k + 1, \dots, m, \quad Ax = b. \quad (6.38)$$

In other words, the affine inequalities do not need to hold with strict inequality. Note that the refined Slater condition (6.38) reduces to feasibility when the constraints are all linear equalities and inequalities, and **dom** f_0 is open.

Slater's condition (and the refinement (6.38)) not only implies strong duality for convex problems. It also implies that the dual optimal value is attained when $d^* > -\infty$, *i.e.*, there exists a dual feasible (λ^*, ν^*) with $g(\lambda^*, \nu^*) = d^* = p^*$. We will prove that strong duality obtains, when the primal problem is convex and Slater's condition holds, in Section 6.6.3.2.

6.6.2.4 Examples

Least-squares solution of linear equations

Recall the problem (6.16):

$$\begin{aligned} & \text{minimize} && x^T x \\ & \text{subject to} && Ax = b. \end{aligned}$$

The associated dual problem is

$$\max_{\mu} -\frac{1}{4}\nu^T A A^T \nu - b^T \nu,$$

which is an unconstrained concave quadratic maximization problem.

Slater's condition is simply that the primal problem is feasible, so $p^* = d^*$ provided $b \in \mathcal{R}(A)$, i.e., $p^* < \infty$. In fact for this problem we always have strong duality, even when $p^* = \infty$. This is the case when $b \notin \mathcal{R}(A)$, so there is a z with $A^T z = 0$, $b^T z \neq 0$. It follows that the dual function is unbounded above along the line $\{tz \mid t \in \mathbb{R}\}$, so $d^* = \infty$ as well.

Lagrange dual of LP

By the weaker form of Slater's condition, we find that strong duality holds for any LP (in standard or inequality form) provided the primal problem is feasible. Applying this result to the duals, we conclude that strong duality holds for LPs if the dual is feasible. This leaves only one possible situation in which strong duality for LPs can fail: both the primal and dual problems are infeasible. This pathological case can occur, see exercise 5.23 of [18].

Lagrange dual of QCQP

We consider the QCQP

$$\begin{aligned} & \text{minimize} && (1/2)x^T P_0 x + q_0^T x + r_0 \\ & \text{subject to} && (1/2)x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \dots, m, \end{aligned} \tag{6.39}$$

with $P_0 \in \mathbf{S}_{++}^n$ and $P_i \in \mathbf{S}_+^n$, $i = 1, \dots, m$. The Lagrangian is

$$L(x, \lambda) = (1/2)x^T P(\lambda)x + q(\lambda)^T x + r(x),$$

where

$$P(\lambda) = P_0 + \sum_{i=1}^m \lambda_i P_i, \quad q(\lambda) = q_0 + \sum_{i=1}^m \lambda_i q_i, \quad r(\lambda) = r_0 + \sum_{i=1}^m \lambda_i r_i.$$

It is possible to derive an expression for $g(\lambda)$ for general λ , but it is quite complicated. If $\lambda \geq 0$, however, we have $P(\lambda) \succ 0$ and

$$g(\lambda) = \inf_x L(x, \lambda) = -(1/2)q(\lambda)^T P(\lambda)^{-1} q(\lambda) + r(\lambda).$$

We can therefore express the dual problem as

$$\begin{aligned} \max_{\lambda} \quad & -(1/2)q(\lambda)^T P(\lambda)^{-1} q(\lambda) + r(\lambda) \\ \text{subject to} \quad & \lambda \geq 0. \end{aligned} \tag{6.40}$$

The Slater condition says that strong duality between (6.40) and (6.39) holds if the quadratic inequality constraints are strictly feasible, *i.e.*, there exists an x with

$$(1/2)x^T P_i x + 2q_i^T x + r_i < 0, \quad i = 1, \dots, m.$$

Entropy maximization

Our next example is the entropy maximization problem (6.24):

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^n x_i \log x_i \\ \text{subject to} \quad & Ax \leq b \\ & \mathbf{1}^T x = 1, \end{aligned}$$

with domain $\mathcal{D} = \mathbb{R}_+^n$. The Lagrange dual function was derived below (6.24) and the dual problem is

$$\begin{aligned} \max_{\lambda} \quad & -b^T \lambda - \nu - e^{-\nu-1} \sum_{i=1}^n e^{-a_i^T \lambda} \\ \text{subject to} \quad & \lambda \geq 0, \end{aligned} \tag{6.41}$$

with variables $\lambda \in \mathbb{R}^m$, $\nu \in \mathbb{R}$. The (weaker) Slater condition for (6.24) tells us that the optimal duality gap is zero if there exists an $x > 0$ with $Ax \leq b$ and $\mathbf{1}^T x = 1$.

We can simplify the dual problem (6.41) by maximizing over the dual variable ν analytically. For fixed λ , the objective function is maximized when the derivative with respect to ν is zero, *i.e.*,

$$\nu = \log \sum_{i=1}^n e^{-a_i^T \lambda} - 1.$$

Substituting this optimal value of ν into the dual problem gives

$$\begin{aligned} \max_{\lambda} \quad & -b^T \lambda - \log \left(\sum_{i=1}^n e^{-a_i^T \lambda} \right) \\ \text{subject to} \quad & \lambda \geq 0, \end{aligned}$$

which is a geometric program (in convex form) with nonnegativity constraints.

Minimum volume covering ellipsoid

We consider the problem (6.25):

$$\begin{aligned} & \text{minimize} && \log \det X^{-1} \\ & \text{subject to} && a_i^T X a_i \leq 1, \quad i = 1, \dots, m, \end{aligned}$$

with domain $\mathcal{D} = \mathbf{S}_{++}^n$. The Lagrange dual function is given by (6.26), so the dual problem can be expressed as

$$\begin{aligned} & \max_{\lambda} && \log \det \left(\sum_{i=1}^m \lambda_i a_i a_i^T \right) - \mathbf{1}^T \lambda + n \\ & \text{subject to} && \lambda \geq 0, \end{aligned} \tag{6.42}$$

where we take $\log \det X = -\infty$ if $X \not\succ 0$.

The (weaker) Slater condition for the problem (6.25) is that there exists an $X \in \mathbf{S}_{++}^n$ with $a_i^T X a_i \leq 1$, for $i = 1, \dots, m$. This is always satisfied, so strong duality always obtains between (6.25) and the dual problem (6.42).

A nonconvex quadratic problem with strong duality

On rare occasions strong duality obtains for a *nonconvex* problem. As an important example, we consider the problem of minimizing a nonconvex quadratic function over the unit ball,

$$\begin{aligned} & \text{minimize} && x^T A x + 2b^T x \\ & \text{subject to} && x^T x \leq 1, \end{aligned} \tag{6.43}$$

where $A \in \mathbf{S}^n$, $A \not\succeq 0$, and $b \in \mathbb{R}^n$. Since $A \not\succeq 0$, this is not a convex problem. This problem is sometimes called the *trust region problem*, and arises in minimizing a second-order approximation of a function over the unit ball, which is the region in which the approximation is assumed to be approximately valid.

The Lagrangian is

$$L(x, \lambda) = x^T A x + 2b^T x + \lambda(x^T x - 1) = x^T (A + \lambda I)x + 2b^T x - \lambda,$$

so the dual function is given by

$$g(\lambda) = \begin{cases} -b^T (A + \lambda I)^\dagger b - \lambda & A + \lambda I \succeq 0, \quad b \in \mathcal{R}(A + \lambda I) \\ -\infty & \text{otherwise,} \end{cases}$$

where $(A + \lambda I)^\dagger$ is the pseudo-inverse of $A + \lambda I$. The Lagrange dual problem is thus

$$\begin{aligned} & \max_{\lambda} && -b^T (A + \lambda I)^\dagger b - \lambda \\ & \text{subject to} && A + \lambda I \succeq 0, \quad b \in \mathcal{R}(A + \lambda I), \end{aligned} \tag{6.44}$$

with variable $\lambda \in \mathbb{R}$. Although it is not obvious from this expression, this is a convex optimization problem. In fact, it is readily solved since it can be expressed as

$$\begin{aligned} \max_{\lambda} \quad & - \sum_{i=1}^n (q_i^T b)^2 / (\lambda_i + \lambda) - \lambda \\ \text{subject to} \quad & \lambda \geq -\lambda_{\min}(A), \end{aligned}$$

where λ_i and q_i are the eigenvalues and corresponding (orthonormal) eigenvectors of A , and we interpret $(q_i^T b)^2/0$ as 0 if $q_i^T b = 0$ and as ∞ otherwise.

Despite the fact that the original problem (6.43) is not convex, we always have zero optimal duality gap for this problem: The optimal values of (6.43) and (6.44) are always the same. In fact, a more general result holds: strong duality holds for any optimization problem with quadratic objective and one quadratic inequality constraint, provided Slater's condition holds, see §B.1 of [18].

6.6.3 Geometric interpretation

6.6.3.1 Weak and strong duality via set of values

We can give a simple geometric interpretation of the dual function in terms of the set

$$\mathcal{G} = \{(f_1(x), \dots, f_m(x), h_1(x), \dots, h_p(x), f_0(x)) \in \mathbb{R}^m \times \mathbb{R}^p \times \mathbb{R} \mid x \in \mathcal{D}\}, \quad (6.45)$$

which is the set of values taken on by the constraint and objective functions. The optimal value p^* of (6.12) is easily expressed in terms of \mathcal{G} as

$$p^* = \inf\{t \mid (u, v, t) \in \mathcal{G}, u \preceq 0, v = 0\}.$$

To evaluate the dual function at (λ, ν) , we minimize the affine function

$$(\lambda, \mu, 1)^T(u, v, t) = \sum_{i=1}^m \lambda_i u_i + \sum_{i=1}^p \nu_i v_i + t$$

over $(u, v, t) \in \mathcal{G}$, i.e., we have

$$g(\lambda, \nu) = \inf\{(\lambda, \nu, 1)^T(u, v, t) \mid (u, v, t) \in \mathcal{G}\}.$$

In particular, we see that if the infimum is finite, then the inequality

$$(\lambda, \nu, 1)^T(u, v, t) \geq g(\lambda, \nu)$$

defines a supporting hyperplane to \mathcal{G} . This is sometimes referred to as a *nonvertical* supporting hyperplane, because the last component of the normal vector is nonzero.

Now suppose $\lambda \geq 0$. Then, obviously, $t \geq (\lambda, \nu, 1)^T(u, v, t)$ if $u \preceq 0$ and $v = 0$. Therefore

$$\begin{aligned} p^* &= \inf\{t \mid (u, v, t) \in \mathcal{G}, u \preceq 0, v = 0\} \\ &\geq \inf\{(\lambda, \nu, 1)^T(u, v, t) \mid (u, v, t) \in \mathcal{G}, u \preceq 0, v = 0\} \\ &\geq \inf\{(\lambda, \nu, 1)^T(u, v, t) \mid (u, v, t) \in \mathcal{G}\} \\ &= g(\lambda, \nu), \end{aligned}$$

i.e., we have weak duality. This interpretation is illustrated in Figures 6.6 and 6.7, for a simple problem with one inequality constraint.

Epigraph variation

In this section we describe a variation on the geometric interpretation of duality in terms of \mathcal{G} , which explains why strong duality obtains for (most) convex problems. We define the set $\mathcal{A} \subseteq \mathbb{R}^m \times \mathbb{R}^p \times \mathbb{R}$ as

$$\mathcal{A} = \mathcal{G} + (\mathbb{R}_+^m \times \{0\} \times \mathbb{R}_+), \quad (6.46)$$

or, more explicitly,

$$\begin{aligned} \mathcal{A} &= \{(u, v, t) \mid \exists x \in \mathcal{D}, f_i(x) \leq u_i, i = 1, \dots, m, \\ &\quad h_i(x) = v_i, i = 1, \dots, p, f_0(x) \leq t\}, \end{aligned}$$

We can think of \mathcal{A} as a sort of epigraph form of \mathcal{G} , since \mathcal{A} includes all the points in \mathcal{G} , as well as points that are ‘worse’, i.e., those with larger objective or inequality constraint function values.

We can express the optimal value in terms of \mathcal{A} as

$$p^* = \inf\{t \mid (0, 0, t) \in \mathcal{A}\}.$$

To evaluate the dual function at a point (λ, ν) with $\lambda \geq 0$, we can minimize the affine function $(\lambda, \nu, 1)^T(u, v, t)$ over \mathcal{A} : If $\lambda \geq 0$, then

$$g(\lambda, \nu) = \inf\{(\lambda, \nu, 1)^T(u, v, t) \mid (u, v, t) \in \mathcal{A}\}.$$

If the infimum is finite, then

$$(\lambda, \nu, 1)^T(u, v, t) \geq g(\lambda, \nu)$$

defines a nonvertical supporting hyperplane to \mathcal{A} .

In particular, since $(0, 0, p^*) \in \text{bd}\mathcal{A}$, we have

$$p^* = (\lambda, \nu, 1)^T(0, 0, p^*) \geq g(\lambda, \nu), \quad (6.47)$$

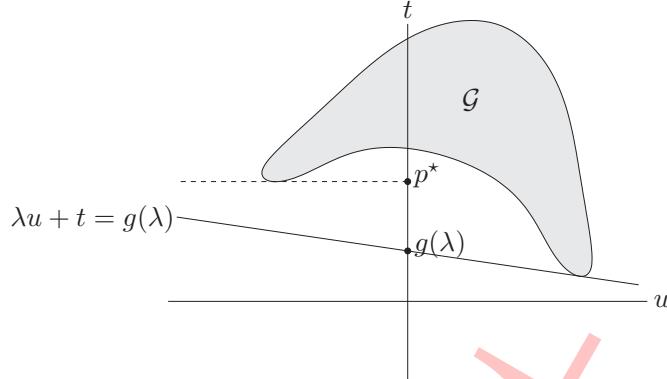


图 6.6: Geometric interpretation of dual function and lower bound $g(\lambda) \leq p^*$, for a problem with one (inequality) constraint. Given λ , we minimize $(\lambda, 1)^T(u, t)$ over $\mathcal{G} = \{(f_1(x), f_0(x)) \mid x \in \mathcal{D}\}$. This yields a supporting hyperplane with slope $-\lambda$. The intersection of this hyperplane with the $u = 0$ axis $g(\lambda)$.

the weak duality lower bound. Strong duality holds if and only if we have equality in (6.47) for some dual feasible (λ, ν) , i.e., there exists a nonvertical supporting hyperplane to \mathcal{A} at its boundary point $(0, 0, p^*)$.

This second interpretation is illustrated in Figure 6.8.

6.6.3.2 Proof of strong duality under constraint qualification

In this section we prove that Slater's constraint qualification guarantees strong duality (and that the dual optimum is attained) for a convex problem. We consider the primal problem (6.36), with f_0, \dots, f_m convex, and assume Slater's condition holds: There exists $\tilde{x} \in \text{relint } \mathcal{D}$ with $f_i(\tilde{x}) < 0$, $i = 1, \dots, m$, and $A\tilde{x} = b$. In order to simplify the proof, we make two additional assumptions: first that \mathcal{D} has nonempty interior (hence, $\text{relint } \mathcal{D} = \text{int } \mathcal{D}$) and second, that $\text{rank } A = p$. We assume that p^* is finite. (Since there is a feasible point, we can only have $p^* = -\infty$ or p^* finite; if $p^* = -\infty$, then $d^* = \infty$ by weak duality.)

The set \mathcal{A} defined in (6.46) is readily shown to be convex if the underlying problem is convex. We define a second convex set \mathcal{B} as

$$\mathcal{B} = \{(0, 0, s) \in \mathbb{R}^m \times \mathbb{R}^p \times \mathbb{R} \mid s < p^*\}.$$

The sets \mathcal{A} and \mathcal{B} do not intersect. To see this, suppose $(u, v, t) \in \mathcal{A} \cap \mathcal{B}$. Since $(u, v, t) \in \mathcal{B}$ we have $u = 0$, $v = 0$, and $t < p^*$. Since $(u, v, t) \in \mathcal{A}$ there exists an x with $f_i(x) \leq 0$, $i =$

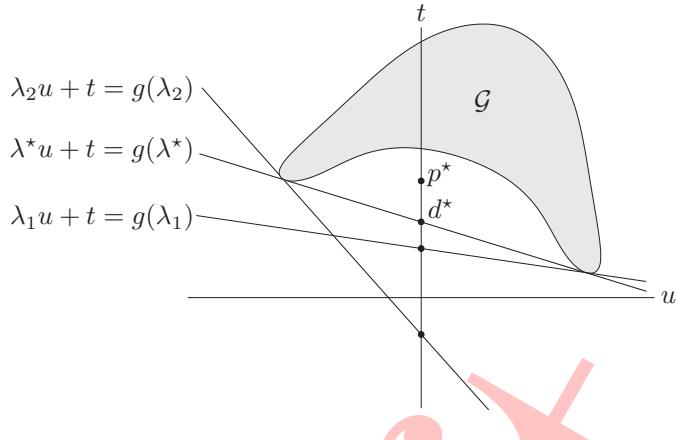


图 6.7: Supporting hyperplanes corresponding to three dual feasible values of λ , including the optimum λ^* . Strong duality does not hold; the optimal duality gap $p^* - d^*$ is positive.

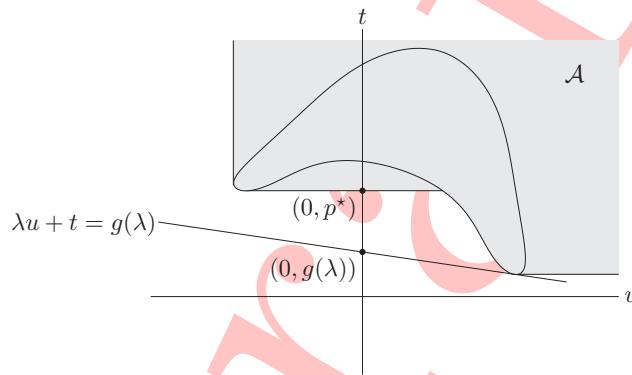


图 6.8: Geometric interpretation of dual function and lower bound $g(\lambda) \leq p^*$, for a problem with one (inequality) constraint. Given λ , we minimize $(\lambda, 1)^T(u, t)$ over $\mathcal{A} = \{(u, t) | \exists x \in \mathcal{D}, f_0(x) \leq t, f_1(x) \leq u\}$. This yields a supporting hyperplane with slope $-\lambda$. The intersection of this hyperplane with the $u = 0$ axis gives $g(\lambda)$.

$1, \dots, m$, $Ax - b = 0$, and $f_0(x) \leq t < p^*$, which is impossible since p^* is the optimal value of the primal problem.

By the separating hyperplane theorem of Section 3.5.1 there exists $(\tilde{\lambda}, \tilde{\nu}, \mu) \neq 0$ and α such that

$$(u, v, t) \in \mathcal{A} \implies \tilde{\lambda}^T u + \tilde{\nu}^T v + \mu t \geq \alpha, \quad (6.48)$$

and

$$(u, v, t) \in \mathcal{B} \implies \tilde{\lambda}^T u + \tilde{\nu}^T v + \mu t \leq \alpha, \quad (6.49)$$

From (6.48) we conclude that $\tilde{\lambda} \geq 0$ and $\mu \geq 0$. (Otherwise $\tilde{\lambda}u + \mu t$ is unbounded below over \mathcal{A} , contradicting (6.48).) The condition (6.49) simply means that $\mu t \leq \alpha$ for all

$t < p^*$, and hence, $\mu p^* \leq \alpha$. Together with (6.48) we conclude that for any $x \in \mathcal{D}$,

$$\sum_{i=1}^m \tilde{\lambda}_i f_i(x) + \tilde{\nu}^T(Ax - b) + \mu f_0(x) \geq \alpha \geq \mu p^*. \quad (6.50)$$

Assume that $\mu > 0$. In that case we can divide (6.50) by μ to obtain

$$L(x, \tilde{\lambda}, \tilde{\nu}/\mu) \geq p^*$$

for all $x \in \mathcal{D}$ from which it follows, by minimizing over x , that $g(\lambda, \nu) \geq p^*$, where we define

$$\lambda = \tilde{\lambda}/\mu, \quad \nu = \tilde{\nu}/\mu.$$

By weak duality we have $g(\lambda, \nu) \leq p^*$, so in fact $g(\lambda, \nu) = p^*$. This shows that strong duality holds, and that the dual optimum is attained, at least in the case when $\mu > 0$.

Now consider the case $\mu = 0$. From (6.50), we conclude that for all $x \in \mathcal{D}$,

$$\sum_{i=1}^m \tilde{\lambda}_i f_i(x) + \tilde{\nu}^T(Ax - b) \geq 0. \quad (6.51)$$

Applying this to the point \tilde{x} that satisfies the Slater condition, we have

$$\sum_{i=1}^m \tilde{\lambda}_i f_i(\tilde{x}) \geq 0.$$

Since $f_i(\tilde{x}) < 0$ and $\tilde{\lambda}_i \geq 0$, we conclude that $\tilde{\lambda} = 0$. From $(\tilde{\lambda}, \tilde{\nu}, \mu) \neq 0$ and $\tilde{\lambda} = 0, \mu = 0$, we conclude that $\tilde{\nu} \neq 0$. Then (6.51) implies that for all $x \in \mathcal{D}$, $\tilde{\nu}^T(Ax - b) \geq 0$. But \tilde{x} satisfies $\tilde{\nu}^T(A\tilde{x} - b) = 0$, and since $\tilde{x} \in \text{int } \mathcal{D}$, there are points in \mathcal{D} with $\tilde{\nu}^T(Ax - b) < 0$ unless $A^T \tilde{\nu} = 0$. This, of course, contradicts our assumption that **rank** $A = p$.

The geometric idea behind the proof is illustrated in Figure 6.9, for a simple problem with one inequality constraint. The hyperplane separating \mathcal{A} and \mathcal{B} defines a supporting hyperplane to \mathcal{A} at $(0, p^*)$. Slater's constraint qualification is used to establish that the hyperplane must be nonvertical (*i.e.*, has a normal vector of the form $(\lambda^*, 1)$). (For a simple example of a convex problem with one inequality constraint for which strong duality fails, see exercise 5.21 of [18].)

6.6.4 Saddle-point interpretation

In this section we give several interpretations of Lagrange duality. The material of this section will not be used in the sequel.

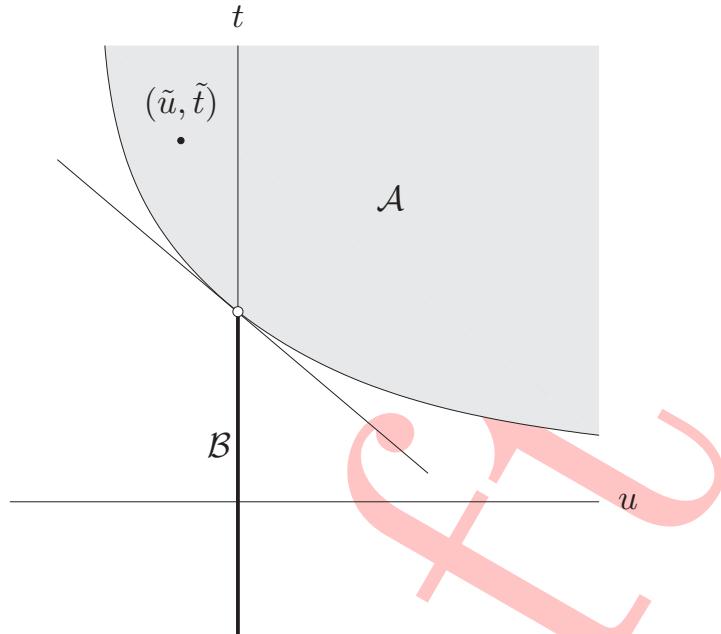


图 6.9: Illustration of strong duality proof, for a convex problem that satisfies Slater's constraint qualification. The set \mathcal{A} is shown shaded, and the set \mathcal{B} is the thick vertical line segment, not including the point $(0, p^*)$, shown as a small open circle. The two sets are convex and do not intersect, so they can be separated by a hyperplane. Slater's constraint qualification guarantees that any separating hyperplane must be nonvertical, since it must pass to the left of the point $(\tilde{u}, \tilde{t}) = (f_1(\tilde{x}), f_0(\tilde{x}))$, where \tilde{x} is strictly feasible.

6.6.4.1 Max-min characterization of weak and strong duality

It is possible to express the primal and the dual optimization problems in a form that is more symmetric. To simplify the discussion we assume there are no equality constraints; the results are easily extended to cover them.

First note that

$$\begin{aligned} \sup_{\lambda \geq 0} L(x, \lambda) &= \sup_{\lambda \geq 0} \left(f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) \right) \\ &= \begin{cases} f_0(x) & f_i(x) \leq 0, \quad i = 1, \dots, m \\ \infty & \text{otherwise,} \end{cases} \end{aligned}$$

Indeed, suppose x is not feasible, and $f_i(x) > 0$ for some i . Then $\sup_{\lambda \geq 0} L(x, \lambda) = \infty$, as can be seen by choosing $\lambda_j = 0$, $j \neq i$, and $\lambda_i \rightarrow \infty$. On the other hand, if $f_i(x) \leq 0$, $i = 1, \dots, m$, then the optimal choice of λ is $\lambda = 0$ and $\sup_{\lambda \geq 0} L(x, \lambda) = f_0(x)$. This

means that we can express the optimal value of the primal problem as

$$p^* = \inf_x \sup_{\lambda \geq 0} L(x, \lambda).$$

By the definition of the dual function, we also have

$$d^* = \sup_{\lambda \geq 0} \inf_x L(x, \lambda).$$

Thus, weak duality can be expressed as the inequality

$$\sup_{\lambda \geq 0} \inf_x L(x, \lambda) \leq \inf_x \sup_{\lambda \geq 0} L(x, \lambda) \quad (6.52)$$

and strong duality as the equality

$$\sup_{\lambda \geq 0} \inf_x L(x, \lambda) = \inf_x \sup_{\lambda \geq 0} L(x, \lambda).$$

Strong duality means that the order of the minimization over x and the maximization over $\lambda \geq 0$ can be switched without affecting the result.

In fact, the inequality (6.52) does not depend on any properties of L : We have

$$\sup_{z \in Z} \inf_{w \in W} f(w, z) \leq \inf_{w \in W} \sup_{z \in Z} f(w, z) \quad (6.53)$$

for any $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ (and any $W \subseteq \mathbb{R}^n$ and $Z \subseteq \mathbb{R}^m$). This general inequality is called the *max-min inequality*. When equality holds, *i.e.*,

$$\sup_{z \in Z} \inf_{w \in W} f(w, z) = \inf_{w \in W} \sup_{z \in Z} f(w, z) \quad (6.54)$$

we say that f (and W and Z) satisfy the *strong max-min property* or the *saddle point property*. Of course the strong max-min property holds only in special cases, for example, when $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ is the Lagrangian of a problem for which strong duality obtains, $W = \mathbb{R}^n$, and $Z = \mathbb{R}_+^m$.

6.6.4.2 Saddle-point interpretation

We refer to a pair $\tilde{w} \in W$, $\tilde{z} \in Z$ as a *saddle-point* for f (and W and Z) if

$$f(\tilde{w}, z) \leq f(\tilde{w}, \tilde{z}) \leq f(w, \tilde{z})$$

for all $w \in W$ and $z \in Z$. In other words, \tilde{w} minimizes $f(w, \tilde{z})$ (over $w \in W$) and \tilde{z} maximizes $f(\tilde{w}, z)$ (over $z \in Z$):

$$f(\tilde{w}, \tilde{z}) = \inf_{w \in W} f(w, \tilde{z}), \quad f(\tilde{w}, \tilde{z}) = \sup_{z \in Z} f(\tilde{w}, z).$$

This implies that the strong max-min property (6.54) holds, and that the common value is $f(\tilde{w}, \tilde{z})$.

Returning to our discussion of Lagrange duality, we see that if x^* and λ^* are primal and dual optimal points for a problem in which strong duality obtains, they form a saddle-point for the Lagrangian. The converse is also true: If (x, λ) is a saddle-point of the Lagrangian, then x is primal optimal, λ is dual optimal, and the optimal duality gap is zero.

6.6.5 Optimality conditions

We remind the reader that we do not assume the problem (6.12) is convex, unless explicitly stated.

6.6.5.1 Certificate of suboptimality and stopping criteria

If we can find a dual feasible (λ, ν) , we establish a lower bound on the optimal value of the primal problem: $p^* \geq g(\lambda, \nu)$. Thus a dual feasible point (λ, ν) provides a *proof or certificate* that $p^* \geq g(\lambda, \nu)$. Strong duality means there exist arbitrarily good certificates.

Dual feasible points allow us to bound how suboptimal a given feasible point is, without knowing the exact value of p^* . Indeed, if x is primal feasible and (λ, ν) is dual feasible, then

$$f_0(x) - p^* \leq f_0(x) - g(\lambda, \nu).$$

In particular, this establishes that x is ϵ -suboptimal, with $\epsilon = f_0(x) - g(\lambda, \nu)$. (It also establishes that (λ, ν) is ϵ -suboptimal for the dual problem.)

We refer to the gap between primal and dual objectives,

$$f_0(x) - g(\lambda, \nu),$$

as the *duality gap* associated with the primal feasible point x and dual feasible point (λ, ν) . A primal dual feasible pair $x, (\lambda, \nu)$ localizes the optimal value of the primal (and dual) problems to an interval:

$$p^* \in [g(\lambda, \nu), f_0(x)], \quad d^* \in [g(\lambda, \nu), f_0(x)],$$

the width of which is the duality gap.

If the duality gap of the primal dual feasible pair $x, (\lambda, \nu)$ is zero, i.e., $f_0(x) = g(\lambda, \nu)$, then x is primal optimal and (λ, ν) is dual optimal. We can think of (λ, ν) as a certificate that proves x is optimal (and, similarly, we can think of x as a certificate that proves (λ, ν) is dual optimal).

These observations can be used in optimization algorithms to provide nonheuristic stopping criteria. Suppose an algorithm produces a sequence of primal feasible $x^{(k)}$ and dual feasible $(\lambda^{(k)}, \nu^{(k)})$, for $k = 1, 2, \dots$, and $\epsilon_{\text{abs}} > 0$ is a given required absolute accuracy. Then the stopping criterion (*i.e.*, the condition for terminating the algorithm)

$$f_0(x^{(k)}) - g(\lambda^{(k)}, \nu^{(k)}) \leq \epsilon_{\text{abs}}$$

guarantees that when the algorithm terminates, $\mathbf{x}^{(k)}$ is ϵ_{abs} -suboptimal. Indeed, $(\lambda^{(k)}, \nu^{(k)})$ is a certificate that proves it. (Of course strong duality must hold if this method is to work for arbitrarily small tolerances ϵ_{abs} .)

A similar condition can be used to guarantee a given relative accuracy. If

$$g(\lambda^{(k)}, \nu^{(k)}) > 0, \quad \frac{f_0(x^{(k)}) - g(\lambda^{(k)}, \nu^{(k)})}{g(\lambda^{(k)}, \nu^{(k)})} \leq \epsilon_{\text{rel}}$$

holds, or

$$f_0(x^{(k)}) < 0, \quad \frac{f_0(x^{(k)}) - g(\lambda^{(k)}, \nu^{(k)})}{-f_0(x^{(k)})} \leq \epsilon_{\text{rel}}$$

holds, then $p^* \neq 0$ and the relative error

$$\frac{f_0(x^{(k)}) - p^*}{|p^*|}$$

is guaranteed to be less than or equal to ϵ_{rel} .

6.6.5.2 Complementary slackness

Suppose that the primal and dual optimal values are attained and equal (so, in particular, strong duality holds). Let x^* be a primal optimal and (λ^*, ν^*) be a dual optimal point. This means that

$$\begin{aligned} f_0(x^*) &= g(\lambda^*, \nu^*) \\ &= \inf_x \left(f_0(x) + \sum_{i=1}^m \lambda_i^* f_i(x) + \sum_{i=1}^p \nu_i^* h_i(x) \right) \\ &\leq f_0(x^*) + \sum_{i=1}^m \lambda_i^* f_i(x^*) + \sum_{i=1}^p \nu_i^* h_i(x^*) \\ &\leq f_0(x^*). \end{aligned}$$

The first line states that the optimal duality gap is zero, and the second line is the definition of the dual function. The third line follows since the infimum of the Lagrangian over x is less than or equal to its value at $x = x^*$. The last inequality follows from $\lambda_i^* \geq 0$,

$f_i(x^*) \leq 0, i = 1, \dots, m$, and $h_i(x^*) = 0, i = 1, \dots, p$. We conclude that the two inequalities in this chain hold with equality.

We can draw several interesting conclusions from this. For example, since the inequality in the third line is an equality, we conclude that x^* minimizes $L(x, \lambda^*, \nu^*)$ over x . (The Lagrangian $L(x, \lambda^*, \nu)^*$ can have other minimizers; x^* is simply a minimizer.)

Another important conclusion is that

$$\sum_{i=1}^m \lambda_i^* f_i(x^*) = 0.$$

Since each term in this sum is nonpositive, we conclude that

$$\lambda_i^* f_i(x^*) = 0, \quad i = 1, \dots, m. \quad (6.55)$$

This condition is known as *complementary slackness*; it holds for any primal optimal x^* and any dual optimal (λ^*, ν^*) (when strong duality holds). We can express the complementary slackness condition as

$$\lambda_i^* > 0 \implies f_i(x^*) = 0,$$

or, equivalently,

$$f_i(x^*) < 0 \implies \lambda_i^* = 0.$$

Roughly speaking, this means the i th optimal Lagrange multiplier is zero unless the i th constraint is active at the optimum.

6.6.5.3 KKT optimality conditions

We now assume that the functions $f_0, \dots, f_m, h_1, \dots, h_p$ are differentiable (and therefore have open domains), but we make no assumptions yet about convexity.

KKT conditions for nonconvex problems

As above, let x^* and (λ^*, ν^*) be any primal and dual optimal points with zero duality gap. Since x^* minimizes $L(x, \lambda^*, \nu^*)$ over x , it follows that its gradient must vanish at x^* , *i.e.*,

$$\nabla f_0(x^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(x^*) + \sum_{i=1}^n \nu_i^* \nabla h_i(x^*) = 0.$$

Thus we have

$$\begin{aligned}
 f_i(x^*) &\leq 0, \quad i = 1, \dots, m \\
 h_i(x^*) &= 0, \quad i = 1, \dots, p \\
 \lambda_i^* &\geq 0, \quad i = 1, \dots, m \\
 \lambda_i^* f_i(x^*) &= 0, \quad i = 1, \dots, m \\
 \nabla f_0(x^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(x^*) + \sum_{i=1}^p \nu_i^* \nabla h_i(x^*) &= 0
 \end{aligned} \tag{6.56}$$

which are called the *Karush-Kuhn-Tucker* (KKT) conditions.

To summarize, for *any* optimization problem with differentiable objective and constraint functions for which strong duality obtains, any pair of primal and dual optimal points must satisfy the KKT conditions (6.56).

KKT conditions for convex problems

When the primal problem is convex, the KKT conditions are also sufficient for the points to be primal and dual optimal. In other words, if f_i are convex and h_i are affine, and \tilde{x} , $\tilde{\lambda}$, $\tilde{\nu}$ are any points that satisfy the KKT conditions

$$\begin{aligned}
 f_i(\tilde{x}) &\leq 0, \quad i = 1, \dots, m \\
 h_i(\tilde{x}) &= 0, \quad i = 1, \dots, p \\
 \tilde{\lambda}_i &\geq 0, \quad i = 1, \dots, m \\
 \tilde{\lambda}_i f_i(\tilde{x}) &= 0, \quad i = 1, \dots, m \\
 \nabla f_0(\tilde{x}) + \sum_{i=1}^m \tilde{\lambda}_i \nabla f_i(\tilde{x}) + \sum_{i=1}^p \tilde{\nu}_i \nabla h_i(\tilde{x}) &= 0,
 \end{aligned}$$

then \tilde{x} and $(\tilde{\lambda}, \tilde{\nu})$ are primal and dual optimal, with zero duality gap.

To see this, note that the first two conditions state that \tilde{x} is primal feasible. Since $\tilde{\lambda}_i \geq 0$, $L(x, \tilde{\lambda}, \tilde{\nu})$ is convex in x ; the last KKT condition states that its gradient with respect to x vanishes at $x = \tilde{x}$, so it follows that \tilde{x} minimizes $L(x, \tilde{\lambda}, \tilde{\nu})$ over x . From this we conclude that

$$\begin{aligned}
 g(\tilde{\lambda}, \tilde{\nu}) &= L(\tilde{x}, \tilde{\lambda}, \tilde{\nu}) \\
 &= f_0(\tilde{x}) + \sum_{i=1}^m \tilde{\lambda}_i f_i(\tilde{x}) + \sum_{i=1}^p \tilde{\nu}_i h_i(\tilde{x}) \\
 &= f_0(\tilde{x}),
 \end{aligned}$$

where in the last line we use $h_i(\tilde{x}) = 0$ and $\tilde{\lambda}_i f_i(\tilde{x}) = 0$. This shows that \tilde{x} and $(\tilde{\lambda}, \tilde{\nu})$ have zero duality gap, and therefore are primal and dual optimal. In summary, for any *convex*

optimization problem with differentiable objective and constraint functions, any points that satisfy the KKT conditions are primal and dual optimal, and have zero duality gap.

If a convex optimization problem with differentiable objective and constraint functions satisfies Slater's condition, then the KKT conditions provide necessary and sufficient conditions for optimality: Slater's condition implies that the optimal duality gap is zero and the dual optimum is attained, so x is optimal if and only if there are (λ, ν) that, together with x , satisfy the KKT conditions.

The KKT conditions play an important role in optimization. In a few special cases it is possible to solve the KKT conditions (and therefore, the optimization problem) analytically. More generally, many algorithms for convex optimization are conceived as, or can be interpreted as, methods for solving the KKT conditions.

6.6.5.4 Solving the primal problem via the dual

We mentioned at the beginning of Section 6.6.5.3 that if strong duality holds and a dual optimal solution (λ^*, ν^*) exists, then any primal optimal point is also a minimizer of $L(x, \lambda^*, \nu^*)$. This fact sometimes allows us to compute a primal optimal solution from a dual optimal solution.

More precisely, suppose we have strong duality and an optimal (λ^*, ν^*) is known. Suppose that the minimizer of $L(x, \lambda^*, \nu^*)$, *i.e.*, the solution of

$$\text{minimize } f_0(x) + \sum_{i=1}^m \lambda_i^* f_i(x) + \sum_{i=1}^p \nu_i^* h_i(x), \quad (6.57)$$

is unique. (For a convex problem this occurs, for example, if $L(x, \lambda^*, \nu^*)$ is a strictly convex function of x .) Then if the solution of (6.57) is primal feasible, it must be primal optimal; if it is not primal feasible, then no primal optimal point can exist, *i.e.*, we can conclude that the primal optimum is not attained. This observation is interesting when the dual problem is easier to solve than the primal problem, for example, because it can be solved analytically, or has some special structure that can be exploited.

Example 505. *Entropy maximization.* We consider the entropy maximization problem

$$\begin{aligned} &\text{minimize} && f_0(x) = \sum_{i=1}^n x_i \log x_i \\ &\text{subject to} && Ax \leq b \\ & && \mathbf{1}^T x = 1 \end{aligned}$$

with domain \mathbb{R}_{++}^n , and its dual problem

$$\begin{aligned} \max_{\lambda} \quad & -b^T \lambda - \nu - e^{-\nu-1} \sum_{i=1}^n e^{-\alpha_i^T \lambda} \\ \text{subject to} \quad & \lambda \geq 0 \end{aligned}$$

(see (6.24) and (6.41)). We assume that the weak form of Slater's condition holds, i.e., there exists an $x \geq 0$ with $Ax \leq b$ and $\mathbf{1}^T x = 1$, so strong duality holds and an optimal solution (λ^*, ν^*) exists.

Suppose we have solved the dual problem. The Lagrangian at (λ^*, ν^*) is

$$L(x, \lambda^*, \nu^*) = \sum_{i=1}^n x_i \log x_i + \lambda^{*T} (Ax - b) + \nu^* (\mathbf{1}^T x - 1)$$

which is strictly convex on \mathcal{D} and bounded below, so it has a unique solution x^* , given by

$$x_i^* = 1 / \exp(\alpha_i^T \lambda^* + \nu^* + 1), \quad i = 1, \dots, n,$$

where α_i are the columns of A . If x^* is primal feasible, it must be the optimal solution of the primal problem (6.24). If x^* is not primal feasible, then we can conclude that the primal optimum is not attained.

Example 506. Minimizing a separable function subject to an equality constraint. We consider the problem

$$\begin{aligned} \text{minimize} \quad & f_0(x) = \sum_{i=1}^n f_i(x_i) \\ \text{subject to} \quad & a^T x = b, \end{aligned}$$

where $a \in \mathbb{R}^n$, $b \in \mathbb{R}^n$ and $f_i : \mathbb{R} \rightarrow \mathbb{R}$ are differentiable and strictly convex. The objective function is called separable since it is a sum of functions of the individual variables x_1, \dots, x_n . We assume that the domain of f_0 intersects the constraint set, i.e., there exists a point $x_0 \in \text{dom } f$ with $a^T x_0 = b$. This implies the problem has a unique optimal point x^* .

The Lagrangian is

$$L(x, \nu) = \sum_{i=1}^n f_i(x_i) + \nu(a^T x - b) = -b\nu + \sum_{i=1}^n (f_i(x_i) + \nu a_i x_i),$$

which is also separable, so the dual function is

$$\begin{aligned} g(\nu) &= -b\nu + \inf_x \left(\sum_{i=1}^n (f_i(x_i) + \nu a_i x_i) \right) \\ &= -b\nu + \sum_{i=1}^n \inf_{x_i} (f_i(x_i) + \nu a_i x_i) \\ &= -b\nu - \sum_{i=1}^n f_i^*(-\nu a_i). \end{aligned}$$

The dual problem is thus

$$\max_{\nu} -b\nu - \sum_{i=1}^n f_i^*(-\nu a_i),$$

with (scalar) variable $\nu \in \mathbb{R}$.

Now suppose we have found an optimal dual variable ν^* . (There are several simple methods for solving a convex problem with one scalar variable, such as the bisection method.) Since each f_i is strictly convex, the function $L(x, \nu)$ is strictly convex in x , and so has a unique minimizer \tilde{x} . But we also know that x^* minimizes $L(x, \nu^*)$, so we must have $\tilde{x} = x^*$. We can recover x^* from $\nabla_x L(x, \nu^*)$, i.e., by solving the equations $f'_i(x^*) = -\nu^* a_i$.

6.6.6 Examples

In this section we show by example that simple equivalent reformulations of a problem can lead to very different dual problems. We consider the following types of reformulations:

- Introducing new variables and associated equality constraints.
- Replacing the objective with an increasing function of the original objective.
- Making explicit constraints implicit, i.e., incorporating them into the domain of the objective.

6.6.6.1 Introducing new variables and equality constraints

Consider an unconstrained problem of the form

$$\text{minimize } f_0(Ax + b). \quad (6.58)$$

Its Lagrange dual function is the constant p^* . So while we do have strong duality, i.e., $p^* = d^*$, the Lagrangian dual is neither useful nor interesting.

Now let us reformulate the problem (6.58) as

$$\begin{aligned} & \text{minimize} && f_0(y) \\ & \text{subject to} && Ax + b = y. \end{aligned} \tag{6.59}$$

Here we have introduced new variables y , as well as new equality constraints $Ax + b = y$. The problems (6.58) and (6.59) are clearly equivalent.

The Lagrangian of the reformulated problem is

$$L(x, y, \nu) = f_0(y) + \nu^T(Ax + b - y).$$

To find the dual function we minimize L over x and y . Minimizing over x we find that $g(\nu) = -\infty$ unless $A^T\nu = 0$, in which case we are left with

$$g(\nu) = b^T\nu + \inf_y (f_0(y) - \nu^T y) = b^T\nu - f_0^*(\nu),$$

where f_0^* is the conjugate of f_0 . The dual problem of (6.59) can therefore be expressed as

$$\begin{aligned} & \max_{\nu} && b^T\nu - f_0^*(\nu) \\ & \text{subject to} && A^T\nu = 0. \end{aligned} \tag{6.60}$$

Thus, the dual of the reformulated problem (6.59) is considerably more useful than the dual of the original problem (6.58).

Example 507. *Unconstrained geometric program. Consider the unconstrained geometric program*

$$\text{minimize } \log \left(\sum_{i=1}^m \exp(a_i^T x + b_i) \right).$$

We first reformulate it by introducing new variables and equality constraints:

$$\begin{aligned} & \text{minimize} && f_0(y) = \log \left(\sum_{i=1}^m \exp y_i \right) \\ & \text{subject to} && Ax + b = y, \end{aligned}$$

where a_i^T are the rows of A . The conjugate of the log-sum-exp function is

$$f_0^*(\nu) = \begin{cases} \sum_{i=1}^m \nu_i \log \nu_i & \nu \geq 0, \quad \mathbf{1}^T \nu = 1 \\ \infty & \text{otherwise} \end{cases}$$

(Example 223), so the dual of the reformulated problem can be expressed as

$$\begin{aligned} \max_{\nu} \quad & b^T \nu - \sum_{i=1}^m \nu_i \log \nu_i \\ \text{subject to} \quad & \mathbf{1}^T \nu = 1 \\ & A^T \nu = 0 \\ & \nu \geq 0, \end{aligned} \tag{6.61}$$

which is an entropy maximization problem.

Example 508. Norm approximation problem. We consider the unconstrained norm approximation problem

$$\text{minimize} \quad \|Ax - b\|, \tag{6.62}$$

where $\|\cdot\|$ is any norm. Here too the Lagrange dual function is constant, equal to the optimal value of (6.62), and therefore not useful.

Once again we reformulate the problem as

$$\begin{aligned} \text{minimize} \quad & \|y\| \\ \text{subject to} \quad & Ax - b = y, \end{aligned}$$

The Lagrange dual problem is, following (6.60),

$$\begin{aligned} \max_{\nu} \quad & b^T \nu \\ \text{subject to} \quad & \|\nu\|_* \leq 1 \\ & A^T \nu = 0, \end{aligned} \tag{6.63}$$

where we use the fact that the conjugate of a norm is the indicator function of the dual norm unit ball (Example 224).

The idea of introducing new equality constraints can be applied to the constraint functions as well. Consider, for example, the problem

$$\begin{aligned} \text{minimize} \quad & f_0(A_0x + b_0) \\ \text{subject to} \quad & f_i(A_i x + b_i) \leq 0, \quad i = 1, \dots, m, \end{aligned} \tag{6.64}$$

where $A_i \in \mathbb{R}^{k_i \times n}$ and $f_i : \mathbb{R}^{k_i} \rightarrow \mathbb{R}$ are convex. (For simplicity we do not include equality constraints here.) We introduce a new variable $y_i \in \mathbb{R}^{k_i}$, for $i = 0, \dots, m$, and reformulate the problem as

$$\begin{aligned} \text{minimize} \quad & f_0(y_0) \\ \text{subject to} \quad & f_i(y_0) \leq 0, \quad i = 1, \dots, m, \\ & A_i x + b_i = y_i, \quad i = 0, \dots, m. \end{aligned} \tag{6.65}$$

The Lagrangian for this problem is

$$L(x, \lambda, \nu_0, \dots, \nu_m) = f_0(y_0) + \sum_{i=1}^m \lambda_i f_i(y_i) + \sum_{i=0}^m \nu_i^T (A_i x + b_i - y_i).$$

To find the dual function we minimize over x and y_i . The minimum over x is $-\infty$ unless

$$\sum_{i=0}^m A_i^T \nu_i = 0,$$

in which case we have, for $\lambda \geq 0$,

$$\begin{aligned} & g(\lambda, \nu_0, \dots, \nu_m) \\ &= \sum_{i=0}^m \nu_i^T b_i + \inf_{y_0, \dots, y_m} \left(f_0(y_0) + \sum_{i=1}^m \lambda_i f_i(y_i) - \sum_{i=0}^m \nu_i^T y_i \right) \\ &= \sum_{i=0}^m \nu_i^T b_i + \inf_{y_0} (f_0(y_0) - \nu_0^T y_0) + \sum_{i=1}^m \lambda_i \inf_{y_i} (f_i(y_i) - (\nu_i / \lambda_i)^T y_i) \\ &= \sum_{i=0}^m \nu_i^T b_i - f_0^*(\nu_0) - \sum_{i=1}^m \lambda_i f_i^*(\nu_i / \lambda_i). \end{aligned}$$

The last expression involves the perspective of the conjugate function, and is therefore concave in the dual variables. Finally, we address the question of what happens when $\lambda \geq 0$, but some λ_i are zero. If $\lambda_i = 0$ and $\nu_i \neq 0$, then the dual function is $-\infty$. If $\lambda_i = 0$ and $\nu_i = 0$, however, the terms involving y_i , ν_i , and λ_i are all zero. Thus, the expression above for g is valid for all $\lambda \geq 0$, if we take $\lambda_i f_i^*(\nu_i / \lambda_i) = 0$ when $\lambda_i = 0$ and $\nu_i = 0$, and $\lambda_i f_i^*(\nu_i / \lambda_i) = -\infty$ when $\lambda_i = 0$ and $\nu_i \neq 0$.

$$\begin{aligned} & \text{maximize} \quad \sum_{i=0}^m \nu_i^T b_i - f_0^*(\nu_0) - \sum_{i=1}^m \lambda_i f_i^*(\nu_i / \lambda_i) \\ & \text{subject to} \quad \lambda \geq 0 \\ & \quad \sum_{i=0}^m A_i^T \nu_i = 0. \end{aligned} \tag{6.66}$$

Example 509. Inequality constrained geometric program. The inequality constrained geometric program

$$\begin{aligned} & \text{minimize} \quad \log \left(\sum_{k=1}^{K_0} e^{a_{0k}^T x + b_{0k}} \right) \\ & \text{subject to} \quad \log \left(\sum_{k=1}^{K_i} e^{a_{ik}^T x + b_{ik}} \right) \leq 0, \quad i = 1, \dots, m \end{aligned}$$

is of the form (6.64) with $f_i : \mathbb{R}^{K_i} \rightarrow \mathbb{R}$ given by $f_i(y) = \log(\sum_{k=1}^{K_i} e^{y_k})$. The conjugate of this function is

$$f_i^*(\nu) = \begin{cases} \sum_{k=1}^{K_i} \nu_k \log \nu_k & \nu \geq 0, \quad \mathbf{1}^T \nu = 1 \\ \infty & \text{otherwise.} \end{cases}$$

Using (6.66) we can immediately write down the dual problem as

$$\begin{aligned} & \text{maximize} && b_0^T \nu_0 - \sum_{k=1}^{K_0} \nu_{0k} \log \nu_{0k} + \sum_{i=1}^m \left(b_i^T \nu_i - \sum_{k=1}^{K_i} \nu_{ik} \log(\nu_{ik}/\lambda_i) \right) \\ & \text{subject to} && \nu_0 \geq 0, \quad \mathbf{1}^T \nu = 1 \\ & && \nu_0 \geq 0, \quad \mathbf{1}^T \nu_i = \lambda_i, \quad i = 1, \dots, m \\ & && \lambda_i \geq 0, \quad i = 1, \dots, m \\ & && \sum_{i=0}^m A_i^T \nu_i = 0, \end{aligned}$$

which further simplifies to

$$\begin{aligned} & \text{maximize} && b_0^T \nu_0 - \sum_{k=1}^{K_0} \nu_{0k} \log \nu_{0k} + \sum_{i=1}^m \left(b_i^T \nu_i - \sum_{k=1}^{K_i} \nu_{ik} \log(\nu_{ik}/\mathbf{1}^T \nu_i) \right) \\ & \text{subject to} && \nu_i \geq 0, \quad i = 0, \dots, m \\ & && \mathbf{1}^T \nu_0 = 1, \\ & && \sum_{i=0}^m A_i^T \nu_i = 0, \end{aligned}$$

6.6.6.2 Transforming the objective

If we replace the objective f_0 by an increasing function of f_0 , the resulting problem is clearly equivalent (see Section 4.1.3 of [18]). The dual of this equivalent problem, however, can be very different from the dual of the original problem.

Example 510. We consider again the minimum norm problem

$$\text{minimize} \quad \|Ax - b\|,$$

where $\|\cdot\|$ is some norm. We reformulate this problem as

$$\begin{aligned} & \text{maximize} && (1/2)\|y\|^2 \\ & \text{subject to} && Ax - b = y. \end{aligned}$$

Here we have introduced new variables, and replaced the objective by half its square. Evidently it is equivalent to the original problem.

The dual of the reformulated problem is

$$\begin{aligned} & \text{maximize} && -(1/2)\|\nu\|_*^2 + b^T \nu \\ & \text{subject to} && A^T \nu = 0, \end{aligned}$$

where we use the fact that the conjugate of $(1/2)\|\cdot\|^2$ is $(1/2)\|\cdot\|_*^2$ (see Example 216).

Note that this dual problem is not the same as the dual problem (6.63) derived earlier.

6.6.6.3 Implicit constraints

The next simple reformulation we study is to include some of the constraints in the objective function, by modifying the objective function to be infinite when the constraint is violated.

Example 511. Linear program with box constraints. We consider the linear program

$$\begin{aligned} & \text{minimize} && c^T x \\ & \text{subject to} && Ax = b \\ & && l \leq x \leq u \end{aligned} \tag{6.67}$$

where $A \in \mathbb{R}^{p \times n}$ and $l \prec u$. The constraints $l \leq x \leq u$ are sometimes called box constraints or variable bounds.

We can, of course, derive the dual of this linear program. The dual will have a Lagrange multiplier ν associated with the equality constraint, λ_1 associated with the inequality constraint $x \leq u$, and λ_2 associated with the inequality constraint $l \leq x$. The dual is

$$\begin{aligned} & \text{maximize} && -b^T \nu - \lambda_1^T u + \lambda_2^T l \\ & \text{subject to} && A^T \nu + \lambda_1 - \lambda_2 + c = 0 \\ & && \lambda_1 \geq 0, \quad \lambda_2 \geq 0. \end{aligned} \tag{6.68}$$

Instead, let us first reformulate the problem (6.67) as

$$\begin{aligned} & \text{minimize} && f_0(x) \\ & \text{subject to} && Ax = b, \end{aligned} \tag{6.69}$$

where we define

$$f_0(x) = \begin{cases} c^T x & l \leq x \leq u \\ \infty & \text{otherwise.} \end{cases}$$

The problem (6.69) is clearly equivalent to (6.67); we have merely made the explicit box constraints implicit.

The dual function for the problem (6.69) is

$$\begin{aligned} g(\nu) &= \inf_{l \leq x \leq u} (c^T x + \nu^T (Ax - b)) \\ &= -b^T \nu - u^T (A^T \nu + c)^- + l^T (A^T \nu + c)^+ \end{aligned}$$

where $y_i^+ = \max\{y_i, 0\}$, $y_i^- = \max\{-y_i, 0\}$. So here we are able to derive an analytical formula for g , which is a concave piecewise-linear function.

The dual problem is the unconstrained problem

$$\text{minimize } -b^T \nu - u^T (A^T \nu + c)^- + l^T (A^T \nu + c)^+, \quad (6.70)$$

which has a quite different form from the dual of the original problem.

(The problems (6.68) and (6.70) are closely related, in fact, equivalent; see exercise 5.8 of [18].)

6.6.7 Theorems of alternatives

6.6.7.1 Weak alternatives via the dual function

In this section we apply Lagrange duality theory to the problem of determining feasibility of a system of inequalities and equalities

$$f_i(x) \leq 0, \quad i = 1, \dots, m, \quad h_i(x) = 0, \quad i = 1, \dots, p. \quad (6.71)$$

We assume the domain of the inequality system (6.71), $\mathcal{D} = \bigcap_{i=1}^m \text{dom } f_i \bigcap_{i=1}^p \text{dom } h_i$, is nonempty. We can think of (6.71) as the standard problem (6.12), with objective $f_0 = 0$, i.e.,

$$\begin{aligned} &\text{minimize} && 0 \\ &\text{subject to} && f_i(x) \leq 0, \quad i = 1, \dots, m \\ & && h_i(x) = 0, \quad i = 1, \dots, p. \end{aligned} \quad (6.72)$$

This problem has optimal value

$$p^* = \begin{cases} 0 & (6.71) \text{ is feasible} \\ \infty & (6.71) \text{ is infeasible,} \end{cases} \quad (6.73)$$

so solving the optimization problem (6.72) is the same as solving the inequality system (6.71).

The dual function

We associate with the inequality system (6.71) the dual function

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} \left(\sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x) \right),$$

which is the same as the dual function for the optimization problem (6.72). Since $f_0 = 0$, the dual function is positive homogeneous in (λ, ν) : For α , $g(\alpha\lambda, \alpha\nu) = \alpha g(\lambda, \nu)$. The dual problem associated with (6.72) is to maximize $g(\lambda, \nu)$ subject to $\lambda \geq 0$. Since g is homogeneous, the optimal value of this dual problem is given by

$$d^* = \begin{cases} \infty & \lambda \geq 0, \quad g(\lambda, \nu) > 0 \text{ is feasible} \\ 0 & \lambda \geq 0, \quad g(\lambda, \nu) > 0 \text{ is infeasible.} \end{cases} \quad (6.74)$$

Weak duality tells us that $d^* \leq p^*$. Combining this fact with (6.73) and (6.74) yields the following: If the inequality system

$$\lambda \geq 0, \quad g(\lambda, \nu) > 0 \quad (6.75)$$

is feasible (which means $d^* = \infty$), then the inequality system (6.71) is infeasible (since we then have $p^* = \infty$). Indeed, we can interpret any solution (λ, ν) of the inequalities (6.75) as a *proof* or *certificate* of infeasibility of the system (6.71).

We can restate this implication in terms of feasibility of the original system: If the original inequality system (6.71) is feasible, then the inequality system (6.75) must be infeasible. We can interpret an x which satisfies (6.71) as a certificate establishing infeasibility of the inequality system (6.75).

Two systems of inequalities (and equalities) are called *weak alternatives* if at most one of the two is feasible. Thus, the systems (6.71) and (6.75) are weak alternatives. This is true whether or not the inequalities (6.71) are convex (i.e., f_i convex, h_i affine); moreover, the alternative inequality system (6.75) is always convex (i.e., g is concave and the constraints $\lambda_i \geq 0$ are convex).

Strict inequalities

We can also study feasibility of the *strict* inequality system

$$f_i(x) < 0, \quad i = 1, \dots, m, \quad h_i(x) = 0, \quad i = 1, \dots, p. \quad (6.76)$$

With g defined as for the nonstrict inequality system, we have the alternative inequality system

$$\lambda \geq 0, \quad \lambda \neq 0, \quad g(\lambda, \nu) \geq 0. \quad (6.77)$$

We can show directly that (6.76) and (6.77) are weak alternatives. Suppose there exists an \tilde{x} with $f_i(\tilde{x}) < 0$, $h_i(\tilde{x}) = 0$. Then for any $\lambda \geq 0$, $\lambda \neq 0$, and ν ,

$$\lambda_1 f_1(\tilde{x}) + \cdots + \lambda_m f_m(\tilde{x}) + \nu_1 h_1(\tilde{x}) + \cdots + \nu_p h_p(\tilde{x}) < 0.$$

It follows that

$$\begin{aligned} g(\lambda, \nu) &= \inf_{x \in \mathcal{D}} \left(\sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x) \right) \\ &\leq \sum_{i=1}^m \lambda_i f_i(\tilde{x}) + \sum_{i=1}^p \nu_i h_i(\tilde{x}) \\ &< 0. \end{aligned}$$

Therefore, feasibility of (6.76) implies that there does not exist (λ, ν) satisfying (6.77).

Thus, we can prove infeasibility of (6.76) by producing a solution of the system (6.77); we can prove infeasibility of (6.77) by producing a solution of the system (6.76).

6.6.7.2 Strong alternatives

When the original inequality system is convex, i.e., f_i are convex and h_i are affine, and some type of constraint qualification holds, then the pairs of weak alternatives described above are *strong alternatives*, which means that *exactly one* of the two alternatives holds. In other words, each of the inequality systems is feasible if and only if the other is infeasible.

In this section we assume that f_i are convex and h_i are affine, so the inequality system (6.71) can be expressed as

$$f_i(x) \leq 0, \quad i = 1, \dots, m, \quad Ax = b,$$

where $A \in \mathbb{R}^{p \times n}$.

Strict inequalities

We first study the strict inequality system

$$f_i(x) < 0, \quad i = 1, \dots, m, \quad Ax = b, \tag{6.78}$$

and its alternative

$$\lambda \geq 0, \quad \lambda \neq 0, \quad g(\lambda, \nu) \geq 0. \tag{6.79}$$

We need one technical condition: There exists an $x \in \text{relint } \mathcal{D}$ with $Ax = b$. In other words we not only assume that the linear equality constraints are consistent, but also that

they have a solution in **relint** \mathcal{D} . (Very often $\mathcal{D} = \mathbb{R}^n$, so the condition is satisfied if the equality constraints are consistent.) Under this condition, exactly one of the inequality systems (6.78) and (6.79) is feasible. In other words, the inequality systems (6.78) and (6.79) are strong alternatives.

We will establish this result by considering the related optimization problem

$$\begin{aligned} & \text{minimize } s \\ & \text{subject to } f_i(x) - s \leq 0, \quad i = 1, \dots, m \\ & \quad Ax = b \end{aligned} \tag{6.80}$$

with variables x, s , and domain $\mathcal{D} \times \mathbb{R}$. The optimal value p^* of this problem is negative if and only if there exists a solution to the strict inequality system (6.78).

The Lagrange dual function for the problem (6.80) is

$$\inf_{x \in \mathcal{D}, s} \left(s + \sum_{i=1}^m \lambda_i (f_i(x) - s) + \nu^T (Ax - b) \right) = \begin{cases} g(\lambda, \nu) & \mathbf{1}^T \lambda = 1 \\ -\infty & \text{otherwise.} \end{cases}$$

Therefore we can express the dual problem of (6.80) as

$$\begin{aligned} & \text{maximize } g(\lambda, \nu) \\ & \text{subject to } \lambda \geq 0, \quad \mathbf{1}^T \lambda = 1. \end{aligned}$$

Now we observe that Slater's condition holds for the problem (6.80). By the hypothesis there exists an $\tilde{x} \in \text{relint } \mathcal{D}$ with $A\tilde{x} = b$. Choosing any $\tilde{s} > \max_i f_i(\tilde{x})$ yields a point (\tilde{x}, \tilde{s}) which is strictly feasible for (6.80). Therefore we have $d^* = p^*$, and the dual optimum d^* is attained. In other words, there exist (λ^*, ν^*) such that

$$g(\lambda^*, \nu^*) = p^* \quad \lambda^* \geq 0, \quad \mathbf{1}^T \lambda^* = 1. \tag{6.81}$$

Now suppose that the strict inequality system (6.78) is infeasible, which means that $p^* \geq 0$. Then (λ^*, ν^*) from (6.81) satisfy the alternate inequality system (6.79). Similarly, if the alternate inequality system (6.78) is feasible, then $p^* < 0$, hence $d^* \leq p^* < 0$. So (6.79) is infeasible. Thus, the inequality systems (6.78) and (6.79) are strong alternatives; each is feasible if and only if the other is not.

Nonstrict inequalities

We now consider the nonstrict inequality system

$$f_i(x) \leq 0, \quad i = 1, \dots, m, \quad Ax = b, \tag{6.82}$$

and its alternative

$$\lambda \geq 0, \quad g(\lambda, \nu) > 0. \quad (6.83)$$

We will show these are strong alternatives, provided the following conditions hold: There exists an $x \in \text{relint } \mathcal{D}$ with $Ax = b$, and the optimal value p^* of (6.80) is attained. This holds, for example, if $\mathcal{D} = \mathbb{R}^n$ and $\max_i f_i(x) \rightarrow \infty$ as $x \rightarrow \infty$. With these assumptions we have, as in the strict case, that $p^* = d^*$, and that both the primal and dual optimal values are attained.

Now suppose that the nonstrict inequality system (6.82) is infeasible, which means that $p^* > 0$. (Here we use the assumption that the primal optimal value is attained.) Then (λ^*, ν^*) from (6.81) satisfy the alternate inequality system (6.83). Similarly, if (6.82) is feasible, then $p^* \leq 0$ and thus $d^* \leq p^* \leq 0$. So (6.83) is infeasible. Thus, the inequality systems (6.82) and (6.83) are strong alternatives; each is feasible if and only if the other is not.

6.6.7.3 Examples

Linear inequalities Consider the system of linear inequalities $Ax \preceq b$. The dual function is

$$g(\lambda) = \inf_x \lambda^T(Ax - b) = \begin{cases} -b^T \lambda & A^T \lambda = 0 \\ -\infty & \text{otherwise.} \end{cases}$$

The alternative inequality system is therefore

$$\lambda \geq 0, \quad A^T \lambda = 0, \quad b^T \lambda < 0.$$

These are, in fact, strong alternatives. This follows since the optimum in the related problem (6.80) is achieved, unless it is unbounded below.

We now consider the system of strict linear inequalities $Ax \prec b$, which has the strong alternative system

$$\lambda \geq 0, \quad \lambda \neq 0, \quad A^T \lambda = 0, \quad b^T \lambda \leq 0.$$

In fact we have encountered (and proved) this result before, in Section 3.5.1; see (3.18) and (3.19).

Intersection of ellipsoids

We consider m ellipsoids, described as

$$\mathcal{E}_i = \{x \mid f_i(x) \leq 0\},$$

with $f_i(x) = x^T A_i x + 2b_i^T x + c_i$, $i = 1, \dots, m$, where $A_i \in \mathbf{S}_{++}^n$. We ask when the intersection of these ellipsoids has nonempty interior. This is equivalent to feasibility of the set of strict quadratic inequalities

$$f_i(x) = x^T A_i x + 2b_i^T x + c_i < 0, \quad i = 1, \dots, m. \quad (6.84)$$

The dual function g is

$$\begin{aligned} g(\lambda) &= \inf_x (x^T A(\lambda)x + 2b(\lambda)^T x + c(\lambda)) \\ &= \begin{cases} -b(\lambda)^T A(\lambda)^{-1} b(\lambda) + c(\lambda) & A(\lambda) \succeq 0, \quad b(\lambda) \in \mathcal{R}(A(\lambda)) \\ -\infty & \text{otherwise,} \end{cases} \end{aligned}$$

where

$$A(\lambda) = \sum_{i=1}^m \lambda_i A_i, \quad b(\lambda) = \sum_{i=1}^m \lambda_i b_i, \quad c(\lambda) = \sum_{i=1}^m \lambda_i c_i.$$

Note that for $\lambda \geq 0$, $\lambda \neq 0$, we have $A(\lambda) \succ 0$, so we can simplify the expression for the dual function as

$$g(\lambda) = -b(\lambda)^T A(\lambda)^{-1} b(\lambda) + c(\lambda).$$

The strong alternative of the system (6.84) is therefore

$$\lambda \geq 0, \quad \lambda \neq 0, \quad -b(\lambda)^T A(\lambda)^{-1} b(\lambda) + c(\lambda) \geq 0. \quad (6.85)$$

We can give a simple geometric interpretation of this pair of strong alternatives. For any nonzero $\lambda \geq 0$, the (possibly empty) ellipsoid

$$\mathcal{E}_\lambda = \{x \mid x^T A(\lambda)x + 2b(\lambda)^T x + c(\lambda) \leq 0\}$$

contains $\mathcal{E}_1 \cap \dots \cap \mathcal{E}_m$, since $f_i(x) \leq 0$ implies $\sum_{i=1}^m \lambda_i f_i(x) \leq 0$. Now, \mathcal{E}_λ has empty interior if and only if

$$\inf_x (x^T A(\lambda)x + 2b(\lambda)^T x + c(\lambda)) = -b(\lambda)^T A(\lambda)^{-1} b(\lambda) + c(\lambda) \geq 0.$$

Therefore the alternative system (6.85) means that \mathcal{E}_λ has empty interior.

Weak duality is obvious: If (6.85) holds, then \mathcal{E}_λ contains the intersection $\mathcal{E}_1 \cap \dots \cap \mathcal{E}_m$, and has empty interior, so naturally the intersection has empty interior. The fact that these are strong alternatives states the (not obvious) fact that if the intersection $\mathcal{E}_1 \cap \dots \cap \mathcal{E}_m$ has empty interior, then we can construct an ellipsoid \mathcal{E}_λ that contains the intersection and has empty interior.

Farkas' lemma In this section we describe a pair of strong alternatives for a mixture of

strict and nonstrict linear inequalities, known as *Farkas' lemma*: The system of inequalities

$$Ax \preceq 0, \quad c^T x < 0, \quad (6.86)$$

where $A \in \mathbb{R}^{m \times n}$ and $c \in \mathbb{R}^n$, and the system of inequalities

$$A^T y + c = 0, \quad y \geq 0, \quad (6.87)$$

are strong alternatives.

We can prove Farkas' lemma directly, using LP duality. Consider the LP

$$\begin{aligned} & \text{minimize} && c^T x \\ & \text{subject to} && Ax \preceq 0, \end{aligned} \quad (6.88)$$

and its dual

$$\begin{aligned} & \text{maximize} && 0 \\ & \text{subject to} && A^T y + c = 0 \\ & && y \geq 0. \end{aligned} \quad (6.89)$$

The primal LP (6.88) is homogeneous, and so has optimal value 0, if (6.86) is not feasible, and optimal value $-\infty$, if (6.86) is feasible. The dual LP (6.89) has optimal value 0, if (6.87) is feasible, and optimal value $-\infty$, if (6.87) is infeasible.

Since $x = 0$ is feasible in (6.88), we can rule out the one case in which strong duality can fail for LPs, so we must have $p^* = d^*$. Combined with the remarks above, this shows that (6.86) and (6.87) are strong alternatives.

Example 512. *Arbitrage-free bounds on price.* We consider a set of n assets, with prices at the beginning of an investment period p_1, \dots, p_n , respectively. At the end of the investment period, the value of the assets is v_1, \dots, v_n . If x_1, \dots, x_n represents the initial investment in each asset (with $x_j < 0$ meaning a short position in asset j), the cost of the initial investment is $p^T x$, and the final value of the investment is $v^T x$. The value of the assets at the end of the investment period, v , is uncertain. We will assume that only m possible scenarios, or outcomes, are possible. If outcome i occurs, the final value of the assets is $v^{(i)}$, and therefore, the overall value of the investments is $v(i)^T x$.

If there is an investment vector x with $p^T x < 0$, and in all possible scenarios, the final value is nonnegative, i.e., $v^{(i)T} x \geq 0$ for $i = 1, \dots, m$, then an arbitrage is said to exist. The condition $p^T x < 0$ means you are paid to accept the investment mix, and

the condition $v^{(i)T}x \geq 0$ for $i = 1, \dots, m$ means that no matter what outcome occurs, the final value is nonnegative, so an arbitrage corresponds to a guaranteed money-making investment strategy. It is generally assumed that the prices and values are such that no arbitrage exists. This means that the inequality system

$$Vx \geq 0, \quad p^T x < 0$$

is infeasible, where $V_{ij} = v_j^{(i)}$. Using Farkas' lemma, we have no arbitrage if and only if there exists y such that

$$-V^T y + p = 0, \quad y \geq 0.$$

We can use this characterization of arbitrage-free prices and values to solve several interesting problems.

Suppose, for example, that the values V are known, and all prices except the last one, p_n , are known. The set of prices p_n that are consistent with the no-arbitrage assumption is an interval, which can be found by solving a pair of LPs. The optimal value of the LP

$$\begin{aligned} & \text{minimize} && p_n \\ & \text{subject to} && V^T y = p, \quad y \geq 0, \end{aligned}$$

with variables p_n and y , gives the smallest possible arbitrage-free price for asset n . Solving the same LP with maximization instead of minimization yields the largest possible price for asset n . If the two values are equal, i.e., the no-arbitrage assumption leads us to a unique price for asset n , we say the market is complete. For an example, see exercise 5.38.

This method can be used to find bounds on the price of a derivative or option that is based on the final value of other underlying assets, i.e., when the value or payoff of asset n is a function of the values of the other assets.

6.6.8 Exercises

Basic definitions

Exercise 513 (A simple example). Consider the optimization problem

$$\begin{aligned} & \min_x x^2 + 1 \\ & \text{s.t. } (x - 2)(x - 4) \leq 0, \end{aligned}$$

with variable $x \in \mathbb{R}$.

- (a) Analysis of primal problem. Give the feasible set, the optimal value, and the optimal solution.
- (b) Lagrangian and dual function. Plot the objective $x^2 + 1$ versus x . On the same plot, show the feasible set, optimal point and value, and plot the Lagrangian $L(x, \lambda)$ versus x for a few positive values of λ . Verify the lower bound property ($p^* \geq \inf_x L(x, \lambda)$ for $\lambda \geq 0$). Derive and sketch the Lagrange dual function g .
- (c) Lagrange dual problem. State the dual problem, and verify that it is a concave maximization problem. Find the dual optimal value and dual optimal solution λ^* . Does strong duality hold?

- (d) Sensitivity analysis. Let $p^*(u)$ denote the optimal value of the problem

$$\begin{aligned} & \min_x x^2 + 1 \\ & \text{s.t. } (x - 2)(x - 4) \leq u, \end{aligned}$$

as a function of the parameter u . Plot $p^*(u)$. Verify that $dp^*(0)/du = -\lambda^*$.

Exercise 514 (Weak duality for unbounded and infeasible problems). *The weak duality inequality, $d^* \leq p^*$, clearly holds when $d^* = -\infty$ or $p^* = \infty$. Show that it holds in the other two cases as well: If $p^* = -\infty$, then we must have $d^* = -\infty$, and also, if $d^* = \infty$, then we must have $p^* = \infty$.*

Exercise 515 (Problems with one inequality constraint). Express the dual problem of

$$\begin{aligned} & \min_{\mathbf{x}} \mathbf{c}^T \mathbf{x} \\ & \text{s.t. } f(\mathbf{x}) \leq 0, \end{aligned}$$

with $\mathbf{c} \neq \mathbf{0}$, in terms of the conjugate f^* . Explain why the problem you give is convex. We do not assume f is convex.

Examples and applications

Exercise 516 (Interpretation of LP dual via relaxed problems). Consider the inequality form LP

$$\begin{aligned} & \min_{\mathbf{x}} \mathbf{c}^T \mathbf{x} \\ & \text{s.t. } \mathbf{A}\mathbf{x} \leq \mathbf{b}, \end{aligned}$$

with $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$. In this exercise we develop a simple geometric interpretation of the dual LP (6.33).

Let $\mathbf{w} \in \mathbb{R}_+^m$. If \mathbf{x} is feasible for the LP, i.e., satisfies $\mathbf{Ax} \leq \mathbf{b}$, then it also satisfies the inequality

$$\mathbf{w}^T \mathbf{Ax} \leq \mathbf{w}^T \mathbf{b}.$$

Geometrically, for any $\mathbf{w} \geq \mathbf{0}$, the halfspace $\mathbf{H}_w = \{\mathbf{x} | \mathbf{w}^T \mathbf{Ax} \leq \mathbf{w}^T \mathbf{b}\}$ contains the feasible set for the LP. Therefore if we minimize the objective $\mathbf{c}^T \mathbf{x}$ over the halfspace \mathbf{H}_w we get a lower bound on p^* .

- (a) Derive an expression for the minimum value of $\mathbf{c}^T \mathbf{x}$ over the halfspace \mathbf{H}_w (which will depend on the choice of $\mathbf{w} \geq \mathbf{0}$).
- (b) Formulate the problem of finding the best such bound, by maximizing the lower bound over $\mathbf{w} \geq \mathbf{0}$.
- (c) Relate the results of (a) and (b) to the Lagrange dual of the LP, given by (6.33).

Exercise 517 (Dual of general LP). Find the dual function of the LP

$$\begin{aligned} & \min_{\mathbf{x}} \mathbf{c}^T \mathbf{x} \\ & \text{s.t. } \mathbf{Gx} \leq \mathbf{h}, \\ & \quad \mathbf{Ax} = \mathbf{b}. \end{aligned}$$

Give the dual problem, and make the implicit equality constraints explicit.

Exercise 518 (Lower bounds in Chebyshev approximation from least-squares). Consider the Chebyshev or ℓ_∞ -norm approximation problem

$$\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_\infty,$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\text{rank}(\mathbf{A}) = n$. Let \mathbf{x}_{ch} denote an optimal solution (there may be multiple optimal solutions; \mathbf{x}_{ch} denotes one of them).

The Chebyshev problem has no closed-form solution, but the corresponding least-squares problem does. Define

$$\mathbf{x}_{ls} = \operatorname{argmin}_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2 = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}.$$

We address the following question. Suppose that for a particular \mathbf{A} and \mathbf{b} we have computed the least-squares solution \mathbf{x}_{ls} (but not \mathbf{x}_{ch}). How suboptimal is \mathbf{x}_{ls} for the Chebyshev problem? In other words, how much larger is $\|\mathbf{Ax}_{ls} - \mathbf{b}\|_\infty$ than $\|\mathbf{Ax}_{ch} - \mathbf{b}\|_\infty$?

(a) Prove the lower bound

$$\|\mathbf{A}\mathbf{x}_{ls} - \mathbf{b}\|_\infty \leq \sqrt{m} \|\mathbf{A}\mathbf{x}_{ch} - \mathbf{b}\|_\infty,$$

using the fact that for all $\mathbf{z} \in \mathbb{R}^m$,

$$\frac{1}{\sqrt{m}} \|\mathbf{z}\|_2 \leq \|\mathbf{z}\|_\infty \leq \|\mathbf{z}\|_2.$$

(b) In Example 508 we derived a dual for the general norm approximation problem.

Applying the results to the ℓ_∞ -norm (and its dual norm, the ℓ_1 -norm), we can state the following dual for the Chebyshev approximation problem:

$$\begin{aligned} & \max_{\mathbf{v}} \mathbf{b}^T \mathbf{v} \\ & \text{s.t. } \|\mathbf{v}\|_1 \leq 1, \\ & \quad \mathbf{A}^T \mathbf{v} = \mathbf{0}. \end{aligned} \tag{6.90}$$

Any feasible \mathbf{v} corresponds to a lower bound $\mathbf{b}^T \mathbf{v}$ on $\|\mathbf{A}\mathbf{x}_{ch} - \mathbf{b}\|_\infty$.

Denote the least-squares residual as $\mathbf{r}_{ls} = \mathbf{b} - \mathbf{A}\mathbf{x}_{ls}$. Assuming $\mathbf{r}_{ls} \neq \mathbf{0}$, show that

$$\hat{\mathbf{v}} = -\mathbf{r}_{ls}/\|\mathbf{r}_{ls}\|_1, \quad \tilde{\mathbf{v}} = \mathbf{r}_{ls}/\|\mathbf{r}_{ls}\|_1,$$

are both feasible in (6.90). By duality $\mathbf{b}^T \hat{\mathbf{v}}$ and $\mathbf{b}^T \tilde{\mathbf{v}}$ are lower bounds on $\|\mathbf{A}\mathbf{x}_{ch} - \mathbf{b}\|_\infty$. Which is the better bound? How do these bounds compare with the bound derived in part (a)?

Exercise 519 (Piecewise-linear minimization). We consider the convex piecewise-linear minimization problem

$$\min_{\mathbf{x}} \max_{i=1,\dots,m} (\mathbf{a}_i^T \mathbf{x} + b_i) \tag{6.91}$$

with variable $\mathbf{x} \in \mathbb{R}^n$.

(a) Derive a dual problem, based on the Lagrange dual of the equivalent problem

$$\begin{aligned} & \min_{\mathbf{x}} \max_{i=1,\dots,m} y_i \\ & \text{s.t. } \mathbf{a}_i^T \mathbf{x} + b_i = y_i, \quad i = 1, \dots, m, \end{aligned}$$

with variables $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$.

(b) Formulate the piecewise-linear minimization problem (6.91) as an LP, and form the dual of the LP. Relate the LP dual to the dual obtained in part (a).

(c) Suppose we approximate the objective function in (6.91) by the smooth function

$$f_0(\mathbf{x}) = \log \left(\sum_{i=1}^m \exp(\mathbf{a}_i^T \mathbf{x} + b_i) \right),$$

and solve the unconstrained geometric program

$$\min_{\mathbf{x}} \log \left(\sum_{i=1}^m \exp(\mathbf{a}_i^T \mathbf{x} + b_i) \right). \quad (6.92)$$

A dual of this problem is given by (6.61). Let p_{pwl}^* and p_{gp}^* be the optimal values of (6.91) and (6.92), respectively. Show that

$$0 \leq p_{gp}^* - p_{pwl}^* \leq \log m.$$

(d) Derive similar bounds for the difference between p_{pwl}^* and the optimal value of

$$\min_{\mathbf{x}} (1/\gamma) \log \left(\sum_{i=1}^m \exp(\gamma(\mathbf{a}_i^T \mathbf{x} + b_i)) \right),$$

where $\gamma > 0$ is a parameter. What happens as we increase γ ?

Exercise 520 (Suboptimality of a simple covering ellipsoid). Recall the problem of determining the minimum volume ellipsoid, centered at the origin, that contains the points $\mathbf{a}_1, \dots, \mathbf{a}_m \in \mathbb{R}^n$:

$$\begin{aligned} \min_{\mathbf{X}} f_0(\mathbf{X}) &= \log \det(\mathbf{X}^{-1}) \\ \text{s.t. } \mathbf{a}_i^T \mathbf{X} \mathbf{a}_i &\leq 1, \quad i = 1, \dots, m, \end{aligned}$$

with $\text{dom} f_0 = \mathbb{S}_{++}^n$. We assume that the vectors $\mathbf{a}_1, \dots, \mathbf{a}_m$ span \mathbb{R}^n (which implies that the problem is bounded below).

(a) Show that the matrix

$$\mathbf{X}_{sim} = \left(\sum_{k=1}^m \mathbf{a}_k \mathbf{a}_k^T \right)^{-1},$$

is feasible. Hint. Show that

$$\begin{pmatrix} \sum_{k=1}^m \mathbf{a}_k \mathbf{a}_k^T & \mathbf{a}_i \\ \mathbf{a}_i^T & 1 \end{pmatrix} \succeq \mathbf{0},$$

and use Schur complements (Section 2.11.7.5) to prove that $\mathbf{a}_i^T \mathbf{X} \mathbf{a}_i \leq 1$ for $i = 1, \dots, m$.

(b) Now we establish a bound on how suboptimal the feasible point \mathbf{X}_{sim} is, via the dual problem

$$\begin{aligned} \max \quad & \log \det \left(\sum_{i=1}^m \lambda_i \mathbf{a}_i \mathbf{a}_i^T \right) - \mathbf{1}^T \lambda + n \\ \text{s.t. } & \boldsymbol{\lambda} \geq \mathbf{0}, \end{aligned}$$

with the implicit constraint $\sum_{i=1}^m \lambda_i \mathbf{a}_i \mathbf{a}_i^T \succ \mathbf{0}$. (This dual is derived below (6.25).) To derive a bound, we restrict our attention to dual variables of the form $\boldsymbol{\lambda} = t \mathbf{1}$, where $t > 0$. Find (analytically) the optimal value of t , and evaluate the dual objective at this $\boldsymbol{\lambda}$. Use this to prove that the volume of the ellipsoid $\{\mathbf{u} | \mathbf{u}^T \mathbf{X}_{sim} \mathbf{u} \leq 1\}$ is no more than a factor $(m/n)^{n/2}$ more than the volume of the minimum volume ellipsoid.

Exercise 521 (Optimal experiment design). The following problems arise in experiment design.

(a) D-optimal design.

$$\begin{aligned} \min_{\mathbf{x}} \quad & \log \det \left(\sum_{i=1}^p x_i \mathbf{v}_i \mathbf{v}_i^T \right)^{-1} \\ \text{s.t. } & \mathbf{x} \geq \mathbf{0}, \mathbf{1}^T \mathbf{x} = 1. \end{aligned}$$

(b) A-optimal design.

$$\begin{aligned} \min_{\mathbf{x}} \quad & \text{tr} \left(\sum_{i=1}^p x_i \mathbf{v}_i \mathbf{v}_i^T \right)^{-1} \\ \text{s.t. } & \mathbf{x} \geq \mathbf{0}, \mathbf{1}^T \mathbf{x} = 1. \end{aligned}$$

The domain of both problems is $\left\{ \mathbf{x} \mid \sum_{i=1}^p x_i \mathbf{v}_i \mathbf{v}_i^T \succ \mathbf{0} \right\}$. The variable is $\mathbf{x} \in \mathbb{R}^p$; the vectors $\mathbf{v}_1, \dots, \mathbf{v}_p \in \mathbb{R}^n$ are given.

Derive dual problems by first introducing a new variable $\mathbf{X} \in \mathbb{S}^n$ and an equality constraint $\mathbf{X} = \sum_{i=1}^p x_i \mathbf{v}_i \mathbf{v}_i^T$, and then applying Lagrange duality. Simplify the dual problems as much as you can.

Exercise 522. Derive a dual problem for

$$\min_{\mathbf{x}} \sum_{i=1}^N \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\|_2 + (1/2) \|\mathbf{x} - \mathbf{x}_0\|_2^2.$$

The problem data are $\mathbf{A}_i \in \mathbb{R}^{m_i \times n}$, $\mathbf{b}_i \in \mathbb{R}^{m_i}$, and $\mathbf{x}_0 \in \mathbb{R}^n$. First introduce new variables $\mathbf{y}_i \in \mathbb{R}^{m_i}$ and equality constraints $\mathbf{y}_i = \mathbf{A}_i \mathbf{x} + \mathbf{b}_i$.

Exercise 523 (Analytic centering). Derive a dual problem for

$$\min_{\mathbf{x}} - \sum_{i=1}^m \log(b_i - \mathbf{a}_i^T \mathbf{x})$$

with domain $\{\mathbf{x} | \mathbf{a}_i^T \mathbf{x} < b_i, i = 1, \dots, m\}$. First introduce new variables y_i and equality constraints $y_i = b_i - \mathbf{a}_i^T \mathbf{x}$.

The solution of this problem is called the analytic center of the linear inequalities $\mathbf{a}_i^T \mathbf{x} \leq b_i, i = 1, \dots, m$. Analytic centers have geometric applications and play an important role in barrier methods.

Exercise 524 (Lagrangian relaxation of Boolean LP). A Boolean linear program is an optimization problem of the form

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t. } \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & x_i \in \{0, 1\}, \quad i = 1, \dots, n, \end{aligned}$$

and is, in general, very difficult to solve. In Exercise 4.15 of [18] we studied the LP relaxation of this problem,

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t. } \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & 0 \leq x_i \leq 1, \quad i = 1, \dots, n, \end{aligned} \tag{6.93}$$

which is far easier to solve, and gives a lower bound on the optimal value of the Boolean LP. In this problem we derive another lower bound for the Boolean LP, and work out the relation between the two lower bounds.

(a) Lagrangian relaxation. The Boolean LP can be reformulated as the problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t. } \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & x_i(1 - x_i) = 0, \quad i = 1, \dots, n, \end{aligned}$$

which has quadratic equality constraints. Find the Lagrange dual of this problem. The optimal value of the dual problem (which is convex) gives a lower bound on the optimal value of the Boolean LP. This method of finding a lower bound on the optimal value is called Lagrangian relaxation.

(b) Show that the lower bound obtained via Lagrangian relaxation, and via the LP relaxation (6.93), are the same. Hint. Derive the dual of the LP relaxation (6.93).

Exercise 525 (A penalty method for equality constraints). We consider the problem

$$\begin{aligned} \min_{\mathbf{x}} & f_0(\mathbf{x}) \\ \text{s.t. } & \mathbf{Ax} = \mathbf{b}, \end{aligned} \tag{6.94}$$

where $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex and differentiable, and $\mathbf{A} \in \mathbb{R}^{m \times n}$ with $\text{rank}(\mathbf{A}) = m$.

In a quadratic penalty method, we form an auxiliary function

$$\phi(\mathbf{x}) = f(\mathbf{x}) + \alpha \|\mathbf{Ax} - \mathbf{b}\|_2^2,$$

where $\alpha > 0$ is a parameter. This auxiliary function consists of the objective plus the penalty term $\|\mathbf{Ax} - \mathbf{b}\|_2^2$. The idea is that a minimizer of the auxiliary function, $\tilde{\mathbf{x}}$, should be an approximate solution of the original problem. Intuition suggests that the larger the penalty weight α , the better the approximation $\tilde{\mathbf{x}}$ to a solution of the original problem. Suppose $\tilde{\mathbf{x}}$ is a minimizer of ϕ . Show how to find, from $\tilde{\mathbf{x}}$, a dual feasible point for (6.94). Find the corresponding lower bound on the optimal value of (6.94).

Exercise 526. Consider the problem

$$\begin{aligned} \min_{\mathbf{x}} & f_0(\mathbf{x}) \\ \text{s.t. } & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m, \end{aligned} \tag{6.95}$$

where the functions $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ are differentiable and convex. Let $h_1, \dots, h_m : \mathbb{R} \rightarrow \mathbb{R}$ be increasing differentiable convex functions. Show that

$$\phi(\mathbf{x}) = f_0(\mathbf{x}) + \sum_{i=1}^m h_i(f_i(\mathbf{x}))$$

is convex. Suppose $\tilde{\mathbf{x}}$ minimizes ϕ . Show how to find from $\tilde{\mathbf{x}}$ a feasible point for the dual of (6.95). Find the corresponding lower bound on the optimal value of (6.95).

Exercise 527 (An exact penalty method for inequality constraints). Consider the problem (6.95). In an exact penalty method, we solve the auxiliary problem

$$\min_{\mathbf{x}} \phi(\mathbf{x}) = f_0(\mathbf{x}) + \alpha \max_{i=1, \dots, m} \max\{0, f_i(\mathbf{x})\}, \tag{6.96}$$

where $\alpha > 0$ is a parameter. The second term in ϕ penalizes deviations of \mathbf{x} from feasibility. The method is called an exact penalty method if for sufficiently large α , solutions of the auxiliary problem (6.96) also solve the original problem (6.95).

(a) Show that ϕ is convex.

(b) The auxiliary problem can be expressed as

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_0(\mathbf{x}) + \alpha \mathbf{y} \\ \text{s.t. } \quad & f_i(\mathbf{x}) \leq y, \quad i = 1, \dots, m, \\ & 0 \leq y, \end{aligned}$$

where the variables are \mathbf{x} and $y \in \mathbb{R}$. Find the Lagrange dual of this problem, and express it in terms of the Lagrange dual function g of (6.95).

(c) Use the result in (b) to prove the following property. Suppose $\boldsymbol{\lambda}^*$ is an optimal solution of the Lagrange dual of (6.95), and that strong duality holds. If $\alpha > \mathbf{1}^T \boldsymbol{\lambda}^*$, then any solution of the auxiliary problem (6.96) is also an optimal solution of (6.95).

Exercise 528 (Robust linear programming with polyhedral uncertainty). Consider the robust LP

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t. } \quad & \sup_{\mathbf{a} \in \mathcal{P}_i} \mathbf{a}^T \mathbf{x} \leq b_i, \quad i = 1, \dots, m, \end{aligned}$$

with variable $\mathbf{x} \in \mathbb{R}^n$, where $\mathcal{P}_i = \{\mathbf{a} | \mathbf{C}_i \mathbf{a} \leq \mathbf{d}_i\}$. The problem data are $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{C}_i \in \mathbb{R}^{m_i \times n}$, $\mathbf{d}_i \in \mathbb{R}^{m_i}$, and $\mathbf{b} \in \mathbb{R}^m$. We assume the polyhedra \mathcal{P}_i are nonempty. Show that this problem is equivalent to the LP

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t. } \quad & \mathbf{d}_i^T \mathbf{z}_i \leq b_i, \quad i = 1, \dots, m \\ & \mathbf{C}_i^T \mathbf{z}_i = \mathbf{x}, \quad i = 1, \dots, m \\ & \mathbf{z}_i \geq \mathbf{0}, \quad i = 1, \dots, m, \end{aligned}$$

with variables $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{z}_i \in \mathbb{R}^{m_i}$, $i = 1, \dots, m$. Hint. Find the dual of the problem of maximizing $\mathbf{a}_i^T \mathbf{x}$ over $\mathbf{a}_i \in \mathcal{P}_i$ (with variable \mathbf{a}_i).

Exercise 529 (Separating hyperplane between two polyhedra). Formulate the following problem as an LP or an LP feasibility problem. Find a separating hyperplane that strictly separates two polyhedra

$$\mathcal{P}_1 = \{\mathbf{x} | \mathbf{A}\mathbf{x} \leq \mathbf{b}\}, \quad \mathcal{P}_2 = \{\mathbf{x} | \mathbf{C}\mathbf{x} \leq \mathbf{d}\},$$

i.e., find a vector $\mathbf{a} \in \mathbb{R}^n$ and a scalar γ such that

$$\mathbf{a}^T \mathbf{x} > \gamma \text{ for } \mathbf{x} \in \mathcal{P}_1, \quad \mathbf{a}^T \mathbf{x} < \gamma \text{ for } \mathbf{x} \in \mathcal{P}_2.$$

You can assume that \mathcal{P}_1 and \mathcal{P}_2 do not intersect. Hint. The vector \mathbf{a} and scalar γ must satisfy

$$\inf_{\mathbf{x} \in \mathcal{P}_1} \mathbf{a}^T \mathbf{x} > \gamma > \sup_{\mathbf{x} \in \mathcal{P}_2} \mathbf{a}^T \mathbf{x}.$$

Use LP duality to simplify the infimum and supremum in these conditions.

Exercise 530 (The sum of the largest elements of a vector). Define $f : \mathbb{R}^n \rightarrow \mathbb{R}$ as

$$f(\mathbf{x}) = \sum_{i=1}^r x_{[i]},$$

where r is an integer between 1 and n , and $x_{[1]} \geq x_{[2]} \geq \dots \geq x_{[r]}$ are the components of \mathbf{x} sorted in decreasing order. In other words, $f(\mathbf{x})$ is the sum of the r largest elements of \mathbf{x} . In this problem we study the constraint

$$f(\mathbf{x}) \leq \alpha.$$

As we have seen in Example 203, this is a convex constraint, and equivalent to a set of $n!/(r!(n-r)!)$ linear inequalities

$$x_{i_1} + \dots + x_{i_r} \leq \alpha, \quad 1 \leq i_1 < i_2 < \dots < i_r \leq n.$$

The purpose of this problem is to derive a more compact representation.

(a) Given a vector $\mathbf{x} \in \mathbb{R}^n$, show that $f(\mathbf{x})$ is equal to the optimal value of the LP

$$\begin{aligned} & \max_{\mathbf{y}} \mathbf{x}^T \mathbf{y} \\ & \text{s.t. } \mathbf{0} \leq \mathbf{y} \leq \mathbf{1} \\ & \quad \mathbf{1}^T \mathbf{y} = r, \end{aligned}$$

with $\mathbf{y} \in \mathbb{R}^n$ as variable.

(b) Derive the dual of the LP in part (a). Show that it can be written as

$$\begin{aligned} & \min_{\mathbf{u}, t} rt + \mathbf{1}^T \mathbf{u} \\ & \text{s.t. } t\mathbf{1} + \mathbf{u} \geq \mathbf{x} \\ & \quad \mathbf{u} \geq \mathbf{0}, \end{aligned}$$

where the variables are $t \in \mathbb{R}$, $\mathbf{u} \in \mathbb{R}^n$. By duality this LP has the same optimal value as the LP in (a), i.e., $f(\mathbf{x})$. We therefore have the following result: \mathbf{x} satisfies $f(\mathbf{x}) \leq \alpha$ if and only if there exist $t \in \mathbb{R}$, $\mathbf{u} \in \mathbb{R}^n$ such that

$$rt + \mathbf{1}^T \mathbf{u} \leq \alpha, \quad t\mathbf{1} + \mathbf{u} \geq \mathbf{x}, \quad \mathbf{u} \geq \mathbf{0}.$$

These conditions form a set of $2n+1$ linear inequalities in the $2n+1$ variables $\mathbf{x}, \mathbf{u}, t$.

(c) As an application, we consider an extension of the classical Markowitz portfolio optimization problem

$$\begin{aligned} & \min_{\mathbf{x}} \mathbf{x}^T \boldsymbol{\Sigma} \mathbf{x} \\ & \text{s.t. } \bar{\mathbf{p}}^T \mathbf{x} \geq r_{\min} \\ & \quad \mathbf{1}^T \mathbf{x} = 1, \mathbf{x} \geq \mathbf{0}, \end{aligned}$$

discussed in Chapter 4, page 155, of [18]. The variable is the portfolio $\mathbf{x} \in \mathbb{R}^n$; $\bar{\mathbf{p}}$ and $\boldsymbol{\Sigma}$ are the mean and covariance matrix of the price change vector \mathbf{p} .

Suppose we add a diversification constraint, requiring that no more than 80% of the total budget can be invested in any 10% of the assets. This constraint can be expressed as

$$\sum_{i=1}^{\lfloor 0.1n \rfloor} x_{[i]} \leq 0.8.$$

Formulate the portfolio optimization problem with diversification constraint as a QP.

Exercise 531 (Dual of channel capacity problem). Derive a dual for the problem

$$\begin{aligned} & \min_{\mathbf{x}} -\mathbf{c}^T \mathbf{x} + \sum_{i=1}^m y_i \log y_i \\ & \text{s.t. } \mathbf{P} \mathbf{x} = \mathbf{y} \\ & \quad \mathbf{x} \geq \mathbf{0}, \mathbf{1}^T \mathbf{x} = 1, \end{aligned}$$

where $\mathbf{P} \in \mathbb{R}^{m \times n}$ has nonnegative elements, and its columns add up to one (i.e., $\mathbf{P}^T \mathbf{1} = 1$). The variables are $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$. (For $c_j = \sum_{i=1}^m p_{ij} \log p_{ij}$, the optimal value is, up to a factor $\log 2$, the negative of the capacity of a discrete memoryless channel with channel transition probability matrix \mathbf{P} .)

Simplify the dual problem as much as possible.

Strong duality and Slater's condition

Exercise 532 (A convex problem in which strong duality fails). Consider the optimization problem

$$\begin{aligned} & \min_{x,y} e^{-x} \\ & \text{s.t. } x^2/y \leq 0 \end{aligned}$$

with variables x and y , and domain $\mathcal{D} = \{f(x, y) | y > 0\}$.

(a) Verify that this is a convex optimization problem. Find the optimal value.

- (b) Give the Lagrange dual problem, and find the optimal solution λ^* and optimal value d^* of the dual problem. What is the optimal duality gap?
- (c) Does Slater's condition hold for this problem?
- (d) What is the optimal value $p^*(u)$ of the perturbed problem

$$\min_u e^{-x}$$

$$s.t. x^2/y \leq u,$$

as a function of u ? Verify that the global sensitivity inequality

$$p^*(u) \geq p^*(0) - \lambda^* u$$

does not hold.

Exercise 533 (Geometric interpretation of duality). For each of the following optimization problems, draw a sketch of the sets

$$\mathcal{G} = \{(u, t) | \exists x \in \mathcal{D}, f_0(x) = t, f_1(x) = u\},$$

$$\mathcal{A} = \{(u, t) | \exists x \in \mathcal{D}, f_0(x) \leq t, f_1(x) \leq u\}.$$

give the dual problem, and solve the primal and dual problems. Is the problem convex? Is Slater's condition satisfied? Does strong duality hold? The domain of the problem is \mathbb{R} unless otherwise stated.

(a) Minimize x subject to $x^2 \leq 1$.

(b) Minimize x subject to $x^2 \leq 0$.

(c) Minimize x subject to $|x| \leq 0$.

(d) Minimize x subject to $f_1(x) \leq 0$ where

$$f_1(x) = \begin{cases} -x + 2, & x \geq 1, \\ x, & -1 \leq x \leq 1, \\ -x - 2, & x \leq -1. \end{cases}$$

(e) Minimize x^3 subject to $-x + 1 \leq 0$.

(f) Minimize x^3 subject to $-x + 1 \leq 0$ with domain $\mathcal{D} = \mathbb{R}_+$.

Exercise 534 (Strong duality in linear programming). We prove that strong duality holds for the LP

$$\begin{aligned} & \min_{\mathbf{x}} \mathbf{c}^T \mathbf{x} \\ & \text{s.t. } \mathbf{A}\mathbf{x} \leq \mathbf{b} \end{aligned}$$

and its dual

$$\begin{aligned} & \max_{\mathbf{z}} -\mathbf{b}^T \mathbf{z} \\ & \text{s.t. } \mathbf{A}^T \mathbf{z} + \mathbf{c} = \mathbf{0}, \mathbf{z} \geq \mathbf{0}, \end{aligned}$$

provided at least one of the problems is feasible. In other words, the only possible exception to strong duality occurs when $p^* = \infty$ and $d^* = -\infty$.

- (a) Suppose p^* is finite and \mathbf{x}^* is an optimal solution. (If finite, the optimal value of an LP is attained.) Let $\mathcal{I} \subseteq \{1, 2, \dots, m\}$ be the set of active constraints at \mathbf{x}^* :

$$\mathbf{a}_i^T \mathbf{x}^* = b_i, i \in \mathcal{I}, \quad \mathbf{a}_i^T \mathbf{x}^* < b_i, i \notin \mathcal{I}.$$

Show that there exists a $\mathbf{z} \in \mathbb{R}^m$ that satisfies

$$z_i \geq 0, i \in \mathcal{I}, \quad z_i = 0, i \notin \mathcal{I}, \quad \sum_{i \in \mathcal{I}} z_i \mathbf{a}_i + \mathbf{c} = \mathbf{0}.$$

Show that \mathbf{z} is dual optimal with objective value $\mathbf{c}^T \mathbf{x}^*$. Hint. Assume there exists no such \mathbf{z} , i.e., $-\mathbf{c} \notin \{\sum_{i \in \mathcal{I}} z_i \mathbf{a}_i \mid z_i \geq 0\}$. Reduce this to a contradiction by applying the strict separating hyperplane theorem of Example 128. Alternatively, you can use Farkas' lemma (see Section 6.6.7.3).

- (b) Suppose $p^* = \infty$ and the dual problem is feasible. Show that $d^* = \infty$. Hint. Show that there exists a nonzero $\mathbf{v} \in \mathbb{R}^m$ such that $\mathbf{A}^T \mathbf{v} = \mathbf{0}$, $\mathbf{v} \geq \mathbf{0}$, $\mathbf{b}^T \mathbf{v} < 0$. If the dual is feasible, it is unbounded in the direction \mathbf{v} .
- (c) Consider the example

$$\begin{aligned} & \min_x x \\ & \text{s.t. } \begin{pmatrix} 0 \\ 1 \end{pmatrix} x \preceq \begin{pmatrix} -1 \\ 1 \end{pmatrix}. \end{aligned}$$

Formulate the dual LP, and solve the primal and dual problems. Show that $p^* = 1$ and $d^* = -\infty$.

Exercise 535 (Weak max-min inequality). Show that the weak max-min inequality

$$\sup_{\mathbf{z} \in \mathcal{Z}} \inf_{\mathbf{w} \in \mathcal{W}} f(\mathbf{w}, \mathbf{z}) \leq \inf_{\mathbf{w} \in \mathcal{W}} \sup_{\mathbf{z} \in \mathcal{Z}} f(\mathbf{w}, \mathbf{z})$$

always holds, with no assumptions on $f : \mathbb{R}^n \rightarrow \mathbb{R}^m \rightarrow \mathbb{R}$, $\mathcal{W} \subseteq \mathbb{R}^n$, or $\mathcal{Z} \subseteq \mathbb{R}^m$.

Exercise 536 (Convex-concave functions and the saddle-point property). We derive conditions under which the saddle-point property

$$\sup_{\mathbf{z} \in \mathcal{Z}} \inf_{\mathbf{w} \in \mathcal{W}} f(\mathbf{w}, \mathbf{z}) = \inf_{\mathbf{w} \in \mathcal{W}} \sup_{\mathbf{z} \in \mathcal{Z}} f(\mathbf{w}, \mathbf{z}) \quad (6.97)$$

holds, where $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $\mathcal{W} \times \mathcal{Z} \subseteq \text{dom } f$, and \mathcal{W} and \mathcal{Z} are nonempty. We will assume that the function

$$g_z(\mathbf{w}) = \begin{cases} f(\mathbf{w}, \mathbf{z}), & \mathbf{w} \in \mathcal{W} \\ \infty, & \text{otherwise} \end{cases}$$

is closed and convex for all $\mathbf{z} \in \mathcal{Z}$, and the function

$$h_w(\mathbf{z}) = \begin{cases} -f(\mathbf{w}, \mathbf{z}), & \mathbf{z} \in \mathcal{Z} \\ \infty, & \text{otherwise} \end{cases}$$

is closed and convex for all $\mathbf{w} \in \mathcal{W}$.

(a) The righthand side of (6.97) can be expressed as $p(\mathbf{0})$, where

$$p(\mathbf{u}) = \inf_{\mathbf{w} \in \mathcal{W}} \sup_{\mathbf{z} \in \mathcal{Z}} (f(\mathbf{w}, \mathbf{z}) + \mathbf{u}^T \mathbf{z}).$$

Show that p is a convex function.

(b) Show that the conjugate of p is given by

$$p^*(v) = \begin{cases} -\inf_{\mathbf{w} \in \mathcal{W}} f(\mathbf{w}, \mathbf{v}), & \mathbf{v} \in \mathcal{Z} \\ \infty, & \text{otherwise.} \end{cases}$$

(c) Show that the conjugate of p^* is given by

$$p^{**}(\mathbf{u}) = \sup_{\mathbf{z} \in \mathcal{Z}} \inf_{\mathbf{w} \in \mathcal{W}} (f(\mathbf{w}, \mathbf{z}) + \mathbf{u}^T \mathbf{z}).$$

Combining this with (a), we can express the max-min equality (6.97) as $p^{**}(\mathbf{0}) = p(\mathbf{0})$.

(d) From Exercises 293 and 304(d), we know that $p^{**}(\mathbf{0}) = p(\mathbf{0})$ if $\mathbf{0} \in \text{int dom } p$. Conclude that this is the case if \mathcal{W} and \mathcal{Z} are bounded.

(e) As another consequence of Exercises 293 and 304, we have $p^{**}(\mathbf{0}) = p(\mathbf{0})$ if $\mathbf{0} \in \text{dom } p$ and p is closed. Show that p is closed if the sublevel sets of g_z are bounded.

Optimality conditions

Exercise 537. Consider the QCQP

$$\begin{aligned} \min_{x_1, x_2} & x_1^2 + x_2^2 \\ \text{s.t. } & (x_1 - 1)^2 + (x_2 - 1)^2 \leq 1 \\ & (x_1 - 1)^2 + (x_2 + 1)^2 \leq 1 \end{aligned}$$

with variable $\mathbf{x} \in \mathbb{R}^2$.

- (a) Sketch the feasible set and level sets of the objective. Find the optimal point \mathbf{x}^* and optimal value p^* .
- (b) Give the KKT conditions. Do there exist Lagrange multipliers λ_1^* and λ_2^* that prove that \mathbf{x}^* is optimal?
- (c) Derive and solve the Lagrange dual problem. Does strong duality hold?

Exercise 538 (Equality constrained least-squares). Consider the equality constrained least-squares problem

$$\begin{aligned} \min_{\mathbf{x}} & \|\mathbf{Ax} - \mathbf{b}\|_2^2, \\ \text{s.t. } & \mathbf{Gx} = \mathbf{h}, \end{aligned}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ with rank $\mathbf{A} = n$, and $\mathbf{G} \in \mathbb{R}^{p \times n}$ with rank $\mathbf{G} = p$. Give the KKT conditions, and derive expressions for the primal solution \mathbf{x}^* and the dual solution $\boldsymbol{\nu}^*$.

Exercise 539. Prove (without using any linear programming code) that the optimal solution of the LP

$$\begin{aligned} \min_{\mathbf{x}} & 47x_1 + 93x_2 + 17x_3 - 93x_4, \\ \text{s.t. } & \begin{pmatrix} -1 & -6 & 1 & 3 \\ -1 & -2 & 7 & 1 \\ 0 & 3 & -10 & -1 \\ -6 & -11 & -2 & 12 \\ 1 & 6 & -1 & -3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \leq \begin{pmatrix} -3 \\ 5 \\ -8 \\ -7 \\ 4 \end{pmatrix} \end{aligned}$$

is unique, and given by $\mathbf{x}^* = (1, 1, 1, 1)^T$.

Exercise 540. The problem

$$\begin{aligned} \min_{\mathbf{x}} & -3x_1^2 + x_2^2 + 2x_3^2 + 2(x_1 + x_2 + x_3) \\ \text{s.t. } & x_1^2 + x_2^2 + x_3^2 = 1, \end{aligned}$$

is a special case of (6.43), so strong duality holds even though the problem is not convex. Derive the KKT conditions. Find all solutions \mathbf{x} , $\boldsymbol{\nu}$ that satisfy the KKT conditions. Which pair corresponds to the optimum?

Exercise 541. Derive the KKT conditions for the problem

$$\begin{aligned} \min_{\mathbf{X}} \quad & \text{tr } \mathbf{X} - \log \det \mathbf{X} \\ \text{s.t. } & \mathbf{X}\mathbf{s} = \mathbf{y}, \end{aligned}$$

with variable $\mathbf{X} \in \mathcal{S}^n$ and domain \mathcal{S}_{++}^n . $\mathbf{y} \in \mathbb{R}^n$ and $\mathbf{s} \in \mathbb{R}^n$ are given, with $\mathbf{s}^T \mathbf{y} = 1$. Verify that the optimal solution is given by

$$\mathbf{X}^* = \mathbf{I} + \mathbf{y}\mathbf{y}^T - \frac{1}{\mathbf{s}^T \mathbf{s}} \mathbf{s}\mathbf{s}^T.$$

Exercise 542 (Supporting hyperplane interpretation of KKT conditions). Consider a convex problem with no equality constraints,

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_0(\mathbf{x}) \\ \text{s.t. } & f_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m. \end{aligned}$$

Assume that $\mathbf{x}^* \in \mathbb{R}^n$ and $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ satisfy the KKT conditions

$$\begin{aligned} f_i(\mathbf{x}^*) &\leq 0, \quad i = 1, \dots, m \\ \lambda_i^* &\geq 0, \quad i = 1, \dots, m \\ \lambda_i^* f_i(\mathbf{x}^*) &= 0, \quad i = 1, \dots, m \\ \nabla f_0(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(\mathbf{x}^*) &= 0. \end{aligned}$$

Show that

$$\nabla f_0(\mathbf{x}^*)^T (\mathbf{x} - \mathbf{x}^*) \geq 0$$

for all feasible \mathbf{x} . In other words the KKT conditions imply the simple optimality criterion of Chapter 4.2.3 of [18].

Theorems of alternatives

Exercise 543 (Alternatives for linear equalities). Consider the linear equations $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{m \times n}$. From linear algebra we know that this equation has a solution if and only if $\mathbf{b} \in \mathcal{R}(\mathbf{A})$, which occurs if and only if $\mathbf{b} \perp \mathcal{N}(\mathbf{A}^T)$. In other words, $\mathbf{Ax} = \mathbf{b}$ has a solution if and only if there exists no $\mathbf{y} \in \mathbb{R}^m$ such that $\mathbf{A}^T \mathbf{y} = \mathbf{0}$ and $\mathbf{b}^T \mathbf{y} \neq 0$. Derive this result from the theorems of alternatives in Section 6.6.7.2.

Exercise 544 (Existence of equilibrium distribution in finite state Markov chain). Let $\mathbf{P} \in \mathbb{R}^{n \times n}$ be a matrix that satisfies

$$p_{ij} \geq 0, \quad i, j = 1, \dots, n, \quad \mathbf{P}^T \mathbf{1} = \mathbf{1},$$

i.e., the coefficients are nonnegative and the columns sum to one. Use Farkas' lemma to prove there exists a $\mathbf{y} \in \mathbb{R}^n$ such that

$$\mathbf{P}\mathbf{y} = \mathbf{y}, \mathbf{y} \geq \mathbf{0}, \mathbf{1}^T \mathbf{y} = 1.$$

(We can interpret \mathbf{y} as an equilibrium distribution of the Markov chain with n states and transition probability matrix \mathbf{P} .)

Exercise 545 (Option pricing). We apply the results of Example 512, to a simple problem with three assets: a riskless asset with fixed return $r > 1$ over the investment period of interest (for example, a bond), a stock, and an option on the stock. The option gives us the right to purchase the stock at the end of the period, for a predetermined price K . We consider two scenarios.

In the first scenario, the price of the stock goes up from S at the beginning of the period, to S_u at the end of the period, where $u > r$. In this scenario, we exercise the option only if $S_u > K$, in which case we make a profit of $S_u - K$. Otherwise, we do not exercise the option, and make zero profit. The value of the option at the end of the period, in the first scenario, is therefore $\max\{0, S_u - K\}$.

In the second scenario, the price of the stock goes down from S to S_d , where $d < 1$. The value at the end of the period is $\max\{0, S_d - K\}$.

In the notation of Example 512,

$$V = \begin{pmatrix} r & S_u & \max\{0, S_u - K\} \\ r & S_d & \max\{0, S_d - K\} \end{pmatrix}, \quad p_1 = 1, \quad p_2 = S, \quad p_3 = C,$$

where C is the price of the option.

Show that for given r, S, K, u, d , the option price C is uniquely determined by the no-arbitrage condition. In other words, the market for the option is complete.

6.7 Problems With Equality Constraints

(Taken from Chapter 20 of [30])

6.7.1 Introduction

In this part we discuss methods for solving a class of nonlinear constrained optimization problems that can be formulated as

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && h_i(\mathbf{x}) = 0, \quad i = i, \dots, m \\ & && g_j(\mathbf{x}) \leq 0, \quad j = 1, \dots, p, \end{aligned}$$

where $\mathbf{x} \in \mathbb{R}^n$, $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $g_j : \mathbb{R}^n \rightarrow \mathbb{R}$, and $m \leq n$. In vector notation, the problem above can be represented in the following standard form:

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & && \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \end{aligned}$$

where $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $g : \mathbb{R}^n \rightarrow \mathbb{R}^p$. As usual, we adopt the following terminology.

Definition 546. Any point satisfying the constraints is called a feasible point. The set of all feasible points,

$$\{\mathbf{x} \in \mathbb{R}^n : \mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}$$

is called a feasible set.

Optimization problems of the above form are not new to us. Indeed, linear programming problems of the form

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} \\ & \text{subject to} && \mathbf{Ax} = \mathbf{b} \\ & && \mathbf{x} \geq \mathbf{0}, \end{aligned}$$

are of this type. There is no loss of generality by considering only minimization problems. For if we are confronted with a maximization problem, it can easily be transformed into the minimization problem by observing that $\text{maximize } f(\mathbf{x}) = \text{minimize } -f(\mathbf{x})$.

We illustrate the problems we study in this part by considering the following simple numerical example.

Example 547. Consider the following optimization problem:

$$\begin{aligned} & \text{minimize} && (x_1 - 1)^2 + x_2 - 2 \\ & \text{subject to} && x_2 - x_1 = 1, \\ & && x_1 + x_2 \leq 2. \end{aligned}$$

This problem is already in the standard form given earlier, with $f(x_1, x_2) = (x_1 - 1)^2 + x_2 - 2$, $h(x_1, x_2) = x_2 - x_1 - 1$, and $g(x_1, x_2) = x_1 + x_2 - 2$. This problem turns out to be simple enough to be solved graphically (see 6.10). In the figure the set of points that satisfy the constraints (the feasible set) is marked by the heavy solid line. The inverted parabolas represent level sets of the objective function — the lower the level set, the smaller the objective function value. Therefore, the solution can be obtained by finding the lowest-level set that intersects the feasible set. In this case, the minimizer lies on the level set with $f = -1/4$. The minimizer of the objective function is $x^* = [1/2, 3/2]^T$.

In the remainder of this chapter we discuss constrained optimization problems with only equality constraints. The general constrained optimization problem is discussed in the chapters to follow.



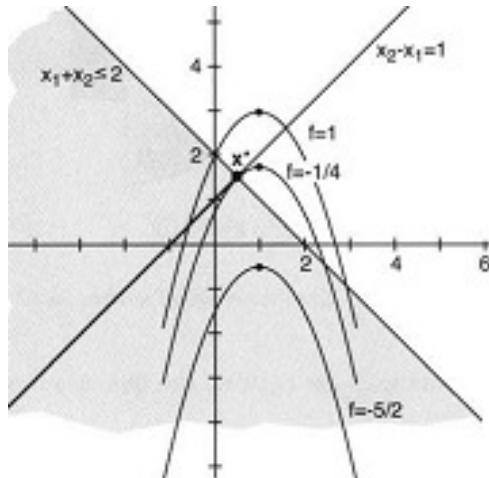


图 6.10: Graphical solution to the problem in Example 547.

6.7.2 Problem Formulation

The class of optimization problems we analyze in this chapter is

$$\begin{aligned} & \text{minimize } f(\mathbf{x}) \\ & \text{subject to } \mathbf{h}(\mathbf{x}) = \mathbf{0}, \end{aligned}$$

where $\mathbf{x} \in \mathbb{R}^n$, $: \mathbb{R}^n \rightarrow \mathbb{R}$, $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $\mathbf{h} = [h_1, \dots, h_m]^T$, and $m \leq n$. We assume that the function \mathbf{h} is continuously differentiable, that is, $\mathbf{h} \in C^1$. We introduce the following definition.

Definition 548. A point x^* satisfying the constraints $h_1(x^*) = 0, \dots, h_m(x^*) = 0$ is said to be a regular point of the constraints if the gradient vectors $\nabla h_1(x^*), \dots, \nabla h_m(x^*)$ are linearly independent.

Let $\mathcal{D}\mathbf{h}(\mathbf{x}^*)$ be the Jacobian matrix of $\mathbf{h} = [h_1, \dots, h_m]^T$ at \mathbf{x}^* , given by

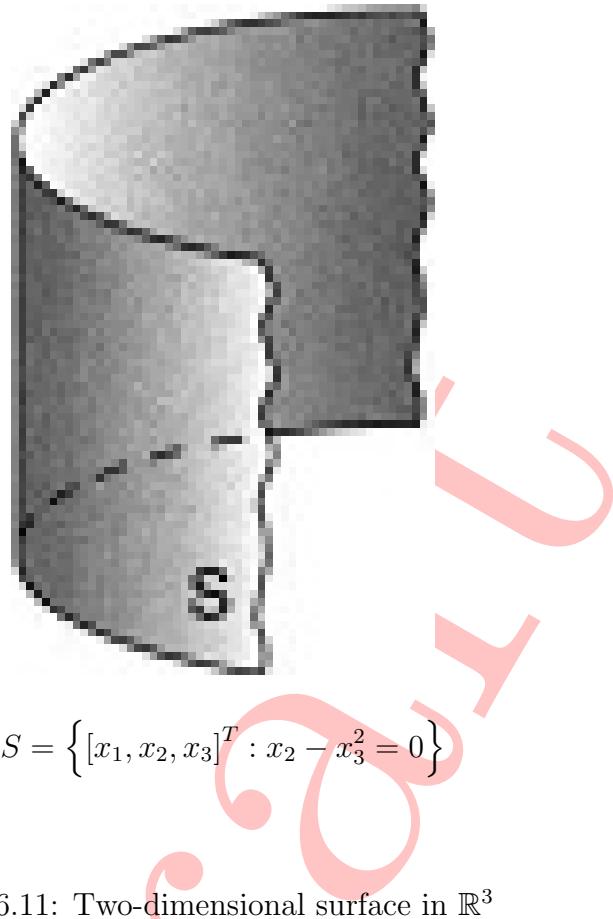
$$\mathcal{D}\mathbf{h}(\mathbf{x}^*) = \begin{bmatrix} \mathcal{D}h_1(\mathbf{x}^*) \\ \vdots \\ \mathcal{D}h_m(\mathbf{x}^*) \end{bmatrix} = \begin{bmatrix} \nabla h_1(\mathbf{x}^*) \\ \vdots \\ \nabla h_m(\mathbf{x}^*) \end{bmatrix}$$

Then, \mathbf{x}^* is regular if and only if $\text{rank}(\mathcal{D}\mathbf{h}(\mathbf{x}^*)) = m$ (i.e., the Jacobian matrix is of full rank).

The set of equality constraints $h_1(\mathbf{x}) = 0, \dots, h_m(\mathbf{x})$, describes a surface

$$S = \{\mathbf{x} \in \mathbb{R}^n : h_1(\mathbf{x}) = 0, \dots, h_m(\mathbf{x}) = 0\}.$$

Assuming that the points in S are regular, the dimension of the surface S is $n - m$.


 图 6.11: Two-dimensional surface in \mathbb{R}^3

Example 549. Let $n = 3$ and $m = 1$ (i.e., we are operating in \mathbb{R}^3). Assuming that all points in S are regular, the set S is a two-dimensional surface. For example, let

$$h_i(\mathbf{x}) = x_2 - x_3^2 = 0.$$

Note that $\nabla h_i(\mathbf{x}) = [0, 1, -2x_3]^T$, and hence for any $\mathbf{x} \in \mathbb{R}^3$, $\nabla h_i(\mathbf{x}) = 0$. In this case,

$$\dim S = \dim \{\mathbf{x} : h_i(\mathbf{x}) = 0\} = n - m = 2.$$

See Figure 6.11 for a graphical illustration.

Example 550. Let $n = 3$ and $m = 2$. Assuming regularity, the feasible set S is a one-dimensional object (i.e., a curve in \mathbb{R}^3). For example, let

$$h_1(\mathbf{x}) = x_1,$$

$$h_2(\mathbf{x}) = x_2 - x_3^2.$$

In this case, $\nabla h_1(\mathbf{x}) = [1, 0, 0]^T$ and $\nabla h_2(\mathbf{x}) = [0, 1, -2x_3]^T$. Hence, the vectors $\nabla h_1(\mathbf{x})$ and $\nabla h_2(\mathbf{x})$ are linearly independent in \mathbb{R}^3 . Thus,

$$\dim S = \dim \{\mathbf{x} : h_1(\mathbf{x}) = 0, h_2(\mathbf{x}) = 0\} = n - m = 1.$$

See Figure 6.12 for a graphical illustration.

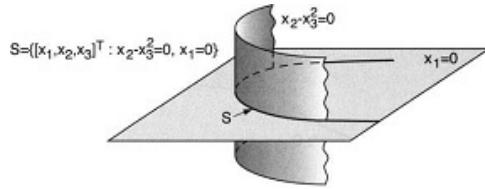
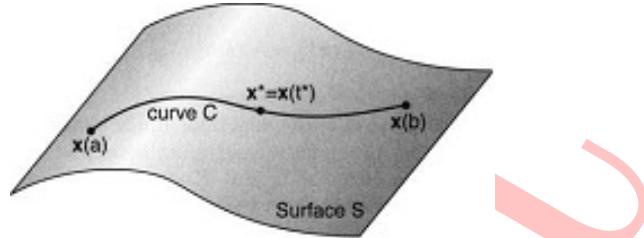

 图 6.12: One-dimensional surface in \mathbb{R}^3


图 6.13: Curve on a surface

6.7.3 Tangent and Normal Spaces

In this section we discuss the notion of a tangent space and normal space at a point on a surface. We begin by defining a curve on a surface S .

Definition 551. A curve C on a surface S is a set of points $\{\mathbf{x}(t) \in S : t \in (a, b)\}$, continuously parameterized by $t \in (a, b)$; that is, $\mathbf{x} : (a, b) \rightarrow S$ is a continuous function.

A graphical illustration of the definition of a curve is given in Figure 6.13. The definition of a curve implies that all the points on the curve satisfy the equation describing the surface. The curve C passes through a point \mathbf{x}^* if there exists $t^* \in (a, b)$ such that $\mathbf{x}(t^*) = \mathbf{x}^*$.

Intuitively, we can think of a curve $C = \{\mathbf{x}(t) : t \in (a, b)\}$ as the path traversed by a point \mathbf{x} traveling on the surface S . The position of the point at time t is given by $\mathbf{x}(t)$.

Definition 552. The curve $C = \{\mathbf{x}(t) : t \in (a, b)\}$ is differentiable if

$$\dot{\mathbf{x}}(t) = \frac{d\mathbf{x}}{dt}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \vdots \\ \dot{x}_n(t) \end{bmatrix}$$

exists for all $t \in (a, b)$. The curve $C = \{\mathbf{x}(t) : t \in (a, b)\}$ is twice differentiable if

$$\ddot{\mathbf{x}}(t) = \frac{d^2\mathbf{x}}{dt^2}(t) = \begin{bmatrix} \ddot{x}_1(t) \\ \vdots \\ \ddot{x}_n(t) \end{bmatrix}$$

exists for all $t \in (a, b)$.

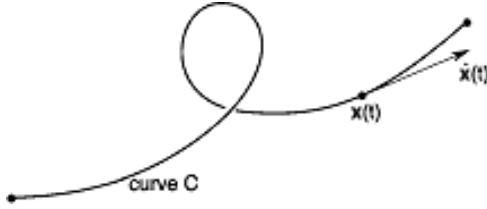


图 6.14: Geometric interpretation of the differentiability of a curve

Note that both $\dot{\mathbf{x}}(t)$ and $\ddot{\mathbf{x}}(t)$ are n -dimensional vectors. We can think of $\dot{\mathbf{x}}(t)$ and $\ddot{\mathbf{x}}(t)$ as the velocity and acceleration, respectively, of a point traversing the curve C with position $\dot{\mathbf{x}}(t)$ at time t . The vector $\dot{\mathbf{x}}(t)$ points in the direction of the instantaneous motion of $\dot{\mathbf{x}}(t)$. Therefore, the vector $\dot{\mathbf{x}}(t^*)$ is tangent to the curve C at \mathbf{x}^* (see Figure 6.14).

We are now ready to introduce the notions of a tangent space. For this recall the set

$$S = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{h}(\mathbf{x}) = \mathbf{0}\},$$

where $\mathbf{h} \in \mathcal{C}^1$. We think of S as a surface in \mathbb{R}^n .

Definition 553. *The tangent space at a point \mathbf{x}^* on the surface $S = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{h}(\mathbf{x}) = \mathbf{0}\}$ is the set $T(\mathbf{x}^*) = \{y : D\mathbf{h}(\mathbf{x}^*)y = \mathbf{0}\}$.*

Note that the tangent space $T(\mathbf{x}^*)$ is the nullspace of the matrix $D\mathbf{h}(\mathbf{x}^*)$:

$$T(\mathbf{x}^*) = \mathcal{N}(D\mathbf{h}(\mathbf{x}^*)).$$

The tangent space is therefore a subspace of R^n .

Assuming that \mathbf{x}^* is regular, the dimension of the tangent space is $n - m$, where m is the number of equality constraints $h_i(\mathbf{x}^*) = 0$. Note that the tangent space passes through the origin. However, it is often convenient to picture the tangent space as a plane that passes through the point \mathbf{x}^* . For this, we define the tangent plane at \mathbf{x}^* to be the set

$$TP(\mathbf{x}^*) = T(\mathbf{x}^*) + \mathbf{x}^* = \{\mathbf{x} + \mathbf{x}^* : \mathbf{x} \in T(\mathbf{x}^*)\}.$$

Figure 6.15 illustrates the notion of a tangent plane, and Figure 6.16, the relationship between the tangent plane and the tangent space.

Example 554. Let

$$S = \{\mathbf{x} \in \mathbb{R}^3 : h_1(\mathbf{x}) = x_1 = 0, h_2(\mathbf{x}) = x_1 - x_2 = 0\}.$$

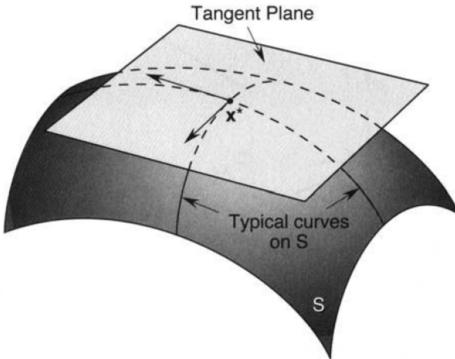


图 6.15: Tangent plane to the surface S at the point \mathbf{x}^*

Then, S is the x_3 -axis in \mathbb{R}^3 (see Figure 6.17). We have

$$D\mathbf{h}(\mathbf{x}) = \begin{bmatrix} \nabla h_1(\mathbf{x})^T \\ \nabla h_2(\mathbf{x})^T \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & -1 & 0 \end{bmatrix}.$$

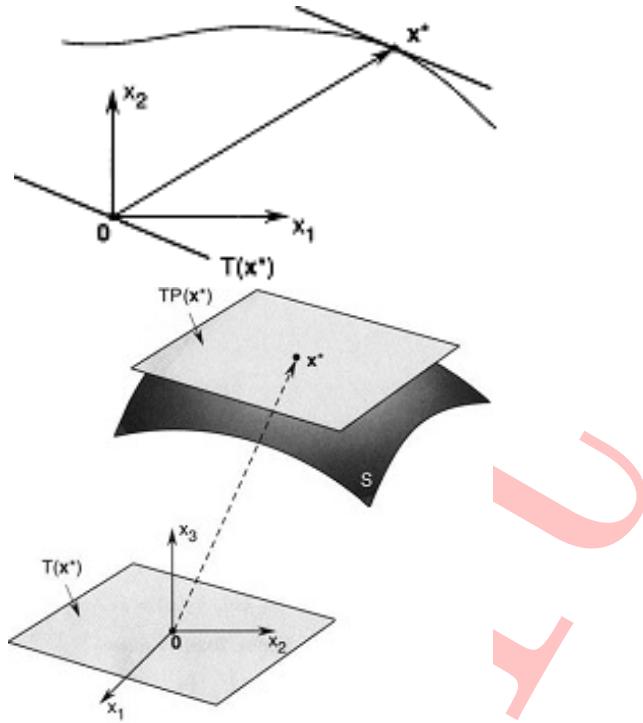
Because ∇h_1 , ∇h_2 are linearly independent when evaluated at any $\mathbf{x} \in S$, all the points of S are regular. The tangent space at an arbitrary point of S is

$$\begin{aligned} T(\mathbf{x}) &= \left\{ \mathbf{y} : \nabla h_1(\mathbf{x})^T \mathbf{y} = 0, \nabla h_2(\mathbf{x})^T \mathbf{y} = 0 \right\} \\ &= \left\{ \mathbf{y} : \begin{bmatrix} 1 & 1 & 0 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \mathbf{0} \right\} \\ &= \left\{ [0, 0, \alpha]^T : \alpha \in \mathbb{R} \right\} \\ &= \text{the } x_3\text{-axis in } \mathbb{R}^3. \end{aligned}$$

In this example, the tangent space $T(\mathbf{x})$ at any point $\mathbf{x} \in S$ is a one-dimensional subspace of \mathbb{R}^3 .

Intuitively, we would expect the definition of the tangent space at a point on a surface to be the collection of all "tangent vectors" to the surface at that point. We have seen that the derivative of a curve on a surface at a point is a tangent vector to the curve, and hence to the surface. The intuition above agrees with our definition whenever \mathbf{x}^* is regular, as stated in the theorem below.

Theorem 555. Suppose that $\mathbf{x}^* \in S$ is a regular point and $T(\mathbf{x}^*)$ is the tangent space at \mathbf{x}^* . Then, $\mathbf{y} \in T(\mathbf{x}^*)$ if and only if there exists a differentiable curve in S passing through \mathbf{x}^* with derivative \mathbf{y} at \mathbf{x}^* .


 图 6.16: Tangent spaces and planes in \mathbb{R}^2 and \mathbb{R}^3

Proof. \Leftarrow : Suppose that there exists a curve $\{\mathbf{x}(t) : t \in (a, b)\}$ in S such that $\mathbf{x}(t^*) = \mathbf{x}^*$ and $\mathbf{x}(t^*) = \mathbf{y}$ for some $t^* \in (a, b)$. Then,

$$\mathbf{h}(\mathbf{x}(t)) = \mathbf{0}$$

for all $t \in (a, b)$. If we differentiate the function $\mathbf{h}(\mathbf{x}(t))$ using the chain rule, we obtain

$$\frac{d}{dt} \mathbf{h}(\mathbf{x}(t)) = D\mathbf{h}(\mathbf{x}(t)) \dot{\mathbf{x}}(t) = \mathbf{0}$$

for all $t \in (a, b)$. Therefore, at t^* we get

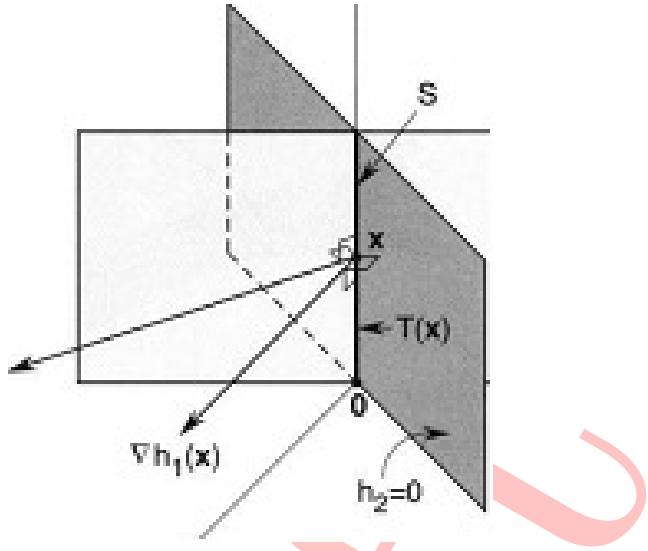
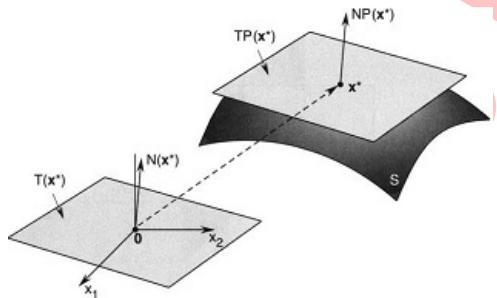
$$D\mathbf{h}(\mathbf{x}^*) \mathbf{y} = \mathbf{0},$$

and hence $\mathbf{y} \in T(\mathbf{x}^*)$.

\Rightarrow : To prove this, we need to use the implicit function theorem. We refer the reader to page 88 of [94]. \square

We now introduce the notion of a normal space.

Definition 556. *The normal space $N(\mathbf{x}^*)$ at a point \mathbf{x}^* on the surface $S = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{h}(\mathbf{x}) = \mathbf{0}\}$ is the set $N(\mathbf{x}^*) = \left\{ \mathbf{x} \in \mathbb{R}^n : \mathbf{x} = D\mathbf{h}(\mathbf{x}^*)^T \mathbf{z}, \mathbf{z} \in \mathbb{R}^m \right\}$.*


 图 6.17: The surface $S = \{\mathbf{x} \in \mathbb{R}^3 : x_1 = 0, x_1 - x_2 = 0\}$

 图 6.18: Normal space in \mathbb{R}^3

We can express the normal space $N(\mathbf{x}^*)$ as

$$N(\mathbf{x}^*) = \mathcal{R}(D\mathbf{h}(\mathbf{x}^*)^T),$$

that is, the range of the matrix $D\mathbf{h}(\mathbf{x}^*)^T$. Note that the normal space $N(\mathbf{x}^*)$ is the subspace of \mathbb{R}^n spanned by the vectors $\nabla h_1(\mathbf{x}^*), \dots, \nabla h_m(\mathbf{x}^*)$; that is,

$$\begin{aligned} N(\mathbf{x}^*) &= \text{span}[\nabla h_1(\mathbf{x}^*), \dots, \nabla h_m(\mathbf{x}^*)] \\ &= \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} = z_1 \nabla h_1(\mathbf{x}^*) + \dots + z_m \nabla h_m(\mathbf{x}^*), z_1, \dots, z_m \in \mathbb{R}\}. \end{aligned}$$

Note that the normal space contains the zero vector. Assuming that \mathbf{x}^* is regular, the dimension of the normal space $N(\mathbf{x}^*)$ is m . As in the case of the tangent space, it is often convenient to picture the normal space $N(\mathbf{x}^*)$ as passing through the point \mathbf{x}^* (rather than through the origin of \mathbb{R}^n). For this, we define the normal plane at \mathbf{x}^* as the set

$$NP(\mathbf{x}^*) = N(\mathbf{x}^*) + \mathbf{x}^* = \{\mathbf{x} + \mathbf{x}^* \in \mathbb{R}^n : \mathbf{x} \in N(\mathbf{x}^*)\}.$$

Figure 6.18 illustrates the normal space and plane in \mathbb{R}^3 (i.e., $n = 3$ and $m = 1$).

We now show that the tangent space and normal space are orthogonal complements of each other (see Section 3.3 of [30]).

Lemma 557. *We have $T(\mathbf{x}^*) = N(\mathbf{x}^*)^\perp$ and $T(\mathbf{x}^*)^\perp = N(\mathbf{x}^*)$.*

Proof. By definition of $T(\mathbf{x}^*)$, we may write

$$T(\mathbf{x}^*) = \{\mathbf{y} \in \mathbb{R}^n : \mathbf{x}^T \mathbf{y} = 0 \text{ for all } \mathbf{x} \in N(\mathbf{x}^*)\}.$$

Hence, by definition of $N(\mathbf{x}^*)$, we also have $T(\mathbf{x}^*) = N(\mathbf{x}^*)^\perp$. By Exercise 3.11 of [30] we also have $T(\mathbf{x}^*)^\perp = N(\mathbf{x}^*)$. \square

By Lemma 557, we can write \mathbb{R}^n as the direct sum decomposition (see Section 3.3 of [30]):

$$\mathbb{R}^n = N(\mathbf{x}^*) \oplus T(\mathbf{x}^*);$$

that is, given any vector $\mathbf{v} \in \mathbb{R}^n$, there are unique vectors $\mathbf{w} \in N(\mathbf{x}^*)$ and $\mathbf{y} \in T(\mathbf{x}^*)$ such that

$$\mathbf{v} = \mathbf{w} + \mathbf{y}.$$

6.7.4 Lagrange Condition

In this section we present a first-order necessary condition for extremum problems with constraints. The result is the well-known Lagrange's theorem. To better understand the idea underlying this theorem, we first consider functions of two variables and only one equality constraint. Let $h : \mathbb{R}^2 \rightarrow \mathbb{R}$ be the constraint function. Recall that at each point \mathbf{x} of the domain, the gradient vector $\nabla h(\mathbf{x})$ is orthogonal to the level set that passes through that point. Indeed, let us choose a point $\mathbf{x}^* = [x_1^*, x_2^*]$ such that $h(\mathbf{x}^*) = 0$, and assume that $h(\mathbf{x}^*) \neq \mathbf{0}$. The level set through the point \mathbf{x}^* is the set $\{\mathbf{x} : h(\mathbf{x}) = 0\}$. We then parameterize this level set in a neighborhood of \mathbf{x}^* by a curve $\{\mathbf{x}(t)\}$, that is, a continuously differentiable vector function $\mathbf{x} : \mathbb{R} \rightarrow \mathbb{R}^2$ such that

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}, t \in (a, b), \mathbf{x}^* = \mathbf{x}^*(t^*), \dot{\mathbf{x}} \neq \mathbf{0}, t^* \in (a, b)$$

We can now show that $\nabla h(\mathbf{x}^*)$ is orthogonal to $\dot{\mathbf{x}}(t^*)$. Indeed, because h is constant on the curve $\{\mathbf{x}(t) : t \in (a, b)\}$, we have that for all $t \in (a, b)$,

$$h(\mathbf{x}(t)) = 0.$$

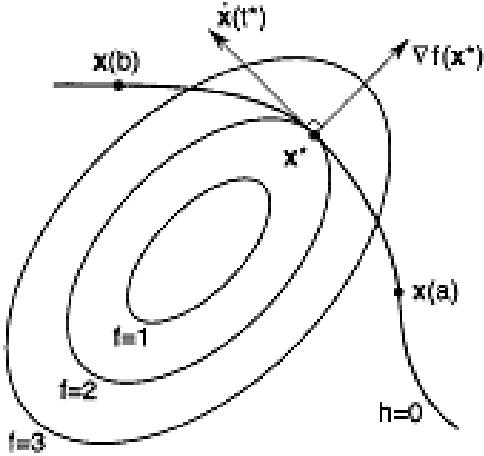


图 6.19: The gradient $\nabla f(\mathbf{x})$ is orthogonal to the curve $\{\mathbf{x}(t)\}$ at the point \mathbf{x}^* that is a minimizer of f on the curve.

Hence, for all $t \in (a, b)$,

$$\frac{d}{dt} h(\mathbf{x}(t)) = 0.$$

Applying the chain rule, we get

$$\frac{d}{dt} h(\mathbf{x}(t)) = \nabla(\mathbf{x}(t))^T \dot{\mathbf{x}}(t) = 0.$$

Therefore, $\nabla h(\mathbf{x}^*)$ is orthogonal to $\dot{\mathbf{x}}(t^*)$.

Now suppose that \mathbf{x}^* is a minimizer of $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ on the set $\{\mathbf{x} : h(\mathbf{x}) = 0\}$. We claim that $\nabla(\mathbf{x}^*)$ is orthogonal to $\dot{\mathbf{x}}(t^*)$. To see this, it is enough to observe that the composite function of t given by

$$\phi(t) = f(\mathbf{x}(t))$$

achieves a minimum at t^* . Consequently, the first-order necessary condition for the unconstrained extremum problem implies that

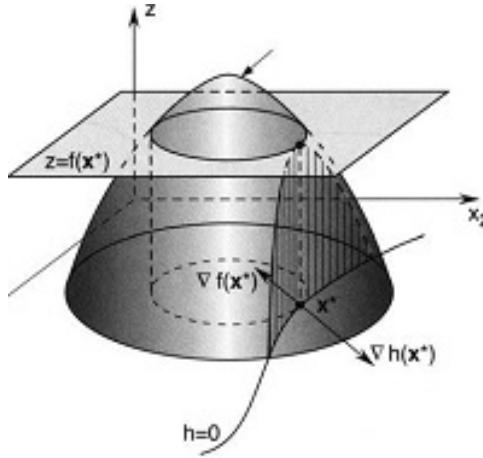
$$\frac{d\phi}{dt}(t^*) = 0.$$

Applying the chain rule yields

$$0 = \frac{d}{dt}\phi(t^*) = \nabla f(\mathbf{x}(t^*))^T \dot{\mathbf{x}}(t^*) = \nabla f(\mathbf{x}^*)^T \dot{\mathbf{x}}(t^*).$$

Thus, $\nabla f(\mathbf{x}^*)$ is orthogonal to $\dot{\mathbf{x}}(t^*)$. The fact that $\dot{\mathbf{x}}(t^*)$ is tangent to the curve $\{\mathbf{x}(t)\}$ at \mathbf{x}^* means that $\nabla f(\mathbf{x}^*)$ is orthogonal to the curve at \mathbf{x}^* (see Figure 6.19).

Recall that $\nabla h(\mathbf{x}^*)$ is also orthogonal to $\dot{\mathbf{x}}(t^*)$. Therefore, the vectors $\nabla h(\mathbf{x}^*)$ and $\nabla f(\mathbf{x}^*)$ are parallel; that is, $\nabla f(\mathbf{x}^*)$ is a scalar multiple of $\nabla h(\mathbf{x}^*)$. The observations


 图 6.20: Lagrange's theorem for $n = 2, m = 1$

above allow us now to formulate Lagrange's theorem for functions of two variables with one constraint.

Theorem 558. *Lagrange's Theorem for $n = 2, m = 1$. Let the point \mathbf{x}^* be a minimizer of $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ subject to the constraint $h(\mathbf{x}) = 0, h : \mathbb{R}^2 \rightarrow \mathbb{R}$. Then, $\nabla f(\mathbf{x}^*)$ and $\nabla h(\mathbf{x}^*)$ are parallel. That is, if $\nabla h(\mathbf{x}^*) \neq \mathbf{0}$, then there exists a scalar λ that*

$$\nabla f(\mathbf{x}^*) + \lambda^* \nabla h(\mathbf{x}^*) = \mathbf{0}.$$

In Theorem 558, we refer to λ^* as the Lagrange multiplier. Note that the theorem also holds for maximizers. Figure 6.20 gives an illustration of Lagrange's theorem for the case where \mathbf{x}^* is a maximizer of f over the set $\{\mathbf{x} : h(\mathbf{x}) = 0\}$.

Lagrange's theorem provides a first-order necessary condition for a point to be a local minimizer. This condition, which we call the Lagrange condition, consists of two equations:

$$\begin{aligned}\nabla f(\mathbf{x}^*) + \lambda^* \nabla h(\mathbf{x}^*) &= \mathbf{0} \\ h(\mathbf{x}^*) &= 0.\end{aligned}$$

Note that the Lagrange condition is necessary but not sufficient. In Figure 6.21 we illustrate a variety of points where the Lagrange condition is satisfied, including a case where the point is not an extremizer (neither a maximizer nor a minimizer).

We now generalize Lagrange's theorem for the case when $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m, m \leq n$.

Theorem 559. *Lagrange's Theorem. Let \mathbf{x}^* be a local minimizer (or maximizer) of $f : \mathbb{R}^n \rightarrow \mathbb{R}$, subject to $\mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m, m \leq n$. Assume that \mathbf{x}^* is a regular*

point. Then, there exists $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ that

$$Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) = \mathbf{0}^T.$$

Proof. We need to prove that

$$\nabla f(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) = \mathbf{0}^T$$

for some $\boldsymbol{\lambda} \in \mathbb{R}^m$; that is, $\nabla f(\mathbf{x}^*) \in \mathcal{R}(D\mathbf{h}(\mathbf{x}^*)^T) = N(\mathbf{x}^*)$. But by Lemma 557, $N(\mathbf{x}^*) = T(\mathbf{x}^*)^\perp$. Therefore, it remains to show that $\nabla f(\mathbf{x}^*) \in T(\mathbf{x}^*)^\perp$.

We proceed as follows. Suppose that

$$\mathbf{y} \in T(\mathbf{x}^*)$$

Then, by Theorem 555, there exists a differentiable curve $\{\mathbf{x}(t) : t \in (a, b)\}$ such that for all $t \in (a, b)$,

$$\mathbf{h}(\mathbf{x}(t)) = \mathbf{0}$$

and there exists $t^* \in (a, b)$ satisfying

$$\mathbf{x}(t^*) = \mathbf{x}^*, \dot{\mathbf{x}}(t^*) = \mathbf{y}$$

Now consider the composite function $\phi(t) = f(\mathbf{x}(t))$. Note that t^* is a local minimizer of this function. By the first-order necessary condition for unconstrained local minimizers (see Theorem 6.1 of [30])

$$\frac{d\phi}{dt}(t^*) = 0$$

Applying the chain rule yields

$$\frac{d\phi}{dt}(t^*) = Df(\mathbf{x}^*) \dot{\mathbf{x}}(t^*) = Df(\mathbf{x}^*) \mathbf{y} = \nabla f(\mathbf{x}^*)^T \mathbf{y} = 0.$$

So all $\mathbf{y} \in T(\mathbf{x}^*)$ satisfy

$$\nabla f(\mathbf{x}^*)^T \mathbf{y} = 0;$$

that is,

$$\nabla f(\mathbf{x}^*) \in T(\mathbf{x}^*)^\perp.$$

This completes the proof. \square

Lagrange's theorem states that if \mathbf{x}^* is an extremizer, then the gradient of the objective function f can be expressed as a linear combination of the gradients of the constraints. We refer to the vector $\boldsymbol{\lambda}^*$ in Theorem 559 as the Lagrange multiplier vector, and its components as *Lagrange multipliers*.

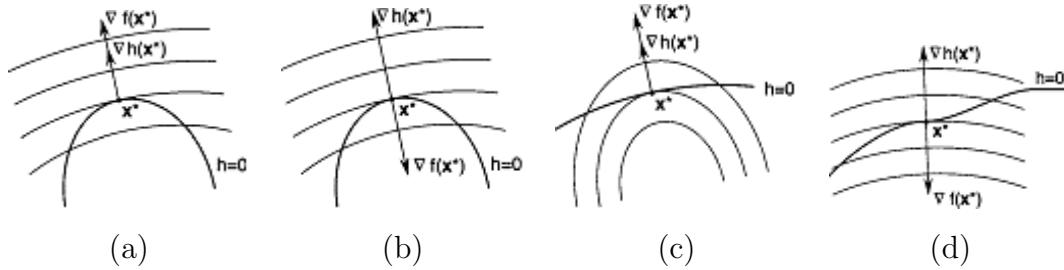


图 6.21: Four examples where the Lagrange condition is satisfied: (a) maximizer, (b) minimizer, (c) minimizer, (d) not an extremizer.

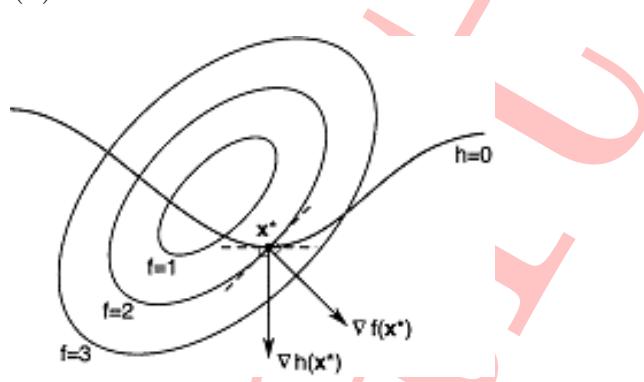


图 6.22: Example where the Lagrange condition does not hold

From the proof of Lagrange's theorem, we see that a compact way to write the necessary condition is $\nabla f(\mathbf{x}^*) \in N(\mathbf{x}^*)$. If this condition fails, then \mathbf{x}^* cannot be an extremizer. This situation is illustrated in Figure 6.22.

Notice that regularity is stated as an assumption in Lagrange's theorem. This assumption plays an essential role, as illustrated in the following example.

Example 560. Consider the following problem:

$$\begin{aligned} & \text{minimize } f(x) \\ & \text{subject to } h(x) = 0 \end{aligned}$$

where $f(x) = x$ and

$$h(x) = \begin{cases} x^2 & \text{if } x \leq 0 \\ 0 & \text{if } 0 \leq x \leq 1 \\ (x-1)^2 & \text{if } x \geq 1. \end{cases}$$

The feasible set is evidently $[0, 1]$. Clearly, $x^* = 0$ is a local minimizer. However, $f'(x^*) = 1$ and $h'(x^*) = 0$. Therefore, x^* does not satisfy the necessary condition in Lagrange's theorem. Note, however, that x^* is not a regular point, which is why Lagrange's theorem does not apply here.

It is convenient to introduce the *Lagrangian function* $l : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$, given by

$$l(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}).$$

The Lagrange condition for a local minimizer \mathbf{x}^* can be represented using the Lagrangian function as

$$Dl(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \mathbf{0}^T$$

for some $\boldsymbol{\lambda}^*$, where the derivative operation D is with respect to the entire argument $[\mathbf{x}^T, \boldsymbol{\lambda}^T]^T$. In other words, the necessary condition in Lagrange's theorem is equivalent to the first-order necessary condition for unconstrained optimization applied to the Lagrangian function. To see the above, denote the derivative of l with respect to \mathbf{x} as $D_x l$ and the derivative of l with respect to $\boldsymbol{\lambda}$ as $D_{\lambda} l$. Then,

$$Dl(\mathbf{x}, \boldsymbol{\lambda}) = [D_x l(\mathbf{x}, \boldsymbol{\lambda}), D_{\lambda} l(\mathbf{x}, \boldsymbol{\lambda})].$$

Note that $D_x l(\mathbf{x}, \boldsymbol{\lambda}) = Df(\mathbf{x}) + \boldsymbol{\lambda}^T D\mathbf{h}(\mathbf{x})$ and $D_{\lambda} l(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{h}(\mathbf{x})^T$. Therefore, Lagrange's theorem for a local minimizer \mathbf{x}^* can be stated as

$$\begin{aligned} D_x l(\mathbf{x}^*, \boldsymbol{\lambda}^*) &= \mathbf{0}^T, \\ D_{\lambda} l(\mathbf{x}^*, \boldsymbol{\lambda}^*) &= \mathbf{0}^T \end{aligned}$$

for some $\boldsymbol{\lambda}^*$, which is equivalent to

$$Dl(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \mathbf{0}^T.$$

In other words, the Lagrange condition can be expressed as $Dl(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \mathbf{0}^T$.

The Lagrange condition is used to find possible extremizers. This entails solving the equations

$$\begin{aligned} D_x l(\mathbf{x}, \boldsymbol{\lambda}) &= \mathbf{0}^T, \\ D_{\lambda} l(\mathbf{x}, \boldsymbol{\lambda}) &= \mathbf{0}^T \end{aligned}$$

The above represents $n+m$ equations in $n+m$ unknowns. Keep in mind that the Lagrange condition is necessary but not sufficient; that is, a point \mathbf{x}^* satisfying the equations above need not be an extremizer.

Example 561. Given a fixed area of cardboard, we wish to construct a closed cardboard box with maximum volume. We can formulate and solve this problem using the Lagrange

condition. Denote the dimensions of the box with maximum volume by x_1 , x_2 , and x_3 , and let the given fixed area of cardboard be A . The problem can then be formulated as

$$\begin{aligned} & \text{maximize} && x_1x_2x_3 \\ & \text{subject to} && x_1x_2 + x_2x_3 + x_3x_1 = \frac{A}{2} \end{aligned}$$

We denote $f(\mathbf{x}) = x_1x_2x_3$ and $h(\mathbf{x}) = x_1x_2 + x_2x_3 + x_3x_1 - \frac{A}{2}$. We have $\nabla f(\mathbf{x}) = -[x_2x_3, x_1x_3, x_1x_2]^T$ and $\nabla h(\mathbf{x}) = [x_2 + x_3, x_1 + x_3, x_1 + x_2]^T$. Note that all feasible points are regular in this case. By the Lagrange condition, the dimensions of the box with maximum volume satisfies

$$\begin{aligned} x_2x_3 - \lambda(x_2 + x_3) &= 0 \\ x_1x_3 - \lambda(x_1 + x_3) &= 0 \\ x_1x_2 - \lambda(x_1 + x_2) &= 0 \\ x_1x_2 + x_2x_3 + x_3x_1 &= \frac{A}{2} \end{aligned}$$

where $\lambda \in \mathbb{R}$.

We now solve these equations. First, we show that x_1, x_2, x_3 , and λ are all nonzero. Suppose that $x_1 = 0$. By the constraints, we have $x_2x_3 = A/2$. However, the second and third equations in the Lagrange condition yield $\lambda x_2 = \lambda x_3 = 0$, which together with the first equation implies that $x_2x_3 = 0$. This contradicts the constraints. A similar argument applies to x_2 and x_3 .

Next, suppose that $\lambda = 0$. Then, the sum of the three Lagrange equations gives $x_2x_3 + x_1x_3 + x_1x_2 = 0$, which contradicts the constraints.

We now solve for x_1 , x_2 , and x_3 in the Lagrange equations. First, multiply the first equation by x_1 and the second by x_2 , and subtract one from the other. We arrive at $x_3\lambda(x_1 - x_2) = 0$. Because neither x_3 nor λ can be zero (by part b), we conclude that $x_1 = x_2$. We similarly deduce that $x_2 = x_3$. From the constraint equation, we obtain $x_1 = x_2 = x_3 = \sqrt{A/6}$.

Notice that we have ignored the constraints that x_1 , x_2 , and x_3 are positive so that we can solve the problem using Lagrange's theorem. However, there is only one solution to the Lagrange equations, and the solution is positive. Therefore, if a solution exists for the problem with positivity constraints on the variables x_1 , x_2 , and x_3 , then this solution must necessarily be equal to the solution above obtained by ignoring the positivity constraints.

Next we provide an example with a quadratic objective function and a quadratic constraint.

Example 562. Consider the problem of extremizing the objective function

$$f(\mathbf{x}) = x_1^2 + x_2^2$$

on the ellipse

$$\left\{ [x_1, x_2]^T : h(\mathbf{x}) = x_1^2 + 2x_2^2 - 1 = 0 \right\}.$$

We have

$$\nabla f(\mathbf{x}) = [2x_1, 2x_2]$$

$$\nabla h(\mathbf{x}) = [2x_1, 4x_2]$$

Thus,

$$D_x l(\mathbf{x}, \lambda) = D_x [f(\mathbf{x}) + \lambda h(\mathbf{x})] = [2x_1 + 2\lambda x_1, 2x_2 + 4\lambda x_2]$$

and

$$D_\lambda l(\mathbf{x}, \lambda) = h(\mathbf{x}) = x_1^2 + 2x_2^2 - 1.$$

Setting $D_x l(\mathbf{x}, \lambda) = \mathbf{0}^T$ and $D_\lambda l(\mathbf{x}, \lambda) = 0$, we obtain three equations in three unknowns

$$2x_1 + 2\lambda x_1 = 0,$$

$$2x_2 + 4\lambda x_2 = 0,$$

$$x_1^2 + 2x_2^2 = 1.$$

All feasible points in this problem are regular. From the first of the equations above, we get either $x_1 = 0$ or $\lambda = -1$. For the case where $x_1 = 0$, the second and third equations imply that $\lambda = -1/2$ and $x_2 = \pm 1/\sqrt{2}$. For the case where $\lambda = -1$, the second and third equations imply that $x_1 = \pm 1$ and $x_2 = 0$. Thus, the points that satisfy the Lagrange condition for extrema are

$$\mathbf{x}^{(1)} = \begin{bmatrix} 0 \\ 1/\sqrt{2} \end{bmatrix}, \mathbf{x}^{(2)} = \begin{bmatrix} 0 \\ -1/\sqrt{2} \end{bmatrix}, \mathbf{x}^{(3)} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \mathbf{x}^{(4)} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}.$$

Because

$$f(\mathbf{x}^{(1)}) = f(\mathbf{x}^{(2)}) = \frac{1}{2}$$

and

$$f(\mathbf{x}^{(3)}) = f(\mathbf{x}^{(4)}) = 1$$

we conclude that if there are minimizers, then they are located at $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ and if there are maximizers, then they are located at $\mathbf{x}^{(3)}$ and $\mathbf{x}^{(4)}$. It turns out that, indeed, $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ are minimizers and $\mathbf{x}^{(3)}$ and $\mathbf{x}^{(4)}$ are maximizers. This problem can be solved graphically, as illustrated in Figure 6.23.

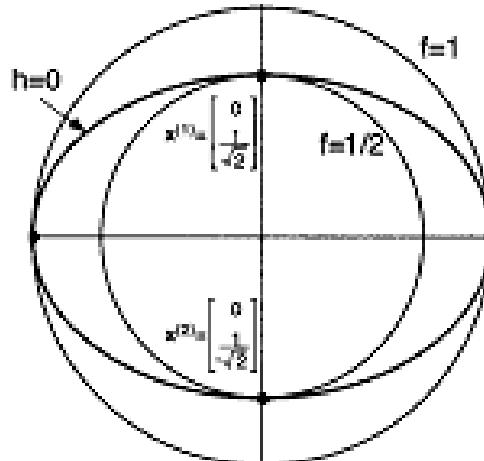


图 6.23: Graphical solution of the problem in Figure 562

In the example above, both the objective function f and the constraint function h are quadratic functions. In the next example we take a closer look at a class of problems where both the objective function f and the constraint h are quadratic functions of n variables.

Example 563. Consider the following problem:

$$\underset{\mathbf{x} \in \mathbb{R}^n}{\text{maximize}} \frac{\mathbf{x}^T \mathbf{Q} \mathbf{x}}{\mathbf{x}^T \mathbf{P} \mathbf{x}}$$

where $\mathbf{Q} = \mathbf{Q}^T \geq 0$ and $\mathbf{P} = \mathbf{P}^T \geq 0$. Note that if a point $\mathbf{x} = [x_1, \dots, x_n]^T$ is a solution to the problem, then so is any nonzero scalar multiple of it,

$$t\mathbf{x} = [tx_1, \dots, tx_n]^T, t \neq 0.$$

Indeed,

$$\frac{(t\mathbf{x})^T \mathbf{Q} (t\mathbf{x})}{(t\mathbf{x})^T \mathbf{P} (t\mathbf{x})} = \frac{t^2 \mathbf{x}^T \mathbf{Q} \mathbf{x}}{t^2 \mathbf{x}^T \mathbf{P} \mathbf{x}} = \frac{\mathbf{x}^T \mathbf{Q} \mathbf{x}}{\mathbf{x}^T \mathbf{P} \mathbf{x}}.$$

Therefore, to avoid the multiplicity of solutions, we further impose the constraint

$$\mathbf{x}^T \mathbf{P} \mathbf{x} = 1.$$

The optimization problem becomes

$$\begin{aligned} & \underset{\mathbf{x} \in \mathbb{R}^n}{\text{maximize}} \quad \mathbf{x}^T \mathbf{Q} \mathbf{x} \\ & \text{subject to} \quad \mathbf{x}^T \mathbf{P} \mathbf{x} = 1. \end{aligned}$$

Let us write

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{Q} \mathbf{x},$$

$$h(\mathbf{x}) = 1 - \mathbf{x}^T \mathbf{P} \mathbf{x}.$$

Any feasible point for this problem is regular. We now apply Lagrange's method. We first form the Lagrangian function

$$l(\mathbf{x}, \lambda) = \mathbf{x}^T \mathbf{Q} \mathbf{x} + \lambda (1 - \mathbf{x}^T \mathbf{P} \mathbf{x}).$$

Applying the Lagrange condition yields

$$D_x l(\mathbf{x}, \lambda) = 2\mathbf{x}^T \mathbf{Q} - 2\lambda \mathbf{x}^T \mathbf{P} = \mathbf{0}^T,$$

$$D_\lambda l(\mathbf{x}, \lambda) = 1 - \mathbf{x}^T \mathbf{P} \mathbf{x} = 0.$$

The first of the equations above can be represented as

$$\mathbf{Q} \mathbf{x} - \lambda \mathbf{P} \mathbf{x} = \mathbf{0}$$

or

$$(\lambda \mathbf{P} - \mathbf{Q}) \mathbf{x} = \mathbf{0}.$$

This representation is possible because $\mathbf{P} = \mathbf{P}^T$ and $\mathbf{Q} = \mathbf{Q}^T$. By assumption $\mathbf{P} \succ 0$, hence \mathbf{P}^{-1} exists. Premultiplying $(\lambda \mathbf{P} - \mathbf{Q}) \mathbf{x} = \mathbf{0}$ by \mathbf{P}^{-1} , we obtain

$$(\lambda \mathbf{I}_n - \mathbf{P}^{-1} \mathbf{Q}) \mathbf{x} = \mathbf{0}$$

or, equivalently,

$$\mathbf{P}^{-1} \mathbf{Q} \mathbf{x} = \lambda \mathbf{x}.$$

Therefore, the solution, if it exists, is an eigenvector of $\mathbf{P}^{-1} \mathbf{Q}$, and the Lagrange multiplier is the corresponding eigenvalue. As usual, let \mathbf{x}^* and λ^* be the optimal solution. Because $\mathbf{x}^{*T} \mathbf{P} \mathbf{x}^* = 1$ and $\mathbf{P}^{-1} \mathbf{Q} \mathbf{x}^* = \lambda^* \mathbf{x}^*$, we have

$$\lambda^* = \mathbf{x}^{*T} \mathbf{Q} \mathbf{x}^*.$$

Hence, λ^* is the maximum of the objective function, and therefore is, in fact, the maximal eigenvalue of $\mathbf{P}^{-1} \mathbf{Q}$. It is also called the maximal generalized eigenvalue.

In the problems above, we are able to find points that are candidates for extremizers of the given objective function subject to equality constraints. These critical points are the only candidates because they are the only points that satisfy the Lagrange condition. To classify such critical points as minimizers, maximizers, or neither, we need a stronger condition?possibly a necessary and sufficient condition. In the next section we discuss a second-order necessary condition and a second-order sufficient condition for minimizers.

6.7.5 Second-Order Conditions

We assume that $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are twice continuously differentiable: $f, \mathbf{h} \in \mathcal{C}^2$. Let

$$l(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}) = f(\mathbf{x}) + \lambda_1 h_1(\mathbf{x}) + \cdots + \lambda_m h_m(\mathbf{x})$$

be the Lagrangian function. Let $\mathbf{L}(\mathbf{x}, \boldsymbol{\lambda})$ be the Hessian matrix of $l(\mathbf{x}, \boldsymbol{\lambda})$ with respect to \mathbf{x} :

$$\mathbf{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{F}(\mathbf{x}) + \lambda_1 \mathbf{H}_1(\mathbf{x}) + \cdots + \lambda_m \mathbf{H}_m(\mathbf{x})$$

where $\mathbf{F}(\mathbf{x})$ is the Hessian matrix of f at \mathbf{x} and $\mathbf{H}_k(\mathbf{x})$ is the Hessian matrix of h_k at $\mathbf{x}, k = 1, \dots, m$, given by

$$\mathbf{H}_k(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 h_k}{\partial x_1^2}(\mathbf{x}) & \cdots & \frac{\partial^2 h_k}{\partial x_n \partial x_1}(\mathbf{x}) \\ \vdots & & \vdots \\ \frac{\partial^2 h_k}{\partial x_1 \partial x_n}(\mathbf{x}) & \cdots & \frac{\partial^2 h_k}{\partial x_n^2}(\mathbf{x}) \end{bmatrix}.$$

We introduce the notation $[\boldsymbol{\lambda} \mathbf{H}(\mathbf{x})]$:

$$[\boldsymbol{\lambda} \mathbf{H}(\mathbf{x})] = \lambda_1 \mathbf{H}_1(\mathbf{x}) + \cdots + \lambda_m \mathbf{H}_m(\mathbf{x}) + \cdots.$$

Using the notation above, we can write

$$\mathbf{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{F}(\mathbf{x}) + [\boldsymbol{\lambda} \mathbf{H}(\mathbf{x})].$$

Theorem 564. Second-Order Necessary Conditions Let \mathbf{x}^* be a local minimizer of $f : \mathbb{R}^n \rightarrow \mathbb{R}$ subject to $\mathbf{h}(\mathbf{x}) = \mathbf{0}$, $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m, m \leq n$, and $f, \mathbf{h} \in \mathcal{C}^2$. Suppose that \mathbf{x}^* is regular. Then, there exists $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ such that:

1. $Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) = \mathbf{0}^T$.
2. for all $\mathbf{y} \in T(\mathbf{x}^*)$, we have $\mathbf{y}^T \mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \mathbf{y} \geq 0$.

Proof. The existence of $\boldsymbol{\lambda}^*$ such that $Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) = \mathbf{0}^T$ follows from Lagrange's theorem. It remains to prove the second part of the result. Suppose that $\mathbf{y} \in T(\mathbf{x}^*)$; that is, \mathbf{y} belongs to the tangent space to $S = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{h}(\mathbf{x}) = \mathbf{0}\}$ at \mathbf{x}^* . Because $\mathbf{h} \in \mathcal{C}^2$, following the argument of Theorem 592, there exists a twice-differentiable curve $\{\mathbf{x}(t) : t \in (a, b)\}$ on S such that

$$\mathbf{x}(t^*) = \mathbf{x}^*, \dot{\mathbf{x}}(t^*) = \mathbf{y}$$

for some $t^* \in (a, b)$. Observe that by assumption, t^* is a local minimizer of the function $\phi(t) = f(\mathbf{x}(t))$. From the second-order necessary condition for unconstrained minimization (see Theorem 6.2 of [30]), we obtain

$$\frac{d^2\phi}{dt^2}(t^*) \geq 0.$$

Using the formula

$$\frac{d}{dt} (\mathbf{y}(t)^T \mathbf{z}(t)) = \mathbf{z}(t)^T \frac{d\mathbf{y}}{dt}(t) + \mathbf{y}(t)^T \frac{d\mathbf{z}}{dt}(t)$$

and applying the chain rule yields

$$\begin{aligned} \frac{d^2\phi}{dt^2}(t^*) &= \frac{d}{dt} [Df(\mathbf{x}(t^*)) \dot{\mathbf{x}}(t^*)] \\ &= \dot{\mathbf{x}}(t^*)^T \mathbf{F}(\mathbf{x}^*) \dot{\mathbf{x}}(t^*) + Df(\mathbf{x}^*) \ddot{\mathbf{x}}(t^*) \\ &= \mathbf{y}^T \mathbf{F}(\mathbf{x}^*) \mathbf{y} + Df(\mathbf{x}^*) \ddot{\mathbf{x}}(t^*) \geq 0. \end{aligned}$$

Because $\mathbf{h}(\mathbf{x}(t)) = \mathbf{0}$ for all $t \in (a, b)$, we have

$$\frac{d^2}{dt^2} \boldsymbol{\lambda}^{*T} \mathbf{h}(\mathbf{x}(t)) = 0.$$

Thus, for all $t \in (a, b)$,

$$\begin{aligned} \frac{d^2}{dt^2} \boldsymbol{\lambda}^{*T} \mathbf{h}(\mathbf{x}(t)) &= \frac{d}{dt} \left[\boldsymbol{\lambda}^{*T} \frac{d}{dt} \mathbf{h}(\mathbf{x}(t)) \right] \\ &= \frac{d}{dt} \left[\sum_{k=1}^m \lambda_k^* \frac{d}{dt} h_k(\mathbf{x}(t)) \right] \\ &= \frac{d}{dt} \left[\sum_{k=1}^m \lambda_k^* D h_k(\mathbf{x}(t)) \dot{\mathbf{x}}(t) \right] \\ &= \sum_{k=1}^m \lambda_k^* \frac{d}{dt} (D h_k(\mathbf{x}(t)) \dot{\mathbf{x}}(t)) \\ &= \sum_{k=1}^m \lambda_k^* \left[\dot{\mathbf{x}}(t)^T \mathbf{H}_k(\mathbf{x}(t)) \dot{\mathbf{x}}(t) + D h_k(\mathbf{x}(t)) \ddot{\mathbf{x}}(t) \right] \\ &= \dot{\mathbf{x}}^T [\boldsymbol{\lambda}^* \mathbf{H}(\mathbf{x}(t))] \dot{\mathbf{x}}(t) + \boldsymbol{\lambda}^{*T} D \mathbf{h}(\mathbf{x}(t)) \ddot{\mathbf{x}}(t) \\ &= 0. \end{aligned}$$

In particular, the above is true for $t = t^*$; that is,

$$\mathbf{y}^T [\boldsymbol{\lambda}^* \mathbf{H}(\mathbf{x}^*)] \mathbf{y} + \boldsymbol{\lambda}^{*T} D \mathbf{h}(\mathbf{x}^*) \ddot{\mathbf{x}}(t^*) = 0.$$

Adding this equation to the inequality

$$\mathbf{y}^T \mathbf{F}(\mathbf{x}^*) \mathbf{y} + Df(\mathbf{x}^*) \ddot{\mathbf{x}}(t^*) \geq 0$$

yields

$$\mathbf{y}^T (\mathbf{F}(\mathbf{x}^*) + [\boldsymbol{\lambda}^* \mathbf{H}(\mathbf{x}^*)]) \mathbf{y} + (Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*)) \ddot{\mathbf{x}}(t) \geq 0.$$

But, by Lagrange's theorem, $Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) = \mathbf{0}^T$. Therefore,

$$\mathbf{y}^T (\mathbf{F}(\mathbf{x}^*) + [\boldsymbol{\lambda}^* \mathbf{H}(\mathbf{x}^*)]) \mathbf{y} = \mathbf{y}^T \mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \mathbf{y} \geq 0,$$

which proves the result. \square

Observe that $\mathbf{L}(\mathbf{x}, \boldsymbol{\lambda})$ plays a similar role as the Hessian matrix $\mathbf{F}(\mathbf{x})$ of the objective function f did in the unconstrained minimization case. However, we now require that $\mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \geq 0$ only on $T(\mathbf{x}^*)$ rather than on \mathbb{R}^n .

The conditions above are necessary, but not sufficient, for a point to be a local minimizer. We now present, without a proof, sufficient conditions for a point to be a strict local minimizer.

Theorem 565. *Second-Order Sufficient Conditions Suppose that $f, \mathbf{h} \in \mathcal{C}^2$ and there exists a point $\mathbf{x}^* \in \mathbb{R}^n$ and $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ such that:*

1. $Df(\mathbf{x}^*) + \boldsymbol{\lambda}^* D\mathbf{h}(\mathbf{x}^*) = \mathbf{0}^T$;
2. for all $\mathbf{y} \in T(\mathbf{x}^*)$, $\mathbf{y} \notin \mathbf{0}$, we have $\mathbf{y}^T \mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \mathbf{y} \geq 0$.

The, \mathbf{x}^ is a strict local minimizer of f subject to $\mathbf{h}(\mathbf{x}) = \mathbf{0}$*

Theorem 565 states that if an \mathbf{b}^* satisfies the Lagrange condition, and $\mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ is positive definite on $T(\mathbf{x}^*)$, then \mathbf{x}^* is a strict local minimizer. A similar result to Theorem 565 holds for a strict local maximizer, the only difference being that $\mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ be negative definite on $T(\mathbf{x}^*)$. We illustrate this condition in the following example.

Example 566. Consider the following problem:

$$\text{maximize } \frac{\mathbf{x}^T \mathbf{Q} \mathbf{x}}{\mathbf{x}^T \mathbf{P} \mathbf{x}}$$

where

$$\mathbf{Q} = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}, \mathbf{P} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$$

As pointed out earlier, we can represent this problem in the equivalent form

$$\begin{aligned} &\text{maximize } \mathbf{x}^T \mathbf{Q} \mathbf{x} \\ &\text{subject to } \mathbf{x}^T \mathbf{P} \mathbf{x} = 1. \end{aligned}$$

The Lagrangian function for the transformed problem is given by

$$l(\mathbf{x}, \lambda) = \mathbf{x}^T \mathbf{Q} \mathbf{x} + \lambda (1 - \mathbf{x}^T \mathbf{P} \mathbf{x}).$$

The Lagrange condition yields

$$(\lambda \mathbf{I} - \mathbf{P}^{-1} \mathbf{Q}) \mathbf{x} = \mathbf{0},$$

where

$$\mathbf{P}^{-1} \mathbf{Q} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}.$$

There are only two values of λ that satisfy $(\lambda \mathbf{I} - \mathbf{P}^{-1} \mathbf{Q}) \mathbf{x} = \mathbf{0}$, namely, the eigenvalues of $\mathbf{P}^{-1} \mathbf{Q}$: $\lambda_1 = 2, \lambda_2 = 1$. We recall from our previous discussion of this problem that the Lagrange multiplier corresponding to the solution is the maximum eigenvalue of $\mathbf{P}^{-1} \mathbf{Q}$ namely, $\lambda^* = \lambda_1 = 2$. The corresponding eigenvector is the maximizer?the solution to the problem. The eigenvector corresponding to the eigenvalue $\lambda^* = 2$ satisfying the constraint $\mathbf{x}^T \mathbf{P} \mathbf{x} = 1$ is $\pm \mathbf{x}^*$, where

$$\mathbf{x}^* = \left[\frac{1}{\sqrt{2}}, 0 \right]^T.$$

At this point, all we have established is that the pairs $(\pm \mathbf{x}^*, \lambda^*)$ satisfy the Lagrange condition. We now show that the points $\pm \mathbf{x}^*$ are, in fact, strict local maximizers. We do this for the point \mathbf{x} . A similar procedure applies to $-\mathbf{x}$. We first compute the Hessian matrix of the Lagrangian function. We have

$$\mathbf{L}(\mathbf{x}^*, \lambda^*) = 2\mathbf{Q} - 2\lambda^* \mathbf{P} = \begin{bmatrix} 0 & 0 \\ 0 & -2 \end{bmatrix}.$$

The tangent space $T(\mathbf{x}^*)$ to $\{\mathbf{x} : 1 - \mathbf{x}^T \mathbf{P} \mathbf{x} = 0\}$ is

$$\begin{aligned} T(\mathbf{x}^*) &= \{\mathbf{y} \in \mathbb{R}^2 : \mathbf{x}^{*T} \mathbf{P} \mathbf{y} = 0\} \\ &= \left\{ \mathbf{y} : \begin{bmatrix} \sqrt{2}, 2 \end{bmatrix} \mathbf{y} = 0 \right\} \\ &= \left\{ \mathbf{y} : \mathbf{y} = [0, a]^T, a \in \mathbb{R} \right\}. \end{aligned}$$

Note that for each $\mathbf{y} \in T(\mathbf{x}^*), \mathbf{y} \neq \mathbf{0}$,

$$\mathbf{y}^T \mathbf{L}(\mathbf{x}^*, \lambda^*) \mathbf{y} = [0, a] \begin{bmatrix} 0 & 0 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} 0 & a \end{bmatrix} = -2a^2 \leq 0.$$

Hence, $\mathbf{L}(\mathbf{x}^*, \lambda^*) \leq 0$ on $T(\mathbf{x}^*)$, and thus $\mathbf{x}^* = [1/\sqrt{2}, 0]^T$ is a strict local maximizer.

The same is true for the point $-\mathbf{x}^*$. Note that

$$\frac{\mathbf{x}^{*T} \mathbf{Q} \mathbf{x}^*}{\mathbf{x}^{*T} \mathbf{P} \mathbf{x}^*} = 2,$$

which, as expected, is the value of the maximal eigenvalue of $\mathbf{P}^{-1}\mathbf{Q}$. Finally, we point out that any scalar multiple $t\mathbf{x}^*$ of \mathbf{x} , $t \neq 0$, is a solution to the original problem of maximizing $\mathbf{x}^T\mathbf{Qx}/\mathbf{x}^T\mathbf{Px}$.

6.7.6 Minimizing Quadratics Subject to Linear Constraints

Consider the problem

$$\text{minimize } \frac{1}{2}\mathbf{x}^T\mathbf{Qx} \quad \text{subject to} \quad \mathbf{Ax} = \mathbf{b},$$

where $\mathbf{Q} \geq 0$, $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m \leq n$, $\text{rank } \mathbf{A} = m$. This problem is a special case of what is called a quadratic programming problem (the general form of a quadratic programming problem includes the constraint $x \geq 0$). Note that the constraint set contains an infinite number of points. We now show, using Lagrange's theorem, that there is a unique solution to the optimization problem above. Following that, we provide an example illustrating the application of this solution to an optimal control problem.

To solve the problem, we first form the Lagrangian function

$$l(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2}\mathbf{x}^T\mathbf{Qx} + \boldsymbol{\lambda}^T(\mathbf{b} - \mathbf{Ax}).$$

The Lagrange condition yields

$$D_x l(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \mathbf{x}^{*T}\mathbf{Q} - \boldsymbol{\lambda}^{*T}\mathbf{A} = \mathbf{0}^T.$$

Rewriting, we get

$$\mathbf{x}^* = \mathbf{Q}^{-1}\mathbf{A}^T\boldsymbol{\lambda}^*.$$

Premultiplying both sides of the above by A gives

$$\mathbf{Ax}^* = \mathbf{AQ}^{-1}\mathbf{A}^T\boldsymbol{\lambda}^*.$$

Using the fact that $\mathbf{Ax}^* = \mathbf{b}$, and noting that $\mathbf{AQ}^{-1}\mathbf{A}^T$ is invertible because $\mathbf{Q} \succ 0$ and $\text{rank } \mathbf{A} = m$, we can solve for $\boldsymbol{\lambda}^*$ to obtain

$$\boldsymbol{\lambda}^* = (\mathbf{AQ}^{-1}\mathbf{A}^T)^{-1}\mathbf{b}.$$

Therefore, we obtain

$$\mathbf{x}^* = \mathbf{Q}^{-1}\mathbf{A}^T(\mathbf{AQ}^{-1}\mathbf{A}^T)^{-1}\mathbf{b}$$

The point \mathbf{x}^* is the only candidate for a minimizer. To establish that \mathbf{x}^* is indeed a minimizer, we verify that \mathbf{x}^* satisfies the second-order sufficient conditions. For this, we first find the Hessian matrix of the Lagrangian function at $\mathbf{x}^*, \boldsymbol{\lambda}^*$. We have

$$\mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \mathbf{Q},$$

which is positive definite. Thus, the point \mathbf{x}^* is a strict local minimizer. We will see in Chapter 22 of [30] that \mathbf{x}^* is in fact a global minimizer.

The special case where $\mathbf{Q} = \mathbf{I}_n$, the $n \times n$ identity matrix, reduces to the problem considered in Section 12.3 of [30]. Specifically, the problem in Section of [30] is to minimize the norm $\|\mathbf{x}\|$ subject to $\mathbf{Ax} = \mathbf{b}$. The objective function here is $f(\mathbf{x}) = \|\mathbf{x}\|$, which is not differentiable at $\mathbf{x} = \mathbf{0}$. This precludes the use of Lagrange's theorem because the theorem requires differentiability of the objective function. We can overcome this difficulty by considering an equivalent optimization problem:

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \|\mathbf{x}\|^2 \\ & \text{subject to} && \mathbf{Ax} = \mathbf{b}. \end{aligned}$$

The objective function $\|\mathbf{x}\|^2/2$ has the same minimizer as the previous objective function $\|\mathbf{x}\|$. Indeed, if \mathbf{x}^* is such that for all $\mathbf{x} \in \mathbb{R}^n$ satisfying $\mathbf{Ax} = \mathbf{b}$, $\|\mathbf{x}^*\| \leq \|\mathbf{x}\|$, then $\|\mathbf{x}^*\|^2/2 \leq \|\mathbf{x}\|^2/2$. The same is true for the converse. Because the problem of minimizing $\|\mathbf{x}\|^2/2$ subject to $\mathbf{Ax} = \mathbf{b}$ is simply the problem considered above with $\mathbf{Q} = \mathbf{I}_n$, we easily deduce the solution to be $\mathbf{x}^* = \mathbf{A}^T (\mathbf{A}\mathbf{A}^T)^{-1} \mathbf{b}$, which agrees with the solution in Section 12.3 of [30].

Example 567. Consider the discrete-time linear system model

$$x_k = ax_{k-1} + b\mu_k, k \geq 1,$$

with initial condition x_0 given. We can think of $\{x_k\}$ as a discrete-time signal that is controlled by an external input signal $\{\mu_k\}$. In the control literature, x_k is called the state at time k . For a given x_0 , our goal is to choose the control signal $\{\mu_k\}$ so that the state remains “small” over a time interval $[1, V]$, but at the same time the control signal is “not too large.” To express the desire to keep the state $\{x_k\}$ small, we choose the control sequence to minimize

$$\frac{1}{2} \sum_{i=1}^N x_i^2.$$

On the other hand, maintaining a control signal that is not too large, we minimize

$$\frac{1}{2} \sum_{i=1}^N \mu_i^2.$$

The two objectives above are conflicting in the sense that they cannot, in general, be achieved simultaneously – minimizing the first may result in a large control effort, while minimizing the second may result in large states. This is clearly a problem that requires

compromise. One way to approach the problem is to minimize a weighted sum of the two functions above. Specifically, we can formulate the problem as

$$\text{minimize} \quad \frac{1}{2} \sum_1^2 (qx_i^2 + r\mu_i^2) \quad \text{subject to} \quad x_k = ax_{k-1} + b\mu_k, k = 1, \dots, N, x_0 \text{ given,}$$

where the parameters q and r reflect the relative importance of keeping the state small versus keeping the control effort not too large. This problem is an instance of the linear quadratic regulator (LQR) problem. Combining the two conflicting objectives of keeping the state small while keeping the control effort small is an instance of the weighted sum approach (see Section 10.4).

To solve the problem above, we can rewrite it as a quadratic programming problem. Define

$$\mathbf{Q} = \begin{bmatrix} q\mathbf{I}_N & \mathbf{0} \\ \mathbf{0} & r\mathbf{I}_N \end{bmatrix},$$

$$\mathbf{A} = \begin{bmatrix} 1 & \cdots & 0 & -b & \cdots & 0 \\ -a & 1 & & & & \\ \ddots & \ddots & \ddots & -b & & \vdots \\ 0 & -a & 1 & 0 & \cdots & -b \end{bmatrix},$$

$$\mathbf{b} = \begin{bmatrix} ax_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

$$\mathbf{z} = [x_1, \dots, x_N, \mu_1, \dots, \mu_N].$$

With these definitions, the problem reduces to the previously considered quadratic programming problem,

$$\text{minimize} \quad \frac{1}{2} \mathbf{z}^T \mathbf{Q} \mathbf{z} \quad \text{subject to} \quad \mathbf{A} \mathbf{z} = \mathbf{b},$$

where \mathbf{Q} is $2N \times 2N$, \mathbf{A} is $N \times 2N$, and $\mathbf{b} \in \mathbb{R}^{nd}$. The solution is

$$\mathbf{z}^* = \mathbf{Q}^{-1} \mathbf{A}^T (\mathbf{A} \mathbf{Q}^{-1} \mathbf{A}^T)^{-1} \mathbf{b}.$$

The first N components of \mathbf{z}^* represent the optimal state signal in the interval $[1, N]$, whereas the second N components represent the optimal control signal.

In practice, computation of the matrix inverses in the formula for \mathbf{z}^* above may be too costly. There are other ways to tackle the problem by exploiting its special structure. This is the study of optimal control.

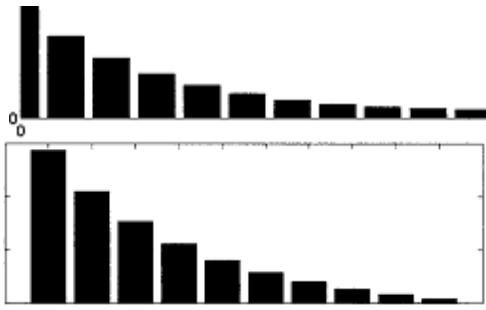


图 6.24: Plots for Example 568 with $q = 1$ and $r = 10$

The following example illustrates an application of the above discussion.

Example 568. Credit-Card Holder Dilemma Suppose that we currently have a credit-card debt of \$10,000. Credit-card debts are subject to a monthly interest rate of 2%, and the account balance is increased by the interest amount every month. Each month we have the option of reducing the account balance by contributing a payment to the account. Over the next 10 months, we plan to contribute a payment every month in such a way as to minimize the overall debt level while minimizing the hardship of making monthly payments.

We solve our problem using the LQR framework described in Example 567. Let the current time be 0, x_k the account balance at the end of month k , and μ_k our payment in month k . We have

$$x_k = 1.02x_{k-1} - \mu_k, k = 1, \dots, 10;$$

that is, the account balance in a given month is equal to the account balance in the previous month plus the monthly interest on that balance minus our payment that month. Our optimization problem is then

$$\begin{aligned} & \text{minimize} \quad \frac{1}{2} \sum_{i=1}^{10} (qx_i^2 + r\mu_i^2) \\ & \text{subject to} \quad x_k = 1.02x_{k-1} - \mu_k, k = 1, \dots, 10, x_0 = 10000, \end{aligned}$$

which is an instance of the LQR problem. The parameters q and r reflect our priority in trading off between debt reduction and hardship in making payments. The more anxious we are to reduce our debt, the larger the value of q relative to r . On the other hand, the more reluctant we are to make payments, the larger the value of r relative to q .

The solution to the problem above is given by the formula derived in Example 567. In Figure 6.24 we plot the monthly account balances and payments over the next 10 months using $q = 1$ and $r = 10$. We can see here that our debt has been reduced to less than

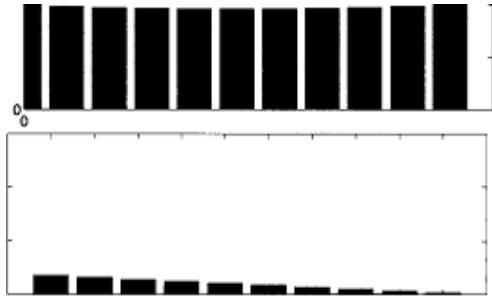


图 6.25: Plots for Example 568 with $q = 1$ and $r = 300$

\$1000 after 10 months, but with a first payment close to \$3000. If we feel that a payment of \$3000 is too high, then we can try to reduce this amount by increasing the value of r relative to q . However, going too far along these lines can lead to trouble. Indeed, if we use $q = 1$ and $r = 300$ (see Figure 6.25), although the monthly payments do not exceed \$400, the account balance is never reduced by much below \$10,000. In this case, the interest on the account balance eats up a significant portion of our monthly payments. In fact, our debt after 10 months will be higher than \$10,000.

6.7.7 Exercises

Exercise 569. Consider the following constraints on \mathbb{R}^2 : $h(x_1, x_2) = (x_1 - 2)^2 = 0$ and $g(x_1, x_2) = (x_2 + 1)^3 \leq 0$. Find the set of feasible points. Are the feasible points regular? Justify your answer.

Exercise 570. Find local extremizers for the following optimization problems:

1.

$$\begin{aligned} & \text{minimize} && x_1^2 + 2x_1x_2 + 3x_2^2 + 4x_1 + 5x_2 + 6x_3 \\ & \text{subject to} && x_1 + 2x_2 = 3 \\ & && 4x_1 + 5x_3 = 6. \end{aligned}$$

2.

$$\begin{aligned} & \text{maximize} && 4x_1 + x_2^2 \\ & \text{subject to} && x_1^2 + x_2^2 = 9. \end{aligned}$$

3.

$$\begin{aligned} & \text{maximize} && x_1x_2 \\ & \text{subject to} && x_1^2 + 4x_2^2 = 1. \end{aligned}$$

Exercise 571. Find minimizers and maximizers of the function

$$f(\mathbf{x}) = (\mathbf{a}^T \mathbf{x})(\mathbf{b}^T \mathbf{x}), \mathbf{x} \in \mathbb{R}^3,$$

subject to

$$x_1 + x_2 = 0$$

$$x_2 + x_3 = 0,$$

where

$$\mathbf{a} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \text{ and } \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}.$$

Exercise 572. Consider the problem

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && h(\mathbf{x}) = 0, \end{aligned}$$

where $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, $h : \mathbb{R}^2 \rightarrow \mathbb{R}$, and $\nabla f(\mathbf{x}) = [x_1, x_1 + 4]^T$. Suppose that \mathbf{x}^* is an optimal solution and $\nabla h(\mathbf{x}^*) = [1, 4]^T$. Find $\nabla f(\mathbf{x}^*)$.

Exercise 573. Consider the problem

$$\begin{aligned} & \text{minimize} && \|\mathbf{x} - \mathbf{x}_0\|^2 \\ & \text{subject to} && \|\mathbf{x}\|^2 = 9, \end{aligned}$$

where $\mathbf{x}_0 = [1, \sqrt{3}]^T$.

1. Find all points satisfying the Lagrange condition for the problem.

2. Using second-order conditions, determine whether or not each of the points in part a is a local minimizer.

Exercise 574. We wish to construct a closed box with minimum surface area that encloses a volume of V cubic feet, where $V > 0$.

1. Let a , b , and c denote the dimensions of the box with minimum surface area (with volume V). Derive the Lagrange condition that must be satisfied by a , b , and c .
2. What does it mean for a point \mathbf{x}^* to be a regular point in this problem? Is the point $\mathbf{x}^* = [a, b, c]^T$ a regular point?

3. Find a , b , and c .
4. Does the point $\mathbf{x}^* = [a, b, c]^T$ found in part c satisfy the second-order sufficient condition?

Exercise 575. Find local extremizers of

1. $f(x_1, x_2, x_3) = x_1^2 + 3x_2^2 + x_3$ subject to $x_1^2 + x_2^2 + x_3^2 = 16$;
2. $f(x_1, x_2) = x_1^2 + x_2^2$ subject to $3x_1^2 + 4x_1x_2 + 6x_2^2 = 140$.

Exercise 576. Consider the problem

$$\begin{aligned} & \text{minimize} && 2x_1 + 3x_2 - 4, x_1, x_2 \in \mathbb{R} \\ & \text{subject to} && x_1x_2 = 6. \end{aligned}$$

1. Use Lagrange's theorem to find all possible local minimizers and maximizers.
2. Use the second-order sufficient conditions to specify which points are strict local minimizers and which are strict local maximizers.
3. Are the points in part b global minimizers or maximizers? Explain.

Exercise 577. Find all maximizers of the function

$$f(x_1, x_2) = \frac{18x_1^2 - 8x_1x_2 + 12x_2^2}{2x_1^2 + 2x_2^2}.$$

Exercise 578. Find all solutions to the problem

$$\begin{aligned} & \text{maximize} && \mathbf{x}^T \begin{bmatrix} 3 & 4 \\ 0 & 3 \end{bmatrix} \mathbf{x} \\ & \text{subject to} && \|\mathbf{x}\|^2 = 1. \end{aligned}$$

Exercise 579. Consider a matrix \mathbf{A} with the property that $\mathbf{A}^T \mathbf{A}$ has eigenvalues ranging from 1 to 20 (i.e., the smallest eigenvalue is 1 and the largest is 20). Let \mathbf{x} be a vector such that $\|\mathbf{x}\| = 1$, and let $\mathbf{y} = \mathbf{A}\mathbf{x}$. Use Lagrange multiplier methods to find the range of values that $\|\mathbf{y}\|$ can take. Hint: What is the largest value that $\|\mathbf{y}\|$ can take? What is the smallest value that $\|\mathbf{y}\|$ can take?

Exercise 580. Consider a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$. Define the induced 2-norm of \mathbf{A} , denoted $\|\mathbf{A}\|^2$, to be the number

$$\|\mathbf{A}\|^2 = \max \{\|\mathbf{Ax}\| : \mathbf{x} \in \mathbb{R}^n, \|\mathbf{x}\| = 1\},$$

where the norm $\|\cdot\|$ on the right-hand side above is the usual Euclidean norm.

Suppose that the eigenvalues of $\mathbf{A}^T \mathbf{A}$ are $\lambda_1, \dots, \lambda_n$ (ordered from largest to smallest). Use Lagrange's theorem to express $\|\mathbf{A}\|^2$ in terms of the eigenvalues above (cf. Theorem 3.8 of [30]).

Exercise 581. Let $\mathbf{P} = \mathbf{P}^T$ be a positive definite matrix. Show that any point \mathbf{x} satisfying $1 - \mathbf{x}^T \mathbf{P} \mathbf{x} = 0$ is a regular point.

Exercise 582. Consider the problem

$$\begin{aligned} & \text{minimize} && x_1 x_2 - 2x_1, x_1, x_2 \in \mathbb{R} \\ & \text{subject to} && x_1^2 - x_2^2 = 0. \end{aligned}$$

1. Apply Lagrange's theorem directly to the problem to show that if a solution exists, it must be either $[1, 1]^T$ or $[-1, 1]^T$.
2. Use the second-order necessary conditions to show that $[-1, 1]^T$ cannot possibly be the solution.
3. Use the second-order sufficient conditions to show that $[1, 1]^T$ is a strict local minimizer.

Exercise 583. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m \leq n$, $\text{rank } \mathbf{A} = m$, and $\mathbf{x}_0 \in \mathbb{R}^n$. Let \mathbf{x}^* be the point on the nullspace of \mathbf{A} that is closest to \mathbf{x}_0 (in the sense of Euclidean norm).

1. Show that \mathbf{x}^* is orthogonal to $\mathbf{x}^* - \mathbf{x}_0$.
2. Find a formula for \mathbf{x}^* in terms of \mathbf{A} and \mathbf{x}_0 .

Exercise 584. Consider the problem

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|^2 \\ & \text{subject to} && \mathbf{Cx} = \mathbf{d}, \end{aligned}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m \geq n$, $\mathbf{C} \in \mathbb{R}^{p \times n}$, $p \leq n$, and both \mathbf{A} and \mathbf{C} are of full rank. We wish to find an expression for the solution (in terms of \mathbf{A} , \mathbf{b} , \mathbf{C} , and \mathbf{d}).

1. Apply Lagrange's theorem to solve this problem.
2. As an alternative, rewrite the given optimization problem in the form of a quadratic programming problem and apply the formula in Section 6.7.6 to obtain the solution.

Exercise 585. Consider the problem of minimizing a general quadratic function subject to a linear constraint:

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{c}^T \mathbf{x} + d \\ & \text{subject to} && \mathbf{A} \mathbf{x} = \mathbf{b}, \end{aligned}$$

where $\mathbf{Q} = \mathbf{Q}^T \geq 0$, $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m \leq n$, $\text{rank } \mathbf{A} = m$, and d is a constant. Derive a closed-form solution to the problem.

Exercise 586. Let \mathbf{L} be an $n \times n$ real symmetric matrix, and let \mathcal{M} be a subspace of \mathbb{R}^n with dimension $m \leq n$. Let $\{\mathbf{b}_1, \dots, \mathbf{b}_m\} \subset \mathbb{R}^n$ be a basis for \mathcal{M} , and let \mathbf{B} be the $n \times m$ matrix with \mathbf{b}_i as the i th column. Let $\mathbf{L}_{\mathcal{M}}$ be the $m \times m$ matrix defined by $\mathbf{L}_{\mathcal{M}} = \mathbf{B}^T \mathbf{L} \mathbf{B}$. Show that \mathbf{L} is positive semidefinite (definite) on \mathcal{M} if and only if $\mathbf{L}_{\mathcal{M}}$ is positive semidefinite (definite). Note: This result is useful for checking that the Hessian of the Lagrangian function at a point is positive definite on the tangent space at that point.

Exercise 587. Consider the sequence $\{x_k\}$, $x_k \in \mathbb{R}$, generated by the recursion

$$x_{k+1} = ax_k + b\mu_k, k \geq 0 \quad (a, b \in \mathbb{R}, a, b \neq 0)$$

where $\mu_0, \mu_1, \mu_2, \dots$ is a sequence of “control inputs,” and the initial condition $x_0 \neq 0$ is given. The recursion above is also called a discrete-time linear system. We wish to find values of control inputs μ_0 and μ_1 such that $x_2 = 0$, and the average input energy $(\mu_0^2 + \mu_1^2)/2$ is minimized. Denote the optimal inputs by μ_0^* and μ_1^* .

1. Find expressions for μ_0^* and μ_1^* in terms of a , b , and x_0 .
2. Use the second-order sufficient conditions to show that the point $\boldsymbol{\mu}^* = [\mu_0^*, \mu_1^*]^T$ in part a is a strict local minimizer.

Exercise 588. Consider the discrete-time linear system $x_k = 2x_{k-1} + \mu_k$, $k \geq 1$, with $x_0 = 1$. Find the values of the control inputs μ_1 and μ_2 to minimize

$$x_2^2 + \frac{1}{2}\mu_1^2 + \frac{1}{3}\mu_2^2.$$

Exercise 589. Consider the discrete-time linear system $x_{k+1} = x_k + 2\mu_k$, $0 \leq k \leq 2$, with $x_0 = 3$. Use the Lagrange multiplier approach to calculate the optimal control sequence $\{\mu_0, \mu_1, \mu_2\}$ that transfers the initial state x_0 to $x_3 = 9$ while minimizing

$$\frac{1}{2} \sum_{k=0}^2 \mu_k^2.$$

6.8 Problems With Inequality Constraints

(Taken from Chapter 21 of [30])

6.8.1 Karush-Kuhn-Tucker Condition

In Chapter 6.7 we analyzed constrained optimization problems involving only equality constraints. In this chapter we discuss extremum problems that also involve inequality constraints. The treatment in this chapter parallels that of Chapter 6.7. In particular, as we shall see, problems with inequality constraints can also be treated using Lagrange multipliers.

We consider the following problem:

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & && \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \end{aligned}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $m \leq n$, and $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^p$. For the general problem above, we adopt the following definitions.

Definition 590. An inequality constraint $g_j(\mathbf{x}) \leq 0$ is said to be active at \mathbf{x}^* if $g_j(\mathbf{x}^*) = 0$. It is inactive at \mathbf{x}^* if $g_j(\mathbf{x}^*) < 0$.

By convention, we consider an equality constraint $h_i(\mathbf{x}) = 0$ to be always active.

Definition 591. Let \mathbf{x}^* satisfy $\mathbf{h}(\mathbf{x}^*) = \mathbf{0}$, $\mathbf{g}(\mathbf{x}^*) \leq \mathbf{0}$, and let $J(\mathbf{x}^*)$ be the index set of active inequality constraints:

$$J(\mathbf{x}^*) \triangleq \{j : g_j(\mathbf{x}^*) = 0\}.$$

Then, we say that \mathbf{x}^* is a regular point if the vectors

$$\nabla h_i(\mathbf{x}^*), \nabla g_j(\mathbf{x}^*), 1 \leq i \leq m, j \in J(\mathbf{x}^*)$$

are linearly independent.

We now prove a first-order necessary condition for a point to be a local minimizer. We call this condition the Karush-Kuhn-Tucker (KKT) condition. In the literature, this condition is sometimes also called the Kuhn-Tucker condition.

Theorem 592. Karush-Kuhn-Tucker(KKT) Theorem Let $f, \mathbf{h}, \mathbf{g} \in \mathcal{C}^1$. Let \mathbf{x}^* be a regular point and a local minimizer for the problem of minimizing f subject to $\mathbf{h}(\mathbf{x}) = \mathbf{0}$, $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$. Then, there exist $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ and $\mu^* \in \mathbb{R}^p$ such that

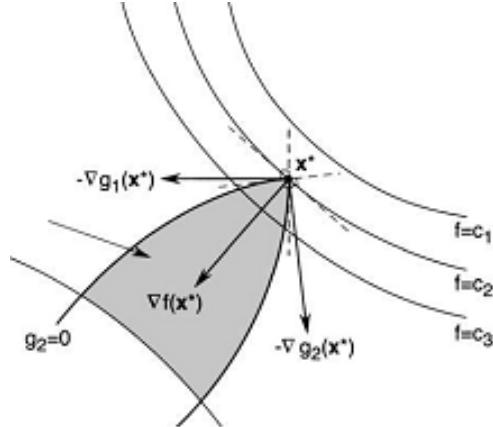


图 6.26: Illustration of the Karush-Kuhn-Tucker (KKT) theorem

1. $\mu^* \geq 0$.
2. $Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) + \mu^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T$.
3. $\mu^{*T} \mathbf{g}(\mathbf{x}^*) = 0$

In Theorem 592, we refer to $\boldsymbol{\lambda}^*$ as the Lagrange multiplier vector and μ as the Karush-Kuhn-Tucker (KKT) multiplier vector. We refer to their components as Lagrange multipliers and Karush-Kuhn-Tucker (KKT) multipliers, respectively.

Before proving this theorem, let us first discuss its meaning. Observe that $\mu_j \geq 0$ (by condition 1) and $g_j(\mathbf{x}^*) < 0$. Therefore, the condition $\mu^{*T} \mathbf{g}(\mathbf{x}^*) = \mu_1^* g_1(\mathbf{x}^*) + \dots + \mu_p^* g_p(\mathbf{x}^*) = 0$ implies that if $g_j(\mathbf{x}^*) < 0$, then $\mu_j^* = 0$; that is, for all $j \notin J(\mathbf{x}^*)$, we have $\mu_j^* = 0$. In other words, the KKT multipliers μ_j^* corresponding to inactive constraints are zero. The other KKT multipliers, μ_i^* , $i \in J(\mathbf{x}^*)$, are nonnegative; they may or may not be equal to zero.

Example 593. A graphical illustration of the KKT theorem is given in Figure 6.26. In this two-dimensional example, we have only inequality constraints $g_j(\mathbf{x}) \leq 0$, $j = 1, 2, 3$. Note that the point \mathbf{x}^* in the figure is indeed a minimizer. The constraint $g_3(\mathbf{x}) \leq 0$ is inactive: $g_3(\mathbf{x}^*) \leq 0$; hence $\mu_3^* = 0$. By the KKT theorem, we have

$$\nabla f(\mathbf{x}^*) + \mu_1^* \nabla g_1(\mathbf{x}^*) + \mu_2^* \nabla g_2(\mathbf{x}^*) = \mathbf{0},$$

or, equivalently,

$$\nabla f(\mathbf{x}^*) = -\mu_1^* \nabla g_1(\mathbf{x}^*) - \mu_2^* \nabla g_2(\mathbf{x}^*)$$

where $\mu_1^* \geq 0$ and $\mu_2^* \geq 0$. It is easy to interpret the KKT condition graphically for this example. Specifically, we can see from Figure 6.26 that $\nabla f(\mathbf{x}^*)$ must be a linear

combination of the vectors $-\nabla g_1(\mathbf{x}^*)$ and $-\nabla g_2(\mathbf{x}^*)$ with positive coefficients. This is reflected exactly in the equation above, where the coefficients μ_1^* and μ_2^* are the KKT multipliers.

We apply the KKT condition in the same way that we apply any necessary condition. Specifically, we search for points satisfying the KKT condition and treat these points as candidate minimizers. To summarize, the KKT condition consists of five parts (three equations and two inequalities):

1. $\boldsymbol{\mu}^* \geq \mathbf{0}$.
2. $Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T$.
3. $\boldsymbol{\mu}^{*T} \mathbf{g}(\mathbf{x}^*) = 0$.
4. $\mathbf{h}(\mathbf{x}^*) = \mathbf{0}$.
5. $\mathbf{g}(\mathbf{x}^*) \leq \mathbf{0}$.

We now prove the KKT theorem.

Proof. (Proof of the Karush-Kuhn-Tucker Theorem.) Let \mathbf{x}^* be a regular local minimizer of f on the set $\{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) < \mathbf{0}\}$. Then, \mathbf{x}^* is also a regular local minimizer of f on the set $\{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0}, g_j(\mathbf{x}) = 0, j \in J(\mathbf{x}^*)\}$. Note that the latter constraint set involves only equality constraints. Therefore, from Lagrange's theorem, it follows that there exist vectors $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ and $\boldsymbol{\mu}^* \in \mathbb{R}^p$ such that

$$Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T,$$

where for all $j \notin J(\mathbf{x}^*)$, we have $\mu_j^* = 0$. To complete the proof it remains to show that for all $j \in J(\mathbf{x}^*)$, we have $\mu_j^* \geq 0$ (and hence for all $j = 1, \dots, p$, we have $\mu_j^* \geq 0$, i.e., $\boldsymbol{\mu}^* \geq \mathbf{0}$). We use a proof by contradiction. So suppose that there exists $j \in J(\mathbf{x}^*)$ such that $\mu_j^* \leq 0$. Let \tilde{S} and $\tilde{T}(\mathbf{x}^*)$ be the surface and tangent space defined by all other active constraints at \mathbf{x}^* . Specifically,

$$\tilde{S} = \{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0}, g_i(\mathbf{x}) = 0, i \in J(\mathbf{x}^*), i \neq j\}$$

and

$$\tilde{T}(\mathbf{x}^*) = \{\mathbf{y} : D\mathbf{h}(\mathbf{x}^*) \mathbf{y} = \mathbf{0}, Dg_i(\mathbf{x}^*) \mathbf{y} = 0, i \in J(\mathbf{x}^*), i \neq j\}.$$

We claim that by the regularity of \mathbf{x}^* , there exists $\mathbf{y} \in \tilde{T}(\mathbf{x}^*)$ such that

$$Dg_j(\mathbf{x}^*) \mathbf{y} \neq 0.$$

To see this, suppose that for all $\mathbf{y} \in \tilde{T}(\mathbf{x}^*)$, $\nabla g_j(\mathbf{x}^*)^T \mathbf{y} = Dg_j(\mathbf{x}^*) \mathbf{y} = 0$. This implies that $\nabla g_j(\mathbf{x}^*) \in \tilde{T}(\mathbf{x}^*)^\perp$. By Lemma 557, this, in turn, implies that

$$\nabla g_j(\mathbf{x}^*) \in \text{span} [\nabla h_k(\mathbf{x}^*), k = 1, \dots, m, \nabla g_i(\mathbf{x}^*), i \in J(\mathbf{x}^*), i \neq j].$$

But this contradicts the fact that \mathbf{x}^* is a regular point, which proves our claim. Without loss of generality, we assume that we have \mathbf{y} such that $Dg_j(\mathbf{x}^*) \mathbf{y} \leq 0$.

Consider the Lagrange condition, rewritten as

$$Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) + \mu^* Dg_j(\mathbf{x}^*) + \sum_{i \neq j} \mu_i^* Dg_i(\mathbf{x}^*) = \mathbf{0}^T.$$

If we postmultiply the above by \mathbf{y} and use the fact that $\mathbf{y} \in \tilde{T}(\mathbf{x}^*)$, we get

$$Df(\mathbf{x}^*) \mathbf{y} = -\mu_j^* Dg_j(\mathbf{x}^*) \mathbf{y}.$$

Because $Dg_j(\mathbf{x}^*) \mathbf{y} \leq 0$ and we have assumed that $\mu_j^* \leq 0$, we have

$$Df(\mathbf{x}^*) \leq 0.$$

Because $\mathbf{y} \in \tilde{T}(\mathbf{x}^*)$, by Theorem 555 we can find a differentiable curve $\{\mathbf{x}(t) : t \in (a, b)\}$ on S such that there exists $t^* \in (a, b)$ with $\mathbf{x}(t^*) = b\mathbf{x}^*$ and $\dot{\mathbf{x}}(t^*) = \mathbf{y}$. Now,

$$\frac{d}{dt} f(\mathbf{x}(t^*)) = Df(\mathbf{x}^*) \mathbf{y} \leq 0.$$

The above means that there is a $\delta \geq 0$ such that for all $t \in (t^*, t^* + \delta]$, we have

$$f(\mathbf{x}(t)) \leq f(\mathbf{x}(t^*)) = f(\mathbf{x}^*).$$

On the other hand,

$$\frac{d}{dt} g_j(\mathbf{x}(t^*)) = Dg_j(\mathbf{x}^*) \mathbf{y} \leq 0,$$

and for some $\varepsilon \geq 0$ and all $t \in [t^*, t^* + \varepsilon]$, we have that $g_j(\mathbf{x}(t)) \leq 0$. Therefore, for all $t \in (t^*, t^* + \min\{\delta, \varepsilon\})$, we have that $g_j(\mathbf{x}(t)) \leq 0$ and $f(\mathbf{x}(t)) \leq f(\mathbf{x}^*)$. Because the points $\mathbf{x}(t)$, $t \in (t^*, t^* + \min\{\delta, \varepsilon\}]$, are in \tilde{S} , they are feasible points with lower objective function values than \mathbf{x}^* . This contradicts the assumption that \mathbf{x}^* is a local minimizer, which completes the proof. \square

Example 594. Consider the circuit in Figure 6.27. Formulate and solve the KKT condition for the following problems.

1. Find the value of the resistor $R \geq 0$ such that the power absorbed by this resistor is maximized.

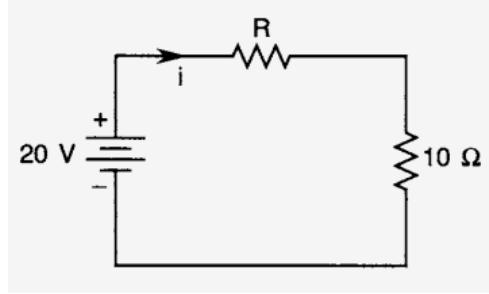


图 6.27: Circuit in Example 594

2. Find the value of the resistor $R \geq 0$ such that the power delivered to the $10 - \Omega$ resistor is maximized.

Solution:

1. The power absorbed by the resistor R is $p = i^2R$, where $i = \frac{20}{10+R}$. The optimization problem can be represented as

$$\begin{aligned} & \text{minimize} && -\frac{400R}{(10+R)^2} \\ & \text{subject to} && -R \leq 0. \end{aligned}$$

The derivative of the objective function is

$$-\frac{400(10+R)^2 - 800R(10+R)}{(10+R)^4} = \frac{400(10-R)}{(10+R)^3}.$$

Thus, the KKT condition is

$$\begin{aligned} & -\frac{400(10-R)}{(10+R)^3} - \mu = 0, \\ & \mu \geq 0, \\ & \mu R = 0, \\ & -R \leq 0. \end{aligned}$$

We consider two cases. In the first case, suppose that $\mu \geq 0$. Then, $R = 0$. But this contradicts the first condition above. Now suppose that $\mu = 0$. Then, by the first condition, we have $R = 10$. Therefore, the only solution to the KKT condition is $R = 10$, $\mu = 0$.

2. The power absorbed by the $10 - \Omega$ resistor is $p = i^210$, where $i = 20 / (10 + R)$. The

optimization problem can be represented as

$$\begin{aligned} & \text{minimize} && -\frac{4000}{(10+R)^2} \\ & \text{subject to} && -R \leq 0. \end{aligned}$$

The derivative of the objective function is

$$\frac{8000}{(10+R)^3}.$$

Thus, the KKT condition is

$$\begin{aligned} \frac{8000}{(10+R)^3} - \mu &= 0, \\ \mu &\geq 0, \\ \mu R &= 0, \\ -R &\leq 0. \end{aligned}$$

As before, we consider two cases. In the first case, suppose that $\mu \geq 0$. Then, $R = 0$, which is feasible. For the second case, suppose that $\mu = 0$. But this contradicts the first condition. Therefore, the only solution to the KKT condition is $R = 0, \mu = 8$.

In the case when the objective function is to be maximized, that is, when the optimization problem has the form

$$\begin{aligned} & \text{maximize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & && \mathbf{g}(\mathbf{x}) \leq \mathbf{0}, \end{aligned}$$

the KKT condition can be written as

1. $\boldsymbol{\mu}^* \geq \mathbf{0}$.
2. $-Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T$.
3. $\boldsymbol{\mu}^{*T} \mathbf{g}(\mathbf{x}^*) = 0$.
4. $\mathbf{h}(\mathbf{x}^*) = \mathbf{0}$.
5. $\mathbf{g}(\mathbf{x}^*) \leq \mathbf{0}$.

The above is easily derived by converting the maximization problem above into a minimization problem, by multiplying the objective function by -1 . It can be further rewritten as

1. $\mu^* \leq \mathbf{0}$.
2. $Df(\mathbf{x}^*) + \lambda^{*T} D\mathbf{h}(\mathbf{x}^*) + \mu^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T$.
3. $\mu^{*T} \mathbf{g}(\mathbf{x}^*) = 0$.
4. $\mathbf{h}(\mathbf{x}^*) = \mathbf{0}$.
5. $\mathbf{g}(\mathbf{x}^*) \leq \mathbf{0}$.

The form shown above is obtained from the preceding one by changing the signs of μ^* and λ^* and multiplying condition 2 by -1 .

We can similarly derive the KKT condition for the case when the inequality constraint is of the form $\mathbf{g}(\mathbf{x}) \geq \mathbf{0}$. Specifically, consider the problem

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & && \mathbf{g}(\mathbf{x}) \geq \mathbf{0}. \end{aligned}$$

We multiply the inequality constraint function by -1 to obtain $-\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$. Thus, the KKT condition for this case is

1. $\mu^* \geq \mathbf{0}$.
2. $Df(\mathbf{x}^*) + \lambda^{*T} D\mathbf{h}(\mathbf{x}^*) - \mu^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T$.
3. $\mu^{*T} \mathbf{g}(\mathbf{x}^*) = 0$.
4. $\mathbf{h}(\mathbf{x}^*) = \mathbf{0}$.
5. $\mathbf{g}(\mathbf{x}^*) \geq \mathbf{0}$.

Changing the sign of μ^* as before, we obtain

1. $\mu^* \leq \mathbf{0}$.
2. $Df(\mathbf{x}^*) + \lambda^{*T} D\mathbf{h}(\mathbf{x}^*) + \mu^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T$.
3. $\mu^{*T} \mathbf{g}(\mathbf{x}^*) = 0$.
4. $\mathbf{h}(\mathbf{x}^*) = \mathbf{0}$.
5. $\mathbf{g}(\mathbf{x}^*) \geq \mathbf{0}$.

For the problem

$$\begin{aligned} & \text{maximize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & && \mathbf{g}(\mathbf{x}) \geq \mathbf{0}, \end{aligned}$$

the KKT condition is exactly the same as in Theorem 592, except for the reversal of the inequality constraint.

Example 595. In Figure 6.28, the two points \mathbf{x}_1 and \mathbf{x}_2 are feasible points; that is, $\mathbf{g}(\mathbf{x}_1) \geq \mathbf{0}$ and $\mathbf{g}(\mathbf{x}_2) \geq \mathbf{0}$, and they satisfy the KKT condition.

The point \mathbf{x}_1 is a maximizer. The KKT condition for this point (with KKT multiplier μ_1) is

1. $\mu_1 \geq 0$.
2. $\nabla f(\mathbf{x}_1) + \mu_1 \nabla g(\mathbf{x}_1) = \mathbf{0}$.
3. $\mu_1 g(\mathbf{x}_1) = 0$.
4. $g(\mathbf{x}_1) \geq 0$.

The point \mathbf{x}_2 is a minimizer of f . The KKT condition for this point (with KKT multiplier μ_2) is

1. $\mu_2 \leq 0$.
2. $\nabla f(\mathbf{x}_2) + \mu_2 \nabla g(\mathbf{x}_2) = \mathbf{0}$.
3. $\mu_2 g(\mathbf{x}_2) = 0$.
4. $g(\mathbf{x}_2) \geq 0$.

Example 596. Consider the problem

$$\begin{aligned} & \text{minimize} && f(x_1, x_2) \\ & \text{subject to} && x_1, x_2 \geq 0, \end{aligned}$$

where

$$f(x_1, x_2) = x_1^2 + x_2^2 + x_1 x_2 - 3x_1.$$

The KKT condition for this problem is

$$1. \boldsymbol{\mu} = [\mu_1, \mu_2]^T \leq \mathbf{0}.$$

$$2. Df(\mathbf{x}) + \boldsymbol{\mu}^T = \mathbf{0}^T.$$

$$3. \boldsymbol{\mu}^T \mathbf{x} = 0.$$

$$4. \mathbf{x} \geq \mathbf{0}.$$

We have

$$Df(\mathbf{x}) = [2x_1 + x_2 - 3, x_1 + 2x_2].$$

This gives

$$2x_1 + x_2 + \mu_1 = 3,$$

$$x_1 + 2x_2 + \mu_2 = 0,$$

$$\mu_1 x_1 + \mu_2 x_2 = 0.$$

We now have four variables, three equations, and the inequality constraints on each variable. To find a solution $(\mathbf{x}^*, \boldsymbol{\mu}^*)$, we first try

$$\mu_1^* = 0, x_2^* = 0,$$

which gives

$$x_1^* = \frac{3}{2}, \mu_2^* = -\frac{3}{2}.$$

The above satisfies all the KKT and feasibility conditions. In a similar fashion, we can try

$$\mu_2^* = 0, x_1^* = 0,$$

which gives

$$x_2^* = 0, \mu_1^* = 3.$$

This point clearly violates the nonpositivity constraint on μ_1^* .

The feasible point above satisfying the KKT condition is only a candidate for a minimizer. However, there is no guarantee that the point is indeed a minimizer, because the KKT condition is, in general, only necessary. A sufficient condition for a point to be a minimizer is given in the next section.

Example 596 is a special case of a more general problem of the form

$$\text{minimize } f(\mathbf{x})$$

$$\text{subject to } \mathbf{x} \geq \mathbf{0}.$$

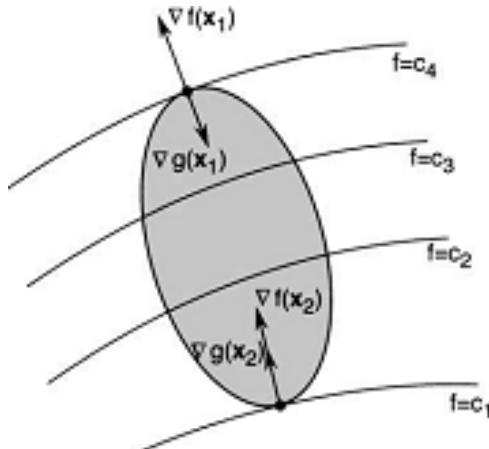


图 6.28: Points satisfying the KKT condition (x_1 is a maximizer and x_2 is a minimizer)

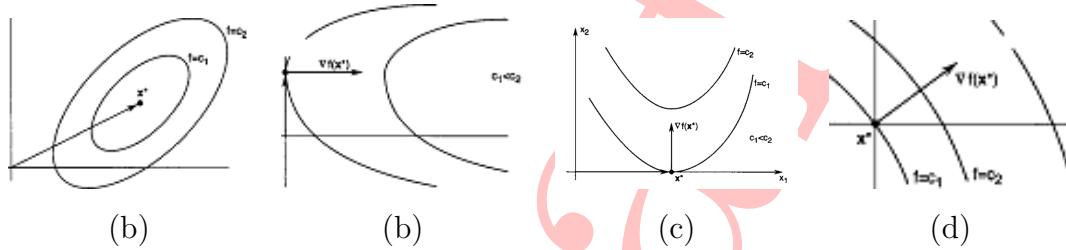


图 6.29: Some possible points satisfying the KKT condition for problems with positive constraints.

The KKT condition for this problem has the form

$$\begin{aligned} \mu &\leq 0, \\ \nabla f(\mathbf{x}) + \mu &= 0, \\ \mu^T \mathbf{x} &= 0, \\ \mathbf{x} &\geq 0. \end{aligned}$$

From the above, we can eliminate μ to obtain

$$\begin{aligned} \nabla f(\mathbf{x}) &\geq \mathbf{0}, \\ \mathbf{x}^T \nabla f(\mathbf{x}) &= 0, \\ \mathbf{x} &\geq \mathbf{0}. \end{aligned}$$

Some possible points in \mathbb{R}^2 that satisfy these conditions are depicted in Figure 6.29.

6.8.1.1 Using KKT Conditions to Deduce Stopping Criteria

Use the example of LADM, see Section 7.4.

6.8.2 Second-Order Conditions

As in the case of extremum problems with equality constraints, we can also give second-order necessary and sufficient conditions for extremum problems involving inequality constraints. For this, we need to define the following matrix:

$$\mathbf{L}(\mathbf{x}, \boldsymbol{\lambda}, \mu) = \mathbf{F}(\mathbf{x}) + [\boldsymbol{\lambda}\mathbf{H}(\mathbf{x})] + [\mu\mathbf{G}(\mathbf{x})],$$

where $\mathbf{F}(\mathbf{x})$ is the Hessian matrix of f at \mathbf{x} , and the notation $[\boldsymbol{\lambda}\mathbf{H}(\mathbf{x})]$ represents

$$[\boldsymbol{\lambda}\mathbf{H}(\mathbf{x})] = \lambda_1\mathbf{H}_1(\mathbf{x}) + \cdots + \lambda_m\mathbf{H}_m(\mathbf{x})$$

as before. Similarly, the notation $[\mu\mathbf{G}(\mathbf{x})]$ represents

$$[\mu\mathbf{G}(\mathbf{x})] = \mu_1\mathbf{G}_1(\mathbf{x}) + \cdots + \mu_p\mathbf{G}_p(\mathbf{x})$$

where $\mathbf{G}_k(\mathbf{x})$ is the Hessian of g_k at \mathbf{x} , given by

$$\mathbf{G}_k(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 g_k}{\partial x_1^2}(\mathbf{x}) & \cdots & \frac{\partial^2 g_k}{\partial x_n \partial x_1}(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 g_k}{\partial x_1 \partial x_n}(\mathbf{x}) & \cdots & \frac{\partial^2 g_k}{\partial x_n^2}(\mathbf{x}) \end{bmatrix}.$$

In the following theorem, we use

$$T(\mathbf{x}^*) = \{\mathbf{y} \in \mathbb{R}^n : D\mathbf{h}(\mathbf{x}^*) = \mathbf{0}, Dg_j(\mathbf{x}^*)\mathbf{y} = 0, j \in J(\mathbf{x}^*)\},$$

that is, the tangent space to the surface defined by active constraints.

Theorem 597. Second-Order Necessary Conditions Let \mathbf{x}^* be a local minimizer of $f : \mathbb{R}^n \rightarrow \mathbb{R}$ subject to $\mathbf{h}(\mathbf{x}) = \mathbf{0}$, $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$, $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m, m \leq n$, $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^p$, and $f, \mathbf{h}, \mathbf{g} \in \mathcal{C}^2$. Suppose that \mathbf{x}^* is regular. Then, there exists $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ and $\boldsymbol{\mu}^* \in \mathbb{R}^p$ such that:

1. $\boldsymbol{\mu}^* \geq \mathbf{0}, Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T}D\mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^{*T}D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T, \boldsymbol{\mu}^{*T}\mathbf{g}(\mathbf{x}^*) = 0$.
2. For all $\mathbf{y} \in T(\mathbf{x}^*)$ we have $\mathbf{y}^T \mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) \mathbf{y} \geq 0$.

Proof. Part 1 is simply a result of the KKT theorem. To prove part 2, we note that because the point \mathbf{x}^* is a local minimizer over $\{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}$, it is also a local minimizer over $\{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0}, g_j(\mathbf{x}) = 0, j \in J(\mathbf{x}^*)\}$; that is, the point \mathbf{x}^* is a local minimizer with active constraints taken as equality constraints. Hence, the second-order necessary conditions for equality constraints (Theorem 20.4 of [30]) are applicable here, which completes the proof. \square

We now state the second-order sufficient conditions for extremum problems involving inequality constraints. In the formulation of the result, we use the following set:

$$\tilde{T}(\mathbf{x}^*, \boldsymbol{\mu}^*) = \left\{ \mathbf{y} : D\mathbf{h}(\mathbf{x}^*) \mathbf{y} = \mathbf{0}, Dg_i(\mathbf{x}^*) \mathbf{y} = 0, i \in \tilde{J}(\mathbf{x}^*, \boldsymbol{\mu}^*) \right\}$$

where $\tilde{J}(\mathbf{x}^*, \boldsymbol{\mu}^*) = \{i : g_i(\mathbf{x}^*) = 0, \mu_i^* \geq 0\}$. Note that $\tilde{J}(\mathbf{x}^*, \boldsymbol{\mu}^*)$ is a subset of $J(\mathbf{x}^*) : (\mathbf{x}^*, \boldsymbol{\mu}^*) \subset J(\mathbf{x}^*)$. This, in turn, implies that $T(\mathbf{x}^*)$ is a subset of $\tilde{T}(\mathbf{x}^*, \boldsymbol{\mu}^*) : T(\mathbf{x}^*) \subset \tilde{T}(\mathbf{x}^*, \boldsymbol{\mu}^*)$.

Theorem 598. *Second-Order Sufficient Conditions Suppose that $f, \mathbf{g}, \mathbf{h} \in \mathcal{C}^2$ and there exist a feasible point $\mathbf{x}^* \in \mathbb{R}^n$ and vectors $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ and $\boldsymbol{\mu}^* \in \mathbb{R}^p$ such that:*

1. $\boldsymbol{\mu}^* \geq \mathbf{0}, Df(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T, \boldsymbol{\mu}^{*T} \mathbf{g}(\mathbf{x}^*) = 0$.
2. For all $\mathbf{y} \in \tilde{T}(\mathbf{x}^*, \boldsymbol{\mu}^*), \mathbf{y} \notin \mathbf{0}$, we have $\mathbf{y}^T \mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) \mathbf{y} \geq 0$.

The, \mathbf{x}^* is a strict local minimizer of f subject to $\mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \leq \mathbf{0}$.

A result similar to Theorem 598 holds for a strict local maximizer, the only difference being that we need $\boldsymbol{\mu}^* \leq \mathbf{0}$ and that $\mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ be negative definite on $\tilde{T}(\mathbf{x}^*, \boldsymbol{\mu}^*)$.

Example 599. Consider the following problem:

$$\begin{aligned} & \text{minimize} && x_1 x_2 \\ & \text{subject to} && x_1 + x_2 \geq 2 \\ & && x_2 \geq x_1. \end{aligned}$$

1. Write down the KKT condition for this problem.
2. Find all points (and KKT multipliers) satisfying the KKT condition. In each case, determine if the point is regular.
3. Find all points in part b that also satisfy the Second Order Necessary Conditions (SONC).
4. Find all points in part c that also satisfy the Second Order Sufficient Conditions (SOSC).
5. Find all points in part d that are local minimizers.

Solution:

1. Write $f(\mathbf{x}) = x_1 x_2$, $g_1(\mathbf{x}) = 2 - x_1 - x_2$, and $g_2(\mathbf{x}) = x_1 - x_2$. The KKT condition is

$$\begin{aligned} x_2 - \mu_1 + \mu_2 &= 0, \\ x_1 - \mu_1 - \mu_2 &= 0, \\ \mu_1(2 - x_1 - x_2) + \mu_2(x_1 + x_2) &= 0, \\ \mu_1\mu_2 &\geq 0, \\ 2 - x_1 - x_2 &\leq 0, \\ x_1 - x_2 &\leq 0. \end{aligned}$$

2. It is easy to check that $\mu_1 \neq 0$ and $\mu_2 \neq 0$. This leave us with only one solution to the KKT condition: $x_1^* = x_2^* = 1$, $\mu_1^* = 1$, $\mu_2^* = 0$. For this point, we have $Dg_1(\mathbf{x}^*) = [-1, -1]$ and $Dg_2(\mathbf{x}^*) = [1, -1]$. Hence, \mathbf{x}^* is regular.
3. Both constraints are active. Hence, because \mathbf{x}^* is regular. $T(\mathbf{x}^*) = \{\mathbf{0}\}$. This implies that the SONC is satisfied.

4. Now,

$$\mathbf{L}(\mathbf{x}^*, \boldsymbol{\mu}^*) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

Moreover, $\tilde{T}(\mathbf{x}^*, \boldsymbol{\mu}^*) = \{\mathbf{y} : [-1, -1] \mathbf{y} = 0\} = \{\mathbf{y} : y_1 = -y_2\}$. Pick $\mathbf{y} = [-1, -1]^T \in \tilde{T}(\mathbf{x}^*, \boldsymbol{\mu}^*)$. We have $\mathbf{y}^T \mathbf{L}(\mathbf{x}^*, \boldsymbol{\mu}^*) \mathbf{y} = -2 \leq 0$, which means that the SOSC fails.

5. In fact, the point \mathbf{x}^* is not a local minimizer. To see this, draw a picture of the constraint set and level sets of the objective function. Moving in the feasible direction $[1, 1]^T$, the objective function increases; but moving in the feasible direction $[-1, 1]^T$, the objective function decreases.

We now solve analytically the problem in Example 547 that we solved graphically earlier.

Example 600. We wish to minimize $f(\mathbf{x}) = (x_1 - 1)^2 + x_2 - 2$ subject to

$$\begin{aligned} h(\mathbf{x}) &= x_2 - x_1 - 1 = 0, \\ g(\mathbf{x}) &= x_1 + x_2 - 2 \leq 0. \end{aligned}$$

For all $\mathbf{x} \in \mathbb{R}^2$, we have

$$Dh(\mathbf{x}) = [-1, 1], Dg(\mathbf{x}) = [1, 1].$$

Thus, $\nabla h(\mathbf{x})$ and $\nabla g(\mathbf{x})$ are linearly independent and hence all feasible points are regular. We first write the KKT condition. Because $Df(\mathbf{x}) = [2x_1 - 2, 1]$, we have

$$\begin{aligned} Df(\mathbf{x}) + \lambda Dh(\mathbf{x}) + \mu Dg(\mathbf{x}) &= [2x_1 - 2 - \lambda + \mu, 1 + \lambda + \mu] = \mathbf{0}^T, \\ \mu(x_1 + x_2 - 2) &= 0, \\ \mu &\geq 0, \\ x_2 - x_1 - 1 &= 0, \\ x_1 + x_2 - 2 &\leq 0. \end{aligned}$$

To find points that satisfy the conditions above, we first try $\mu \geq 0$, which implies that $x_1 + x_2 - 2 = 0$. Thus, we are faced with a system of four linear equations

$$\begin{aligned} 2x_1 - 2 - \lambda + \mu &= 0, \\ 1 + \lambda + \mu &= 0. \\ x_2 - x_1 - 1 &= 0. \\ x_1 + x_2 - 2 &= 0. \end{aligned}$$

Solving the system of equations above, we obtain

$$x_1 = \frac{1}{2}, x_2 = \frac{3}{2}, \lambda = -1, \mu = 0.$$

However, the above is not a legitimate solution to the KKT condition, because we obtained $\mu = 0$, which contradicts the assumption that $\mu \geq 0$.

In the second try, we assume that $\mu = 0$. Thus, we have to solve the system of equations

$$\begin{aligned} 2x_1 - 2 - \lambda &= 0, \\ 1 + \lambda &= 0, \\ x_2 + x_1 - 1 &= 0, \end{aligned}$$

and the solutions must satisfy

$$g(x_1, x_2) = x_1 + x_2 - 2 \leq 0.$$

Solving the equations above, we obtain

$$x_1 = \frac{1}{2}, x_2 = \frac{3}{2}, \lambda = -1.$$

Note that $\mathbf{x}^* = [1/2, 3/2]^T$ satisfies the constraint $g(\mathbf{x}^*) < 0$. The point \mathbf{x}^* satisfying the KKT necessary condition is therefore the candidate for being a minimizer.

We now verify if $\mathbf{x}^* = [1/2, 3/2]^T$, $\lambda^* = -1$, satisfy the second-order sufficient conditions. For this, we form the matrix

$$\begin{aligned}\mathbf{L}(\mathbf{x}^*, \lambda^*, \mu^*) &= \mathbf{F}(\mathbf{x}^*) + \lambda^* \mathbf{H}(\mathbf{x}^*) + \mu^* \mathbf{G}(\mathbf{x}^*) \\ &= \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} + (-1) \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} + (0) \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}\end{aligned}$$

We then find the subspace

$$\tilde{T}(\mathbf{x}^*, \mu^*) = \{\mathbf{y} : Dh(\mathbf{x}^*)\mathbf{y} = 0\}$$

Note that because $\mu^* = 0$, the active constraint $g(\mathbf{x}^*)$ does not enter the computation of $\tilde{T}(\mathbf{x}^*, \mu^*) = \{\mathbf{y} : Dh(\mathbf{x}^*)\mathbf{y} = 0\}$. We have

$$\tilde{T}(\mathbf{x}^*, \mu^*) = \{\mathbf{y} : [-1, 1]\mathbf{y} = 0\} = \left\{ [a, a]^T : a \in \mathbb{R} \right\}.$$

We then check for positive definiteness of $\mathbf{L}(\mathbf{x}^*, \lambda^*, \mu^*)$ on $\tilde{T}(\mathbf{x}^*, \mu^*)$. We have

$$\mathbf{y}^T \mathbf{L}(\mathbf{x}^*, \lambda^*, \mu^*) \mathbf{y} = [a, a] \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} a \\ a \end{bmatrix} = 2a^2.$$

Thus, $\mathbf{L}(\mathbf{x}^*, \lambda^*, \mu^*)$ is positive definite on $\tilde{T}(\mathbf{x}^*, \mu^*)$. Observe that $\mathbf{L}(\mathbf{x}^*, \lambda^*, \mu^*)$ is, in fact, only positive semidefinite on \mathbb{R}^2 .

By the second-order sufficient conditions, we conclude that $\mathbf{x}^* = [1/2, 3/2]^T$ is a strict local minimizer.

6.8.3 Exercises

Exercise 601. Consider the optimization problem

$$\begin{aligned}&\text{minimize} && x_1^2 + 4x_2^2 \\ &\text{subject to} && x_1^2 + 2x_2^2 \geq 4.\end{aligned}$$

1. Find all the points that satisfy the KKT conditions.
2. Apply the SOSC to determine the nature of the critical points from the previous part.

Exercise 602. Find local extremizers for:

$$1. \ x_1^2 + x_2^2 - 2x_1 - 10x_2 + 26 \text{ subject to } \frac{1}{5}x_2 - x_1^2 \leq 0, 5x_1 + \frac{1}{2}x_2 \leq 5.$$

$$2. \ x_1^2 + x_2^2 \text{ subject to } x_1 \geq 0, x_2 \geq 0, x_1 + x_2 \geq 5.$$

$$3. \ x_1^2 + 6x_1x_2 - 4x_1 - 2x_2 \text{ subject to } x_1^2 + 2x_2 \leq 1, 2x_1 - 2x_2 \leq 1.$$

Exercise 603. Find local minimizers for $x_1^2 + x_2^2$ subject to $x_1^2 + 2x_1x_2 + x_2^2 = 1, x_1^2 - x_2 \leq 0$.

Exercise 604. Write down the Karush-Kuhn-Tucker condition for the optimization problem in Exercise 15.8 of [30].

Exercise 605. Consider the problem

$$\begin{aligned} & \text{minimize} && x_2 - (x_1 - 2)^3 + 3 \\ & \text{subject to} && x_2 \geq 1, \end{aligned}$$

where x_1 and x_2 are real variables. Answer each of the following questions, making sure that you give complete reasoning for your answers.

1. Write down the KKT condition for the problem, and find all points that satisfy the condition. Check whether or not each point is regular.
2. Determine whether or not the point(s) in part a satisfy the second-order necessary condition.
3. Determine whether or not the point(s) in part b satisfy the second-order sufficient condition.

Exercise 606. Consider the problem

$$\begin{aligned} & \text{minimize} && x_2 \\ & \text{subject to} && x_2 \geq -(x_1 - 1)^2 + 3. \end{aligned}$$

1. Find all points satisfying the KKT condition for the problem.
2. For each point \mathbf{x}^* in part a, find $T(\mathbf{x}^*)$, $N(\mathbf{x}^*)$, and $T(\mathbf{x}^*)$.
3. Find the subset of points from part a that satisfy the second-order necessary condition.

Exercise 607. Consider the problem of optimizing (either minimizing or maximizing) $(x_1 - 2)^2 + (x_2 - 1)^2$ subject to

$$x_2 - x_1^2 \geq 0$$

$$2 - x_1 - x_2 \geq 0$$

$$x_1 \geq 0.$$

The point $\mathbf{x}^* = \mathbf{0}$ satisfies the KKT conditions.

1. Does \mathbf{x}^* satisfy the FONC for minimization or maximization? What are the KKT multipliers?
2. Does \mathbf{x}^* satisfy the SOSC? Carefully justify your answer.

Exercise 608. Consider the problem

$$\begin{aligned} & \text{minimize } f(\mathbf{x}) \\ & \text{subject to } \mathbf{x} \in \Omega, \end{aligned}$$

where $f(\mathbf{x}) = x_1 x_2^2$, where $\mathbf{x} = [x_1, x_2]^T$, and $\Omega = \{\mathbf{x} \in \mathbb{R}^2 : x_1 = x_2, x_1 \geq 0\}$.

1. Find all points satisfying the KKT condition.
2. Do each of the points found in part a satisfy the second-order necessary condition?
3. Do each of the points found in part a satisfy the second-order sufficient condition?

Exercise 609. Consider the problem

$$\begin{aligned} & \text{minimize } \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|^2 \\ & \text{subject to } x_1 + \cdots + x_n = 1, \\ & \quad x_1, \dots, x_n \geq 0. \end{aligned}$$

1. Write down the KKT condition for the problem.
2. Define what it means for a feasible point \mathbf{x}^* to be regular in this particular problem. Are there any feasible points in this problem that are not regular? If yes, find them. If not, explain why not.

Exercise 610. Let $g : \mathbb{R}^n \rightarrow \mathbb{R}$ and $\mathbf{x}_0 \in \mathbb{R}^n$ be given, where $g(\mathbf{x}_0) \geq 0$. Consider the problem

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \|\mathbf{x} - \mathbf{x}_0\|^2 \\ & \text{subject to} && g(\mathbf{x}) \leq 0. \end{aligned}$$

Suppose that \mathbf{x}^* is a solution to the problem and $g \in \mathcal{C}^1$. Use the KKT theorem to decide which of the following equations/inequalities hold:

1. $g(\mathbf{x}^*) \leq 0$.
2. $g(\mathbf{x}^*) = 0$.
3. $(\mathbf{x}^* - \mathbf{x}_0)^T \nabla g(\mathbf{x}^*) \leq 0$.
4. $(\mathbf{x}^* - \mathbf{x}_0)^T \nabla g(\mathbf{x}^*) = 0$.
5. $(\mathbf{x}^* - \mathbf{x}_0)^T \nabla g(\mathbf{x}^*) \geq 0$.

Exercise 611. Consider a square room with corners located at $[0, 0]^T$, $[0, 2]^T$, $[2, 0]^T$, and $[2, 2]^T$ (in \mathbb{R}^2). We wish to find the point in the room that is closest to the point $[3, 4]^T$.

1. Guess which point in the room is the closest point in the room to the point $[3, 4]^T$.
2. Use the second-order sufficient conditions to prove that the point you have guessed is a strict local minimizer.

Hint: Minimizing the distance is the same as minimizing the square distance.

Exercise 612. Consider the quadratic programming problem

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} \\ & \text{subject to} && \mathbf{A} \mathbf{x} \leq \mathbf{b}, \end{aligned}$$

where $\mathbf{Q} = \mathbf{Q}^T \geq 0$, $\mathbf{A} \in \mathbb{R}^{m \times n}$, and $\mathbf{b} \geq \mathbf{0}$. Find all points satisfying the KKT condition.

Exercise 613. Consider the linear programming problem

$$\begin{aligned} & \text{minimize} && ax_1 + bx_2 \\ & \text{subject to} && cx_1 + dx_2 = e \\ & && x_1, x_2 \geq 0, \end{aligned}$$

where $a, b, c, d, e \in \mathbb{R}$ are all nonzero constants. Suppose that \mathbf{x}^* is an optimal basic feasible solution to the problem.

1. Write down the Karush-Kuhn-Tucker condition involving \mathbf{x}^* (specifying clearly the number of Lagrange and KKT multipliers).
2. Is \mathbf{x}^* regular? Explain.
3. Find the tangent space $T(\mathbf{x}^*)$ (defined by the active constraints) for this problem.
4. Assume that the relative cost coefficients of all nonbasic variables are strictly positive. Does \mathbf{x}^* satisfy the second-order sufficient condition? Explain.

Exercise 614. Consider the problem

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} \\ & \text{subject to} && \mathbf{A} \mathbf{x} \leq \mathbf{0}, \end{aligned}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m \leq n$, is of full rank. Use the KKT theorem to show that if there exists a solution, then the optimal objective function value is 0.

Exercise 615. Consider a linear programming problem in standard form.

1. Write down the Karush-Kuhn-Tucker condition for the problem.
2. Use part a to show that if there exists an optimal feasible solution to the linear program, then there exists a feasible solution to the corresponding dual problem that achieves an objective function value that is the same as the optimal value of the primal (compare this with Theorem 17.1 of [30]).
3. Use parts a and b to prove that if \mathbf{x}^* is an optimal feasible solutions of the primal, then there exists a feasible solution $\boldsymbol{\lambda}^*$ to the dual such that $(\mathbf{c}^T - \boldsymbol{\lambda}^{*T} \mathbf{A}) \mathbf{x}^* = 0$ (compare this with Theorem 17.3 of [30]).

Exercise 616. Consider the constraint set $S = \{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}$. Let $\mathbf{x}^* \in S$ be a regular local minimizer of f over S and $J(\mathbf{x}^*)$ the index set of active inequality constraints. Show that \mathbf{x}^* is also a regular local minimizer of over the set $S' = \{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0}, g_j(\mathbf{x}) = 0, j \in J(\mathbf{x}^*)\}$.

Exercise 617. Solve the following optimization problem using the second-order sufficient conditions:

$$\begin{aligned} & \text{minimize} && x_1^2 + x_2^2 \\ & \text{subject to} && x_1^2 - x_2 - 4 \leq 0 \\ & && x_2 - x_1 - 2 \leq 0. \end{aligned}$$

See Figure 6.30 for a graphical illustration of the problem.

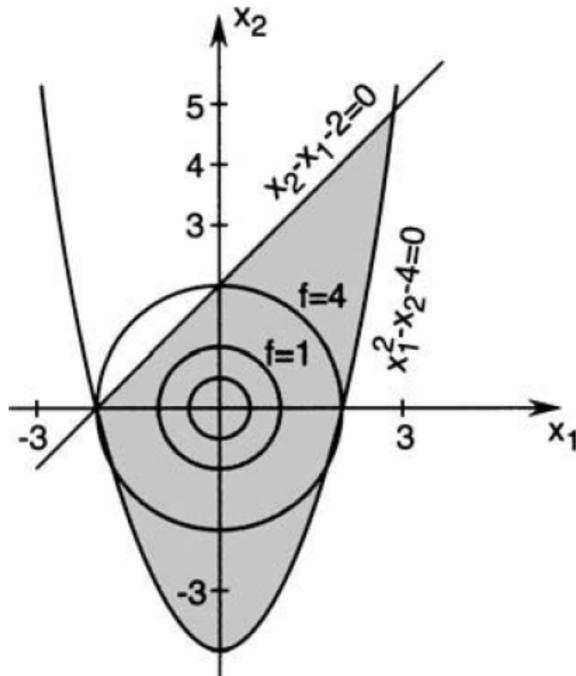


图 6.30: Illustration of the function in Exercise 617.

Exercise 618. Solve the following optimization problem using the second-order sufficient conditions:

$$\begin{aligned} & \text{minimize} && x_1^2 + x_2^2 \\ & \text{subject to} && x_1 - x_2^2 - 4 \geq 0 \\ & && x_1 - 10 \leq 0. \end{aligned}$$

See Figure 6.31 for a graphical illustration of the problem.

Exercise 619. Consider the problem

$$\begin{aligned} & \text{minimize} && x_1^2 + x_2^2 \\ & \text{subject to} && 4 - x_1 - x_2^2 \leq 0 \\ & && 3x_2 - x_1 \leq 0. \\ & && -3x_2 - x_1 \leq 0. \end{aligned}$$

Figure 6.32 gives a graphical illustration of the problem. Deduce from the figure that the problem has two strict local minimizers, and use the second-order sufficient conditions to verify the graphical solutions.

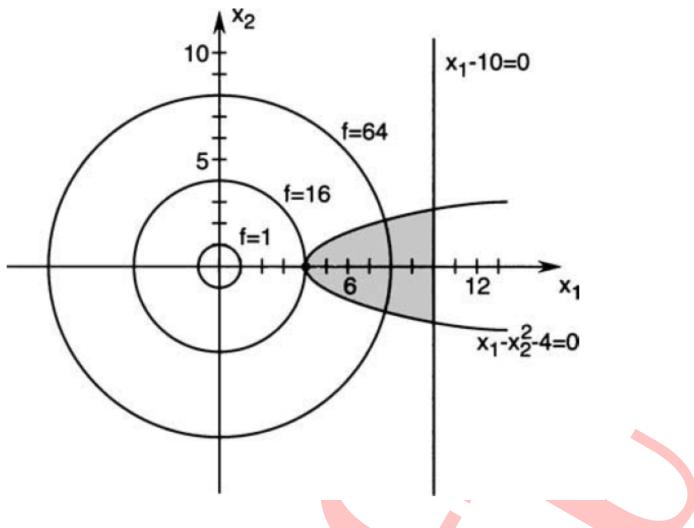


图 6.31: Illustration of the function in Exercise 618.

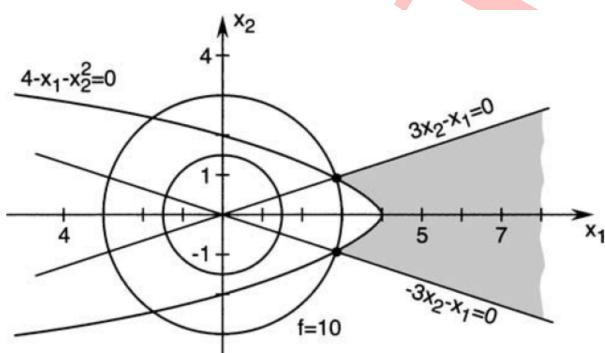


图 6.32: Illustration of the function in Exercise 619.

Exercise 620. Consider the following optimization problem with an inequality constraint:

$$\begin{aligned} & \text{minimize} && 3x_1 \\ & \text{subject to} && x_1 + x_2^2 \geq 2. \end{aligned}$$

1. Does the point $\mathbf{x}^* = [2, 0]^T$ satisfy the KKT (first-order necessary) condition?
2. Does the point $\mathbf{x}^* = [2, 0]^T$ satisfy the second-order necessary condition (for problems with inequality constraints)?
3. Is the point $\mathbf{x}^* = [2, 0]^T$ a local minimizer?

(See Exercise 6.15 of [30] for a similar problem treated using set-constrained methods.)

Exercise 621. Consider the problem

$$\begin{aligned} & \text{minimize} && \frac{1}{2}\|\mathbf{x}\|^2 \\ & \text{subject to} && \mathbf{a}^T \mathbf{x} = b \\ & && \mathbf{x} \geq \mathbf{0}, \end{aligned}$$

where $\mathbf{a} \in \mathbb{R}^n$, $\mathbf{a} \geq \mathbf{0}$, and $b \in \mathbb{R}$, $b \geq 0$. Show that if a solution to the problem exists, then it is unique, and find an expression for it in terms of \mathbf{a} and b .

Exercise 622. Consider the problem

$$\begin{aligned} & \text{minimize} && (x_1 - a)^2 + (x_2 - b)^2, x_1, x_2 \in \mathbb{R} \\ & \text{subject to} && x_1^2 + x_2^2 \leq 1, \end{aligned}$$

where $a, b \in \mathbb{R}$ are given constants satisfying $a^2 + b^2 \geq 1$.

1. Let $\mathbf{x}^* = [x_1^*, x_2^*]^T$ be a solution to the problem. Use the first-order necessary conditions for unconstrained optimization to show that $(x_1^*)^2 + (x_2^*)^2 = 1$.
2. Use the KKT theorem to show that the solution $\mathbf{x}^* = [x_1^*, x_2^*]^T$ is unique and has the form $x_1^* = \alpha a$, $x_2^* = \alpha b$, where $\alpha \in \mathbb{R}$ is a positive constant.
3. Find an expression for α (from part b) in terms of a and b .

Exercise 623. Consider the problem

$$\begin{aligned} & \text{minimize} && x_1^2 + (x_2 + 1)^2, x_1, x_2 \in \mathbb{R} \\ & \text{subject to} && x_2 \geq \exp(x_1). \end{aligned}$$

Let $\mathbf{x}^* = [x_1^*, x_2^*]^T$ be the solution to the problem.

1. Write down the KKT condition that must be satisfied by \mathbf{x}^* .
2. Prove that $x_2^* = \exp(x_1^*)$.
3. Prove that $-2 \leq x_1^* \leq 0$.

Exercise 624. Consider the problem

$$\begin{aligned} & \text{minimize} && \mathbf{c}^T \mathbf{x} + 8 \\ & \text{subject to} && \frac{1}{2}\|\mathbf{x}\|^2 \leq 1, \end{aligned}$$

where $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{c} \neq \mathbf{0}$. Suppose that $\mathbf{x}^* = a\mathbf{e}$ is a solution to the problem, where $a \in \mathbb{R}$ and $\mathbf{e} = [1, \dots, 1]^T$, and the corresponding objective value is 4.

1. Show that $\|\mathbf{x}^*\|^2 = 2$.
2. Find α and c (they may depend on n).

Exercise 625. Consider the problem with equality constraint

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{h}(\mathbf{x}) = \mathbf{0} \end{aligned}$$

We can convert the above into the equivalent optimization problem

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \frac{1}{2}\|\mathbf{h}(\mathbf{x})\|^2 \leq 0. \end{aligned}$$

Write down the KKT condition for the equivalent problem (with inequality constraint) and explain why the KKT theorem cannot be applied in this case.

第七章 Constrained Optimization

7.1 Algorithms for Constrained Optimization

(Taken from Chapter 23 of [30])

7.1.1 Introduction

In Part II we discussed algorithms for solving unconstrained optimization problems. In this chapter we present some simple algorithms for solving special constrained optimization problems. The methods here build on those of Part II.

We begin our presentation in the next section with a discussion of projected methods, including a treatment of projected gradient methods for problems with linear equality constraints. We then consider Lagrangian methods. Finally, we consider penalty methods. This chapter is intended as an introduction to ideas underlying methods for solving constrained optimization problems. For an in-depth coverage of the subject, we refer the reader to [12].

7.1.2 Projections

The optimization algorithms considered in Part II have the general form

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}, \quad (7.1)$$

where $\mathbf{d}^{(k)}$ is typically a function of $\nabla f(\mathbf{x}^{(k)})$. The value of $\mathbf{x}^{(k)}$ is not constrained to lie inside any particular set. Such an algorithm is not immediately applicable to solving constrained optimization problems in which the decision variable is required to lie within a prespecified constraint set.

Consider the optimization problem

$$\min f(\mathbf{x}) \quad (7.2)$$

$$s.t. \quad \mathbf{x} \in \Omega. \quad (7.3)$$

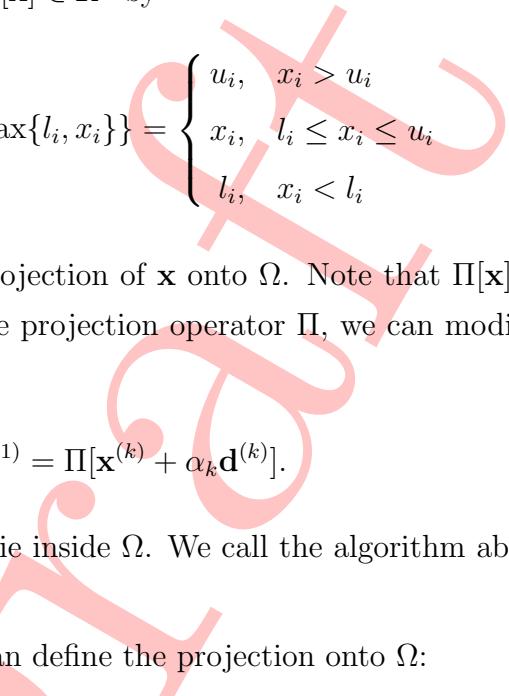
If we use the algorithm above to solve this constrained problem, the iterates $\mathbf{x}^{(k)}$ may not satisfy the constraints. Therefore, we need to modify the algorithms to take into account the presence of the constraints. A simple modification involves the introduction of a projection. The idea is as follows. If $\mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$ is in Ω , then we set $\mathbf{x}^{(k+1)} =$

$\mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$ as usual. If, on the other hand, $\mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$ is not in Ω , then we “project” it back into Ω before setting $\mathbf{x}^{(k+1)}$.

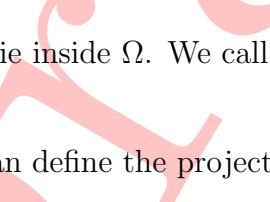
To illustrate the projection method, consider the case where the constraint set $\Omega \subseteq \mathbb{R}^n$ is given by

$$\Omega = \{\mathbf{x} : l_i \leq x_i \leq u_i, i = 1, \dots, n\}. \quad (7.4)$$

In this case, Ω is a “box” in \mathbb{R}^n ; for this reason, this form of Ω is called a box constraint. Given a point $\mathbf{x} \in \mathbb{R}^n$, define $\mathbf{y} = \Pi[\mathbf{x}] \in \mathbb{R}^n$ by

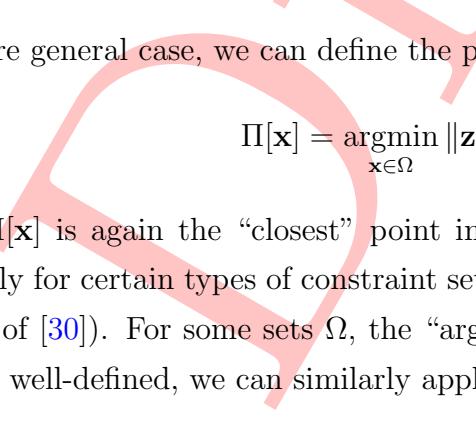
$$y_i = \min\{u_i, \max\{l_i, x_i\}\} = \begin{cases} u_i, & x_i > u_i \\ x_i, & l_i \leq x_i \leq u_i \\ l_i, & x_i < l_i \end{cases} \quad (7.5)$$


The point $\Pi[\mathbf{x}]$ is called the projection of \mathbf{x} onto Ω . Note that $\Pi[\mathbf{x}]$ is actually the “closest” point in Ω to \mathbf{x} . Using the projection operator Π , we can modify the previous unconstrained algorithm as follows:

$$\mathbf{x}^{(k+1)} = \Pi[\mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}]. \quad (7.6)$$


Note that the iterates $\mathbf{x}^{(k)}$ now all lie inside Ω . We call the algorithm above a projected algorithm.

In the more general case, we can define the projection onto Ω :

$$\Pi[\mathbf{x}] = \underset{\mathbf{x} \in \Omega}{\operatorname{argmin}} \|\mathbf{z} - \mathbf{x}\|. \quad (7.7)$$


In this case, $\Pi[\mathbf{x}]$ is again the “closest” point in Ω to \mathbf{x} . This projection operator is well-defined only for certain types of constraint sets: for example, closed convex sets (see Exercise 22.19 of [30]). For some sets Ω , the “argmin” above is not well-defined. If the projection Π is well-defined, we can similarly apply the projected algorithm

$$\mathbf{x}^{(k+1)} = \Pi[\mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}]. \quad (7.8)$$

In some cases, there is a formula for computing $\Pi[\mathbf{x}]$. For example, if Ω represents a box constraint as described above, then the formula given previously can be used. Another example is where Ω is a linear variety, which is discussed in the next section. In general, even if the projection Π is well-defined, computation of $\Pi[\mathbf{x}]$ for a given \mathbf{x} may not be easy. Often, the projection $\Pi[\mathbf{x}]$ may have to be computed numerically. However, the numerical computation of $\Pi[\mathbf{x}]$ itself entails solving an optimization algorithm. Indeed,

the computation of $\Pi[\mathbf{x}]$ may be as difficult as the original optimization problem, as is the case in the following example:

$$\begin{aligned} \min \quad & \|\mathbf{x}\|^2 \\ \text{s.t.} \quad & \mathbf{x} \in \Omega. \end{aligned}$$

Note that the solution to the problem in this case can be written as $\Pi[\mathbf{0}]$. Therefore, if $\mathbf{0} \notin \Omega$, the computation of a projection is equivalent to solving the given optimization problem.

As an example, consider the projection method applied specifically to the gradient algorithm (see Chapter 8 of [30]). Recall that the vector $-\nabla f(\mathbf{x})$ points in the direction of maximum rate of decrease of f at \mathbf{x} . This was the basis for gradient methods for unconstrained optimization, which have the form $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \nabla f(\mathbf{x}^{(k)})$, where α_k is the step size. The choice of the step size α_k depends on the particular gradient algorithm. For example, recall that in the steepest descent algorithm, $\alpha_k = \operatorname{argmin}_{\alpha \geq 0} f(\mathbf{x}^{(k)} - \alpha \nabla f(\mathbf{x}^{(k)}))$.

The projected version of the gradient algorithm has the form

$$\mathbf{x}^{(k+1)} = \Pi[\mathbf{x}^{(k)} - \alpha_k \nabla f(\mathbf{x}^{(k)})]. \quad (7.9)$$

We refer to the above as the *projected gradient algorithm*.

Example 626. Consider the problem

$$\begin{aligned} \min \quad & \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} \\ \text{s.t.} \quad & \|\mathbf{x}\|^2 = 1, \end{aligned}$$

where $\mathbf{Q} = \mathbf{Q}^T \succ \mathbf{0}$. Suppose that we apply a fixed-step-size projected gradient algorithm to this problem.

a. Derive a formula for the update equation for the algorithm (i.e., write down an explicit formula for $\mathbf{x}^{(k+1)}$ as a function of \mathbf{x}^k , \mathbf{Q} , and the fixed step size α). You may assume that the argument in the projection operator to obtain $\mathbf{x}^{(k)}$ is never zero.

b. Is it possible for the algorithm not to converge to an optimal solution even if the step size $\alpha > 0$ is taken to be arbitrarily small?

c. Show that for $0 < \alpha < 1/\lambda_{\max}$ (where λ_{\max} is the largest eigenvalue of \mathbf{Q}), the fixed-step-size projected gradient algorithm (with step size α) converges to an optimal solution, provided that $\mathbf{x}^{(0)}$ is not orthogonal to the eigenvectors of \mathbf{Q} corresponding to the smallest eigenvalue. (Assume that the eigenvalues are distinct.)

Solution. a. The projection operator in this case simply maps any vector to the closest point on the unit circle. Therefore, the projection operator is given by $\Pi[\mathbf{x}] = \mathbf{x}/\|\mathbf{x}\|$, provided that $\mathbf{x} \neq \mathbf{0}$. The update equation is

$$\mathbf{x}^{(k+1)} = \beta_k(\mathbf{x}^{(k)} - \alpha\mathbf{Q}\mathbf{x}^{(k)}) = \beta_k(\mathbf{I} - \alpha\mathbf{Q})\mathbf{x}^{(k)}, \quad (7.10)$$

where $\beta_k = 1/\|(I - \alpha\mathbf{Q})\mathbf{x}^{(k)}\|$ (i.e., it is whatever constant scaling is needed to make $\mathbf{x}^{(k+1)}$ have unit norm).

b. If we start with $\mathbf{x}^{(0)}$ being an eigenvector of \mathbf{Q} , then $\mathbf{x}^{(k)} = \mathbf{x}^{(0)}$ for all k . Therefore, if the corresponding eigenvalue is not the smallest, then clearly the algorithm is stuck at a point that is not optimal.

c. We have

$$\begin{aligned} \mathbf{x}^{(k+1)} &= \beta_k(\mathbf{I} - \alpha\mathbf{Q})\mathbf{x}^{(k)} \\ &= \beta_k(\mathbf{I} - \alpha\mathbf{Q})(y_1^{(k)}\mathbf{v}_1 + \cdots + y_n^{(k)}\mathbf{v}_n) \\ &= \beta_k(y_1^{(k)}(\mathbf{I} - \alpha\mathbf{Q})\mathbf{v}_1 + \cdots + y_n^{(k)}(\mathbf{I} - \alpha\mathbf{Q})\mathbf{v}_n). \end{aligned} \quad (7.11)$$

But $(\mathbf{I} - \alpha\mathbf{Q})\mathbf{v}_i = (1 - \alpha\lambda_i)\mathbf{v}_i$, where λ_i is the eigenvalue corresponding to \mathbf{v}_i . Hence,

$$\mathbf{x}^{(k+1)} = \beta_k(y_1^{(k)}(1 - \alpha\lambda_1)\mathbf{v}_1 + \cdots + y_n^{(k)}(1 - \alpha\lambda_n)\mathbf{v}_n), \quad (7.12)$$

which means that $y_i^{(k+1)} = \beta_k y_i^{(k)}(1 - \alpha\lambda_i)$. In other words, $y_i^{(k)} = \beta^{(k)} y_i^{(0)}(1 - \alpha\lambda_i)^k$, where $\beta^{(k)} = \prod_{i=1}^{k-1} \beta_i$. We rewrite $\mathbf{x}^{(k)}$ as

$$\begin{aligned} \mathbf{x}^{(k)} &= \sum_{i=1}^n y_i^{(k)}\mathbf{v}_i \\ &= y_1^{(k)} \left(\mathbf{v}_1 + \sum_{i=2}^n \frac{y_i^{(k)}}{y_1^{(k)}} \mathbf{v}_i \right). \end{aligned} \quad (7.13)$$

Assuming that $y_1^{(0)} \neq 0$, we obtain

$$\frac{y_i^{(k)}}{y_1^{(k)}} = \frac{y_i^{(0)}(1 - \alpha\lambda_i)^k}{y_1^{(0)}(1 - \alpha\lambda_1)^k} = \frac{y_i^{(0)}}{y_1^{(0)}} \left(\frac{1 - \alpha\lambda_i}{1 - \alpha\lambda_1} \right)^k. \quad (7.14)$$

Using the fact that $(1 - \alpha\lambda_i)/(1 - \alpha\lambda_1) < 1$ (because the $\lambda_i > \lambda_1$ for $i > 1$ and $\alpha < 1/\lambda_{max}$), we deduce that

$$\frac{y_i^{(k)}}{y_1^{(k)}} \rightarrow 0, \quad (7.15)$$

which implies that $\mathbf{x}^{(k)} \rightarrow \mathbf{v}_1$, as required. \square

7.1.3 Projected Gradient Methods with Linear Constraints

In this section we consider optimization problems of the form

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b}, \end{aligned}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m < n$, $\text{rank } \mathbf{A} = m$, $\mathbf{b} \in \mathbb{R}^m$. We assume throughout that $f \in \mathcal{C}^1$. In the problem above, the constraint set is $\Omega = \{\mathbf{x} : \mathbf{A}\mathbf{x} = \mathbf{b}\}$. The specific structure of the constraint set allows us to compute the projection operator $\mathbf{\Pi}$ using the orthogonal projector (see Section 3.3 of [30]). Specifically, $\mathbf{\Pi}[\mathbf{x}]$ can be defined using the orthogonal projector matrix \mathbf{P} given by

$$\mathbf{P} = \mathbf{I}_n - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A}, \quad (7.16)$$

(see Example 12.5 of [30]). Two important properties of the orthogonal projector \mathbf{P} that we use in this section are (see Theorem 3.5 of [30]):

1. $\mathbf{P} = \mathbf{P}^T$.
2. $\mathbf{P}^2 = \mathbf{P}$.

Another property of the orthogonal projector that we need in our discussion is given in the following lemma.

Lemma 627. *Let $\mathbf{v} \in \mathbb{R}^n$. Then, $\mathbf{P}\mathbf{v} = \mathbf{0}$ if and only if $\mathbf{v} \in \mathcal{R}(\mathbf{A}^T)$. In other words, $\mathcal{N}(\mathbf{P}) = \mathcal{R}(\mathbf{A}^T)$. Moreover, $\mathbf{A}\mathbf{v} = \mathbf{0}$ if and only if $\mathbf{v} \in \mathcal{R}(\mathbf{P})$; that is, $\mathcal{N}(\mathbf{A}) = \mathcal{R}(\mathbf{P})$.*

Proof. \Rightarrow : We have

$$\begin{aligned} \mathbf{P}\mathbf{v} &= (\mathbf{I}_n - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A})\mathbf{v} \\ &= \mathbf{v} - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{v}. \end{aligned} \quad (7.17)$$

If $\mathbf{P}\mathbf{v} = \mathbf{0}$, then

$$\mathbf{v} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{v} \quad (7.18)$$

and hence $\mathbf{v} \in \mathcal{R}(\mathbf{A}^T)$. \square

\Leftarrow : Suppose that there exists $\mathbf{u} \in \mathbb{R}^m$ such that $\mathbf{v} = \mathbf{A}^T\mathbf{u}$. Then,

$$\begin{aligned} \mathbf{P}\mathbf{v} &= (\mathbf{I}_n - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A})\mathbf{A}^T\mathbf{u} \\ &= \mathbf{A}^T\mathbf{u} - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{A}^T\mathbf{u} \\ &= \mathbf{0}. \end{aligned} \quad (7.19)$$

Hence, we have proved that $\mathcal{N}(\mathbf{P}) = \mathcal{R}(\mathbf{A}^T)$.

Using an argument similar to that above, we can show that $\mathcal{N}(\mathbf{A}) = \mathcal{R}(\mathbf{P})$.

Recall that in unconstrained optimization, the first-order necessary condition for a point \mathbf{x}^* to be a local minimizer is $\nabla f(\mathbf{x}^*) = \mathbf{0}$ (see Section 6.2 of [30]). In optimization problems with equality constraints, the Lagrange condition plays the role of the first-order necessary condition (see Section 6.7.4). When the constraint set takes the form $\{\mathbf{x} : \mathbf{Ax} = \mathbf{b}\}$, the Lagrange condition can be written as $\mathbf{P}\nabla f(\mathbf{x}^*) = \mathbf{0}$, as stated in the following proposition.

Proposition 628. *Let $\mathbf{x}^* \in \mathbb{R}^n$ be a feasible point. Then, $\mathbf{P}\nabla f(\mathbf{x}^*) = \mathbf{0}$ if and only if \mathbf{x}^* satisfies the Lagrange condition.*

Proof. By Lemma 627, $\mathbf{P}\nabla f(\mathbf{x}^*) = \mathbf{0}$ if and only if we have $\nabla f(\mathbf{x}^*) \in \mathcal{R}(\mathbf{A}^T)$. This is equivalent to the condition that there exists $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ such that $\nabla f(\mathbf{x}^*) + \mathbf{A}^T \boldsymbol{\lambda}^* = \mathbf{0}$, which together with the feasibility equation $\mathbf{Ax} = \mathbf{b}$, constitutes the Lagrange condition. \square

Recall that the projected gradient algorithm has the form

$$\mathbf{x}^{(k+1)} = \mathbf{\Pi}[\mathbf{x}^{(k)} - \alpha_k \nabla f(\mathbf{x}^{(k)})]. \quad (7.20)$$

For the case where the constraints are linear, it turns out that we can express the projection $\mathbf{\Pi}$ in terms of the matrix \mathbf{P} as follows:

$$\mathbf{\Pi}[\mathbf{x}^{(k)} - \alpha_k \nabla f(\mathbf{x}^{(k)})] = \mathbf{x}^{(k)} - \alpha_k \mathbf{P}\nabla f(\mathbf{x}^{(k)}), \quad (7.21)$$

assuming that $\mathbf{x}^{(k)} \in \Omega$. Although the formula above can be derived algebraically (see Exercise 655), it is more insightful to derive the formula using a geometric argument, as follows. In our constrained optimization problem, the vector $-\nabla f(\mathbf{x})$ is not necessarily a feasible direction. In other words, if $\mathbf{x}^{(k)}$ is a feasible point and we apply the algorithm $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \nabla f(\mathbf{x}^{(k)})$, then $\mathbf{x}^{(k+1)}$ need not be feasible. This problem can be overcome by replacing $-\nabla f(\mathbf{x}^{(k)})$ by a vector that points in a feasible direction. Note that the set of feasible directions is simply the nullspace $\mathcal{N}(\mathbf{A})$ of the matrix \mathbf{A} . Therefore, we should first project the vector $-\nabla f(\mathbf{x})$ onto $\mathcal{N}(\mathbf{A})$. This projection is equivalent to multiplication by the matrix \mathbf{P} . In summary, in the projection gradient algorithm, we update $\mathbf{x}^{(k)}$ according to the equation

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{P}\nabla f(\mathbf{x}^{(k)}). \quad (7.22)$$

The projected gradient algorithm has the following property.

Proposition 629. In projected gradient algorithm, if $\mathbf{x}^{(0)}$ is feasible, then each $\mathbf{x}^{(k)}$ is feasible; that is, for each $k > 0$, $\mathbf{A}\mathbf{x}^{(k)} = \mathbf{b}$.

Proof. We proceed by induction. The result holds for $k = 0$ by assumption. Suppose now that $\mathbf{A}\mathbf{x}^{(k)} = \mathbf{b}$. We now show that $\mathbf{A}\mathbf{x}^{(k+1)} = \mathbf{b}$. To show this, first observe that $\mathbf{P}\nabla f(\mathbf{x}^{(k)}) \in \mathcal{N}(\mathbf{A})$. Therefore,

$$\begin{aligned}\mathbf{A}\mathbf{x}^{(k+1)} &= \mathbf{A}(\mathbf{x}^{(k)} - \alpha_k \mathbf{P}\nabla f(\mathbf{x}^{(k)})) \\ &= \mathbf{A}\mathbf{x}^{(k)} - \alpha_k \mathbf{A}\mathbf{P}\nabla f(\mathbf{x}^{(k)}) \\ &= \mathbf{b},\end{aligned}\tag{7.23}$$

which completes the proof. \square

The projected gradient algorithm updates $\mathbf{x}^{(k)}$ in the direction of $-\mathbf{P}\nabla f(\mathbf{x}^{(k)})$. This vector points in the direction of maximum rate of decrease of f at $\mathbf{x}^{(k)}$ along the surface defined by $\mathbf{A}\mathbf{x} = \mathbf{b}$, as described in the following argument. Let \mathbf{x} be any feasible point and \mathbf{d} a feasible direction such that $\|\mathbf{d}\| = 1$. The rate of increase of f at \mathbf{x} in the direction \mathbf{d} is $\langle \nabla f(\mathbf{x}), \mathbf{d} \rangle$. Next, we note that because \mathbf{d} is a feasible direction, it lies in $\mathcal{N}(\mathbf{A})$ and hence by Lemma 627, we have $\mathbf{d} \in \mathcal{R}(\mathbf{P}) = \mathcal{R}(\mathbf{P}^T)$. So, there exists \mathbf{v} such that $\mathbf{d} = \mathbf{P}\mathbf{v}$. Hence,

$$\langle \nabla f(\mathbf{x}), \mathbf{d} \rangle = \langle \nabla f(\mathbf{x}), \mathbf{P}^T \mathbf{v} \rangle = \langle \mathbf{P}\nabla f(\mathbf{x}), \mathbf{v} \rangle.\tag{7.24}$$

By the Cauchy-Schwarz inequality,

$$\langle \mathbf{P}\nabla f(\mathbf{x}), \mathbf{v} \rangle \leq \|\mathbf{P}\nabla f(\mathbf{x})\| \|\mathbf{v}\| \tag{7.25}$$

with equality if and only if the direction of \mathbf{v} is parallel with the direction of $\mathbf{P}\nabla f(\mathbf{x})$. Therefore, the vector $-\mathbf{P}\nabla f(\mathbf{x})$ points in the direction of maximum rate of decrease of f at \mathbf{x} among all feasible directions.

Following the discussion in Chapter 8 of [30] for gradient methods in unconstrained optimization, we suggest the following gradient method for our constrained problem. Suppose that we have a starting point $\mathbf{x}^{(0)}$ which we assume is feasible; that is, $\mathbf{A}\mathbf{x}^{(0)} = \mathbf{b}$. Consider the point $\mathbf{x} = \mathbf{x}^{(0)} - \alpha \mathbf{P}\nabla f(\mathbf{x}^{(0)})$, where $\alpha \in \mathbb{R}$. As usual, the scalar α is called the step size. By the discussion above, \mathbf{x} is also a feasible point. Using a Taylor series expansion of f about $\mathbf{x}^{(0)}$ and the fact that $\mathbf{P} = \mathbf{P}^2 = \mathbf{P}^T \mathbf{P}$, we get

$$\begin{aligned}f(\mathbf{x}^{(0)} - \alpha \mathbf{P}\nabla f(\mathbf{x}^{(0)})) &= f(\mathbf{x}^{(0)}) - \alpha \nabla f(\mathbf{x}^{(0)})^T \mathbf{P}\nabla f(\mathbf{x}^{(0)}) + o(\alpha) \\ &= f(\mathbf{x}^{(0)}) - \alpha \|\mathbf{P}\nabla f(\mathbf{x}^{(0)})\|^2 + o(\alpha).\end{aligned}\tag{7.26}$$

Thus, if $\mathbf{P}\nabla f(\mathbf{x}^{(0)}) \neq 0$, that is, $x^{(0)}$ does not satisfy the Lagrange condition, then we can choose an α sufficiently small such that $f(\mathbf{x}) < f(\mathbf{x}^{(0)})$, which means that $\mathbf{x} = \mathbf{x}^{(0)} - \alpha\mathbf{P}\nabla f(\mathbf{x}^{(0)})$ is an improvement over $\mathbf{x}^{(0)}$. This is the basis for the projected gradient algorithm $\mathbf{x} = \mathbf{x}^{(k)} - \alpha_k \mathbf{P}\nabla f(\mathbf{x}^{(k)})$, where the initial point $\mathbf{x}^{(0)}$ satisfies $\mathbf{A}\mathbf{x}^{(0)} = \mathbf{b}$ and α_k is some step size. As for unconstrained gradient methods, the choice of α_k determines the behavior of the algorithm. For small step sizes, the algorithm progresses slowly, while large step sizes may result in a zigzagging path. A well-known variant of the projected gradient algorithm is the projected steepest descent algorithm, where α_k is given by

$$\alpha_k = \operatorname{argmin}_{\alpha \geq 0} f(\mathbf{x}^{(k)} - \alpha \mathbf{P}\nabla f(\mathbf{x}^{(k)})). \quad (7.27)$$

The following theorem states that the projected steepest descent algorithm is a descent algorithm, in the sense that at each step the value of the objective function decreases.

Theorem 630. *If $\{\mathbf{x}^{(k)}\}$ is the sequence of points generated by the projected steepest descent algorithm and if $\mathbf{P}\nabla f(\mathbf{x}^{(k)}) \neq 0$, then $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$.*

Proof. First, recall that

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{P}\nabla f(\mathbf{x}^{(k)}), \quad (7.28)$$

where $\alpha_k \geq 0$ is the minimizer of

$$\phi_k(\alpha) = f(\mathbf{x}^{(k)} - \alpha \mathbf{P}\nabla f(\mathbf{x}^{(k)})) \quad (7.29)$$

over all $\alpha \geq 0$. Thus, for $\alpha \geq 0$, we have

$$\phi_k(\alpha_k) \leq \phi_k(\alpha). \quad (7.30)$$

By the chain rule,

$$\begin{aligned} \phi'_k(0) &= \frac{d\phi_k}{d\alpha}(0) \\ &= -\nabla f(\mathbf{x}^{(k)} - 0 \mathbf{P}\nabla f(\mathbf{x}^{(k)}))^T \mathbf{P}\nabla f(\mathbf{x}^{(k)}) \\ &= -\nabla f(\mathbf{x}^{(k)})^T \mathbf{P}\nabla f(\mathbf{x}^{(k)}). \end{aligned} \quad (7.31)$$

Using the fact that $\mathbf{P} = \mathbf{P}^2 = \mathbf{P}^T \mathbf{P}$, we get

$$\phi'_k(0) = -\nabla f(\mathbf{x}^{(k)})^T \mathbf{P}^T \mathbf{P} \nabla f(\mathbf{x}^{(k)}) = -\|\mathbf{P}\nabla f(\mathbf{x}^{(k)})\|^2 < 0, \quad (7.32)$$

because $\mathbf{P}\nabla f(\mathbf{x}^{(k)}) \neq \mathbf{0}$ by assumption. Thus, there exists $\bar{\alpha} > 0$ such that $\phi_k(0) > \phi_k(\alpha)$ for all $\alpha \in (0, \bar{\alpha}]$. Hence,

$$f(\mathbf{x}^{(k+1)}) = \phi_k(\alpha_k) \leq \phi_k(\bar{\alpha}) < \phi_k(0) = f(\mathbf{x}^{(k)}), \quad (7.33)$$

which completes the proof of the theorem. \square

In Theorem 630 we needed the assumption that $\mathbf{P}\nabla f(\mathbf{x}^{(k)}) \neq \mathbf{0}$ to prove that the algorithm possesses the descent property. If for some k , we have $\mathbf{P}\nabla f(\mathbf{x}^{(k)}) = \mathbf{0}$, then by Proposition 628 the point $\mathbf{x}^{(k)}$ satisfies the Lagrange condition. This condition can be used as a stopping criterion for the algorithm. Note that in this case, $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}$. For the case where f is a convex function, the condition $\mathbf{P}\nabla f(\mathbf{x}^{(k)}) = \mathbf{0}$ is, in fact, equivalent to $\mathbf{x}^{(k)}$ being a global minimizer of f over the constraint set $\{\mathbf{x} : \mathbf{Ax} = \mathbf{b}\}$. We show this in the following proposition.

Proposition 631. *The point $\mathbf{x} \in \mathbb{R}^n$ is a global minimizer of a convex function f over $\{\mathbf{x} : \mathbf{Ax} = \mathbf{b}\}$ if and only if $\mathbf{P}\nabla f(\mathbf{x}^*) = \mathbf{0}$.*

Proof. We first write $h(\mathbf{x}) = \mathbf{Ax} - \mathbf{b}$. Then, the constraints can be written as $h(\mathbf{x}) = \mathbf{0}$, and the problem is of the form considered in earlier chapters. Note that $Dh(\mathbf{x}) = \mathbf{A}$. Hence, $\mathbf{x}^* \in \mathbb{R}^n$ is a global minimizer of f if and only if the Lagrange condition holds (see Theorem 22.8 of [30]). By Proposition 628, this is true if and only if $\mathbf{P}\nabla f(\mathbf{x}^*) = \mathbf{0}$, and this completes the proof. \square

For an application of the projected steepest descent algorithm to minimum fuel and minimum amplitude control problems in linear discrete systems, see [70].

7.1.4 Equality constrained minimization

(Taken from Chapter 10 of [18])

7.1.4.1 Equality constrained minimization problems

In this section we describe methods for solving a convex optimization problem with equality constraints,

$$\min_{\mathbf{x}} f(\mathbf{x}), \quad s.t. \quad \mathbf{Ax} = \mathbf{b}, \quad (7.34)$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex and twice continuously differentiable, and $\mathbf{A} \in \mathbb{R}^{p \times n}$ with $\text{rank } \mathbf{A} = p < n$. The assumptions on \mathbf{A} mean that there are fewer equality constraints than variables, and that the equality constraints are independent. We will assume that an optimal solution \mathbf{x}^* exists, and use p^* to denote the optimal value, $p^* = \inf\{f(\mathbf{x}) | \mathbf{Ax} = \mathbf{b}\} = f(\mathbf{x}^*)$.

Recall that a point $x^* \in \text{dom } f$ is optimal for (7.34) if and only if there is a $\boldsymbol{\nu} \in \mathbb{R}^p$ such that

$$\mathbf{Ax}^* = \mathbf{b}, \quad \nabla f(\mathbf{x}^*) + \mathbf{A}^T \boldsymbol{\nu}^* = \mathbf{0}. \quad (7.35)$$

Solving the equality constrained optimization problem (7.34) is therefore equivalent to finding a solution of the KKT equations (7.35), which is a set of $n + p$ equations in the

$n+p$ variables $\mathbf{x}^*, \boldsymbol{\nu}^*$. The first set of equations, $\mathbf{Ax}^* = \mathbf{b}$, are called the primal feasibility equations, which are linear. The second set of equations, $\nabla f(\mathbf{x}^*) + \mathbf{A}^T \boldsymbol{\nu}^* = \mathbf{0}$, are called the dual feasibility equations, and are in general nonlinear. As with unconstrained optimization, there are a few problems for which we can solve these optimality conditions analytically. The most important special case is when f is quadratic, which we examine in Section 7.1.4.2.

Any equality constrained minimization problem can be reduced to an equivalent unconstrained problem by eliminating the equality constraints, after which the methods of chapter 5 can be used to solve the problem. Another approach is to solve the dual problem (assuming the dual function is twice differentiable) using an unconstrained minimization method, and then recover the solution of the equality constrained problem (7.34) from the dual solution. The elimination and dual methods are briefly discussed in Sections 7.1.4.5 and 7.1.4.6, respectively.

The bulk of this chapter is devoted to extensions of Newton's method that directly handle equality constraints. In many cases these methods are preferable to methods that reduce an equality constrained problem to an unconstrained one. One reason is that problem structure, such as sparsity, is often destroyed by elimination (or forming the dual); in contrast, a method that directly handles equality constraints can exploit the problem structure. Another reason is conceptual: methods that directly handle equality constraints can be thought of as methods for directly solving the optimality conditions (7.35).

7.1.4.2 Equality constrained convex quadratic minimization

Consider the equality constrained convex quadratic minimization problem

$$\min_{\mathbf{x}} \frac{1}{2} \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{q}^T \mathbf{x} + r, \quad s.t. \quad \mathbf{Ax} = \mathbf{b}, \quad (7.36)$$

where $\mathbf{P} \in \mathbb{S}_+^n$ and $\mathbf{A} \in \mathbb{R}^{p \times n}$. This problem is important on its own, and also because it forms the basis for an extension of Newton's method to equality constrained problems.

Here the optimality conditions (7.35) are

$$\mathbf{Ax}^* = \mathbf{b}, \quad \mathbf{Px}^* + \mathbf{q} + \mathbf{A}^T \boldsymbol{\nu}^* = \mathbf{0},$$

which we can write as

$$\begin{pmatrix} \mathbf{P} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{x}^* \\ \boldsymbol{\nu}^* \end{pmatrix} = \begin{pmatrix} -\mathbf{q} \\ \mathbf{b} \end{pmatrix}. \quad (7.37)$$

This set of $n+p$ linear equations in the $n+p$ variables $\mathbf{x}^*, \boldsymbol{\nu}^*$ is called the *KKT system* for the equality constrained quadratic optimization problem (7.36). The coefficient matrix is

called the *KKT matrix*. When the KKT matrix is nonsingular, there is a unique optimal primal-dual pair $(\mathbf{x}^*, \boldsymbol{\nu}^*)$. If the KKT matrix is singular, but the KKT system is solvable, any solution yields an optimal pair $(\mathbf{x}^*, \boldsymbol{\nu}^*)$. If the KKT system is not solvable, the quadratic optimization problem is unbounded below or infeasible. Indeed, in this case there exist $\mathbf{v} \in \mathbb{R}^n$ and $\mathbf{w} \in \mathbb{R}^p$ such that

$$\mathbf{P}\mathbf{v} + \mathbf{A}^T\mathbf{w} = \mathbf{0}, \quad \mathbf{A}\mathbf{v} = \mathbf{0}, \quad -\mathbf{q}^T\mathbf{v} + \mathbf{b}^T\mathbf{w} > 0.$$

Let $\hat{\mathbf{x}}$ be any feasible point. The point $\mathbf{x} = \hat{\mathbf{x}} + t\mathbf{v}$ is feasible for all t and

$$\begin{aligned} f(\hat{\mathbf{x}} + t\mathbf{v}) &= f(\hat{\mathbf{x}}) + t(\mathbf{v}^T \mathbf{P}\hat{\mathbf{x}} + \mathbf{q}^T \mathbf{v}) + \frac{1}{2}t^2 \mathbf{v}^T \mathbf{P}\mathbf{v} \\ &= f(\hat{\mathbf{x}}) + t(-\hat{\mathbf{x}}^T \mathbf{A}^T \mathbf{w} + \mathbf{q}^T \mathbf{v}) - \frac{1}{2}t^2 \mathbf{w}^T \mathbf{A}\mathbf{v} \\ &= f(\hat{\mathbf{x}}) + t(-\mathbf{b}^T \mathbf{w} + \mathbf{q}^T \mathbf{v}), \end{aligned}$$

which decreases without bound as $t \rightarrow \infty$.

Nonsingularity of the KKT matrix

Recall our assumption that $\mathbf{P} \in \mathbb{S}_+^n$ and $\text{rank } \mathbf{A} = p < n$. There are several conditions equivalent to nonsingularity of the KKT matrix (see Exercise 10.1 of [18]):

- $\mathcal{N}(\mathbf{P}) \cap \mathcal{N}(\mathbf{A}) = \{\mathbf{0}\}$, i.e., \mathbf{P} and \mathbf{A} have no nontrivial common nullspace.
- $\mathbf{A}\mathbf{x} = \mathbf{0}, \mathbf{x} \neq \mathbf{0} \implies \mathbf{x}^T \mathbf{P}\mathbf{x} > 0$, i.e., \mathbf{P} is positive definite on the nullspace of \mathbf{A} .
- $\mathbf{F}^T \mathbf{P} \mathbf{F} \succ \mathbf{0}$, where $\mathbf{F} \in \mathbb{R}^{n \times (n-p)}$ is a matrix for which $\mathcal{R}(\mathbf{F}) = \mathcal{N}(\mathbf{A})$.

As an important special case, we note that if $\mathbf{P} \succ \mathbf{0}$, the KKT matrix must be nonsingular.

7.1.4.3 Solving KKT systems

(Taken from Section 10.4.2 of [18])

In this section we describe methods that can be used to compute the Newton step or infeasible Newton step, both of which involve solving a set of linear equations with KKT form

$$\begin{pmatrix} \mathbf{H} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{v} \\ \mathbf{w} \end{pmatrix} = - \begin{pmatrix} \mathbf{g} \\ \mathbf{h} \end{pmatrix}. \quad (7.38)$$

Here we assume $\mathbf{H} \in \mathbb{S}_+^n$ and $\mathbf{A} \in \mathbb{R}^{p \times n}$ with $\text{rank } \mathbf{A} = p < n$. Similar methods can be used to compute the Newton step for a convex-concave game, in which the bottom right entry of the coefficient matrix is negative semidefinite.

Solving full KKT systems

One straightforward approach is to simply solve the KKT system (7.38), which is a set of

$n+p$ linear equations in $n+p$ variables. The KKT matrix is symmetric, but not positive definite, so a good way to do this is to use an LDL^T factorization. If no structure of the matrix is exploited, the cost is $(1/3)(n+p)^3$ flops. This can be a reasonable approach when the problem is small (i.e., n and p are not too large), or when \mathbf{A} and \mathbf{H} are sparse.

Solving KKT system via elimination

A method that is often better than directly solving the full KKT system is based on eliminating the variable \mathbf{v} . We start by describing the simplest case, in which $\mathbf{H} \succ \mathbf{0}$. Starting from the first of the KKT equations

$$\mathbf{H}\mathbf{v} + \mathbf{A}^T\mathbf{b} = -\mathbf{g}, \quad \mathbf{A}\mathbf{v} = -\mathbf{h},$$

we solve for \mathbf{v} to obtain

$$\mathbf{v} = -\mathbf{H}^{-1}(\mathbf{g} + \mathbf{A}^T\mathbf{w}).$$

Substituting this into the second KKT equation yields $\mathbf{A}\mathbf{H}^{-1}(\mathbf{g} + \mathbf{A}^T\mathbf{w}) = \mathbf{h}$, so we have

$$\mathbf{w} = (\mathbf{A}\mathbf{H}^{-1}\mathbf{A}^T)^{-1}(\mathbf{h} - \mathbf{A}\mathbf{H}^{-1}\mathbf{g}).$$

These formulae give us a method for computing \mathbf{v} and \mathbf{w} .

Algorithm 4 Solving KKT system by block elimination

Given: KKT system with $\mathbf{H} \succ \mathbf{0}$.

1. Form $\mathbf{H}^{-1}\mathbf{A}^T$ and $\mathbf{H}^{-1}\mathbf{g}$.
 2. Form Schur complement $\mathbf{S} = -\mathbf{A}\mathbf{H}^{-1}\mathbf{A}^T$.
 3. Determine \mathbf{w} by solving $\mathbf{S}\mathbf{w} = \mathbf{A}\mathbf{H}^{-1}\mathbf{g} - \mathbf{h}$.
 4. Determine \mathbf{v} by solving $\mathbf{H}\mathbf{v} = -\mathbf{A}^T\mathbf{w} - \mathbf{g}$.
-

Step 1 can be done by a Cholesky factorization of \mathbf{H} , followed by $p+1$ solves, which costs $f + (p+1)s$, where f is the cost of factoring \mathbf{H} and s is the cost of an associated solve. Step 2 requires a $p \times n$ by $n \times p$ matrix multiplication. If we exploit no structure in this calculation, the cost is p^2n flops. (Since the result is symmetric, we only need to compute the upper triangular part of \mathbf{S} .) In some cases special structure in \mathbf{A} and \mathbf{H} can be exploited to carry out step 2 more efficiently. Step 3 can be carried out by Cholesky factorization of $-\mathbf{S}$, which costs $(1/3)p^3$ flops if no further structure of \mathbf{S} is exploited. Step 4 can be carried out using the factorization of \mathbf{H} already calculated in step 1, so the cost is $2np + s$ flops. The total flop count, assuming that no structure is exploited in forming or factoring the Schur complement, is

$$f + ps + p^2n + (1/3)p^3$$

flops (keeping only dominant terms). If we exploit structure in forming or factoring \mathbf{S} , the last two terms are even smaller.

If \mathbf{H} can be factored efficiently, then block elimination gives us a flop count advantage over directly solving the KKT system using an LDL^T factorization. For example, if \mathbf{H} is diagonal (which corresponds to a separable objective function), we have $f = 0$ and $s = n$, so the total cost is $p^2n + (1/3)p^3$ flops, which grows only linearly with n . If \mathbf{H} is banded with bandwidth $k \ll n$, then $f = nk^2$, $s = 4nk$, so the total cost is around $nk^2 + 4nkp + p^2n + (1/3)p^3$ which still grows only linearly with n . Other structures of \mathbf{H} that can be exploited are block diagonal (which corresponds to block separable objective function), sparse, or diagonal plus low rank.

Example 632 (Equality constrained analytic center). *We consider the problem*

$$\min_{\mathbf{x}} - \sum_{i=1}^n \log x_i, \quad \text{s.t.} \quad \mathbf{Ax} = \mathbf{b}.$$

Here the objective is separable, so the Hessian at \mathbf{x} is diagonal:

$$\mathbf{H} = \text{Diag}(x_1^{-2}, \dots, x_n^{-2}).$$

If we compute the Newton direction using a generic method such as an LDL^T factorization of the KKT matrix, the cost is $(1/3)(n+p)^3$ flops.

If we compute the Newton step using block elimination, the cost is $np^2 + (1/3)p^3$ flops. This is much smaller than the cost of the generic method. In fact this cost is the same as that of computing the Newton step for the dual problem, described in Example 635. For the (unconstrained) dual problem, the Hessian is

$$\mathbf{H}_{\text{dual}} = -\mathbf{ADA}^T,$$

where \mathbf{D} is diagonal, with $D_{ii} = (\mathbf{A}^T \boldsymbol{\nu})_i^{-2}$. Forming this matrix costs np^2 flops, and solving for the Newton step by a Cholesky factorization of $-\mathbf{H}_{\text{dual}}$ costs $(1/3)p^3$ flops.

Example 633 (Minimum length piecewise-linear curve subject to equality constraints). We consider a piecewise-linear curve in \mathbb{R}^2 with knot points $(0, 0)$, $(1, x_1)$, \dots , (n, x_n) . To find the minimum length curve that satisfies the equality constraints $\mathbf{Ax} = \mathbf{b}$, we form the problem

$$\min_{\mathbf{x}} (1 + x_1^2)^{1/2} + \sum_{i=1}^{n-1} (1 + (x_{i+1} - x_i)^2)^{1/2}, \quad \text{s.t.} \quad \mathbf{Ax} = \mathbf{b},$$

with variable $\mathbf{x} \in \mathbb{R}^n$, and $\mathbf{A} \in \mathbb{R}^{p \times n}$. In this problem, the objective is a sum of functions of pairs of adjacent variables, so the Hessian \mathbf{H} is tridiagonal. Using block elimination, we can compute the Newton step in around $p^2n + (1/3)p^3$ flops.

Elimination with singular \mathbf{H}

The block elimination method described above obviously does not work when \mathbf{H} is singular, but a simple variation on the method can be used in this more general case. The more general method is based on the following result: The KKT matrix is nonsingular if and only $\mathbf{H} + \mathbf{A}^T \mathbf{Q} \mathbf{A} \succ \mathbf{0}$ for some $\mathbf{Q} \succeq \mathbf{0}$, in which case, $\mathbf{H} + \mathbf{A}^T \mathbf{Q} \mathbf{A} \succ \mathbf{0}$ for all $\mathbf{Q} \succ \mathbf{0}$. We conclude, for example, that if the KKT matrix is nonsingular, then $\mathbf{H} + \mathbf{A}^T \mathbf{A} \succ \mathbf{0}$.

Let $\mathbf{Q} \succeq \mathbf{0}$ be a matrix for which $\mathbf{H} + \mathbf{A}^T \mathbf{Q} \mathbf{A} \succ \mathbf{0}$. Then the KKT system (7.38) is equivalent to

$$\begin{pmatrix} \mathbf{H} + \mathbf{A}^T \mathbf{Q} \mathbf{A} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{v} \\ \mathbf{w} \end{pmatrix} = -\begin{pmatrix} \mathbf{g} + \mathbf{A}^T \mathbf{Q} \mathbf{h} \\ \mathbf{h} \end{pmatrix}, \quad (7.39)$$

which can be solved using elimination since $\mathbf{H} + \mathbf{A}^T \mathbf{Q} \mathbf{A} \succ \mathbf{0}$.

7.1.4.4 Examples

In this section we describe some longer examples, showing how structure can be exploited to efficiently compute the Newton step. We also include some numerical results.

Equality constrained analytic centering

We consider the equality constrained analytic centering problem

$$\min_{\mathbf{x}} f(\mathbf{x}) = -\sum_{i=1}^n \log x_i, \quad s.t. \quad \mathbf{A}\mathbf{x} = \mathbf{b}.$$

(See Examples 635 and 632.) We compare three methods, for a problem of size $p = 100, n = 500$.

The first method is Newton's method with equality constraints (Section 7.1.5). The Newton step $\Delta \mathbf{x}_{nt}$ is defined by the KKT system (7.54):

$$\begin{bmatrix} \mathbf{H} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_{nt} \\ \mathbf{w} \end{bmatrix} = \begin{bmatrix} -\mathbf{g} \\ \mathbf{0} \end{bmatrix},$$

where $\mathbf{H} = \text{Diag}(1/x_1^2, \dots, 1/x_n^2)$, and $\mathbf{g} = -(1/x_1, \dots, 1/x_n - n)$. As explained in Example 632, the KKT system can be efficiently solved by elimination, i.e., by solving

$$\mathbf{A} \mathbf{H}^{-1} \mathbf{A}^T \mathbf{w} = -\mathbf{A} \mathbf{H}^{-1} \mathbf{g},$$

and setting $\Delta \mathbf{x}_{nt} = -\mathbf{H}^{-1}(\mathbf{A}^T \mathbf{w} + \mathbf{g})$. In other words,

$$\Delta \mathbf{x}_{nt} = -\text{Diag}(\mathbf{x})^2 \mathbf{A}^T \mathbf{w} + \mathbf{x},$$

where \mathbf{w} is the solution of

$$\mathbf{A} \text{Diag}(\mathbf{x})^2 \mathbf{A}^T \mathbf{w} = \mathbf{b}. \quad (7.40)$$

Figure 7.1 shows the error versus iteration. The different curves correspond to four different starting points. We use a backtracking line search with $\alpha = 0.1$, $\beta = 0.5$.

The second method is Newton's method applied to the dual

$$\max_{\boldsymbol{\nu}} g(\boldsymbol{\nu}) = -\mathbf{b}^T \boldsymbol{\nu} + \sum_{i=1}^n \log(\mathbf{A}^T \boldsymbol{\nu})_i + n$$

(see Example 635). Here the Newton step is obtained from solving

$$\mathbf{A} \operatorname{Diag}(\mathbf{y})^2 \mathbf{A}^T \Delta \boldsymbol{\nu}_{nt} = -\mathbf{b} + \mathbf{A} \mathbf{y}, \quad (7.41)$$

where $\mathbf{y} = (1/(\mathbf{A}^T \boldsymbol{\nu})_1, \dots, 1/(\mathbf{A}^T \boldsymbol{\nu})_n)^T$. Comparing (7.41) and (7.40) we see that both methods have the same complexity. In Figure 7.2 we show the error for four different starting points. We use a backtracking line search with $\alpha = 0.1$, $\beta = 0.5$.

The third method is the infeasible start Newton method of Section 10.3 of [18], applied to the optimality conditions

$$\nabla f(\mathbf{x}^*) + \mathbf{A}^T \boldsymbol{\nu}^* = \mathbf{0}, \quad \mathbf{A} \mathbf{x}^* = \mathbf{b}.$$

The Newton step is obtained by solving

$$\begin{bmatrix} \mathbf{H} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_{nt} \\ \Delta \boldsymbol{\nu}_{nt} \end{bmatrix} = - \begin{bmatrix} \mathbf{g} + \mathbf{A}^T \boldsymbol{\nu} \\ \mathbf{Ax} - \mathbf{b} \end{bmatrix},$$

where $\mathbf{H} = \operatorname{Diag}(1/x_1^2, \dots, 1/x_n^2)$, and $\mathbf{g} = -(1/x_1, \dots, 1/x_n)$. This KKT system can be efficiently solved by elimination, at the same cost as (7.40) or (7.41). For example, if we first solve

$$\mathbf{A} \operatorname{Diag}(\mathbf{x})^2 \mathbf{A}^T \mathbf{w} = 2\mathbf{Ax} - \mathbf{b},$$

then $\Delta \boldsymbol{\nu}_{nt}$ and $\Delta \mathbf{x}_{nt}$ follow from

$$\Delta \boldsymbol{\nu}_{nt} = \mathbf{w} - \boldsymbol{\nu}, \quad \Delta \mathbf{x}_{nt} = \mathbf{x} - \operatorname{Diag}(\mathbf{x})^2 \mathbf{A}^T \mathbf{w}.$$

Figure 7.3 shows the norm of the residual

$$r(\mathbf{x}, \boldsymbol{\nu}) = (\nabla f(\mathbf{x}) + \mathbf{A}^T \boldsymbol{\nu}, \mathbf{Ax} - \mathbf{b})$$

versus iteration, for four different starting points. We use a backtracking line search with $\alpha = 0.1$, $\beta = 0.5$.

The figures show that for this problem, the dual method appears to be faster, but only by a factor of two or three. It takes about six iterations to reach the region of quadratic

convergence, as opposed to 12-15 in the primal method and 10-20 in the infeasible start Newton method.

The methods also differ in the initialization they require. The primal method requires knowledge of a primal feasible point, i.e., satisfying $\mathbf{A}\mathbf{x}^{(0)} = \mathbf{b}$, $\mathbf{x}^{(0)} > \mathbf{0}$. The dual method requires a dual feasible point, i.e., $\mathbf{A}^T \boldsymbol{\nu}^{(0)} > \mathbf{0}$. Depending on the problem, one or the other might be more readily available. The infeasible start Newton method requires no initialization; the only requirement is that $\mathbf{x}^{(0)} > \mathbf{0}$.

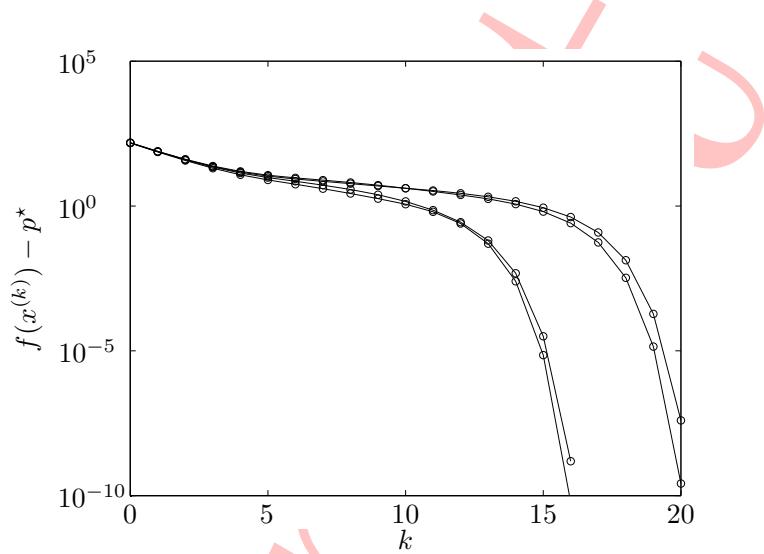


图 7.1: Error $|g(\boldsymbol{\nu}^{(k)}) - p^*|$ in Newton's method, applied to the equality constrained analytic centering problem of size $p = 100$, $n = 500$. The different curves correspond to four different starting points. Final quadratic convergence is clearly evident.

Optimal network flow

We consider a connected directed graph or network with n edges and $p + 1$ nodes. We let x_j denote the flow or traffic on arc j , with $x_j > 0$ meaning flow in the direction of the arc, and $x_j < 0$ meaning flow in the direction opposite the arc. There is also a given external source (or sink) flow s_i that enters (if $s_i > 0$) or leaves (if $s_i < 0$) node i . The flow must satisfy a conservation equation, which states that at each node, the total flow entering the node, including the external sources and sinks, is zero. This conservation equation can be expressed as $\tilde{\mathbf{A}}\mathbf{x} = \mathbf{s}$ where $\tilde{\mathbf{A}} \in \mathbb{R}^{(p+1) \times n}$ is the *node incidence matrix* of the graph,

$$\tilde{A}_{ij} = \begin{cases} 1, & \text{arc } j \text{ leaves node } i, \\ -1, & \text{arc } j \text{ enters node } i, \\ 0, & \text{otherwise.} \end{cases}$$

The flow conservation equation $\tilde{\mathbf{A}}\mathbf{x} = \mathbf{s}$ is inconsistent unless $\mathbf{1}^T \mathbf{s} = 0$, which we assume

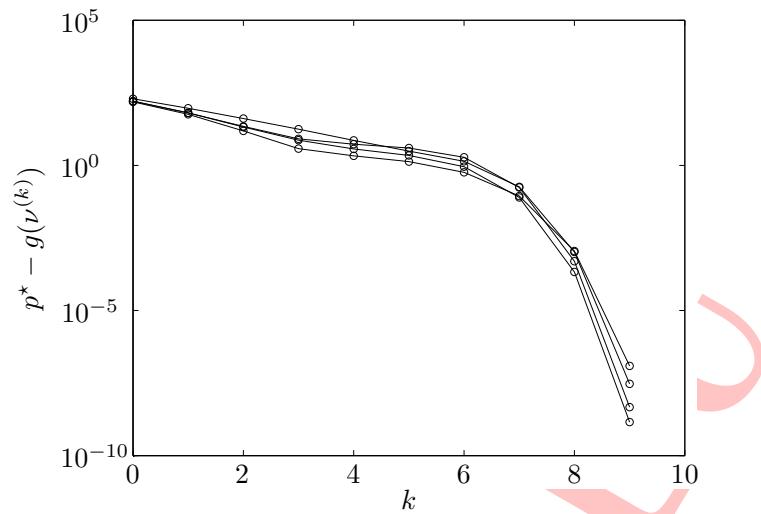


图 7.2: Error $|g(\nu^{(k)}) - p^*|$ in Newton's method, applied to the dual of the equality constrained analytic centering problem.

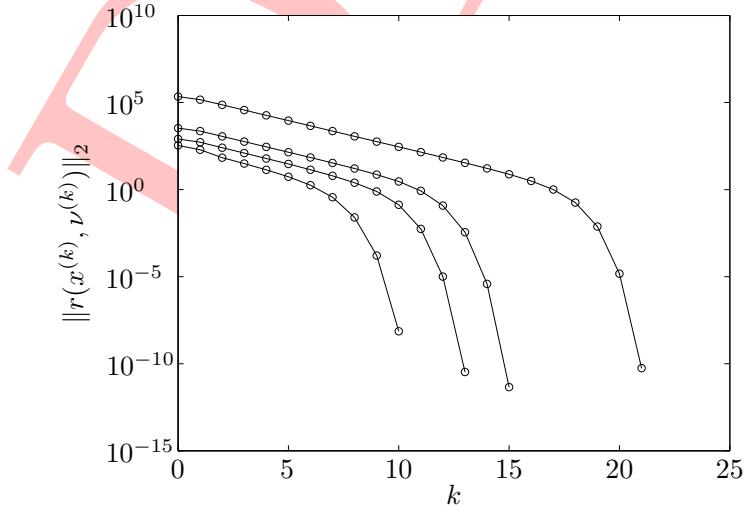


图 7.3: Residual $\|r(\mathbf{x}^{(k)}, \nu^{(k)})\|_2$ in the infeasible start Newton method, applied to the equality constrained analytic centering problem.

is the case. (In other words, the total of the source flows must equal the total of the sink flows.) The flow conservation equations $\tilde{\mathbf{A}}\mathbf{x} = \mathbf{s}$ are also redundant, since $\mathbf{1}^T \tilde{\mathbf{A}} = \mathbf{0}$. To obtain an independent set of equations we can delete any one equation, to obtain $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{p \times n}$ is the reduced node incidence matrix of the graph (i.e., the node incidence matrix with one row removed) and $\mathbf{b} \in \mathbb{R}^p$ is reduced source vector (i.e., \mathbf{s} with the associated entry removed).

In summary, flow conservation is given by $\mathbf{Ax} = \mathbf{b}$, where \mathbf{A} is the reduced node incidence matrix of the graph and \mathbf{b} is the reduced source vector. The matrix \mathbf{A} is very sparse, since each column has at most two nonzero entries (which can only be $+1$ or -1).

We will take traffic flows \mathbf{x} as the variables, and the sources as given. We introduce the objective function

$$f(\mathbf{x}) = \sum_{i=1}^n \phi_i(x_i),$$

where $\phi_i : \mathbb{R} \rightarrow \mathbb{R}$ is the flow cost function for arc i . We assume that the flow cost functions are strictly convex and twice differentiable.

The problem of choosing the best flow, that satisfies the flow conservation requirement, is

$$\min_{\mathbf{x}} \sum_{i=1}^n \phi_i(x_i), \quad s.t. \quad \mathbf{Ax} = \mathbf{b}. \quad (7.42)$$

Here the Hessian \mathbf{H} is diagonal, since the objective is separable.

We have several choices for computing the Newton step for the optimal network flow problem (7.42). The most straightforward is to solve the full KKT system, using a sparse LDL^T factorization. For this problem it is probably better to compute the Newton step using block elimination. We can characterize the sparsity pattern of the Schur complement $\mathbf{S} = -\mathbf{AH}^{-1}\mathbf{A}^T$ in terms of the graph: We have $S_{ij} \neq 0$ if and only if node i and node j are connected by an arc. It follows that if the network is sparse, i.e., if each node is connected by an arc to only a few other nodes, then the Schur complement \mathbf{S} is sparse. In this case, we can exploit sparsity in forming \mathbf{S} , and in the associated factorization and solve steps, as well. We can expect the computational complexity of computing the Newton step to grow approximately linearly with the number of arcs (which is the number of variables).

Optimal control

We consider the problem

$$\begin{aligned} \min_{\mathbf{z}, \mathbf{u}} \quad & \sum_{t=1}^N \phi_t(z(t)) + \sum_{t=0}^{N-1} \psi_t(u(t)), \\ \text{s.t. } z(t+1) = & \mathbf{A}_t z(t) + \mathbf{B}_t u(t), t = 0, \dots, N-1. \end{aligned}$$

Here

- $z(t) \in \mathbb{R}^k$ is the system state at time t
- $u(t) \in \mathbb{R}^l$ is the input or control action at time t
- $\phi_t : \mathbb{R}^k \rightarrow \mathbb{R}$ is the state cost function
- $\psi_t : \mathbb{R}^l \rightarrow \mathbb{R}$ is the input cost function
- N is called the *time horizon* for the problem.

We assume that the input and state cost functions are strictly convex and twice differentiable. The variables in the problem are $u(0), \dots, u(N-1)$, and $z(1), \dots, z(N)$. The initial state $z(0)$ is given. The linear equality constraints are called the *state equations* or *dynamic evolution equations*. We define the overall optimization variable \mathbf{x} as

$$\mathbf{x} = (u(0), z(1), u(1), \dots, u(N-1), z(N))^T \in \mathbb{R}^{N(k+l)}.$$

Since the objective is block separable (i.e., a sum of functions of $z(t)$ and $u(t)$), the Hessian is block diagonal:

$$\mathbf{H} = \text{Diag}(R_0, Q_1, \dots, R_{N-1}, Q_N),$$

where

$$R_t = \nabla^2 \psi_t(u(t)), t = 0, \dots, N-1, \quad Q_t = \nabla^2 \phi_t(z(t)), t = 1, \dots, N.$$

We can collect all the equality constraints (i.e., the state equations) and express them as $\mathbf{Ax} = \mathbf{b}$ where

$$\mathbf{A} = \begin{bmatrix} -\mathbf{B}_0 & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{A}_1 & -\mathbf{B}_1 & \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_2 & -\mathbf{B}_2 & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -\mathbf{A}_{N-1} & -\mathbf{B}_{N-1} & \mathbf{I} \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{A}_0 z(0) \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}.$$

The number of rows of \mathbf{A} (i.e., equality constraints) is Nk .

Directly solving the KKT system for the Newton step, using a dense LDL^T factorization, would cost

$$(1/3)(2Nk + Nl)^3 = (1/3)N^3(2k + l)^3$$

flops. Using a sparse LDL^T factorization would give a large improvement, since the method would exploit the many zero entries in \mathbf{A} and \mathbf{H} .

In fact we can do better by exploiting the special block structure of \mathbf{H} and \mathbf{A} , using block elimination to compute the Newton step. The Schur complement $\mathbf{S} = -\mathbf{AH}^{-1}\mathbf{A}^T$ turns out to be block tridiagonal, with $k \times k$ blocks:

$$\mathbf{S} = -\mathbf{AH}^{-1}\mathbf{A}^T = \begin{bmatrix} \mathbf{S}_{11} & \mathbf{Q}_1^{-1}\mathbf{A}_1^T & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{A}_1\mathbf{Q}_1^{-1} & \mathbf{S}_{22} & \mathbf{Q}_2^{-1}\mathbf{A}_2^T & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2\mathbf{Q}_2^{-1} & \mathbf{S}_{33} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{S}_{N-1,N-1} & \mathbf{Q}_{N-1}^{-1}\mathbf{A}_{N-1}^T \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_{N-1}\mathbf{Q}_{N-1}^{-1} & \mathbf{S}_{NN} \end{bmatrix}$$

where

$$\begin{aligned} \mathbf{S}_{11} &= -\mathbf{B}_0\mathbf{R}_0^{-1}\mathbf{B}_0^T - \mathbf{Q}_1^{-1}, \\ \mathbf{S}_{ii} &= -\mathbf{A}_{i-1}\mathbf{Q}_{i-1}^{-1}\mathbf{A}_{i-1}^T - \mathbf{B}_{i-1}\mathbf{R}_{i-1}^{-1}\mathbf{B}_{i-1}^T - \mathbf{Q}_i^{-1}, i = 2, \dots, N. \end{aligned}$$

In particular, \mathbf{S} is banded, with bandwidth $2k - 1$, so we can factor it in order k^3N flops. Therefore we can compute the Newton step in order k^3N flops, assuming $k \ll N$. Note that this grows linearly with the time horizon N , whereas for a generic method, the flop count grows like N^3 .

For this problem we could go one step further and exploit the block tridiagonal structure of \mathbf{S} . Applying a standard block tridiagonal factorization method would result in the classic Riccati recursion for solving a quadratic optimal control problem. Still, using only the banded nature of \mathbf{S} yields an algorithm that is the same order.

Analytic center of a linear matrix inequality

We consider the problem

$$\begin{aligned} \min_{\mathbf{X}} f(\mathbf{X}) &= -\log \det \mathbf{X}, \\ \text{s.t. } \text{tr}(\mathbf{A}_i \mathbf{X}) &= b_i, i = 1, \dots, p, \end{aligned} \tag{7.43}$$

where $\mathbf{X} \in \mathbb{S}^n$ is the variable, $\mathbf{A}_i \in \mathbb{S}^n$, $b_i \in \mathbb{R}$, and $\text{dom } f = \mathbb{S}_{++}^n$. The KKT conditions for this problem are

$$-\mathbf{X}^{*-1} + \sum_{i=1}^m \nu_i^* \mathbf{A}_i = \mathbf{0}, \quad \text{tr}(\mathbf{A}_i \mathbf{X}^*) = b_i, i = 1, \dots, p. \quad (7.44)$$

The dimension of the variable \mathbf{X} is $n(n+1)/2$. We could simply ignore the special matrix structure of \mathbf{X} , and consider it as (vector) variable $\mathbf{x} \in \mathbb{R}^{n(n+1)/2}$, and solve the problem (7.43) using a generic method for a problem with $n(n+1)/2$ variables and p equality constraints. The cost for computing a Newton step would then be at least

$$(1/3)(n(n+1)/2 + p)^3$$

flops, which is order n^6 in n . We will see that there are a number of far more attractive alternatives.

A first option is to solve the dual problem. The conjugate of f is

$$f^*(\mathbf{Y}) = \log \det(-\mathbf{Y})^{-1} - n$$

with $\text{dom } f^* = -\mathbb{S}_{++}^n$ (see Example 221), so the dual problem is

$$\max_{\boldsymbol{\nu}} -\mathbf{b}^T \boldsymbol{\nu} + \log \det \left(\sum_{i=1}^p \nu_i \mathbf{A}_i \right) + n, \quad (7.45)$$

with domain $\left\{ \boldsymbol{\nu} \mid \sum_{i=1}^p \nu_i \mathbf{A}_i \succ \mathbf{0} \right\}$. This is an unconstrained problem with variable $\boldsymbol{\nu} \in \mathbb{R}^p$. The optimal \mathbf{X}^* can be recovered from the optimal $\boldsymbol{\nu}^*$ by solving the first (dual feasibility) equation in (7.44), i.e., $\mathbf{X}^* = \left(\sum_{i=1}^p \nu_i \mathbf{A}_i \right)^{-1}$.

Let us work out the cost of computing the Newton step for the dual problem (7.45). We have to form the gradient and Hessian of g , and then solve for the Newton step. The gradient and Hessian are given by

$$\begin{aligned} \nabla^2 g(\boldsymbol{\nu})_{ij} &= -\text{tr}(\mathbf{A}^{-1} \mathbf{A}_i \mathbf{A}^{-1} \mathbf{A}_j), \quad i, j = 1, \dots, p, \\ \nabla g(\boldsymbol{\nu})_i &= \text{tr}(\mathbf{A}^{-1} \mathbf{A}_i) - b_i, \quad i = 1, \dots, p, \end{aligned}$$

where $\mathbf{A} = \sum_{i=1}^p \nu_i \mathbf{A}_i$. To form $\nabla^2 g(\boldsymbol{\nu})$ and $\nabla g(\boldsymbol{\nu})$ we proceed as follows. We first form \mathbf{A} (pn^2 flops), and $\mathbf{A}^{-1} \mathbf{A}_j$ for each j ($2pn^3$ flops). Then we form the matrix $\nabla^2 g(\boldsymbol{\nu})$. Each of the $p(p+1)/2$ entries of $\nabla^2 g(\boldsymbol{\nu})$ is the inner product of two matrices in \mathbb{S}^n , each of which costs $n(n+1)$ flops, so the total is (dropping dominated terms) $(1/2)p^2n^2$ flops. Forming $\nabla g(\boldsymbol{\nu})$ is cheap since we already have the matrices $\mathbf{A}^{-1} \mathbf{A}_i$. Finally, we

solve for the Newton step $-\nabla^2 g(\boldsymbol{\nu})^{-1} \nabla g(\boldsymbol{\nu})$, which costs $(1/3)p^3$ flops. All together, and keeping only the leading terms, the total cost of computing the Newton step is $2pn^3 + (1/2)p^2n^2 + (1/3)p^3$. Note that this is order n^3 in n , which is far better than the simple primal method described above, which is order n^6 .

We can also solve the primal problem more efficiently, by exploiting its special matrix structure. To derive the KKT system for the Newton step $\Delta \mathbf{X}_{nt}$ at a feasible \mathbf{X} , we replace \mathbf{X}^* in the KKT conditions by $\mathbf{X} + \Delta \mathbf{X}_{nt}$ and $\boldsymbol{\nu}^*$ by \mathbf{w} , and linearize the first equation using the first-order approximation

$$(\mathbf{X} + \Delta \mathbf{X}_{nt})^{-1} \approx \mathbf{X}^{-1} - \mathbf{X}^{-1} \Delta \mathbf{X}_{nt} \mathbf{X}^{-1}.$$

This gives the KKT system

$$-\mathbf{X}^{-1} + \mathbf{X}^{-1} \Delta \mathbf{X}_{nt} \mathbf{X}^{-1} + \sum_{i=1}^p w_i \mathbf{A}_i = \mathbf{0}, \quad \text{tr}(\mathbf{A}_i \Delta \mathbf{X}_{nt}) = 0, i = 1, \dots, p. \quad (7.46)$$

This is a set of $n(n+1)/2 + p$ linear equations in the variables $\Delta \mathbf{X}_{nt} \in \mathbb{S}^n$ and $\mathbf{w} \in \mathbb{R}^p$. If we solved these equations using a generic method, the cost would be order n^6 .

We can use block elimination to solve the KKT system (7.46) far more efficiently. We eliminate the variable $\Delta \mathbf{X}_{nt}$, by solving the first equation to get

$$\Delta \mathbf{X}_{nt} = \mathbf{X} - \mathbf{X} \left(\sum_{i=1}^p w_i \mathbf{A}_i \right) \mathbf{X} = \mathbf{X} - \sum_{i=1}^p w_i \mathbf{X} \mathbf{A}_i \mathbf{X}. \quad (7.47)$$

Substituting this expression for $\Delta \mathbf{X}_{nt}$ into the other equation gives

$$\text{tr}(\mathbf{A}_j \Delta \mathbf{X}_{nt}) = \text{tr}(\mathbf{A}_j \mathbf{X}) - \sum_{i=1}^p w_i \text{tr}(\mathbf{A}_j \mathbf{X} \mathbf{A}_i \mathbf{X}) = 0, \quad j = 1, \dots, p.$$

This is a set of p linear equations in \mathbf{w} :

$$\mathbf{C}\mathbf{w} = \mathbf{d}$$

where $C_{ij} = \text{tr}(\mathbf{A}_i \mathbf{X} \mathbf{A}_j \mathbf{X})$, $d_i = \text{tr}(\mathbf{A}_i \mathbf{X})$. The coefficient matrix \mathbf{C} is symmetric and positive definite, so a Cholesky factorization can be used to find \mathbf{w} . Once we have \mathbf{w} , we can compute $\Delta \mathbf{X}_{nt}$ from (7.47).

The cost of this method is as follows. We form the products $\mathbf{A}_i \mathbf{X}$ ($2pn^3$ flops), and then form the matrix \mathbf{C} . Each of the $p(p+1)/2$ entries of \mathbf{C} is the inner product of two matrices in $\mathbb{R}^{n \times n}$, so forming \mathbf{C} costs $p^2 n^2$ flops. Then we solve for $\mathbf{w} = \mathbf{C}^{-1} \mathbf{d}$, which costs $(1/3)p^3$. Finally we compute $\Delta \mathbf{X}_{nt}$. If we use the first expression in (7.47), i.e., first compute the sum and then pre- and post-multiply with \mathbf{X} , the cost is approximately

$pn^2 + 3n^3$. All together, the total cost is $2pn^3 + p^2n^2 + (1/3)p^3$ flops to form the Newton step for the primal problem, using block elimination. This is far better than the simple method, which is order n^6 . Note also that the cost is the same as that of computing the Newton step for the dual problem.

7.1.4.5 Eliminating equality constraints

One general approach to solving the equality constrained problem (7.34) is to eliminate the equality constraints and then solve the resulting unconstrained problem using methods for unconstrained minimization. We first find a matrix $\mathbf{F} \in \mathbb{R}^{n \times (n-p)}$ and vector $\hat{\mathbf{x}} \in \mathbb{R}^n$ that parametrize the (affine) feasible set:

$$\{\mathbf{x} | \mathbf{A}\mathbf{x} = \mathbf{b}\} = \{\mathbf{F}\mathbf{z} + \hat{\mathbf{x}} | \mathbf{z} \in \mathbb{R}^{n-p}\}.$$

Here $\hat{\mathbf{x}}$ can be chosen as any particular solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$, and $\mathbf{F} \in \mathbb{R}^{n \times (n-p)}$ is any matrix whose range is the nullspace of \mathbf{A} . We then form the reduced or eliminated optimization problem

$$\min_{\mathbf{z}} \tilde{f}(\mathbf{z}) \triangleq f(\mathbf{F}\mathbf{z} + \hat{\mathbf{x}}), \quad (7.48)$$

which is an unconstrained problem with variable $\mathbf{z} \in \mathbb{R}^{n-p}$. From its solution \mathbf{z}^* , we can find the solution of the equality constrained problem as $\mathbf{x}^* = \mathbf{F}\mathbf{z}^* + \hat{\mathbf{x}}$.

We can also construct an optimal dual variable $\boldsymbol{\nu}^*$ for the equality constrained problem, as

$$\boldsymbol{\nu}^* = -(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A}\nabla f(\mathbf{x}^*).$$

To show that this expression is correct, we must verify that the dual feasibility condition

$$\nabla f(\mathbf{x}^*) + \mathbf{A}^T(-(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A}\nabla f(\mathbf{x}^*)) = \mathbf{0} \quad (7.49)$$

holds. To show this, we note that

$$\begin{bmatrix} \mathbf{F}^T \\ \mathbf{A} \end{bmatrix} (\nabla f(\mathbf{x}^*) - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A}\nabla f(\mathbf{x}^*)) = \mathbf{0},$$

where in the top block we use $\mathbf{F}^T\nabla f(\mathbf{x}^*) = \nabla \tilde{f}(\mathbf{z}^*) = \mathbf{0}$ and $\mathbf{A}\mathbf{F} = \mathbf{0}$. Since the matrix on the left is nonsingular, this implies (7.49).

Example 634 (Optimal allocation with resource constraint). *We consider the problem*

$$\begin{aligned} \min_{\mathbf{x}} \quad & \sum_{i=1}^n f_i(x_i), \\ \text{s.t.} \quad & \sum_{i=1}^n x_i = b, \end{aligned}$$

where the functions $f_i : \mathbb{R} \rightarrow \mathbb{R}$ are convex and twice differentiable, and $b \in \mathbb{R}$ is a problem parameter. We interpret this as the problem of optimally allocating a single resource, with a fixed total amount b (the budget) to n otherwise independent activities.

We can eliminate x_n (for example) using the parametrization

$$x_n = b - x_1 - \cdots - x_{n-1},$$

which corresponds to the choices

$$\hat{\mathbf{x}} = b\mathbf{e}_n, \quad \mathbf{F} = \begin{bmatrix} \mathbf{I} \\ -\mathbf{1}^T \end{bmatrix} \in \mathbb{R}^{n \times (n-1)}.$$

The reduced problem is then

$$\min_{\mathbf{x}} f_n(b - x_1 - \cdots - x_{n-1}) + \sum_{i=1}^{n-1} f_i(x_i).$$

7.1.4.6 Solving equality constrained problems via the dual

Another approach to solving (7.34) is to solve the dual, and then recover the optimal primal variable \mathbf{x}^* , as described in Section 6.6.5.4. The dual function of (7.34) is

$$\begin{aligned} g(\boldsymbol{\nu}) &= -\mathbf{b}^T \boldsymbol{\nu} + \inf_{\mathbf{x}} (f(\mathbf{x}) + \boldsymbol{\nu}^T \mathbf{A}\mathbf{x}) \\ &= -\mathbf{b}^T \boldsymbol{\nu} - \sup_{\mathbf{x}} ((-\mathbf{A}^T \boldsymbol{\nu})^T \mathbf{x} - f(\mathbf{x})) \\ &= -\mathbf{b}^T \boldsymbol{\nu} - f^*(-\mathbf{A}^T \boldsymbol{\nu}), \end{aligned}$$

where f^* is the conjugate of f . So the dual problem is:

$$\max_{\boldsymbol{\nu}} -\mathbf{b}^T \boldsymbol{\nu} - f^*(-\mathbf{A}^T \boldsymbol{\nu}).$$

Since by assumption there is an optimal point, the problem is strictly feasible, so Slater's condition holds. Therefore strong duality holds, and the dual optimum is attained, i.e., there exists a $\boldsymbol{\nu}^*$ with $g(\boldsymbol{\nu}^*) = p^*$.

If the dual function g is twice differentiable, then the methods for unconstrained minimization can be used to maximize g . (In general, the dual function g need not be twice differentiable, even if f is.) Once we find an optimal dual variable $\boldsymbol{\nu}^*$, we reconstruct an optimal primal solution \mathbf{x}^* from it. (This is not always straightforward; see Section 6.6.5.4.)

Example 635 (Equality constrained analytic center). We consider the problem

$$\min_{\mathbf{x}} f(\mathbf{x}) = -\sum_{i=1}^n \log x_i, \quad s.t. \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \tag{7.50}$$

where $\mathbf{A} \in \mathbb{R}^{p \times n}$, with implicit constraint $\mathbf{x} > \mathbf{0}$. Using

$$f^*(\mathbf{y}) = \sum_{i=1}^n (-1 - \log(-y_i)) = -n - \sum_{i=1}^n \log(-y_i)$$

(with $\text{dom } f^* = -\mathbb{R}_{++}^n$), the dual problem is

$$\max_{\boldsymbol{\nu}} g(\boldsymbol{\nu}) = -\mathbf{b}^T \boldsymbol{\nu} + n + \sum_{i=1}^n \log(\mathbf{A}^T \boldsymbol{\nu})_i, \quad (7.51)$$

with implicit constraint $\mathbf{A}^T \boldsymbol{\nu} > \mathbf{0}$. Here we can easily solve the dual feasibility equation, i.e., find the \mathbf{x} that minimizes $L(\mathbf{x}, \boldsymbol{\nu})$:

$$\nabla f(\mathbf{x}) + \mathbf{A}^T \boldsymbol{\nu} = -(1/x_1, \dots, 1/x_n) + \mathbf{A}^T \boldsymbol{\nu} = \mathbf{0},$$

and so

$$x_i(\boldsymbol{\nu}) = 1/(\mathbf{A}^T \boldsymbol{\nu})_i. \quad (7.52)$$

To solve the equality constrained analytic centering problem (7.50), we solve the (unconstrained) dual problem (7.51), and then recover the optimal solution of (7.50) via (7.52).

7.1.5 Newton's method with equality constraints

In this section we describe an extension of Newton's method to include equality constraints. The method is almost the same as Newton's method without constraints, except for two differences: The initial point must be feasible (i.e., satisfy $\mathbf{x} \in \text{dom } f$ and $\mathbf{Ax} = \mathbf{b}$), and the definition of Newton step is modified to take the equality constraints into account. In particular, we make sure that the Newton step $\Delta \mathbf{x}_{nt}$ is a feasible direction, i.e., $\mathbf{A}\Delta \mathbf{x}_{nt} = \mathbf{0}$.

7.1.5.1 The Newton step

Definition via second-order approximation

To derive the Newton step $\Delta \mathbf{x}_{nt}$ for the equality constrained problem (7.34) at the feasible point \mathbf{x} , we replace the objective with its second-order Taylor approximation near \mathbf{x} , to form the problem

$$\begin{aligned} \min_{\mathbf{v}} \hat{f}(\mathbf{x} + \mathbf{v}) &= f(\mathbf{x}) + \nabla f(\mathbf{x})^T \mathbf{v} + (1/2)\mathbf{v}^T \nabla^2 f(\mathbf{x}) \mathbf{v}, \\ \text{s.t. } \mathbf{A}(\mathbf{x} + \mathbf{v}) &= \mathbf{b}. \end{aligned} \quad (7.53)$$

This is a (convex) quadratic minimization problem with equality constraints, and can be solved analytically. We define $\Delta \mathbf{x}_{nt}$, the Newton step at \mathbf{x} , as the solution of the convex

quadratic problem (7.53), assuming the associated KKT matrix is nonsingular. In other words, the Newton step $\Delta\mathbf{x}_{nt}$ is what must be added to \mathbf{x} to solve the problem when the quadratic approximation is used in place of f .

From our analysis in Section 7.1.4.2 of the equality constrained quadratic problem, the Newton step $\Delta\mathbf{x}_{nt}$ is characterized by

$$\begin{pmatrix} \nabla^2 f(\mathbf{x}) & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \Delta\mathbf{x}_{nt} \\ \mathbf{w} \end{pmatrix} = \begin{pmatrix} -\nabla f(\mathbf{x}) \\ \mathbf{0} \end{pmatrix}, \quad (7.54)$$

where w is the associated optimal dual variable for the quadratic problem. The Newton step is defined only at points for which the KKT matrix is nonsingular.

As in Newton's method for unconstrained problems, we observe that when the objective f is exactly quadratic, the Newton update $\mathbf{x} + \Delta\mathbf{x}_{nt}$ exactly solves the equality constrained minimization problem, and in this case the vector \mathbf{w} is the optimal dual variable for the original problem. This suggests, as in the unconstrained case, that when f is nearly quadratic, $\mathbf{x} + \Delta\mathbf{x}_{nt}$ should be a very good estimate of the solution \mathbf{x}^* , and \mathbf{w} should be a good estimate of the optimal dual variable $\boldsymbol{\nu}^*$.

Solution of linearized optimality conditions

We can interpret the Newton step $\Delta\mathbf{x}_{nt}$, and the associated vector w , as the solutions of a linearized approximation of the optimality conditions

$$\mathbf{A}\mathbf{x}^* = \mathbf{b}, \quad \nabla f(\mathbf{x}^*) + \mathbf{A}^T \boldsymbol{\nu}^* = \mathbf{0}.$$

We substitute $\mathbf{x} + \Delta\mathbf{x}_{nt}$ for \mathbf{x}^* and \mathbf{w} for $\boldsymbol{\nu}^*$, and replace the gradient term in the second equation by its linearized approximation near \mathbf{x} , to obtain the equations

$$\mathbf{A}(\mathbf{x} + \Delta\mathbf{x}_{nt}) = \mathbf{b}, \quad \nabla f(\mathbf{x} + \Delta\mathbf{x}_{nt}) + \mathbf{A}^T \mathbf{w} \approx \nabla f(\mathbf{x}) + \nabla^2 f(\mathbf{x}) \Delta\mathbf{x}_{nt} + \mathbf{A}^T \mathbf{w} = \mathbf{0}.$$

Using $\mathbf{A}\mathbf{x} = \mathbf{b}$, these become

$$\mathbf{A}\Delta\mathbf{x}_{nt} = \mathbf{0}, \quad \nabla^2 f(\mathbf{x}) \Delta\mathbf{x}_{nt} + \mathbf{A}^T \mathbf{w} = -\nabla f(\mathbf{x}), \quad (7.55)$$

which are precisely the equations (7.54) that define the Newton step.

The Newton decrement

We define the Newton decrement for the equality constrained problem as

$$\lambda(\mathbf{x}) = (\Delta\mathbf{x}_{nt} \nabla^2 f(\mathbf{x}) \Delta\mathbf{x}_{nt})^{1/2}. \quad (7.56)$$

Then

$$f(\mathbf{x}) - \inf_{\mathbf{v}} \{\hat{f}(\mathbf{x} + \mathbf{v}) | \mathbf{A}(\mathbf{x} + \mathbf{v}) = \mathbf{b}\} = \frac{1}{2} \lambda^2(\mathbf{x}).$$

So $\frac{1}{2}\lambda^2(\mathbf{x})$ gives an estimate of $f(\mathbf{x}) - p^*$ and can serve as a good stopping criterion.

Feasible descent direction

$\Delta\mathbf{x}_{nt}$ is feasible as $\mathbf{A}\Delta\mathbf{x}_{nt} = \mathbf{0}$. It also gives a descent direction because by (7.55)

$$\nabla f(\mathbf{x})^T \Delta\mathbf{x}_{nt} = -\lambda^2(\mathbf{x}) < 0.$$

7.1.5.2 Newton's method with equality constraints

The outline of Newton's method with equality constraints is exactly the same as for unconstrained problems, as summarized in Algorithm 5.

Algorithm 5 Newton's method for equality constrained minimization

Given: starting point $\mathbf{x} \in \text{dom } f$ with $\mathbf{Ax} = \mathbf{b}$, tolerance $\epsilon > 0$.

Repeat

1. Compute the Newton step and decrement $\Delta\mathbf{x}_{nt}$, $\lambda(\mathbf{x})$.
 2. *Stopping criterion.* Quit if $\lambda^2/2 \leq \epsilon$.
 3. *Line search.* Choose step size t by backtracking line search.
 4. *Update.* $\mathbf{x} := \mathbf{x} + t\Delta\mathbf{x}_{nt}$.
-

The method is called a *feasible descent* method, since all the iterates are feasible, with $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$ (unless $\mathbf{x}^{(k)}$ is optimal). Newton's method requires that the KKT matrix be invertible at each \mathbf{x} ; we will be more precise about the assumptions required for convergence in Section 10.2.4 of [18].

7.1.6 Lagrangian Algorithms

In this section we consider an optimization method based on the Lagrangian function (see Section 6.7.4). The basic idea is to use gradient algorithms to update simultaneously the decision variable and Lagrange multiplier vector. We consider first the case with equality constraints, followed by inequality constraints.

7.1.6.1 Dual Derivatives and Subgradients

(Taken from Section 6.1 of [12])

We focus on the primal problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{s.t. } & \mathbf{x} \in \mathcal{X}, g_j(\mathbf{x}) \leq 0, j = 1, \dots, r, \end{aligned} \tag{7.57}$$

and its dual

$$\begin{aligned} \max_{\boldsymbol{\mu}} q(\boldsymbol{\mu}) \\ s.t. \quad \boldsymbol{\mu} \geq \mathbf{0}, \end{aligned} \tag{7.58}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $g_j : \mathbb{R}^n \rightarrow \mathbb{R}$ are given functions, \mathcal{X} is a subset of \mathbb{R}^n , and

$$q(\boldsymbol{\mu}) = \inf_{\mathbf{x} \in \mathcal{X}} L(\mathbf{x}, \boldsymbol{\mu}) = \inf_{\mathbf{x} \in \mathcal{X}} \{f(\mathbf{x}) + \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x})\}$$

is the dual function. We may also consider additional equality constraints in the primal problem, and on occasion we pause to discuss special issues regarding their treatment.

It is worth reflecting on the potential incentives for solving the dual problem in place of the primal. These are:

- a) The dual is a concave problem (concave cost, convex constraint set). By contrast, the primal need not be convex.
- b) The dual may have smaller dimension and/or simpler constraints than the primal.
- c) If there is no duality gap and the dual is solved exactly to yield a Lagrange multiplier $\boldsymbol{\mu}^*$, all optimal primal solutions can be obtained by minimizing the Lagrangian $L(\mathbf{x}, \boldsymbol{\mu}^*)$ over $\mathbf{x} \in \mathcal{X}$ (however, there may be additional minimizers of $L(\mathbf{x}, \boldsymbol{\mu}^*)$ than the primal-infeasible). Furthermore, if the dual is solved approximately to yield an approximate Lagrange multiplier $\boldsymbol{\mu}$, and \mathbf{x}_μ minimizes $L(\mathbf{x}, \boldsymbol{\mu})$ over $\mathbf{x} \in \mathcal{X}$, then it can be seen by applying Proposition 5.1.5 of [12], that \mathbf{x}_μ also solves the problem

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) \\ s.t. \quad \mathbf{x} \in \mathcal{X}, g_j(\mathbf{x}) \leq g_j(\mathbf{x}_\mu), j = 1, \dots, r. \end{aligned} \tag{7.59}$$

Thus if the constraint violations $g_j(\mathbf{x}_\mu)$ are not much larger than zero, \mathbf{x}_μ may be an acceptable practical solution.

- d) Even if there is a duality gap, for every $\boldsymbol{\mu} \geq \mathbf{0}$, the dual value $q(\boldsymbol{\mu})$ is a lower bound to the optimal primal value (the weak duality theorem). This lower bound may be useful in the context of discrete optimization and branch and bound procedures.

We should also consider some of the difficulties in solving the dual problem. The most important ones are the following:

- a) To evaluate the dual function at any $\boldsymbol{\mu}$ requires minimization of the Lagrangian $L(\mathbf{x}, \boldsymbol{\mu})$ over $\mathbf{x} \in \mathcal{X}$. In effect, this restricts the utility of dual methods to problems where this minimization can either be done in closed form or else is relatively simple; for example, when there is special structure that allows decomposition, as in the separable problems of Section 5.1.6 of [12] and the monotropic programming problems of Section 5.4.1 of [12].
- b) In many types of problems, the dual function is nondifferentiable, in which algorithms for smooth objective functions do not apply.
- c) Even if we find an optimal dual solution $\boldsymbol{\mu}^*$, it may be difficult to obtain a primal feasible vector \mathbf{x} from the minimization of $L(\mathbf{x}, \boldsymbol{\mu}^*)$ over $\mathbf{x} \in \mathcal{X}$ as required by the primal-dual optimality conditions of Proposition 5.1.5 of [12], since this minimization can also yield primal-infeasible vectors.

Another important point regarding large-scale optimization problems is that there are several different ways to introduce duality in their solution. For example an alternative strategy to take advantage of separability, often called *partitioning*, is to divide the variables in two subsets, and minimizing first with respect to one subset while taking advantage of whatever simplification may arise by fixing the variables in the other subset. In particular, the problem

$$\begin{aligned} & \min_{\mathbf{x}, \mathbf{y}} F(\mathbf{x}) + G(\mathbf{y}) \\ & \text{s.t. } \mathbf{Ax} + \mathbf{By} = \mathbf{c}, \mathbf{x} \in \mathcal{X}, \mathbf{y} \in \mathcal{Y} \end{aligned} \tag{7.60}$$

can be written as

$$\begin{aligned} & \min_{\mathbf{x}} F(\mathbf{x}) + \inf_{\mathbf{By}=\mathbf{c}-\mathbf{Ax}, \mathbf{y} \in \mathcal{Y}} G(\mathbf{y}) \\ & \text{s.t. } \mathbf{x} \in \mathcal{X} \end{aligned} \tag{7.61}$$

or

$$\begin{aligned} & \min_{\mathbf{x}} F(\mathbf{x}) + \tilde{p}(\mathbf{c} - \mathbf{Ax}) \\ & \text{s.t. } \mathbf{x} \in \mathcal{X} \end{aligned} \tag{7.62}$$

where $\tilde{p}(\cdot)$ is the primal function of the minimization problem involving \mathbf{y} above:

$$\tilde{p}(\mathbf{u}) = \inf_{\mathbf{By}=\mathbf{u}, \mathbf{y} \in \mathcal{Y}} G(\mathbf{y}).$$

Assuming no duality gap, this primal function and its subgradients can be calculated using the corresponding dual function and associated Lagrange multipliers.

Naturally, the differentiability properties of dual functions are a very important determinant of the type of dual method that is appropriate for a given problem. We consequently develop these properties.

For a given $\boldsymbol{\mu} \in \mathbb{R}^r$, suppose that \mathbf{x}_μ minimizes the Lagrangian $L(\mathbf{x}, \boldsymbol{\mu})$ over $\mathbf{x} \in \mathcal{X}$,

$$\mathbf{x}_\mu = \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmin}} L(\mathbf{x}, \boldsymbol{\mu}) = \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmin}} \{f(\mathbf{x}) + \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x})\}.$$

An important fact for our purposes is that $\mathbf{g}(\mathbf{x}_\mu)$ is a subgradient of the dual function q at $\boldsymbol{\mu}$, i.e.,

$$q(\bar{\boldsymbol{\mu}}) \leq q(\boldsymbol{\mu}) + (\bar{\boldsymbol{\mu}} - \boldsymbol{\mu})^T \mathbf{g}(\mathbf{x}_\mu), \quad \forall \bar{\boldsymbol{\mu}} \in \mathbb{R}^r. \quad (7.63)$$

To see this, we use the definition of q and \mathbf{x}_μ to write for all $\bar{\boldsymbol{\mu}} \in \mathbb{R}^r$,

$$\begin{aligned} q(\bar{\boldsymbol{\mu}}) &= \inf_{\mathbf{x} \in \mathcal{X}} \{f(\mathbf{x}) + \bar{\boldsymbol{\mu}}^T \mathbf{g}(\mathbf{x})\} \\ &\leq f(\mathbf{x}_\mu) + \bar{\boldsymbol{\mu}}^T \mathbf{g}(\mathbf{x}_\mu) \\ &= f(\mathbf{x}_\mu) + \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x}_\mu) + (\bar{\boldsymbol{\mu}} - \boldsymbol{\mu})^T \mathbf{g}(\mathbf{x}_\mu) \\ &= q(\boldsymbol{\mu}) + (\bar{\boldsymbol{\mu}} - \boldsymbol{\mu})^T \mathbf{g}(\mathbf{x}_\mu). \end{aligned}$$

Note that this calculation is valid for all $\boldsymbol{\mu} \in \mathbb{R}^r$ for which there is a minimizing vector \mathbf{x}_μ , regardless of whether $\boldsymbol{\mu} \geq \mathbf{0}$.

What is particularly important here is that we need to compute \mathbf{x}_μ anyway in order to evaluate the dual function at $\boldsymbol{\mu}$, so a subgradient $\mathbf{g}(\mathbf{x}_\mu)$ is obtained *essentially at no cost*. All of the dual methods to be discussed solve the dual problems by computing the dual function value and a subgradient at a sequence of vectors $\{\boldsymbol{\mu}^k\}$. It is not necessary to compute the set of all subgradients at $\boldsymbol{\mu}^k$ in these methods; a single subgradient is sufficient.

Despite the fact that the full set of subgradients at a point is not needed for the application, of the following methodology it is still useful to have characterizations of this set. For example, it is important to derive conditions under which q is differentiable. We know from our preceding discussion that if q is differentiable at $\boldsymbol{\mu}$, there can be at most one value of $\mathbf{g}(\mathbf{x}_\mu)$ corresponding to vectors $\mathbf{x}_\mu \in \mathcal{X}$ minimizing $L(\mathbf{x}, \boldsymbol{\mu})$. This suggests that q is everywhere differentiable (as well real-valued and concave) if for all $\boldsymbol{\mu}$, $L(\mathbf{x}, \boldsymbol{\mu})$ is minimized at a unique $\mathbf{x}_\mu \in \mathcal{X}$. Indeed this can be inferred under some assumptions from the convexity. In particular, we have the following proposition.

Proposition 636. Let \mathcal{X} be a compact set, and let f and \mathbf{g} be continuous over \mathcal{X} . Assume also that for every $\boldsymbol{\mu} \in \mathbb{R}^r$, $L(\mathbf{x}, \boldsymbol{\mu})$ is minimized over $\mathbf{x} \in \mathcal{X}$ at a unique point \mathbf{x}_μ . Then q is everywhere continuously differentiable and

$$\nabla q(\boldsymbol{\mu}) = \mathbf{g}(\mathbf{x}_\mu), \quad \forall \boldsymbol{\mu} \in \mathbb{R}^r.$$

Proof. To assert the uniqueness of the subgradient of q at $\boldsymbol{\mu}$, apply Danskin's theorem (Theorem 193) with the identifications $\mathbf{x} \sim \mathbf{z}$, $\mathcal{X} \sim \mathcal{Z}$, $\boldsymbol{\mu} \sim \mathbf{x}$, and $-L(\mathbf{x}, \boldsymbol{\mu}) \sim \phi(\mathbf{x}, \mathbf{z})$. The assumptions of this theorem are satisfied because \mathcal{X} is compact, while $L(\mathbf{x}, \boldsymbol{\mu})$ is continuous as a function of \mathbf{x} and concave (in fact linear) as a function of $\boldsymbol{\mu}$. The continuity of the dual gradient ∇q follows from Proposition B.23 of [12]. \square

Note that if the constraint functions g_j are linear, \mathcal{X} is convex and compact, and f is strictly convex, then the assumptions of Proposition 636 are satisfied and the dual function g is differentiable.

In the case where \mathcal{X} is a discrete set, as for example in integer programming, the continuity and compactness assumptions of Proposition 636 are satisfied, but there typically exist some $\boldsymbol{\mu}$ for which $L(\mathbf{x}, \boldsymbol{\mu})$ has multiple minima, leading to nondifferentiabilities. In fact, it can be shown that if there exists a duality gap, the dual function is nondifferentiable at every dual optimal solution; see Exercise 6.1.1 of [12]. Thus, nondifferentiabilities tend to arise at the most interesting points and cannot be ignored in dual methods.

Even though a subgradient may not be a direction of ascent at points $\boldsymbol{\mu}$ where $q(\boldsymbol{\mu})$ is nondifferentiable, it still maintains an important property of the gradient: it makes an angle less than 90 degrees with all ascent directions at $\boldsymbol{\mu}$, i.e., all the vectors $\alpha(\bar{\boldsymbol{\mu}} - \boldsymbol{\mu})$ such that $\alpha > 0$ and $q(\bar{\boldsymbol{\mu}}) > q(\boldsymbol{\mu})$. In particular, a small move from $\boldsymbol{\mu}$ along any subgradient at $\boldsymbol{\mu}$ decreases the distance to any maximizer $\boldsymbol{\mu}^*$ of q . This property follows from Eqn. (7.63). It will form the basis for a number of dual methods that use subgradients (see Section 6.3 of [12]).

7.1.6.2 Lagrangian Algorithm for Equality Constraints

(Since the algorithm is not the classic Lagrangian algorithm, this section will be replaced.)

Consider the following optimization problem with equality constraints:

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{x}) = \mathbf{0}. \end{aligned}$$

where $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{M}^m$. Recall that for this problem the Lagrangian function is given by

$$l(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}), \quad (7.64)$$

Assume that $f, h \in \mathcal{C}^2$; as usual, denote the Hessian of the Lagrangian by $\mathbf{L}(\mathbf{x}, \boldsymbol{\lambda})$.

The Lagrangian algorithm for this problem is given by

$$\begin{aligned} \mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} - \alpha_k (\nabla f(\mathbf{x}^{(k)}) + D\mathbf{h}(\mathbf{x}^{(k)})^T \boldsymbol{\lambda}^{(k)}), \\ \boldsymbol{\lambda}^{(k+1)} &= \boldsymbol{\lambda}^{(k)} + \beta_k \mathbf{h}(\mathbf{x}^{(k)}). \end{aligned} \quad (7.65)$$

Notice that the update equation for $\mathbf{x}^{(k)}$ is a gradient algorithm for minimizing the Lagrangian with respect to its \mathbf{x} argument, and the update equation for $\boldsymbol{\lambda}^{(k)}$ is a gradient algorithm for maximizing the Lagrangian with respect to its $\boldsymbol{\lambda}$ argument. Because only the gradient is used, the method is also called the first-order Lagrangian algorithm.

The following lemma establishes that if the algorithm converges, the limit must satisfy the Lagrange condition. More specifically, the lemma states that any fixed point of the algorithm must satisfy the Lagrange condition. A fixed point of an update algorithm is simply a point with the property that when updated using the algorithm, the resulting point is equal to the given point. For the case of the Lagrangian algorithm, which updates both $\mathbf{x}^{(k)}$ and $\boldsymbol{\lambda}^{(k)}$ vectors, a fixed point is a pair of vectors. If the Lagrangian algorithm converges, the limit must be a fixed point. We omit the proof of the lemma because it follows easily by inspection.

Lemma 637. *For the Lagrangian algorithm for updating $\mathbf{x}^{(k)}$ and $\boldsymbol{\lambda}^{(k)}$, the pair $(\mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k)})$ is a fixed point if and only if it satisfies the Lagrange condition.*

Below, we use $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ to denote a pair satisfying the Lagrange condition. Assume that $\mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \succ 0$. Also assume that \mathbf{x}^* is a regular point. With these assumptions, we are now ready to state and prove that the algorithm is locally convergent. For simplicity, we will take α_k and β_k to be fixed constants (not depending on k), denoted α and β , respectively.

Theorem 638. *For the Lagrangian algorithm for updating $\mathbf{x}^{(k)}$ and $\boldsymbol{\lambda}^{(k)}$, provided that α and β are sufficiently small, there is a neighborhood of $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ such that if the pair $(\mathbf{x}^0, \boldsymbol{\lambda}^0)$ is in this neighborhood, then the the algorithm converges to $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ with at least a linear order of convergence.*

Proof. We can rescale \mathbf{x} and $\boldsymbol{\lambda}$ by appropriate constants (so that the assumptions are preserved) and effectively change the relative values of the step sizes for the update equations. Therefore, without loss of generality, we can take $\beta = \alpha$.

We will set up our proof by introducing some convenient notation. Given a pair $(\mathbf{x}, \boldsymbol{\lambda})$, let $\mathbf{w} = [\mathbf{x}^T, \boldsymbol{\lambda}^T]^T$ be the $(n + m)$ -vector constructed by concatenating \mathbf{x} and $\boldsymbol{\lambda}$. Similarly define $\mathbf{w}^{(k)} = [\mathbf{x}^{(k)T}, \boldsymbol{\lambda}^{(k)T}]^T$ and $\mathbf{w}^* = [\mathbf{x}^{*T}, \boldsymbol{\lambda}^{*T}]^T$. Define the map $\mathbf{U} : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^{n \times m}$ by

$$\mathbf{U}(\mathbf{w}) = \begin{bmatrix} \mathbf{x} - \alpha(\nabla f(\mathbf{x}) + D\mathbf{h}(\mathbf{x})^T \boldsymbol{\lambda}) \\ \boldsymbol{\lambda} + \alpha\mathbf{h}(\mathbf{x}) \end{bmatrix} \quad (7.66)$$

Then, the Lagrangian algorithm can be rewritten as

$$\mathbf{w}^{(k+1)} = \mathbf{U}(\mathbf{w}^{(k)}). \quad (7.67)$$

We now write $\|\mathbf{w}^{(k+1)} - \mathbf{w}^*\|$ in terms of $\|\mathbf{w}^{(k)} - \mathbf{w}^*\|$, where $\|\cdot\|$ denotes the usual Euclidean norm. By Lemma 637, $\mathbf{w}^* = [\mathbf{x}^{*T}, \boldsymbol{\lambda}^{*T}]^T$ is a fixed point of $\mathbf{w}^{(k+1)} = \mathbf{U}(\mathbf{w}^{(k)})$. Therefore,

$$\|\mathbf{w}^{(k+1)} - \mathbf{w}^*\| = \|\mathbf{U}(\mathbf{w}^{(k)}) - \mathbf{U}(\mathbf{w}^*)\|. \quad (7.68)$$

Let $D\mathbf{U}$ be the (matrix) derivative of \mathbf{U} :

$$D\mathbf{U}(\mathbf{w}) = \mathbf{I} + \alpha \begin{bmatrix} -\mathbf{L}(\mathbf{x}, \boldsymbol{\lambda}) & -D\mathbf{h}(\mathbf{x})^T \\ D\mathbf{h}(\mathbf{x}) & \mathbf{0} \end{bmatrix}. \quad (7.69)$$

By the mean value theorem (see Theorem 5.9),

$$\mathbf{U}(\mathbf{w}^{(k)}) - \mathbf{U}(\mathbf{w}^*) = \mathbf{G}(\mathbf{w}^{(k)})(\mathbf{w}^{(k)} - \mathbf{w}^*), \quad (7.70)$$

where $\mathbf{G}(\mathbf{w}^{(k)})$ is a matrix whose rows are the rows of $D\mathbf{U}$ evaluated at points that lie on the line segment joining $\mathbf{w}^{(k)}$ and \mathbf{w}^* (these points may differ from row to row). Taking norms of both sides of the equation above,

$$\|\mathbf{U}(\mathbf{w}^{(k)}) - \mathbf{U}(\mathbf{w}^*)\| = \|\mathbf{G}(\mathbf{w}^{(k)})\| \|(\mathbf{w}^{(k)} - \mathbf{w}^*)\|. \quad (7.71)$$

We now claim that for sufficiently small $\alpha > 0$, $\|D\mathbf{U}(\mathbf{w}^*)\| < 1$. Our argument here follows (Section 4.4 of [12]). Let

$$\mathbf{M} = \begin{bmatrix} -\mathbf{L}(\mathbf{x}, \boldsymbol{\lambda}) & -D\mathbf{h}(\mathbf{x})^T \\ D\mathbf{h}(\mathbf{x}) & \mathbf{0} \end{bmatrix}, \quad (7.72)$$

so that $D\mathbf{U}(\mathbf{w}^*) = \mathbf{I} + \alpha\mathbf{M}$. Hence, to prove the claim, it suffices to show that the eigenvalues of \mathbf{M} all lie in the open left-half complex plane.

For any complex vector \mathbf{y} , let \mathbf{y}^H represent its complex conjugate transpose (or Hermitian) and $\mathcal{R}(\mathbf{y})$ its real part. Let λ be an eigenvalue of \mathbf{M} and $\mathbf{w} = [\mathbf{x}^T, \boldsymbol{\lambda}^T]^T \neq 0$

be a corresponding eigenvector. Now, $\mathcal{R}(\mathbf{w}^H \mathbf{M} \mathbf{w}) = \mathcal{R}(\lambda) \|\mathbf{w}\|^2$. However, from the structure of \mathbf{M} , we can readily see that

$$\begin{aligned}\mathcal{R}(\mathbf{w}^H \mathbf{M} \mathbf{w}) &= -\mathcal{R}(\mathbf{x}^H \mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \mathbf{x}) - \mathcal{R}(\mathbf{x}^H D\mathbf{h}(\mathbf{x}^*)^T \boldsymbol{\lambda}) + \mathcal{R}(\boldsymbol{\lambda}^H D\mathbf{h}(\mathbf{x}^*) \mathbf{x}) \\ &= -\mathcal{R}(\mathbf{x}^H \mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \mathbf{x}).\end{aligned}\quad (7.73)$$

By the assumption that $\mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) > 0$, we know that $\mathcal{R}(\mathbf{x}^H \mathbf{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \mathbf{x}) > 0$ if $\mathbf{x} \neq \mathbf{0}$. Therefore, comparing the two equations above, we deduce that $\mathcal{R}(\lambda) < 0$, as required, provided that \mathbf{x} is nonzero, as we now demonstrate.

Now, suppose that $\mathbf{x} = \mathbf{0}$. Because \mathbf{w} is an eigenvector of \mathbf{M} , we have $\mathbf{M}\mathbf{w} = \lambda\mathbf{w}$. Extracting the first n components, we have $D\mathbf{h}(\mathbf{x}^*)^T \boldsymbol{\lambda} = \mathbf{0}$. By the regularity assumption, we deduce that $\boldsymbol{\lambda} = \mathbf{0}$. This contradicts the assumption that $\mathbf{w} \neq \mathbf{0}$. Hence we conclude that $\mathbf{x} \neq \mathbf{0}$, which completes the proof of our claim that for sufficiently small $\alpha > 0$, $\|D\mathbf{U}(\mathbf{w}^*)\| < 1$.

The result of the foregoing claim allows us to pick constants $\eta > 0$ and $\kappa < 1$ such that for all $\mathbf{w} = [\mathbf{x}^T, \boldsymbol{\lambda}^T]^T$ satisfying $\|\mathbf{w} - \mathbf{w}^*\| < \eta$, we have $\|\mathbf{G}(\mathbf{w})\| < \kappa$ (this follows from the continuity of $D\mathbf{U}$ and norms).

To complete the proof, suppose that $\|\mathbf{w}^{(0)} - \mathbf{w}^*\| \leq \eta$. We will show by induction that for all $k \geq 0$, $\|\mathbf{w}^{(k)} - \mathbf{w}^*\| < \eta$ and $\|\mathbf{w}^{(k+1)} - \mathbf{w}^*\| \leq \kappa \|\mathbf{w}^{(k)} - \mathbf{w}^*\|$, from which we conclude that $\mathbf{w}^{(k)}$ converges to \mathbf{w}^* with at least linear order of convergence. For $k = 0$, the result follows because $\|\mathbf{w}^{(0)} - \mathbf{w}^*\| \leq \eta$ by assumption, and

$$\|\mathbf{w}^{(1)} - \mathbf{w}^*\| \leq \|\mathbf{G}(\mathbf{w}^{(0)})\| \|\mathbf{w}^{(0)} - \mathbf{w}^*\| \leq \kappa \|\mathbf{w}^{(0)} - \mathbf{w}^*\|, \quad (7.74)$$

which follows from $\|\mathbf{w}^{(0)} - \mathbf{w}^*\| \leq \eta$. So suppose that the result holds for k . This implies that $\|\mathbf{G}(\mathbf{w}^{(k)})\| \leq \kappa$. To show the $k+1$ case, we write

$$\|\mathbf{w}^{(k+1)} - \mathbf{w}^{(*)}\| \leq \|\mathbf{G}(\mathbf{w}^{(k)})\| \|\mathbf{w}^{(k)} - \mathbf{w}^*\| \leq \kappa \|\mathbf{w}^{(k)} - \mathbf{w}^*\| \leq \eta. \quad (7.75)$$

This means that $\|\mathbf{G}(\mathbf{w}^{(k+1)})\| \leq \kappa$, from which we can write

$$\|\mathbf{w}^{(k+2)} - \mathbf{w}^{(*)}\| \leq \|\mathbf{G}(\mathbf{w}^{(k+1)})\| \|\mathbf{w}^{(k+1)} - \mathbf{w}^*\| \leq \kappa \|\mathbf{w}^{(k+1)} - \mathbf{w}^*\|. \quad (7.76)$$

This completes the proof. \square

7.1.6.3 Lagrangian Algorithm for Inequality Constraints

Consider the following optimization problem with inequality constraints:

$$\begin{aligned}\min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{g}(\mathbf{x}) \leq \mathbf{0},\end{aligned}$$

where $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{M}^m$. Recall that for this problem the Lagrangian function is given by

$$l(\mathbf{x}, \boldsymbol{\mu}) = f(\mathbf{x}) + \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x}). \quad (7.77)$$

As before, assume that $f, g \in \mathcal{C}^2$; as usual, denote the Hessian of the Lagrangian by $\mathbf{L}(\mathbf{x}, \boldsymbol{\mu})$.

The Lagrangian algorithm for this problem is given by

$$\begin{aligned}\mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} - \alpha_k (\nabla f(\mathbf{x}^{(k)}) + Dg(\mathbf{x}^{(k)})^T \boldsymbol{\mu}^{(k)}), \\ \boldsymbol{\mu}^{(k+1)} &= [\mathbf{u}^{(k)} + \beta_k \mathbf{g}(\mathbf{x}^{(k)})]_+, \end{aligned}$$

where $[\cdot]_+ = \max\{\cdot, 0\}$ (applied componentwise). Notice that, as before, the update equation for $\mathbf{x}^{(k)}$ is a gradient algorithm for minimizing the Lagrangian with respect to its \mathbf{x} argument. The update equation for $\boldsymbol{\mu}^{(k)}$ is a projected gradient algorithm for maximizing the Lagrangian with respect to its $\boldsymbol{\mu}$ argument. The reason for the projection is that the KKT multiplier vector is required to be nonnegative to satisfy the KKT condition.

The following lemma establishes that if the algorithm converges, the limit must satisfy the KKT condition. As before, we use the notion of a fixed point to state the result formally. The proof is omitted because the result follows easily by inspection.

Lemma 639. *For the Lagrangian algorithm for updating $\mathbf{x}^{(k)}$ and $\boldsymbol{\mu}^{(k)}$, the pair $(\mathbf{x}^*, \boldsymbol{\mu}^*)$ is a fixed point if and only if it satisfies the KKT condition.*

As before, we use the notation $(\mathbf{x}^*, \boldsymbol{\mu}^*)$ to denote a pair satisfying the KKT condition. Assume that $\mathbf{L}(\mathbf{x}^*, \boldsymbol{\mu}^*) \succ \mathbf{0}$. Also assume that \mathbf{x}^* is a regular point. With these assumptions, we are now ready to state and prove that the algorithm is locally convergent. As before, we will take α_k and β_k to be fixed constants (not depending on k), denoted α and β , respectively. Our analysis examines the behavior of the algorithm in two phases. In the first phase, the “nonactive” multipliers decrease to zero in finite time and remain at zero thereafter. In the second phase, the $\mathbf{x}^{(k)}$ iterates and the “active” multipliers converge jointly to their respective solutions, with at least a linear order of convergence.

Theorem 640. *For the Lagrangian algorithm for updating $\mathbf{x}^{(k)}$ and $\boldsymbol{\mu}^{(k)}$ provided that α and β are sufficiently small, there is a neighborhood of $(\mathbf{x}^*, \boldsymbol{\mu}^*)$ such that if the pair $(\mathbf{x}^{(0)}, \boldsymbol{\mu}^{(0)})$ is in this neighborhood, then (1) the nonactive multipliers reduce to zero in finite time and remain at zero thereafter and (2) the algorithm converges to $(\mathbf{x}^*, \boldsymbol{\mu}^*)$ with at least a linear order of convergence.*

Proof. As in the proof of Theorem 638, we can rescale \mathbf{x} and $\boldsymbol{\mu}$ by appropriate constants (so that the assumptions are preserved) and effectively change the relative values of the step sizes for the update equations. Therefore, without loss of generality, we can take $\beta = \alpha$.

We set up our proof using the same vector notation as before. Given a pair $(\mathbf{x}, \boldsymbol{\mu})$, let $\mathbf{w} = [\mathbf{x}^T, \boldsymbol{\mu}^T]^T$ be the $(n+p)$ -vector constructed by concatenating \mathbf{x} and $\boldsymbol{\mu}$. Similarly define $\mathbf{w}^{(k)} = [\mathbf{x}^{(k)T}, \boldsymbol{\mu}^{(k)T}]^T$ and $\mathbf{w}^* = [\mathbf{x}^{*T}, \boldsymbol{\mu}^{*T}]^T$. Define the map \mathbf{U} as

$$\mathbf{U}(\mathbf{w}) = \begin{bmatrix} \mathbf{x} - \alpha(\nabla f(\mathbf{x}) + D\mathbf{g}(\mathbf{x})^T \boldsymbol{\mu}) \\ \boldsymbol{\mu} + \alpha\mathbf{g}(\mathbf{x}) \end{bmatrix}. \quad (7.78)$$

Also, define the map $\mathbf{\Pi}$ by

$$\mathbf{\Pi}[\mathbf{w}] = \begin{bmatrix} \mathbf{x} \\ [\boldsymbol{\mu}_+] \end{bmatrix}. \quad (7.79)$$

Then, the update equations can be rewritten as

$$\mathbf{w}^{(k+1)} = \mathbf{\Pi}[\mathbf{U}(\mathbf{w}^{(k)})]. \quad (7.80)$$

Because $\mathbf{\Pi}$ is a projection onto the convex set $\{\mathbf{w} = [\mathbf{x}^T, \boldsymbol{\mu}^T] : \boldsymbol{\mu} \geq 0\}$, it is a nonexpansive map (see [12, Proposition 3.2]), which means that $\|\mathbf{\Pi}[\mathbf{v}] - \mathbf{\Pi}[\mathbf{w}]\| \leq \|\mathbf{v} - \mathbf{w}\|$.

We now write $\|\mathbf{w}^{(k+1)} - \mathbf{w}^*\|$ in terms of $\|\mathbf{w}^{(k)} - \mathbf{w}^*\|$, where $\|\cdot\|$ denotes the usual Euclidean norm. By Lemma 639, $\mathbf{w}^* = [\mathbf{x}^{*T}, \boldsymbol{\mu}^{*T}]^T$ is a fixed point of $\mathbf{w}^{(k+1)} = \mathbf{U}(\mathbf{w}^{(k)})$. Therefore,

$$\begin{aligned} \|\mathbf{w}^{(k+1)} - \mathbf{w}^*\| &= \|\mathbf{\Pi}[\mathbf{U}(\mathbf{w}^{(k)})] - \mathbf{\Pi}[\mathbf{U}(\mathbf{w}^*)]\| \\ &\leq \|\mathbf{U}(\mathbf{w}^{(k)}) - \mathbf{U}(\mathbf{w}^*)\| \end{aligned} \quad (7.81)$$

by the nonexpansiveness of $\mathbf{\Pi}$. Let $D\mathbf{U}$ be the (matrix) derivative of \mathbf{U} :

$$D\mathbf{U}(\mathbf{w}) = \mathbf{I} + \alpha \begin{bmatrix} -\mathbf{L}(\mathbf{x}, \boldsymbol{\mu}) & -D\mathbf{g}(\mathbf{x})^T \\ D\mathbf{g}(\mathbf{x}) & \mathbf{0} \end{bmatrix}. \quad (7.82)$$

By the mean value theorem,

$$\mathbf{U}(\mathbf{w}^{(k)}) - \mathbf{U}(\mathbf{w}^*) \leq \mathbf{G}(\mathbf{w}^{(k)})(\mathbf{w}^{(k)} - \mathbf{w}^*), \quad (7.83)$$

where $\mathbf{G}(\mathbf{w}^{(k)})$ is a matrix whose rows are the rows of $D\mathbf{U}$ evaluated at points that lie on the line segment joining $\mathbf{w}^{(k)}$ and \mathbf{w}^* (these points may differ from row to row). Taking norms of both sides of the equation above yields

$$\|\mathbf{U}(\mathbf{w}^{(k)}) - \mathbf{U}(\mathbf{w}^*)\| \leq \|\mathbf{G}(\mathbf{w}^{(k)})\| \|(\mathbf{w}^{(k)} - \mathbf{w}^*)\|, \quad (7.84)$$

Let \mathbf{g}_A represent those rows of \mathbf{g} corresponding to active constraints (at \mathbf{x}^*) and $\mathbf{g}_{\bar{A}}$ represent the remaining rows of \mathbf{g} . [Recall that by regularity, $D\mathbf{g}_A(\mathbf{x}^*)$ has full rank.] Given a vector $\boldsymbol{\mu}$, we divide it into two subvectors $\boldsymbol{\mu}_A$ and $\boldsymbol{\mu}_{\bar{A}}$, according to active and nonactive components, respectively. (Note that $\boldsymbol{\mu}_{\bar{A}}^* = \mathbf{0}$, the zero vector.) Next, write $\mathbf{w}_A = [\mathbf{x}^T, \boldsymbol{\mu}_A^T]^T$ and

$$\mathbf{U}_A(\mathbf{w}_A) = \begin{bmatrix} \mathbf{x} - \alpha(\nabla f(\mathbf{x}) + D\mathbf{g}_A(\mathbf{x})^T \boldsymbol{\mu}_A) \\ \boldsymbol{\mu}_A + \alpha\mathbf{g}_A(\mathbf{x}) \end{bmatrix}. \quad (7.85)$$

so that

$$D\mathbf{U}_A(\mathbf{w}_A) = \mathbf{I} + \alpha \begin{bmatrix} -\mathbf{L}(\mathbf{x}, \boldsymbol{\mu}_A) & -D\mathbf{g}_A(\mathbf{x})^T \\ D\mathbf{g}_A(\mathbf{x}) & \mathbf{0} \end{bmatrix}. \quad (7.86)$$

Finally, let \mathbf{G}_A be such that $\mathbf{U}_A(\mathbf{w}_A^{(k)}) - \mathbf{U}_A(\mathbf{w}_A^*) = \mathbf{G}_A(\mathbf{w}_A^{(k)})(\mathbf{w}_A^{(k)} - \mathbf{w}_A^*)$ (by the mean value theorem as before).

We organize the remainder of our proof into four claims.

Proposition 641. *For sufficiently small $\alpha > 0$, $\|D\mathbf{U}_A(\mathbf{w}_A^*)\| < 1$.*

The argument here parallels that of the proof of Theorem 638. So for the sake of brevity we omit the details.

The result of claim 1 allows us to pick constants $\eta > 0, \delta > 0$, and $\kappa_A < 1$ such that for all $\mathbf{w} = [\mathbf{x}^T, \boldsymbol{\mu}^T]^T$ satisfying $\|\mathbf{w} - \mathbf{w}^*\| \leq \eta$, $\|\mathbf{G}_A(\mathbf{w}_A)\| \leq \kappa_A$, and $\mathbf{g}_{\bar{A}}(\mathbf{x}) \leq -\delta\mathbf{e}$, where \mathbf{e} is the vector with all components equal to 1. The first inequality follows from claim 1 and the continuity of $D\mathbf{U}_A(\cdot)$ and $\|\cdot\|$. The second follows from the fact that the components of $\mathbf{g}_{\bar{A}}(\mathbf{x}^*)$ are negative.

Let $\kappa = \max\{\|\mathbf{G}(\mathbf{w})\| : \|\mathbf{w} - \mathbf{w}^*\| \leq \eta\}$, which we assume to be at least 1; otherwise, set $\kappa = 1$. Now pick $\epsilon > 0$ such that $\epsilon\kappa^{\epsilon/(\alpha\delta)} \leq \eta$. We can do this because the left side of this inequality goes to 0 as $\epsilon \rightarrow 0$. Assume for convenience that $k_0 = \epsilon/(\alpha\delta)$ is an integer; otherwise, replace all instances of $\epsilon/(\alpha\delta)$ by the smallest integer that exceeds it (i.e., round it up to the closest integer).

For the remainder of this proof, let $\mathbf{w}^{(0)}$ satisfy $\|\mathbf{w}^{(0)} - \mathbf{w}^*\| \leq \epsilon$.

Proposition 642. *For $k = 0, \dots, k_0$, $\|\mathbf{w}^{(k)} - \mathbf{w}^*\| \leq \eta$.*

To prove the claim, we show by induction that $\|\mathbf{w}^{(k)} - \mathbf{w}^*\| < \epsilon\kappa^k$ (which is bounded above by η provided that $k \leq k_0$). For $k = 0$, by assumption $\|\mathbf{w}^{(0)} - \mathbf{w}^*\| \leq \epsilon = \epsilon\kappa^0$, as required. For the inductive step, suppose that $\|\mathbf{w}^{(k)} - \mathbf{w}^*\| \leq \epsilon\kappa^k$ for $k < k_0$. Now, using $\|\mathbf{w}^{(k+1)} - \mathbf{w}^*\| \leq \|\mathbf{G}(\mathbf{w}^{(k)})\|\|\mathbf{w}^{(k)} - \mathbf{w}^*\|$ and the fact that $\|\mathbf{w}^{(k)} - \mathbf{w}^*\| \leq \eta$,

$$\|\mathbf{U}(\mathbf{w}^{(k+1)}) - \mathbf{U}(\mathbf{w}^*)\| \leq \|\mathbf{G}(\mathbf{w}^{(k)})\|\|(\mathbf{w}^{(k)} - \mathbf{w}^*)\| \leq \kappa(\epsilon\kappa^k) = \epsilon\kappa^{k+1}, \quad (7.87)$$

and the result now follows by induction.

Proposition 643. For $k = 0, \dots, k_0$, $\boldsymbol{\mu}_{\bar{A}}^{(k)}$ is monotonically nonincreasing in k , and $\boldsymbol{\mu}_{\bar{A}}^{(k_0)} = \mathbf{0}$ (which is equal to $\boldsymbol{\mu}_{\bar{A}}^*$).

By claim 2, $\mathbf{g}_{\bar{A}}(\mathbf{x}^{(k)}) \leq -\delta \mathbf{e}$ for all $k = 0, \dots, k_0$. Hence, for $k < k_0$,

$$\begin{aligned}\boldsymbol{\mu}_{\bar{A}}^{(k+1)} &= [\mathbf{u}_{\bar{A}}^{(k)} + \alpha \mathbf{g}_{\bar{A}}(\mathbf{x}^{(k)})]_+ \\ &\leq [\boldsymbol{\mu}_{\bar{A}}^{(k)} - \alpha \delta \mathbf{e}]_+ \\ &\leq \boldsymbol{\mu}_{\bar{A}}^{(k)}\end{aligned}\tag{7.88}$$

which establishes nonincreasing monotonicity.

To show that $\boldsymbol{\mu}_{\bar{A}}^{(k_0)}$, suppose that for some nonactive component l , we have $\boldsymbol{\mu}_l^{(k_0)} > 0$. By the monotonicity above, $\boldsymbol{\mu}_l^{(k)} > 0$ for $k = 0, \dots, k_0$. Hence,

$$\begin{aligned}\boldsymbol{\mu}_l^{(k_0)} &= \boldsymbol{\mu}_l^{(k_0-1)} + \alpha g_l(\mathbf{x}^{(k_0-1)}) \\ &= \boldsymbol{\mu}_l^{(0)} + \sum_{k=0}^{k_0-1} \alpha g_l(\mathbf{x}^{(k)})\end{aligned}\tag{7.89}$$

But by claim 2, $g_l(\mathbf{x}^{(k)}) \leq -\delta$ for all $k = 0, \dots, k_0 - 1$. Hence, $\boldsymbol{\mu}_l^{(k_0)} \leq \epsilon - k_0 \alpha \delta \leq 0$, which is a contradiction.

Finally, we will state and prove claim 4, which completes the proof of the theorem.

Proposition 644. For $k \geq k_0$, we have $\boldsymbol{\mu}_{\bar{A}}^{(k)} = \mathbf{0} = \boldsymbol{\mu}_{\bar{A}}^*$, $\|\mathbf{w}_A^{(k+1)} - \mathbf{w}_A^*\| \leq \kappa_A \|\mathbf{w}_A^{(k)} - \mathbf{w}_A^*\|$, and $\|\mathbf{w}^{(k)} - \mathbf{w}^*\| \leq \eta$.

We use induction. For $k = k_0$, we have $\|\mathbf{w}(k_0) - \mathbf{w}^*\| \leq \eta$ by claim 2, $\boldsymbol{\mu}_{\bar{A}}^{(k_0)} = \mathbf{0}$ by claim 3. Hence,

$$\mathbf{w}_A^{(k_0+1)} = \mathbf{\Pi}[\mathbf{U}_A(\mathbf{w}_A^{(k_0)}) + \alpha D \mathbf{g}_{\bar{A}}(\mathbf{x}^{(k_0)})^T \boldsymbol{\mu}_{\bar{A}}^{(k_0)}] = \mathbf{\Pi}[\mathbf{U}_A(\mathbf{w}_A^{(k_0)})].\tag{7.90}$$

Because $\boldsymbol{\mu}_{\bar{A}}^* = \mathbf{0}$, it is, similarly, also true that $\mathbf{w}_A^* = \mathbf{\Pi}[\mathbf{U}_A(\mathbf{w}_A^*)]$. Thus,

$$\begin{aligned}\|\mathbf{w}_A^{(k_0+1)} - \mathbf{w}_A^*\| &= \|\mathbf{\Pi}[\mathbf{U}_A(\mathbf{w}_A^{(k_0)})] - \mathbf{\Pi}[\mathbf{U}_A(\mathbf{w}_A^*)]\| \\ &\leq \|\mathbf{U}_A(\mathbf{w}_A^{(k_0)}) - \mathbf{U}_A(\mathbf{w}_A^*)\| \\ &\leq \|\mathbf{G}_A(\mathbf{w}_A^{(k_0)})\| \|\mathbf{w}_A^{(k_0)} - \mathbf{w}_A^*\|,\end{aligned}\tag{7.91}$$

where $\|\mathbf{G}_A(\mathbf{w}_A^{(k_0)})\| \leq \kappa_A$ because $\|\mathbf{w}(k_0) - \mathbf{w}^*\| \leq \eta$. Hence, $\|\mathbf{w}_A^{(k_0+1)} - \mathbf{w}_A^*\| \leq \kappa_A \|\mathbf{w}_A^{(k_0)} - \mathbf{w}_A^*\|$, as required.

For the inductive step, suppose that the result holds for $k \geq k_0$. Now, $\mathbf{g}_{\bar{A}}(\mathbf{x}^{(k)}) \leq -\delta \mathbf{e}$ and

$$\boldsymbol{\mu}_{\bar{A}}^{(k+1)} = [\boldsymbol{\mu}_{\bar{A}}^{(k)} + \alpha \mathbf{g}_{\bar{A}}(\mathbf{x}^{(k)})]_+ \leq [\mathbf{0} - \alpha \delta \mathbf{e}]_+ \mathbf{0},\tag{7.92}$$

which implies that $\boldsymbol{\mu}_{\bar{A}}^{(k+1)} = \mathbf{0}$. It follows that

$$\begin{aligned}\mathbf{w}_A^{(k+2)} &= \mathbf{\Pi}[\mathbf{U}_A(\mathbf{w}_A^{(k+1)}) + \alpha D \mathbf{g}_{\bar{A}}(\mathbf{x}^{(k+1)})^T \boldsymbol{\mu}_{\bar{A}}^{(k+1)}] \\ &= \mathbf{\Pi}[\mathbf{U}_A(\mathbf{w}_A^{(k+1)})],\end{aligned}\tag{7.93}$$

and now using the same argument as in the case of $k = k_0$ above we get $\|\mathbf{w}_A^{(k+2)} - \mathbf{w}_A^*\| \leq \kappa_A \|\mathbf{w}_A^{(k+1)} - \mathbf{w}_A^*\|$. Finally,

$$\|\mathbf{w}^{(k+1)} - \mathbf{w}^*\| = \|\mathbf{w}_A^{(k+1)} - \mathbf{w}_A^*\| \leq \kappa_A \|\mathbf{w}_A^{(k)} - \mathbf{w}_A^*\| \leq \eta.\tag{7.94}$$

Because $\kappa_A < 1$, claim 4 implies that $\mathbf{w}^{(k)}$ converges to \mathbf{w}^* , with at least a linear order of convergence. \square

An application of Lagrangian algorithms to a problem in decentralized rate control for sensor networks appears in [28, 29, 100]. The proof above is based on [29].

7.1.7 Penalty Methods

Consider a general constrained optimization problem

$$\begin{aligned}\min f(\mathbf{x}) \\ s.t. \mathbf{x} \in \Omega.\end{aligned}\tag{7.95}$$

We now discuss a method for solving this problem using techniques from unconstrained optimization. Specifically, we approximate the constrained optimization problem above by the unconstrained optimization problem

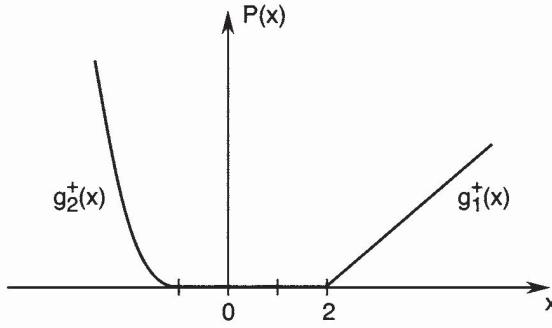
$$\min f(\mathbf{x}) + \gamma P(\mathbf{x}),\tag{7.96}$$

where $\gamma \in \mathbb{R}$ is a positive constant and $P : \mathbb{R}^n \rightarrow \mathbb{R}$ is a given function. We then solve the associated unconstrained optimization problem and use the solution as an approximation to the minimizer of the original problem. The constant γ is called the penalty parameter, and the function P is called the penalty function. Formally, we define a penalty function as follows

Definition 645. A function $P : \mathbb{R}^n \rightarrow \mathbb{R}$ is called a *penalty function* for the constrained optimization problem above if it satisfies the following three conditions:

1. P is continuous.

2. $P(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in \mathbb{R}^n$.


 图 7.4: g^+ for Example 646.

3. $P(\mathbf{x}) = 0$ if and only if \mathbf{x} is feasible (i.e., $\mathbf{x} \in \Omega$).

Clearly, for the unconstrained problem above to be a good approximation to the original problem, the penalty function P must be chosen appropriately. The role of the penalty function is to “penalize” points that are outside the feasible set.

To illustrate how we choose penalty functions, consider a constrained optimization problem of the form

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g_i(\mathbf{x}) \leq 0, i = 1, \dots, p, \end{aligned}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}, g_i : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, \dots, p$. Considering only inequality constraints is not restrictive, because an equality constraint of the form $\mathbf{h}(\mathbf{x}) = \mathbf{0}$ is equivalent to the inequality constraint $\|\mathbf{h}(\mathbf{x})\|^2 \leq 0$ (however, see Exercise 625 for a caveat). For the constrained problem above, it is natural that the penalty function be defined in terms of the constraint functions g_1, \dots, g_p . A possible choice for P is

$$P(\mathbf{x}) = \sum_{i=1}^p g_i^+(\mathbf{x}), \quad (7.97)$$

where

$$g_i^+(\mathbf{x}) = \max\{0, g_i(\mathbf{x})\} = \begin{cases} 0, & g_i(\mathbf{x}) \leq 0 \\ g_i(\mathbf{x}), & g_i(\mathbf{x}) > 0. \end{cases} \quad (7.98)$$

We refer to this penalty function as the absolute value penalty function, because it is equal to $\sum |g_i(\mathbf{x})|$, where the summation is taken over all constraints that are violated at \mathbf{x} . We illustrate this penalty function in the following example.

Example 646. Let $g_1, g_2 : \mathbb{R} \rightarrow \mathbb{R}$ be defined by $g_1(x) = x - 2, g_2(x) = -(x + 1)^3$. The feasible set defined by $\{\mathbf{x} \in \mathbb{R} : g_1(\mathbf{x}) \leq 0, g_2(\mathbf{x}) \leq 0\}$ is simply the interval $[-1, 2]$. In

this example, we have

$$g_1^+(x) = \max\{0, g_1(x)\} = \begin{cases} 0, & x \leq 2 \\ x - 2, & \text{otherwise,} \end{cases} \quad (7.99)$$

$$g_2^+(x) = \max\{0, g_2(x)\} = \begin{cases} 0, & x \geq -1 \\ -(x + 1)^3, & \text{otherwise,} \end{cases} \quad (7.100)$$

and

$$P(x) = g_1^+(x) + g_2^+(x) = \begin{cases} x - 2, & x > 2 \\ 0, & -1 \leq x \leq 2 \\ -(x + 1)^3, & x < -1. \end{cases} \quad (7.101)$$

Figure 7.4 provides a graphical illustration of g^+ for this example.

The absolute value penalty function may not be differentiable at points \mathbf{x} where $g_i(\mathbf{x}) = 0$, as is the case at the point $x = 2$ in Example 646 (notice, though, that in Example 646, P is differentiable at $x = -1$). Therefore, in such cases we cannot use techniques for optimization that involve derivatives. A form of the penalty function that is guaranteed to be differentiable is the *Courant-Beltrami penalty function*, given by

$$P(\mathbf{x}) = \sum_{i=1}^p (g_i^+(\mathbf{x}))^2. \quad (7.102)$$

In the following discussion we do not assume any particular form of the penalty function P . We only assume that P satisfies conditions 1 to 3 given in Definition 645.

The penalty function method for solving constrained optimization problems involves constructing and solving an associated unconstrained optimization problem and using the solution to the unconstrained problem as the solution to the original constrained problem. Of course, the solution to the unconstrained problem (the approximated solution) may not be exactly equal to the solution to the constrained problem (the true solution). Whether or not the solution to the unconstrained problem is a good approximation to the true solution depends on the penalty parameter γ and the penalty function P . We would expect that the larger the value of the penalty parameter γ , the closer the approximated solution will be to the true solution, because points that violate the constraints are penalized more heavily. Ideally, in the limit as $\gamma \rightarrow \infty$, the penalty method should yield the true solution to the constrained problem. In the remainder of this section, we analyze this property of the penalty function method.

Example 647. Consider the problem

$$\begin{aligned} \min \quad & \mathbf{x}^T \mathbf{Q} \mathbf{x} \\ \text{s.t.} \quad & \|\mathbf{x}\|^2 = 1, \end{aligned}$$

where $\mathbf{Q} = \mathbf{Q}^T > 0$.

- a. Using the penalty function $P(\mathbf{x}) = (\|\mathbf{x}\|^2 - 1)^2$ and penalty parameter γ , write down an unconstrained optimization problem whose solution \mathbf{x}_γ approximates the solution to this problem.
- b. Show that for any γ , \mathbf{x}_γ is an eigenvector of \mathbf{Q} .
- c. Show that $\|\mathbf{x}_\gamma\|^2 - 1 = O(1/\gamma)$ as $\gamma \rightarrow \infty$.

Solution. a. The unconstrained problem based on the given penalty function is

$$\min \mathbf{x}^T \mathbf{Q} \mathbf{x} + \gamma(\|\mathbf{x}\|^2 - 1)^2. \quad (7.103)$$

b. By the FONC, \mathbf{x}_γ satisfies

$$2\mathbf{Q}\mathbf{x}_\gamma + 4\gamma(\|\mathbf{x}_\gamma\|^2 - 1)\mathbf{x}_\gamma = 0. \quad (7.104)$$

Rearranging, we obtain

$$\mathbf{Q}\mathbf{x}_\gamma = 2\gamma(1 - \|\mathbf{x}_\gamma\|^2)\mathbf{x}_\gamma = \lambda_\gamma \mathbf{x}_\gamma, \quad (7.105)$$

where λ_γ is a scalar. Hence, \mathbf{x}_γ is an eigenvector of \mathbf{Q} . (This agrees with Example 563.)

- c. Now, $\lambda_\gamma = 2\gamma(1 - \|\mathbf{x}_\gamma\|^2) \leq \lambda_{max}$, where λ_{max} is the largest eigenvalue of \mathbf{Q} . Hence, $0 \leq 1 - \|\mathbf{x}_\gamma\|^2 \leq \lambda_{max}/(2\gamma)$ and thus $\|\mathbf{x}_\gamma\|^2 - 1 = O(1/\gamma)$ as $\gamma \rightarrow \infty$.

□

We now analyze the penalty method in a more general setting. In our analysis, we adopt the following notation. Denote by \mathbf{x}^* a solution (global minimizer) to the problem. Let P be a penalty function for the problem. For each $k = 1, 2, \dots$, let $\gamma_k \in \mathbb{R}$ be a given positive constant. Define an associated function $q(\gamma_k, \cdot) : \mathbb{R}^n \rightarrow \mathbb{R}$ by

$$q(\gamma_k, \mathbf{x}) = f(\mathbf{x}) + \gamma_k P(\mathbf{x}). \quad (7.106)$$

For each k , we can write the following associated unconstrained optimization problem:

$$\min q(\gamma_k, \mathbf{x}). \quad (7.107)$$

Denote by $\mathbf{x}^{(k)}$ a minimizer of $q(\gamma_k, \mathbf{x})$. The following technical lemma describes certain useful relationships between the constrained problem and the associated unconstrained problems.

Lemma 648. Suppose that $\{\gamma_k\}$ is a nondecreasing sequence; that is, for each k , we have $\gamma_k < \gamma_{k+1}$. Then, for each k we have

1. $q(\gamma_{k+1}, \mathbf{x}^{(k+1)}) \geq q(\gamma_k, \mathbf{x}^{(k)})$.
2. $P(\mathbf{x}^{(k+1)}) \leq P(\mathbf{x}^{(k)})$.
3. $f(\mathbf{x}^{(k+1)}) \geq f(\mathbf{x}^{(k)})$.
4. $f(\mathbf{x}^*) \geq q(\gamma_k, \mathbf{x}^{(k)}) \geq f(\mathbf{x}^{(k)})$.

Proof. We first prove part 1. From the definition of q and the fact that $\{\gamma_k\}$ is an increasing sequence, we have

$$q(\gamma_{k+1}, \mathbf{x}^{(k+1)}) = f(\mathbf{x}^{(k+1)}) + \gamma_{k+1}P(\mathbf{x}^{(k+1)}) \geq f(\mathbf{x}^{(k+1)}) + \gamma_kP(\mathbf{x}^{(k+1)}) \quad (7.108)$$

Now, because $\mathbf{x}^{(k)}$ is a minimizer of $q(\gamma_k, \mathbf{x})$,

$$q(\gamma_k, \mathbf{x}^{(k)}) = f(\mathbf{x}^{(k)}) + \gamma_kP(\mathbf{x}^{(k)}) \leq f(\mathbf{x}^{(k+1)}) + \gamma_kP(\mathbf{x}^{(k+1)}) \quad (7.109)$$

Combining the above, we get part 1.

We next prove part 2. Because $\mathbf{x}^{(k)}$ and $\mathbf{x}^{(k+1)}$ minimize $q(\gamma_k, \mathbf{x})$ and $q(\gamma_{k+1}, \mathbf{x})$, respectively, we can write

$$q(\gamma_k, \mathbf{x}^{(k)}) = f(\mathbf{x}^{(k)}) + \gamma_kP(\mathbf{x}^{(k)}) \geq f(\mathbf{x}^{(k+1)}) + \gamma_kP(\mathbf{x}^{(k+1)}) \quad (7.110)$$

$$q(\gamma_{k+1}, \mathbf{x}^{(k+1)}) = f(\mathbf{x}^{(k+1)}) + \gamma_{k+1}P(\mathbf{x}^{(k+1)}) \geq f(\mathbf{x}^{(k)}) + \gamma_{k+1}P(\mathbf{x}^{(k)}) \quad (7.111)$$

Adding the inequalities above yields

$$\gamma_kP(\mathbf{x}^{(k)}) + \gamma_{k+1}P(\mathbf{x}^{(k+1)}) \leq \gamma_{k+1}P(\mathbf{x}^{(k)}) + \gamma_kP(\mathbf{x}^{(k+1)}) \quad (7.112)$$

Rearranging, we get

$$(\gamma_{k+1} - \gamma_k)P(\mathbf{x}^{(k+1)}) \leq (\gamma_{k+1} - \gamma_k)P(\mathbf{x}^{(k)}) \quad (7.113)$$

We know by assumption that $\gamma_{k+1} \geq \gamma_k$. If $\gamma_{k+1} > \gamma_k$, then we get $P(\mathbf{x}^{(k+1)}) \leq P(\mathbf{x}^{(k)})$. If, on the other hand, $\gamma_{k+1} = \gamma_k$, then clearly $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}$ and so $P(\mathbf{x}^{(k+1)}) = P(\mathbf{x}^{(k)})$. Therefore, in either case, we arrive at part 2.

We now prove part 3. Because $\mathbf{x}^{(k)}$ is a minimizer of $q(\gamma_k, \mathbf{x})$, we obtain

$$q(\gamma_k, \mathbf{x}^{(k)}) = f(\mathbf{x}^{(k)}) + \gamma_kP(\mathbf{x}^{(k)}) \leq f(\mathbf{x}^{(k+1)}) + \gamma_kP(\mathbf{x}^{(k+1)}). \quad (7.114)$$

Therefore,

$$f(\mathbf{x}^{(k+1)}) \geq f(\mathbf{x}^{(k)}) + \gamma_k(P(\mathbf{x}^{(k)}) - P(\mathbf{x}^{(k+1)})) \quad (7.115)$$

From part 2 we have $P(\mathbf{x}^{(k)}) - P(\mathbf{x}^{(k+1)}) \geq 0$, and $\gamma_k > 0$ by assumption; therefore, we get

$$f(\mathbf{x}^{(k+1)}) \geq f(\mathbf{x}^{(k)}) \quad (7.116)$$

Finally, we now prove part 4. Because $\mathbf{x}^{(k)}$ is a minimizer of $q(\gamma_k, \mathbf{x})$, we get

$$f(\mathbf{x}^*) + \gamma_k P(\mathbf{x}^*) \geq q(\gamma_k, \mathbf{x}^{(k)}) = f(\mathbf{x}^{(k)}) + \gamma_k P(\mathbf{x}^{(k)}) \quad (7.117)$$

Because \mathbf{x}^* is a minimizer for the constrained optimization problem, we have $P(\mathbf{x}^*) = 0$. Therefore,

$$f(\mathbf{x}^*) \geq f(\mathbf{x}^{(k)}) + \gamma_k P(\mathbf{x}^{(k)}) \quad (7.118)$$

Because $P(\mathbf{x}^{(k)}) \geq 0$ and $\gamma_k \geq 0$,

$$f(\mathbf{x}^*) \geq q(\gamma_k, \mathbf{x}^{(k)}) \geq f(\mathbf{x}^{(k)}) \quad (7.119)$$

which completes the proof. \square

With the above lemma, we are now ready to prove the following theorem.

Theorem 649. Suppose that the objective function f is continuous and $\gamma_k \rightarrow \infty$ as $k \rightarrow \infty$. Then, the limit of any convergent subsequence of the sequence $\{\mathbf{x}^{(k)}\}$ is a solution to the constrained optimization problem.

Proof. Suppose that $\{\mathbf{x}^{(m_k)}\}$ is a convergent subsequence of the sequence $\{\mathbf{x}^{(k)}\}$. (See Section 5.1 of [30] for a discussion of sequences and subsequences.) Let $\bar{\mathbf{x}}$ be the limit of $\{\mathbf{x}^{(m_k)}\}$. By Lemma 648, the sequence $\{q(\gamma_k, \mathbf{x}^{(k)})\}$ is nondecreasing and bounded above by $f(\mathbf{x}^*)$. Therefore, the sequence $\{q(\gamma_k, \mathbf{x}^{(k)})\}$ has a limit $q^* = \lim_{k \rightarrow \infty} \{q(\gamma_k, \mathbf{x}^{(k)})\}$ such that $q^* \leq f(\mathbf{x}^*)$ (see Theorem 5.3 of [30]). Because the function f is continuous and $f(\mathbf{x}^{(m_k)}) \leq f(\mathbf{x}^*)$ by Lemma 648, we have

$$\lim_{k \rightarrow \infty} f(\mathbf{x}^{(m_k)}) = f\left(\lim_{k \rightarrow \infty} \mathbf{x}^{(m_k)}\right) = f(\bar{\mathbf{x}}) \leq f(\mathbf{x}^*). \quad (7.120)$$

Because the sequences $\{f(\mathbf{x}^{(m_k)})\}$ and $\{q(\gamma_k, \mathbf{x}^{(k)})\}$ both converge, the sequence $\{\gamma_{m_k} P(\mathbf{x}^{(m_k)})\} = \{q(\gamma_k, \mathbf{x}^{(k)}) - f(\mathbf{x}^{(m_k)})\}$ also converges, with

$$\lim_{k \rightarrow \infty} \gamma_{m_k} P(\mathbf{x}^{(m_k)}) = q^* - f(\bar{\mathbf{x}}). \quad (7.121)$$

By Lemma 648, the sequence $\{P(\mathbf{x}^{(k)})\}$ is nonincreasing and bounded from below by 0. Therefore, $\{P(\mathbf{x}^{(k)})\}$ converges (again see Theorem 14), and hence so does $\{P(\mathbf{x}^{(k)})\}$. Because $\gamma_{m_k} \rightarrow \infty$ we conclude that

$$\lim_{k \rightarrow \infty} P(\mathbf{x}^{(m_k)}) = 0. \quad (7.122)$$

By continuity of P , we have

$$0 = \lim_{k \rightarrow \infty} P(\mathbf{x}^{(m_k)}) = P\left(\lim_{k \rightarrow \infty} \mathbf{x}^{(m_k)}\right) = P(\bar{\mathbf{x}}), \quad (7.123)$$

and hence $\bar{\mathbf{x}}$ is a feasible point. Because $f(\mathbf{x}^*) \geq f(\bar{\mathbf{x}})$ from above, we conclude that $\bar{\mathbf{x}}$ must be a solution to the constrained optimization problem. \square

If we perform an infinite number of minimization runs, with the penalty parameter $\gamma_k \rightarrow \infty$, then Theorem 649 ensures that the limit of any convergent subsequence is a minimizer \mathbf{x}^* to the original constrained optimization problem. There is clearly a practical limitation in applying this theorem. It is certainly desirable to find a minimizer to the original constrained optimization problem using a single minimization run for the unconstrained problem that approximates the original problem using a penalty function. In other words, we desire an exact solution to the original constrained problem by solving the associated unconstrained problem $\min_{\mathbf{x}} f(\mathbf{x}) + \gamma P(\mathbf{x})$ with a finite $\gamma > 0$. It turns out that indeed this can be accomplished, in which case we say that the penalty function is *exact*. However, it is necessary that exact penalty functions be nondifferentiable, as shown in [11], and illustrated in the following example.

Example 650. Consider the problem

$$\begin{aligned} & \min f(x) \\ & \text{s.t. } x \in [0, 1], \end{aligned}$$

where $f(x) = 5 - 3x$. Clearly, the solution is $x^* = 1$.

Suppose that we use the penalty method to solve the problem, with a penalty function P that is differentiable at $x^* = 1$. Then, $P'(x^*) = 0$, because $P(x) = 0$ for all $x \in [0, 1]$. Hence, if we let $g = f + \gamma P$, then $g'(x^*) = f'(x^*) + \gamma P'(x^*) \neq 0$ for all finite $\gamma > 0$. Hence, $x^* = 1$ does not satisfy the first-order necessary condition to be a local minimizer of g . Thus, P is not an exact penalty function.

Here, we prove a result on the necessity of nondifferentiability of exact penalty functions for a special class of problems.

Proposition 651. Consider the problem

$$\min f(\mathbf{x}) \quad (7.124)$$

$$\text{s.t. } \mathbf{x} \in \Omega, \quad (7.125)$$

with $\Omega \subset \mathbb{R}^n$ convex. Suppose that the minimizer \mathbf{x}^* lies on the boundary of Ω and there exists a feasible direction \mathbf{d} at \mathbf{x}^* such that $\mathbf{d}^T \nabla f(\mathbf{x}^*) > 0$. If P is an exact penalty function, then P is not differentiable at \mathbf{x}^* .

Proof. We use contraposition. Suppose that P is differentiable at \mathbf{x}^* . Then, $\mathbf{d}^T \nabla P(\mathbf{x}^*) = 0$, because $P(\mathbf{x}) = 0$ for all $\mathbf{x} \in \Omega$. Hence, if we let $g = f + \gamma P$, then $\mathbf{d}^T \nabla g(\mathbf{x}^*) > 0$ for all finite $\gamma > 0$, which implies that $\nabla g(\mathbf{x}^*) \neq 0$. Hence, \mathbf{x}^* is not a local minimizer of g , and thus P is not an exact penalty function. \square

Note that the result of Proposition 651 does not hold if we remove the assumption that $\mathbf{d}^T \nabla f(\mathbf{x}^*) > 0$. Indeed, consider a convex problem where $\nabla f(\mathbf{x}^*) = \mathbf{0}$. Choose P to be differentiable. Clearly, in this case we have $\nabla g(\mathbf{x}^*) = \nabla f(\mathbf{x}^*) + \gamma \nabla P(\mathbf{x}^*) = \mathbf{0}$. The function P is therefore an exact penalty function, although differentiable.

For further reading on the subject of optimization of nondifferentiable functions, see, for example, [39]. References [12] and [102] provide further discussions on the penalty method, including nondifferentiable exact penalty functions. These references also discuss exact penalty methods involving differentiable functions; these methods go beyond the elementary type of penalty method introduced in this chapter.

7.1.8 Exercises

Exercise 652. Consider the constrained optimization problem

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \|\mathbf{x}\| = 1, \end{aligned} \tag{7.126}$$

where $f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x}$ and $\mathbf{Q} = \mathbf{Q}^T$. We wish to apply a fixed-step-size projected gradient algorithm to this problem:

$$\mathbf{x}^{(k+1)} = \mathbf{\Pi}[\mathbf{x}^{(k)} + \alpha \nabla f(\mathbf{x}^{(k)})], \tag{7.127}$$

where $\alpha > 0$ and $\mathbf{\Pi}$ is the usual projection operator defined by $\mathbf{\Pi}[\mathbf{x}] = \operatorname{argmin}_{\mathbf{x} \in \Omega} \|\mathbf{z} - \mathbf{x}\|$ and Ω is the constraint set.

a. Find a simple formula for $\mathbf{\Pi}[\mathbf{x}]$ in this problem (an explicit expression in terms of \mathbf{x}), assuming that $\mathbf{x} \neq \mathbf{0}$.

b. For the remainder of the question, suppose that

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}. \tag{7.128}$$

Find the solution(s) to this optimization problem.

c. Let $y^{(k)} = x_1^{(k)} / x_2^{(k)}$. Derive an expression for $y^{(k+1)}$ in terms of $y^{(k)}$ and α .

- d. Assuming that $x_2^{(0)} \neq 0$, use parts b and c to show that for any $\alpha > 0$, $\mathbf{x}^{(k)}$ converges to a solution to the optimization problem (i.e., the algorithm works).
- e. In part d, what if $x_2^{(0)} = 0$?

Exercise 653. Consider the constrained optimization problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \Omega, \end{aligned} \tag{7.129}$$

where $f(\mathbf{x}) = \mathbf{c}^T \mathbf{x}$ and $\mathbf{c} \in \mathbb{R}^n$ is a given nonzero vector. (Linear programming is a special case of this problem.) We wish to apply a fixed-step-size projected gradient algorithm

$$\mathbf{x}^{(k+1)} = \mathbf{\Pi}[\mathbf{x}^{(k)} - \nabla f(\mathbf{x}^{(k)})], \tag{7.130}$$

where, as usual, $\mathbf{\Pi}$ is the projection operator onto Ω (assume that for any \mathbf{y} , $\mathbf{\Pi}[\mathbf{y}] = \operatorname{argmin}_{\mathbf{x} \in \Omega} \|\mathbf{y} - \mathbf{x}\|^2$ is unique).

- a. Suppose that for some k , $\mathbf{x}^{(k)}$ is a global minimizer of the problem. Is it necessarily the case that $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}$? Explain fully.
- b. Suppose that for some k , $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}$. Is it necessarily the case that $\mathbf{x}^{(k)}$ is a local minimizer of the problem? Explain fully.

Exercise 654. Consider the constrained optimization problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \Omega, \end{aligned}$$

where $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, $f \in \mathbb{R}^1$, and $\Omega = [-1, 1]^2 = \{\mathbf{x} : -1 \leq x_i \leq 1, i = 1, 2\}$. Consider the projected steepest descent algorithm applied to this problem:

$$\mathbf{x}^{(k+1)} = \mathbf{\Pi}[\mathbf{x}^{(k)} - \alpha_k \nabla f(\mathbf{x}^{(k)})], \tag{7.131}$$

where $\mathbf{\Pi}$ represents the projection operator with respect to Ω and $\alpha_k = \operatorname{argmin}_{\alpha \geq 0} f(\mathbf{x}^{(k)} - \alpha \nabla f(\mathbf{x}^{(k)}))$. Our goal is to prove the following statement:

$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}$ if and only if $\mathbf{x}^{(k)}$ satisfies the first-order necessary condition

We will do this in two parts.

- a. Prove the statement above for the case where $\mathbf{x}^{(k)}$ is an interior point of Ω .

b. Prove the statement for the case where $\mathbf{x}^{(k)}$ is a boundary point of Ω .

Hint: Consider two further subcases: (i) $\mathbf{x}^{(k)}$ is a corner point, and (ii) $\mathbf{x}^{(k)}$ is not a corner point. For subcase (i) it suffices to take $\mathbf{x}^{(k)} = [1, 1]^T$. For subcase (ii) it suffices to take $\mathbf{x}^{(k)} \in \{\mathbf{x} : x_1 = 1, -1 < x_2 < 1\}$.

Exercise 655. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m < n$, $\text{rank } \mathbf{A} = m$, and $\mathbf{b} \in \mathbb{R}^m$. Define $\Omega = \{\mathbf{x} : \mathbf{Ax} = \mathbf{b}\}$ and let $x_0 \in \Omega$. Show that for any $\mathbf{y} \in \mathbb{R}^n$,

$$\Pi[\mathbf{x}_0 + \mathbf{y}] = \mathbf{x}_0 + \mathbf{Py}, \quad (7.132)$$

where $\mathbf{P} = \mathbf{I} - \mathbf{A}^T(\mathbf{AA}^T)^{-1}\mathbf{A}$.

Exercise 656. Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be given by $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x} - \mathbf{x}^T \mathbf{c}$, where $\mathbf{Q} = \mathbf{Q}^T > 0$. We wish to minimize f over $\{\mathbf{x} : \mathbf{Ax} = \mathbf{B}\}$, where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m < n$, and $\text{rank } \mathbf{A} = m$. Show that the projected steepest descent algorithm for this case takes the form

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \frac{\mathbf{g}^{(k)T} \mathbf{Pg}^{(k)}}{\mathbf{g}^{(k)T} \mathbf{PQ} \mathbf{Pg}^{(k)}} \mathbf{Pg}^{(k)}, \quad (7.133)$$

where

$$\mathbf{g}^{(k)} = \nabla f(\mathbf{x}^{(k)}) = \mathbf{Q} \mathbf{x}^{(k)} - \mathbf{c}, \quad (7.134)$$

and $\mathbf{P} = \mathbf{I}_n - \mathbf{A}^T(\mathbf{AA}^T)^{-1}\mathbf{A}$.

Exercise 657. Consider the problem

$$\begin{aligned} & \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x}\|^2, \\ & \text{s.t. } \mathbf{Ax} = \mathbf{b}, \end{aligned}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m < n$, and $\text{rank } \mathbf{A} = m$. Show that if $\mathbf{x}^{(0)} \in \{\mathbf{x} : \mathbf{Ax} = \mathbf{b}\}$, then the projected steepest descent algorithm converges to the solution in one step.

Exercise 658. Show that in the projected steepest descent algorithm, we have that for each k :

a. $\mathbf{g}^{(k+1)T} \mathbf{Pg}^{(k)} = 0$.

b. The vector $\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}$ is orthogonal to the vector $\mathbf{x}^{(k+2)} - \mathbf{x}^{(k+1)}$.

Exercise 659. Consider the optimization problem

$$\begin{aligned} & \min_{\mathbf{x}} f(\mathbf{x}), \\ & \text{s.t. } \mathbf{x} \in \Omega, \end{aligned}$$

where $\Omega \subset \mathbb{R}^n$. Suppose that we apply the penalty method to this problem, which involves solving an associated unconstrained optimization problem with penalty function P and penalty parameter $\gamma > 0$.

- a. Write down the unconstrained problem associated with penalty function P and penalty parameter γ .
- b. Let \mathbf{x}^* be a global minimizer of the given constrained problem, and let \mathbf{x}^γ be a global minimizer of the associated unconstrained optimization problem (in part a) with penalty parameter γ . Show that if $\mathbf{x}^\gamma \notin \Omega$, then $f(\mathbf{x}^\gamma) < f(\mathbf{x}^*)$.

Exercise 660. Use the penalty method to solve the following problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & x_1^2 + 2x_2^2 \\ \text{s.t.} \quad & x_1 + x_2 = 3. \end{aligned}$$

Exercise 661. Consider the simple optimization problem

$$\begin{aligned} \min_x \quad & x \\ \text{s.t.} \quad & x \geq a, \end{aligned}$$

where $a \in \mathbb{R}$. Suppose that we use the penalty method to solve this problem, with penalty function

$$P(x) = (\max\{a - x, 0\})^2 \tag{7.135}$$

(the Courant-Beltrami penalty function). Given a number $\epsilon > 0$, find the smallest value of the penalty parameter γ such that the solution obtained using the penalty method is no further than ϵ from the true solution to the given problem. (Think of ϵ as the desired accuracy.)

Exercise 662. Consider the problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & \frac{1}{2} \|\mathbf{x}\|_2 \\ \text{s.t.} \quad & \mathbf{Ax} = \mathbf{b}, \end{aligned}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^m$, $m \leq n$, and $\text{rank } \mathbf{A} = m$. Let \mathbf{x}^* be the solution. Suppose that we solve the problem using the penalty method, with the penalty function

$$P(\mathbf{x}) = \|\mathbf{Ax} - \mathbf{b}\|^2. \tag{7.136}$$

Let \mathbf{x}_γ^* be the solution to the associated unconstrained problem with the penalty parameter $\gamma > 0$; that is, \mathbf{x}_γ^* is the solution to

$$\min_{\mathbf{x}} \quad \frac{1}{2} \|\mathbf{x}\|_2 + \gamma \|\mathbf{Ax} - \mathbf{b}\|^2. \tag{7.137}$$

a. Suppose that

$$\mathbf{A} = [1 \quad 1], \quad \mathbf{b} = [1]. \quad (7.138)$$

Verify that \mathbf{x}_γ^* converges to the solution \mathbf{x}^* of the original constrained problem as $\gamma \rightarrow \infty$.

b. Prove that $\mathbf{x}_\gamma^* \rightarrow \mathbf{x}^*$ as $\gamma \rightarrow \infty$ holds in general.

Hint: Use the following result: There exist orthogonal matrices $\mathbf{U} \in \mathbb{R}^{m \times m}$ and $\mathbf{V}^T \in \mathbb{R}^{n \times n}$ such that

$$\mathbf{A} = \mathbf{U}[\mathbf{S}, \mathbf{O}]\mathbf{V}^T, \quad (7.139)$$

where

$$\mathbf{S} = \text{diag} \left(\sqrt{\lambda_1(\mathbf{A}\mathbf{A}^T)}, \dots, \sqrt{\lambda_m(\mathbf{A}\mathbf{A}^T)} \right) \quad (7.140)$$

is a diagonal matrix with diagonal elements that are the square roots of the eigenvalues of $\mathbf{A}\mathbf{A}^T$.

The result above is called the singular value decomposition (see, e.g., [61]).

7.2 Frank-Wolfe Algorithm

(Taken from Chapter 4.1.2 of [83])

For problems with a convex set constraint:

$$\min_{\mathbf{x}} f(\mathbf{x}), \quad \text{s.t.} \quad \mathbf{x} \in \mathcal{D}, \quad (7.141)$$

where f is convex and continuously differentiable and \mathcal{D} is a compact convex set, Frank-Wolfe-type algorithms [49, 63] suggest that the following iteration:

$$\begin{aligned} \mathbf{g}_k &= \underset{\mathbf{g} \in \mathcal{D}}{\text{argmin}} \langle \mathbf{g}, \nabla f(\mathbf{x}_k) \rangle, \\ \mathbf{x}_{k+1} &= (1 - \gamma_k)\mathbf{x}_k + \gamma_k \mathbf{g}_k, \quad \text{where } \gamma_k = \frac{2}{k+2}, \end{aligned} \quad (7.142)$$

can be used to solve (7.141). Intuitively, the procedure can be explained as follows. At a current position \mathbf{x}_k , the procedure considers the linearization of object function, moving towards the minimizer of such a linear function in the domain \mathcal{D} (see Figure 7.5). When the constraint set \mathcal{D} is a ball of bounded nuclear norm, \mathbf{g}_k can be relatively easily computed by finding the singular vectors associated to the leading singular values of $\nabla f(\mathbf{x}_k)$ [63]. Such a particular problem can also be efficiently solved by transforming it into a positive semi-definite program [64], where only the eigenvector corresponding to the largest eigenvalue of a matrix is needed.

The following theorem guarantees the convergence of the Frank-Wolfe algorithm.

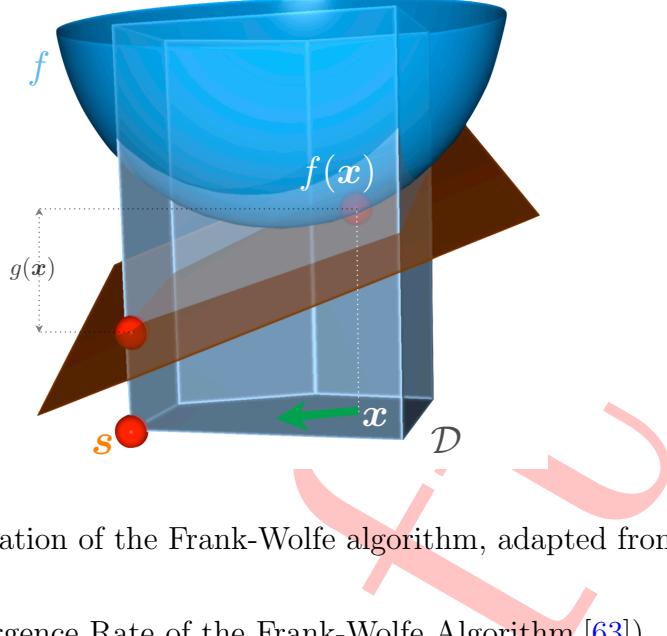


图 7.5: Illustration of the Frank-Wolfe algorithm, adapted from [63].

Theorem 663 (Convergence Rate of the Frank-Wolfe Algorithm [63]). *For each $k \geq 1$, the iterate \mathbf{x}_k in procedure (7.142) satisfies*

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{2C_f}{k+2}, \quad (7.143)$$

where $\mathbf{x}^* \in \mathcal{D}$ is an optimal solution to problem (7.141) and C_f is the curvature constant defined as

$$C_f = \sup_{\mathbf{x}, \mathbf{s} \in \mathcal{D}, \gamma \in [0, 1], \mathbf{y} = \mathbf{x} + \gamma(\mathbf{s} - \mathbf{x})} \frac{2}{\gamma^2} (f(\mathbf{y}) - f(\mathbf{x}) - \langle \mathbf{y} - \mathbf{x}, \nabla f(\mathbf{x}) \rangle). \quad (7.144)$$

The following lemma studies the improvement in each iteration, expressing the improvement in terms of current duality gap.

Lemma 664. *For an iteration $\mathbf{x}_{k+1} = \mathbf{x}_k + \gamma(\mathbf{g}_k - \mathbf{x}_k)$ with an arbitrary stepsize $\gamma \in [0, 1]$, it holds that*

$$f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k) - \gamma g(\mathbf{x}_k) + \frac{\gamma^2}{2} C_f, \quad (7.145)$$

if \mathbf{g}_k is an approximate linear minimizer, i.e., $\langle \mathbf{g}_k, \nabla f(\mathbf{x}_k) \rangle = \min_{\hat{\mathbf{g}}_k \in \mathcal{D}} \langle \hat{\mathbf{g}}_k, \nabla f(\mathbf{x}_k) \rangle$. Here $g(\mathbf{x})$ is the duality gap defined as

$$g(\mathbf{x}) = \max_{\mathbf{g} \in \mathcal{D}} \langle \mathbf{x} - \mathbf{g}, \nabla f(\mathbf{x}) \rangle. \quad (7.146)$$

Proof. By the definition of curvature constant C_f , we have

$$\begin{aligned} f(\mathbf{x}_{k+1}) &= f(\mathbf{x}_k + \gamma(\mathbf{g}_k - \mathbf{x}_k)) \\ &\leq f(\mathbf{x}_k) + \gamma \langle \mathbf{g}_k - \mathbf{x}_k, \mathbf{d}_x \rangle + \frac{\gamma^2}{2} C_f. \end{aligned} \quad (7.147)$$

Now we use the fact that the choice of \mathbf{g}_k is a good “descent direction” for the linear approximation of f at \mathbf{x}_k . Formally, we are given a point \mathbf{g}_k that satisfies $\langle \mathbf{g}_k, \mathbf{d}_x \rangle = \min_{\mathbf{y} \in \mathcal{D}} \langle \mathbf{y}, \mathbf{d}_x \rangle$. This is equivalent to

$$\langle \mathbf{g}_k - \mathbf{x}_k, \mathbf{d}_x \rangle = \min_{\mathbf{y} \in \mathcal{D}} \langle \mathbf{y}, \mathbf{d}_x \rangle - \langle \mathbf{x}_k, \mathbf{d}_x \rangle = -g(\mathbf{x}_k). \quad (7.148)$$

Here we have utilized the definition (7.146) of duality gap $g(\mathbf{x})$. So (7.145) is obtained. \square

Now we are ready to prove Theorem 663.

Proof. By Lemma 664, for each step of the Frank-Wolfe algorithm, it holds that $f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k) - \gamma g(\mathbf{x}_k) + \gamma^2 C$, where $C = \frac{C_f}{2}$. Denote by $h(\mathbf{x}) = f(\mathbf{x}) - f(\mathbf{x}^*)$ the primal error at any point \mathbf{x} . Thus

$$\begin{aligned} h(\mathbf{x}_{k+1}) &\leq h(\mathbf{x}_k) - \gamma g(\mathbf{x}_k) + \gamma^2 C \\ &\leq h(\mathbf{x}_k) - \gamma h(\mathbf{x}_k) + \gamma^2 C \\ &= (1 - \gamma)h(\mathbf{x}_k) + \gamma^2 C, \end{aligned} \quad (7.149)$$

where the second inequality holds because the linearization $f(\mathbf{x}) + \langle \mathbf{g} - \mathbf{x}, \nabla f(\mathbf{x}) \rangle$ always lies below the graph of the function f , which implies that $g(\mathbf{x}) \geq f(\mathbf{x}) - f(\mathbf{x}^*)$. We will now use induction over k to prove the desired bound, namely,

$$h(\mathbf{x}_{k+1}) \leq \frac{4C}{k+1+2}, \quad k = 0, 1, \dots \quad (7.150)$$

The base case $k = 0$ follows from (7.149) with $\gamma = \frac{2}{0+2} = 1$ by algorithm. Now assume that

$$h(\mathbf{x}_k) \leq \frac{4C}{k+2}. \quad (7.151)$$

Considering the case of $k+1$, bound (7.149) guarantees that

$$\begin{aligned} h(\mathbf{x}_{k+1}) &\leq (1 - \gamma_k)h(\mathbf{x}_k) + \gamma_k^2 C \\ &= \left(1 - \frac{2}{k+2}\right)h(\mathbf{x}_k) + \left(\frac{2}{k+2}\right)^2 C \\ &\leq \left(1 - \frac{2}{k+2}\right)\frac{4C}{k+2} + \left(\frac{2}{k+2}\right)^2 C \\ &= \frac{4C}{k+2} \left(1 - \frac{1}{k+2}\right) \\ &= \frac{4C}{k+2} \frac{k+1}{k+2} \\ &\leq \frac{4C}{k+2} \frac{k+2}{k+3} = \frac{4C}{k+3}, \end{aligned} \quad (7.152)$$

as desired. \square

7.2.1 An application of conditional gradient descent: Least-squares regression with structured sparsity

(Taken from Chapter 3.3 of [20])

This example is inspired by Lugosi [2010] (see also Jones [1992]). Consider the problem of approximating a signal $\mathbf{y} \in \mathbb{R}^n$ by a “small” combination of dictionary elements $\mathbf{d}_1, \dots, \mathbf{d}_N \in \mathbb{R}^n$. One way to do this is to consider a LASSO type problem in dimension N of the following form (with $\lambda \in \mathbb{R}$ fixed)

$$\min_{\mathbf{x} \in \mathbb{R}^N} \left\| \mathbf{y} - \sum_{i=1}^N x_i \mathbf{d}_i \right\|_2^2 + \lambda \|\mathbf{x}\|_1. \quad (7.153)$$

Let $\mathbf{D} \in \mathbb{R}^{n \times N}$ be the dictionary matrix with the i^{th} column given by \mathbf{d}_i . Instead of considering the penalized version of the problem one could look at the following constrained problem (with $s \in \mathbb{R}$ fixed) on which we will now focus, see e.g. Friedlander and Tseng [2007],

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^N} \|\mathbf{y} - \mathbf{Dx}\|_2^2 &\Leftrightarrow \min_{\mathbf{x} \in \mathbb{R}^N} \|\mathbf{y}/s - \mathbf{Dx}\|_2^2 \\ s.t. \quad \|\mathbf{x}\|_1 \leq s & s.t. \quad \|\mathbf{x}\|_1 \leq 1. \end{aligned} \quad (7.154)$$

We make some assumptions on the dictionary. We are interested in situations where the size of the dictionary N can be very large, potentially exponential in the ambient dimension n . Nonetheless we want to restrict our attention to algorithms that run in reasonable time with respect to the ambient dimension n , that is we want polynomial time algorithms in n . Of course in general this is impossible, and we need to assume that the dictionary has some structure that can be exploited. Here we make the assumption that one can do linear optimization over the dictionary in polynomial time in n . More precisely we assume that one can solve in time $p(n)$ (where p is polynomial) the following problem for any $\mathbf{y} \in \mathbb{R}^n$:

$$\min_{1 \leq i \leq N} \mathbf{y}^\top \mathbf{d}_i. \quad (7.155)$$

This assumption is met for many combinatorial dictionaries. For instance the dictionary elements could be vector of incidence of spanning trees in some fixed graph, in which case the linear optimization problem can be solved with a greedy algorithm.

Finally, for normalization issues, we assume that the l_2 -norm of the dictionary elements are controlled by some $m > 0$, that is $\|\mathbf{d}_i\|_2 \leq m, \forall i \in [N]$.

Our problem of interest (7.154) corresponds to minimizing the function $f(x) = \frac{1}{2} \|\mathbf{y} - \mathbf{Dx}\|_2^2$ on the l_1 -ball of \mathbb{R}^N in polynomial time in n . At first sight this task may seem completely impossible, indeed one is not even allowed to write down entirely a vector

$\mathbf{x} \in \mathbb{R}^N$ (since this would take time linear in N). The key property that will save us is that this function admits sparse minimizers as we discussed in the previous section, and this will be exploited by the conditional gradient descent method.

First let us study the computational complexity of the t^{th} step of conditional gradient descent. Observe that

$$\nabla f(\mathbf{x}) = \mathbf{D}^\top (\mathbf{D}\mathbf{x} - \mathbf{y}). \quad (7.156)$$

Now assume that $\mathbf{z}_t = \mathbf{D}\mathbf{x}_t - \mathbf{y} \in \mathbb{R}^n$ is already computed, then to compute $(\mathbf{u}_t \in \operatorname{argmin}_{\mathbf{u} \in \mathcal{X}} \nabla f(\mathbf{x}_t)^\top \mathbf{u})$ one needs to find the coordinate $i_t \in [N]$ that maximizes $|[\nabla f(\mathbf{x}_t)](i)|$ which can be done by maximizing $\mathbf{d}_i^\top \mathbf{z}_t$ and $-\mathbf{d}_i^\top \mathbf{z}_t$. Thus $\mathbf{u}_t \in \operatorname{argmin}_{\mathbf{u} \in \mathcal{X}} \nabla f(\mathbf{x}_t)^\top \mathbf{u}$ takes time $\mathcal{O}(p(n))$. Computing \mathbf{x}_{t+1} from \mathbf{x}_t and i_t takes time $\mathcal{O}(t)$ since $\|\mathbf{x}_t\|_0 \leq t$, and computing \mathbf{z}_{t+1} from \mathbf{z}_t and i_t takes time $\mathcal{O}(n)$. Thus the overall time complexity of running t steps is (we assume $p(n) = \Omega(n)$)

$$\mathcal{O}(tp(n) + t^2). \quad (7.157)$$

To derive a rate of convergence it remains to study the smoothness of f . This can be done as follows:

$$\begin{aligned} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_\infty &= \|\mathbf{D}^\top \mathbf{D}(\mathbf{x} - \mathbf{y})\|_\infty \\ &= \max_{1 \leq i \leq N} \left| \mathbf{d}_i^\top \left(\sum_{j=1}^N \mathbf{d}_j (x(j) - y(j)) \right) \right| \\ &\leq m^2 \|\mathbf{x} - \mathbf{y}\|_1, \end{aligned}$$

which means that f is m^2 -smooth with respect to the l_1 -norm. Thus we get the following rate of convergence:

$$f(\mathbf{x}_t) - f(\mathbf{x}^*) \leq \frac{8m^2}{t+1}. \quad (7.158)$$

Putting together (7.157) and (7.158) we proved that one can get an ε -optimal solution to (7.154) with a computational effort of $\mathcal{O}(m^2 p(n)/\varepsilon + m^4/\varepsilon^2)$ using the conditional gradient descent.

7.2.2 Exercises

Exercise 665. Use Frank-Wolfe's method to solve:

$$\begin{aligned} \min_{\mathbf{x}} \quad & 2x_1^2 + 2x_2^2 - 2x_1x_2 - 4x_1 - 6x_2 \\ \text{s.t. } \quad & x_1 + x_2 \leq 2 \\ & x_1 + 5x_2 \leq 5 \\ & -x_1 \leq 0 \\ & -x_2 \leq 0. \end{aligned}$$

Exercise 666. Use Frank-Wolfe's method to solve:

$$\begin{aligned} \min_{\mathbf{x}} \quad & (x_1 - 1)^2 + (x_2 - 1)^2 + (x_3 + 1)^2 \\ \text{s.t. } & x_1 + x_2 + x_3 \leq 3 \\ & x_1 \geq 0, x_2 \geq 0, x_3 \geq 0. \end{aligned}$$

7.3 Alternating Direction Method

(Taken from Chapter 4.1.3 of [83])

ADM fits for convex problems with separable objective functions and linear constraints:

$$\min_{\mathbf{x}, \mathbf{y}} f(\mathbf{x}) + g(\mathbf{y}), \quad \text{s.t. } \mathcal{A}(\mathbf{x}) + \mathcal{B}(\mathbf{y}) = \mathbf{c}, \quad (7.159)$$

where f and g are convex functions and \mathcal{A} and \mathcal{B} are linear mappings. It is a variant of the Lagrange Multiplier method. It first constructs an augmented Lagrangian function [79]:

$$\mathcal{L}(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}) = f(\mathbf{x}) + g(\mathbf{y}) + \langle \boldsymbol{\lambda}, \mathcal{A}(\mathbf{x}) + \mathcal{B}(\mathbf{y}) - \mathbf{c} \rangle + \frac{\beta}{2} \|\mathcal{A}(\mathbf{x}) + \mathcal{B}(\mathbf{y}) - \mathbf{c}\|^2, \quad (7.160)$$

where $\boldsymbol{\lambda}$ is the Lagrange multiplier and $\beta > 0$ is the penalty parameter, then updates the two variables alternately by minimizing the augmented Lagrangian function with the other variable fixed [79]:

$$\mathbf{x}_{k+1} = \operatorname{argmin}_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{y}_k, \boldsymbol{\lambda}_k) \quad (7.161)$$

$$= \operatorname{argmin}_{\mathbf{x}} f(\mathbf{x}) + \frac{\beta}{2} \|\mathcal{A}(\mathbf{x}) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c} + \boldsymbol{\lambda}_k/\beta\|^2,$$

$$\mathbf{y}_{k+1} = \operatorname{argmin}_{\mathbf{y}} \mathcal{L}(\mathbf{x}_{k+1}, \mathbf{y}, \boldsymbol{\lambda}_k) \quad (7.162)$$

$$= \operatorname{argmin}_{\mathbf{y}} g(\mathbf{y}) + \frac{\beta}{2} \|\mathcal{B}(\mathbf{y}) + \mathcal{A}(\mathbf{x}_{k+1}) - \mathbf{c} + \boldsymbol{\lambda}_k/\beta\|^2.$$

Finally, it updates the Lagrange multiplier [79]:

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \beta(\mathcal{A}(\mathbf{x}_{k+1}) + \mathcal{B}(\mathbf{y}_{k+1}) - \mathbf{c}). \quad (7.163)$$

7.3.1 Applying ADM to RPCA

ADM is arguably one of the most successful solvers for low-rank optimization problems, e.g., for Relaxed RPCA:

$$(\text{Relaxed RPCA}) \quad \min_{\mathbf{A}, \mathbf{E}} \|\mathbf{A}\|_* + \lambda \|\mathbf{E}\|_1, \quad \text{s.t. } \mathbf{D} = \mathbf{A} + \mathbf{E}. \quad (7.164)$$

In particular, applying procedures (7.161) and (7.162) to Relaxed RPCA, both can have closed-form solutions. More specifically, the subproblem to update \mathbf{A} is:

$$\mathbf{A}_{k+1} = \operatorname{argmin}_{\mathbf{A}} \|\mathbf{A}\|_* + \frac{\beta}{2} \|\mathbf{D} - \mathbf{A} - \mathbf{E}_k + \Lambda_k/\beta\|_F^2, \quad (7.165)$$

where Λ is the Lagrange multiplier, while that for updating \mathbf{E} is:

$$\mathbf{E}_{k+1} = \operatorname{argmin}_{\mathbf{E}} \lambda \|\mathbf{E}\|_1 + \frac{\beta}{2} \|\mathbf{D} - \mathbf{A}_{k+1} - \mathbf{E} + \Lambda_k/\beta\|_F^2. \quad (7.166)$$

(7.166) has a closed-form solution [79] as follows:

$$\mathbf{E}_{k+1} = \mathcal{S}_{\lambda\beta^{-1}}(\mathbf{D} - \mathbf{A}_k + \Lambda_k/\beta), \quad (7.167)$$

where

$$\mathcal{S}_\varepsilon(x) = \operatorname{sgn}(x) \max(|x| - \varepsilon, 0) = \begin{cases} x - \varepsilon, & \text{if } x > \varepsilon, \\ x + \varepsilon, & \text{if } x < -\varepsilon, \\ 0, & \text{if } -\varepsilon \leq x \leq \varepsilon, \end{cases} \quad (7.168)$$

is the soft thresholding operator. (7.165) also has a closed-form solution offered by Singular Value Thresholding (SVT) [22]: suppose that the SVD of $\mathbf{W} = \mathbf{D} - \mathbf{E}_k + \Lambda_k/\beta$ is $\mathbf{W} = \mathbf{U}\Sigma\mathbf{V}^T$, then the optimal solution is $\mathbf{A} = \mathbf{U}\mathcal{S}_{\beta^{-1}}(\Sigma)\mathbf{V}^T$. So solving low-rank models with nuclear norm, SVD is usually indispensable. For $m \times n$ matrices, the time complexity of SVD is $O(mn \min(m, n))$ [55]. So in general the computation cost for solving low-rank models with nuclear norm is high when m and n are large. Fortunately, from (7.168) one can see that it is unnecessary to compute the singular values not exceeding β^{-1} and their associated singular vectors, because these singular values will be shrunk to zeros, thus do not contribute to \mathbf{A} . So we only need to compute singular values greater than β^{-1} and their corresponding singular vectors. This can be achieved by `svds()` in MATLAB or using PROPACK [74] and accordingly the computation cost reduces to $O(rmn)$, where r is the expected rank of the optimal \mathbf{A} . It is worth noting that `svds()` and PROPACK can only provide expected number of leading singular values and their singular vectors. So we have to dynamically predict the value of r when calling `svds()` or PROPACK [79].

Algorithm 6 summarizes the ADM for Relaxed RPCA (7.164) [81].

The convergence of Algorithm 6 is guaranteed by the following theorem.

Theorem 667 (Convergence of Algorithm 6 [81]). *For Algorithm 6, if $\{\beta_k\}$ is nondecreasing and $\sum_{k=1}^{+\infty} \beta_k^{-1} = +\infty$, then $(\mathbf{A}_k, \mathbf{E}_k)$ converges to an optimal solution $(\mathbf{A}^*, \mathbf{E}^*)$ to the Relaxed RPCA problem.*

Algorithm 6 Solving Relaxed RPCA via ADM

- 1: **Input:** Observation matrix $\mathbf{D} \in \mathbb{R}^{m \times n}$, λ .
- 2: **Initialize:** $\mathbf{Y}_0 = \mathbf{D} / \max(\|\mathbf{D}\|, \lambda^{-1}\|\mathbf{D}\|_\infty)$, $\mathbf{E}_0 = \mathbf{0}$, $\beta_0 > 0$, $\rho > 1$, $k = 0$.
- 3: **while** not converged **do**
- 4: Lines 5-6 solve $\mathbf{A}_{k+1} = \underset{\mathbf{A}}{\operatorname{argmin}} \mathcal{L}(\mathbf{A}, \mathbf{E}_k, \boldsymbol{\Lambda}_k, \beta_k)$.
- 5: $(\mathbf{U}, \mathbf{S}, \mathbf{V}) = \operatorname{svd}(\mathbf{D} - \mathbf{E}_k + \beta_k^{-1} \boldsymbol{\Lambda}_k)$.
- 6: $\mathbf{A}_{k+1} = \mathbf{U} \mathcal{S}_{\beta_k^{-1}}(\mathbf{S}) \mathbf{V}^T$.
- 7: Line 8 solves $\mathbf{E}_{k+1} = \underset{\mathbf{E}}{\operatorname{argmin}} \mathcal{L}(\mathbf{A}_{k+1}, \mathbf{E}, \boldsymbol{\Lambda}_k, \beta_k)$.
- 8: $\mathbf{E}_{k+1} = \mathcal{S}_{\lambda \beta_k^{-1}}(\mathbf{D} - \mathbf{A}_{k+1} + \beta_k^{-1} \boldsymbol{\Lambda}_k)$.
- 9: $\boldsymbol{\Lambda}_{k+1} = \boldsymbol{\Lambda}_k + \beta_k (\mathbf{D} - \mathbf{A}_{k+1} - \mathbf{E}_{k+1})$.
- 10: Update β_k to β_{k+1} .
- 11: $k \leftarrow k + 1$.
- 12: **end while**
- 13: **Output:** $(\mathbf{A}_k, \mathbf{E}_k)$.

We can further prove that the condition $\sum_{k=1}^{+\infty} \beta_k^{-1} = +\infty$ is also necessary to ensure the convergence:

Theorem 668 ([81]). *If $\sum_{k=1}^{+\infty} \beta_k^{-1} < +\infty$, then the sequence $\{(\mathbf{A}_k, \mathbf{E}_k)\}$ produced by Algorithm 6 may not converge to the optimal solution of the Relaxed RPCA problem.*

7.3.1.1 Experiments

For the Relaxed RPCA problem, Lin et al. [81] randomly generated square matrices for synthetic experiments. Denote the ground truth solution by $(\mathbf{A}_0, \mathbf{E}_0) \in \mathbb{R}^{m \times m} \times \mathbb{R}^{m \times m}$. They generated the rank- r matrix \mathbf{A}_0 as a product $\mathbf{X}\mathbf{Y}^T$, where \mathbf{X} and \mathbf{Y} are independent $m \times r$ standard Gaussian random matrices. The matrix \mathbf{E}_0 was generated as a sparse matrix whose support was chosen uniformly at random, and whose nonzero entries are i.i.d. uniform in the interval $[-500, 500]$. The matrix $\mathbf{D} = \mathbf{A}_0 + \mathbf{E}_0$ is the input to the algorithms, and $(\mathbf{A}^*, \mathbf{E}^*)$ denotes the output. A fixed weighting parameter $\lambda = m^{-1/2}$ was chosen for the RPCA problem [24]. The codes were provided by their corresponding authors. A brief comparison of APG and ADM is presented in Tables 7.1 and 7.2. We can see that ADM is at least five times faster than APG. Moreover, the accuracies of ADM are higher than those of APG. In particular, APG often overestimates $\|\mathbf{E}_0\|_0$, the number of nonzeros in \mathbf{E}_0 , quite a bit. In comparison, the estimated $\|\mathbf{E}_0\|_0$ by ADM is much closer to the ground truth.

7.3.2 Exercises

(Taken from Chapter 7 of [109])

Exercise 669. Solve

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &= (x_1 - 1)^2 + (x_2 - 2)^2 \\ \text{s.t. } &x_1 + 2x_2 - 2 = 0 \end{aligned}$$

using the augmented Lagrange multiplier method with a fixed value of penalty parameter $\beta = 1$. Use a maximum of three iterations.

Exercise 670. Solve the following optimization problem using the augmented Lagrange multiplier method keeping penalty parameter $\beta = 1$ throughout the iterative process and $\lambda_1 = 0$:

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &= (x_1 - 1)^2 + (x_2 - 2)^2 \\ \text{s.t. } &-x_1 + 2x_2 = 2. \end{aligned}$$

Exercise 671. Solve the following optimization problem using the augmented Lagrange multiplier method keeping penalty parameter $\beta = 1$ throughout the iterative process and $\lambda_1 = 0$:

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &= x_1^3 - 6x_1^2 + 11x_1 + x_3 \\ \text{s.t. } &x_1^2 + x_2^2 - x_3^2 \leq 0, \\ &4 - x_1^2 - x_2^2 - x_3^2 \leq 0, \\ &x_3 \leq 5, \\ &x_i \geq 0, \quad i = 1, 2, 3. \end{aligned}$$

(Taken from Chapter 4.2 of [12])

Exercise 672. Consider the problem

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &= \frac{1}{2}(x_1^2 - x_2^2) - 2x_2, \\ \text{s.t. } &x_2 = 0. \end{aligned}$$

- (a) Calculate the optimal solution and the Lagrange multiplier.
- (b) For $k = 0, 1, 2$ and $\beta_k = 10^{k+1}$ calculate and compare the iterates of the quadratic penalty method with $\lambda_k = 0$ for all k and the method of multipliers with $\lambda_0 = 0$.
- (c) Suppose that β is taken to be constant in the method of multipliers. For what values of β would be the augmented Lagrangian have a minimum and for what values of β would the method converge?

Exercise 673. Consider the problem

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &= \frac{1}{2}(x_1^2 + |x_2|^\rho) + 2x_2, \\ \text{s.t. } x_2 &= 0, \end{aligned}$$

where $\rho > 1$.

- (a) Calculate the optimal solution and the Lagrange multiplier.
- (b) Write a computer program to calculate the iterates of the multiplier method with $\lambda_0 = 0$ and $\beta_k = 1$ for all k . Show computationally that the rate of convergence is sublinear if $\rho = 1.5$, linear if $\rho = 2$ and superlinear if $\rho = 3$.
- (c) Give a heuristic argument why the rate of convergence is sublinear if $\rho < 2$, linear if $\rho = 2$ and superlinear if $\rho > 2$. What happens in the limit where $\rho = 1$?

Exercise 674. Consider the problem in Exercise 672. Verify that the second order method of multipliers converges in one iteration provided β is sufficiently large, and estimate the threshold value for β .

Exercise 675 (Convergence Threshold and Convergence Rate of the Method of Multipliers). Consider the quadratic problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x}, \\ \text{s.t. } \mathbf{A} \mathbf{x} &= \mathbf{b}, \end{aligned}$$

where \mathbf{Q} is symmetric and \mathbf{A} is an $m \times n$ matrix of rank m . Let f^* be the optimal value of the problem and assume that the problem has a unique minimum \mathbf{x}^* with associated Lagrange multiplier $\boldsymbol{\lambda}^*$. Verify that for sufficiently large β , the penalized dual function is

$$q_\beta(\boldsymbol{\lambda}) = -\frac{1}{2}(\boldsymbol{\lambda} - \boldsymbol{\lambda}^*)^T \mathbf{A} (\mathbf{Q} + \beta \mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T (\boldsymbol{\lambda} - \boldsymbol{\lambda}^*) + f^*.$$

Consider the first order of multipliers.

- (a) Use the theory in Section 1.3 of [12] to show that for all k

$$\|\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}^*\| \leq r_k \|\boldsymbol{\lambda}_k - \boldsymbol{\lambda}^*\|,$$

where

$$r_k = \max\{|1 - \beta_k E_{\beta_k}|, |1 - \beta_k e_{\beta_k}|\}$$

and E_β and e_β denote the maximum and the minimum eigenvalues of the matrix $\mathbf{A}(\mathbf{Q} + \beta \mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$.

- (b) Using the Sherman-Morrison-Woodbury identity to relate the eigenvalues of the matrix $\mathbf{A}(\mathbf{Q} + \beta_k \mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$ with those of the matrix $\mathbf{A}\mathbf{Q}^{-1}\mathbf{A}^T$. Show that if $\gamma_1, \dots, \gamma_m$ are the eigenvalues of $\mathbf{A}\mathbf{Q}^{-1}\mathbf{A}^T$, we have

$$r_k = \max_i \left\| \frac{\gamma_i}{\gamma_i + \beta_k} \right\|.$$

Show that the method converges to λ^* if $c > \bar{c}$, where $\bar{c} = 0$ if $\gamma_i \geq 0$ for all i , and $\bar{c} = -2 \min\{\gamma_1, \dots, \gamma_m\}$ otherwise.

Exercise 676 (Stepsize Analysis of the Method of Multipliers). Consider the problem in Exercise 675. Use the results of that exercise to analyze the convergence and rate of convergence of the generalized method of multipliers;

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}^k + \alpha_k (\mathbf{A}\mathbf{x}_k - \mathbf{b}),$$

where α_k is a positive stepsize. Show in particular that if \mathbf{Q} is positive definite and $\beta_k = \beta$ for all k , convergence is guaranteed if $\delta \leq \alpha_k \leq 2\beta$ for all k , where δ is some positive scalar.

Exercise 677. A weakness of the quadratic penalty method is that the augmented Lagrangian may not have a global minimum. As an example, show that the scalar problem

$$\begin{aligned} \min_x \quad & -x^4, \\ \text{s.t. } & x = 0, \end{aligned}$$

has the unique global minimum $x^* = 0$ but its augmented Lagrangian

$$L_{\beta_k}(x, \lambda_k) = -x^4 + \lambda_k x + \frac{\beta_k}{2} x^2$$

has no global minimum for every β_k and λ_k . To overcome this issue, consider a penalty function of the form

$$\frac{\beta}{2} \|h(x)\|^2 + \|h(x)\|^\rho,$$

where $\rho > 4$, instead of $\frac{\beta}{2} \|h(x)\|^2$. Show that $L_{\beta_k}(x, \lambda_k)$ has a global minimum for every λ_k and $\beta_k > 0$.

Exercise 678. Consider the quadratic penalty method ($\beta_k \rightarrow \infty$) for the equality constrained problem of minimizing $f(\mathbf{x})$ subject to $h(\mathbf{x}) = 0$, and assume that the generated sequence converges to a local minimum \mathbf{x}^* that is also a regular point. Show that the condition number of the Hessian $\nabla_{xx}^2 L_{\beta_k}(\mathbf{x}_k, \boldsymbol{\lambda}_k)$ tends to ∞ .

(Taken from Chapter 4.3 of [12])

Exercise 679. Let \mathbf{H} be a positive definite symmetric matrix. Show that the pair $(\mathbf{x}^*, \boldsymbol{\mu}^*)$ satisfies the first order necessary conditions for the problem

$$\begin{aligned} & \min_{\mathbf{x}} f(\mathbf{x}) \\ & \text{s.t. } g_j(\mathbf{x}) \leq 0, \quad j = 1, \dots, r, \end{aligned}$$

if and only if $(\mathbf{0}, \boldsymbol{\mu}^*)$ is a global minimum-Lagrange multiplier pair of the quadratic problem:

$$\begin{aligned} & \min_{\mathbf{d}} \nabla f(\mathbf{x}^*)^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T \mathbf{H} \mathbf{d}, \\ & \text{s.t. } g_j(\mathbf{x}^*) + \nabla g_j(\mathbf{x}^*)^T \mathbf{d} \leq 0, \quad j = 1, \dots, r. \end{aligned}$$

Exercise 680. Show that if $(\mathbf{d}, \boldsymbol{\mu})$ is a global minimum-Lagrange multiplier pair of the quadratic program

$$\begin{aligned} & \min_{\mathbf{x}, \mathbf{d}} \nabla f(\mathbf{x})^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T \mathbf{H} \mathbf{d}, \\ & \text{s.t. } g_j(\mathbf{x}) + \nabla g_j(\mathbf{x})^T \mathbf{d} \leq 0, \quad j = 1, \dots, r, \end{aligned}$$

where \mathbf{H} is positive definite symmetric and

$$c \geq \sum_{j=1}^r \mu_j,$$

then $(\mathbf{d}, \xi = 0, \bar{\boldsymbol{\mu}})$ is a global minimum-Lagrange multiplier pair of the quadratic problem

$$\begin{aligned} & \min_{\mathbf{x}, \mathbf{d}} \nabla f(\mathbf{x})^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T \mathbf{H} \mathbf{d} + c \xi, \\ & \text{s.t. } g_j(\mathbf{x}) + \nabla g_j(\mathbf{x})^T \mathbf{d} \leq \xi, \quad j = 1, \dots, r, \end{aligned}$$

where $\bar{\mu}_j = \mu_j$ for $j = 1, \dots, r$, $\bar{\mu}_0 = c - \sum_{j=1}^r \mu_j$ and $g_0(\mathbf{x}) \equiv 0$.

Exercise 681. Show that if $(\mathbf{x}^*, \boldsymbol{\mu}^*)$ satisfies the first order necessary conditions of the problem

$$\begin{aligned} & \min_{\mathbf{x}} f(\mathbf{x}), \\ & \text{s.t. } g_j(\mathbf{x}) \leq 0, \quad j = 1, \dots, r, \end{aligned}$$

then (\mathbf{x}^*) is a stationary point of $f + cP$ for all $c \geq \sum_{j=1}^r \mu_j^*$, where

$$P(x) = \max\{0, g_1(\mathbf{x}), \dots, g_r(\mathbf{x})\}.$$

Exercise 682 (Marato's Effect). This example illustrates a fundamental difficulty in attaining superlinear convergence using the nondifferentiable exact penalty function for

monitoring descent. This difficulty does not arise for differentiable exact penalty functions.) Consider the problem

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) &= x_1, \\ \text{s.t. } h(\mathbf{x}) &= x_1^2 + x_2^2 - 1 = 0, \end{aligned}$$

with optimal solution $\mathbf{x}^* = (-1, 0)^T$ and Lagrange multiplier $\lambda = 1/2$. For any \mathbf{x} , let \mathbf{b}, λ be an optimal solution-Lagrange multiplier pair of the problem

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{d}} \nabla f(\mathbf{x}) + \frac{1}{2} \mathbf{d}^T \nabla^2 L(\mathbf{x}^*, \lambda^*) \mathbf{d}, \\ \text{s.t. } h(\mathbf{x}) + \nabla h(\mathbf{x})^T \mathbf{d} = 0, \end{aligned}$$

Show that for all c ,

$$f(\mathbf{x} + \mathbf{d}) + c|h(\mathbf{x} + \mathbf{d})| - f(\mathbf{x}) - c|h(\mathbf{x})| = \lambda h(\mathbf{x}) - c|h(\mathbf{x})| + (c - \lambda^*)\|\mathbf{d}\|^2.$$

Conclude that for $c > 2\lambda$, there are points \mathbf{x} arbitrarily close to \mathbf{x}^* for which the exact penalty function $f(\mathbf{x}) + c|h(\mathbf{x})|$ is not reduced by a pure Newton step.

A more detailed description of Marato's Effect can be found in Chapter 15.5 of [104].

(Taken from Chapter 17 of [104])

Exercise 683. Verify that the KKT conditions for the bound-constrained problem

$$\min_{\mathbf{x} \in \mathbb{R}^n} \phi(\mathbf{x}), \quad \text{s.t. } l \leq \mathbf{x} \leq u$$

are equivalent to the compactly stated condition

$$\mathbf{x} - \mathcal{P}(\mathbf{x} - \nabla \phi(\mathbf{x}), l, u) = \mathbf{0},$$

where the operator \mathcal{P} is the projection onto the rectangular box $[l, u]$.

7.4 Linearized Alternating Direction Method with Adaptive Penalty

(Taken from Chapter 4.1.4 of [83])

The advantage of ADM is that its subproblems are simpler than the original problem. They may even have closed-form solutions. When the subproblems are not easily solvable, one may consider approximating the squared constraint $\frac{\beta}{2} \|\mathcal{A}(\mathbf{x}) + \mathcal{B}(\mathbf{y}) - \mathbf{c}\|_2^2$ in the augmented Lagrangian function with its first order Taylor expansion plus a proximal term, to make the subproblem even simpler. This technique is called the Linearized Alternating

Direction Method (LADM) [79]. Specifically, by linearizing the quadratic term in (7.161) at \mathbf{x}_k and adding a proximal term, LADM solves the following approximation:

$$\begin{aligned}\mathbf{x}_{k+1} &= \underset{\mathbf{x}}{\operatorname{argmin}} f(\mathbf{x}) + \langle \mathcal{A}^*(\boldsymbol{\lambda}_k) + \beta \mathcal{A}^*(\mathcal{A}(\mathbf{x}_k) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c}), \mathbf{x} - \mathbf{x}_k \rangle + \frac{\beta \eta_A}{2} \|\mathbf{x} - \mathbf{x}_k\|_2^2, \\ &= \underset{\mathbf{x}}{\operatorname{argmin}} f(\mathbf{x}) + \frac{\beta \eta_A}{2} \|\mathbf{x} - \mathbf{x}_k + \mathcal{A}^*(\boldsymbol{\lambda}_k + \beta(\mathcal{A}(\mathbf{x}_k) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c})) / (\beta \eta_A)\|_2^2,\end{aligned}\quad (7.169)$$

where \mathcal{A}^* is the adjoint operator of \mathcal{A} and $\eta_A > 0$ is a parameter. Similarly, the subproblem (7.162) can be approximated by

$$\mathbf{y}_{k+1} = \underset{\mathbf{y}}{\operatorname{argmin}} g(\mathbf{y}) + \frac{\beta \eta_A}{2} \|\mathbf{y} - \mathbf{y}_k + \mathcal{B}^*(\boldsymbol{\lambda}_k + \beta(\mathcal{A}(\mathbf{x}_{k+1}) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c})) / (\beta \eta_B)\|_2^2. \quad (7.170)$$

We assume that subproblems (7.169) and (7.170), which are proximal operators of $f(\mathbf{x})$ and $g(\mathbf{y})$, respectively, are easily solvable. The update of Lagrange multiplier still goes as (7.163). Lin et al. [79] further suggested updating the penalty parameter β as follows:

$$\beta_{k+1} = \min(\beta_{\max}, \rho \beta_k), \quad (7.171)$$

where β_{\max} is an upper bound of $\{\beta_k\}$. The value of ρ is defined as

$$\rho = \begin{cases} \rho_0, & \text{if } \beta_k \max(\sqrt{\eta_A} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2, \sqrt{\eta_B} \|\mathbf{y}_{k+1} - \mathbf{y}_k\|_2) / \|\mathbf{c}\|_2 < \varepsilon_2, \\ 1, & \text{otherwise,} \end{cases} \quad (7.172)$$

where $\rho_0 > 1$ is a constant¹. The condition for assigning $\rho = \rho_0$ comes from the analysis on the stopping criteria shown below. The advantage of using the above adaptive penalty is that the parameters β_0 and ρ_0 can be easily tuned. A basic guideline is that $\rho = \rho_0$ should be frequently invoked, which can be easily fulfilled if neither β_0 nor ρ_0 is too large. If a fixed penalty is used, it will be hard to tune a good β that fits for different data sets. The complete algorithm is shown in Algorithm 7.

Stopping Criteria: We now analyze the stopping criteria for Algorithm 7. They are based on the Karush-Kuhn-Tucker (KKT) conditions of problem (7.159), which is the existence of a triplet $(\mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\lambda}^*)$ such that

$$\mathcal{A}(\mathbf{x}^*) + \mathcal{B}(\mathbf{y}^*) - \mathbf{c} = \mathbf{0}, \quad (7.173)$$

$$-\mathcal{A}^*(\boldsymbol{\lambda}^*) \in \partial f(\mathbf{x}^*), \quad -\mathcal{B}^*(\boldsymbol{\lambda}^*) \in \partial g(\mathbf{y}^*). \quad (7.174)$$

¹If $\mathbf{c} = \mathbf{0}$, we may change $\beta_k \max(\sqrt{\eta_A} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2, \sqrt{\eta_B} \|\mathbf{y}_{k+1} - \mathbf{y}_k\|_2) / \|\mathbf{c}\|_2 < \varepsilon_2$ to $\beta_k \max(\sqrt{\eta_A} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_\infty, \sqrt{\eta_B} \|\mathbf{y}_{k+1} - \mathbf{y}_k\|_\infty) < \varepsilon_2$. Such a treatment applies to the discussions that follow.

Algorithm 7 Linearized Alternating Direction Method with Adaptive Penalty (LADMAP)

- 1: **Initialize:** Set $\varepsilon_1 > 0$, $\varepsilon_2 > 0$, $\beta_{\max} \gg \beta_0 > 0$, $\eta_A > \|\mathcal{A}\|^2$, $\eta_B > \|\mathcal{B}\|^2$, \mathbf{x}_0 , \mathbf{y}_0 , $\boldsymbol{\lambda}_0$, and $k \leftarrow 0$.
 - 2: **while** (7.175) or (7.178) is not satisfied **do**
 - 3: Update \mathbf{x} by solving (7.169).
 - 4: Update \mathbf{y} by solving (7.170).
 - 5: Update $\boldsymbol{\lambda}$ by (7.163).
 - 6: Update β by (7.171) and (7.172).
 - 7: $k \leftarrow k + 1$.
 - 8: **end while**
 - 9: **Output:** $(\mathbf{x}_k, \mathbf{y}_k)$.
-

Such a triplet $(\mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\lambda}^*)$ is called a KKT point. Thus the first stopping criterion is to monitor the fulfilment of linear constraint:

$$\|\mathcal{A}(\mathbf{x}_{k+1}) + \mathcal{B}(\mathbf{y}_{k+1}) - \mathbf{c}\|_2 / \|\mathbf{c}\|_2 < \varepsilon_1. \quad (7.175)$$

The condition for assigning $\rho = \rho_0$ in (7.172) comes from the second KKT condition. By comparing the first part of (7.179) below with the left part of (7.174) above, we can see that if we approximate $\boldsymbol{\lambda}^*$ by $\tilde{\boldsymbol{\lambda}}_{k+1}$ then the difference $\beta_k \eta_A (\mathbf{x}_{k+1} - \mathbf{x}_k)$ should be small. Next, we rewrite the second part of (7.179) as:

$$-\beta_k [\eta_B (\mathbf{y}_{k+1} - \mathbf{y}_k) + \mathcal{B}^*(\mathcal{A}(\mathbf{x}_{k+1} - \mathbf{x}_k))] - \mathcal{B}^*(\tilde{\boldsymbol{\lambda}}_{k+1}) \in \partial g(\mathbf{y}_{k+1}), \quad (7.176)$$

and compare it with the right part of (7.174). Then we can see that the difference $\beta_k [\eta_B (\mathbf{y}_{k+1} - \mathbf{y}_k) + \mathcal{B}^*(\mathcal{A}(\mathbf{x}_{k+1} - \mathbf{x}_k))]$ should also be small. Thus both $\beta_k \eta_A \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2$ and $\beta_k \eta_B \|\mathbf{y}_{k+1} - \mathbf{y}_k\|_2$ should be small enough and we arrive at the second stopping criterion as follows:

$$\beta_k \max(\eta_A \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 / \|\mathcal{A}^*(\mathbf{c})\|_2, \eta_B \|\mathbf{y}_{k+1} - \mathbf{y}_k\|_2 / \|\mathcal{B}^*(\mathbf{c})\|_2) \leq \varepsilon'_2. \quad (7.177)$$

By estimating $\|\mathcal{A}^*(\mathbf{c})\|_2$ and $\|\mathcal{B}^*(\mathbf{c})\|_2$ by $\sqrt{\eta_A} \|\mathbf{c}\|_2$ and $\sqrt{\eta_B} \|\mathbf{c}\|_2$, respectively, we obtain the second stopping criterion which is more convenient to verify:

$$\beta_k \max(\sqrt{\eta_A} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2, \sqrt{\eta_B} \|\mathbf{y}_{k+1} - \mathbf{y}_k\|_2) / \|\mathbf{c}\|_2 \leq \varepsilon_2. \quad (7.178)$$

7.4.1 Convergence Analysis

To prove the convergence of LADM with Adaptive Penalty (LADMAP), we first have the following propositions.

Proposition 684 ([79]).

$$\begin{aligned} -\beta_k \eta_A(\mathbf{x}_{k+1} - \mathbf{x}_k) - \mathcal{A}^*(\tilde{\boldsymbol{\lambda}}_{k+1}) &\in \partial f(\mathbf{x}_{k+1}), \\ -\beta_k \eta_B(\mathbf{y}_{k+1} - \mathbf{y}_k) - \mathcal{B}^*(\hat{\boldsymbol{\lambda}}_{k+1}) &\in \partial g(\mathbf{y}_{k+1}), \end{aligned} \quad (7.179)$$

where $\tilde{\boldsymbol{\lambda}}_{k+1} = \boldsymbol{\lambda}_k + \beta_k [\mathcal{A}(\mathbf{x}_k) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c}]$, $\hat{\boldsymbol{\lambda}}_{k+1} = \boldsymbol{\lambda}_k + \beta_k [\mathcal{A}(\mathbf{x}_{k+1}) + \mathcal{B}(\mathbf{y}_k) - \mathbf{c}]$, and ∂f and ∂g are subgradients of f and g , respectively.

This can be easily proved by checking the optimality conditions of (7.169) and (7.170).

Proposition 685 ([79]). Denote the operator norms of \mathcal{A} and \mathcal{B} as $\|\mathcal{A}\|$ and $\|\mathcal{B}\|$, respectively. If $\{\beta_k\}$ is non-decreasing and upper bounded, $\eta_A > \|\mathcal{A}\|^2$, $\eta_B > \|\mathcal{B}\|^2$, and $(\mathbf{x}^*, \mathbf{y}^*, \boldsymbol{\lambda}^*)$ is any KKT point of problem (7.159), then:

- (1) $\{\eta_A \|\mathbf{x}_k - \mathbf{x}^*\|^2 - \|\mathcal{A}(\mathbf{x}_k - \mathbf{x}^*)\|^2 + \eta_B \|\mathbf{y}_k - \mathbf{y}^*\|^2 + \beta_k^{-2} \|\boldsymbol{\lambda}_k - \boldsymbol{\lambda}^*\|^2\}$ is non-increasing.
- (2) $\|\mathbf{x}_{k+1} - \mathbf{x}_k\| \rightarrow 0$, $\|\mathbf{y}_{k+1} - \mathbf{y}_k\| \rightarrow 0$, $\|\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k\| \rightarrow 0$.

Then we can prove the convergence of LADMAP, as stated in the following theorem.

Theorem 686 (Convergence of LADMAP [79]). If $\{\beta_k\}$ is non-decreasing and upper bounded, $\eta_A > \|\mathcal{A}\|^2$, and $\eta_B > \|\mathcal{B}\|^2$, then the sequence $\{(\mathbf{x}_k, \mathbf{y}_k, \boldsymbol{\lambda}_k)\}$ generated by LADMAP converges to a KKT point of problem (7.159).

7.4.2 Applying LADMAP to LRR

The Relaxed LRR problem (7.180):

$$(\text{Relaxed LRR}) \quad \min_{\mathbf{Z}, \mathbf{E}} \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_{2,1}, \quad \text{s.t.} \quad \mathbf{D} = \mathbf{DZ} + \mathbf{E}, \quad (7.180)$$

is a special case of problem (7.159). So LADMAP can be directly applied to it. The two subproblems both have closed-form solutions. With some algebra, the subproblem for updating \mathbf{E} is:

$$\mathbf{E}_{k+1} = \operatorname{argmin}_{\mathbf{E}} \lambda \|\mathbf{E}\|_{2,1} + \frac{\beta_k}{2} \|\mathbf{E} - \mathbf{M}_k\|_F^2, \quad (7.181)$$

where $\mathbf{M}_k = -\mathbf{DZ}_k + \mathbf{D} - \boldsymbol{\Lambda}_k / \beta_k$ and $\boldsymbol{\Lambda}_k$ is the Lagrange multiplier. (7.181) also has a closed-form solution: $(\mathbf{E}_{k+1})_{:i} = \mathcal{H}_{\lambda \beta_k^{-1}}((\mathbf{M}_k)_{:i})$, where

$$\mathcal{H}_\varepsilon(\mathbf{x}) = \begin{cases} \frac{\|\mathbf{x}\|_2 - \varepsilon}{\|\mathbf{x}\|_2} \mathbf{x}, & \text{if } \|\mathbf{x}\|_2 > \varepsilon, \\ \mathbf{0}, & \text{otherwise,} \end{cases} \quad (7.182)$$

is the $\ell_{2,1}$ -norm shrinkage operator [84]. In the subproblem for updating \mathbf{Z} , one has to apply the singular value shrinkage operator [22], with a threshold $(\beta_k \eta_D)^{-1}$, to matrix $\mathbf{N}_k = \mathbf{Z}_k - \eta_D^{-1} \mathbf{D}^T (\mathbf{D}\mathbf{Z}_k + \mathbf{E}_{k+1} - \mathbf{D} + \mathbf{\Lambda}_k / \beta_k)$, where $\eta_D > \|\mathbf{D}\|_2^2$.

Unfortunately, naively applying LADMAP to Relaxed LRR still results in a complexity of $O(n^3)$ (suppose that the data matrix is square, $n \times n$), even if partial SVD is used when applying the singular value shrinkage operator (see Section 7.3.1), which brings down the complexity of computing SVD to $O(rn^2)$, where r is the estimated rank of true \mathbf{Z} . This is because forming \mathbf{M}_k and \mathbf{N}_k explicitly requires full sized matrix-matrix multiplications, e.g., $\mathbf{D}\mathbf{Z}_k$. To further reduce the complexity, we choose not to form \mathbf{M}_k , \mathbf{N}_k , and \mathbf{Z}_k explicitly. By representing \mathbf{Z}_k as its skinny SVD: $\mathbf{Z}_k = \mathbf{U}_k \mathbf{\Sigma}_k \mathbf{V}_k^T$, some of the full sized matrix-matrix multiplications can be dismissed. They are replaced by successive reduced sized matrix-matrix multiplications. For example, when updating \mathbf{E} , $\mathbf{D}\mathbf{Z}_k$ is computed as $((\mathbf{D}\mathbf{U}_k)\mathbf{\Sigma}_k)\mathbf{V}_k^T$, reducing the complexity to $O(rn^2)$. When computing the partial SVD of \mathbf{N}_k , we need more advanced techniques. If we form \mathbf{N}_k explicitly, we will face with computing $\mathbf{D}^T(\mathbf{D} + \mathbf{\Lambda}_k / \beta_k)$, which is neither low-rank nor sparse. This can be bypassed by dig into the process of partial SVD, which is based on the Lanczos procedure [55] that bi-diagonalizes a given matrix. The Lanczos procedure on \mathbf{N}_k only requires to compute matrix-vector multiplications $\mathbf{N}_k \mathbf{v}$ and $\mathbf{u}^T \mathbf{N}_k$, where \mathbf{u} and \mathbf{v} are some vectors. So we may compute $\mathbf{N}_k \mathbf{v}$ and $\mathbf{u}^T \mathbf{N}_k$ by multiplying the vectors \mathbf{u} and \mathbf{v} successively with the component matrices in \mathbf{N}_k , rather than forming \mathbf{N}_k explicitly. Consequently, the computation complexity of partial SVD of \mathbf{N}_k is still $O(rn^2)$. With these acceleration techniques, the complexity of accelerated LADMAP (denoted as LADMAP(A) for short) for Relaxed LRR in each iteration is reduced to $O(rn^2)$. We summarize LADMAP(A) in Algorithm 8.

7.4.3 Experiments

Lin et al. [79] generated the synthetic test data as follows. There are four parameters, (s, p, d, \tilde{r}) , for the data. They first constructed s independent subspaces $\{\mathcal{S}_i\}_{i=1}^s$, whose bases $\{\mathbf{U}_i\}_{i=1}^s$ were generated by $\mathbf{U}_{i+1} = \mathbf{T}\mathbf{U}_i$, $1 \leq i \leq s-1$, where \mathbf{T} is a random rotation matrix and \mathbf{U}_1 is a $d \times \tilde{r}$ random column orthonormal matrix. So the rank of each subspace is \tilde{r} and the ambient dimension of data is d . Then they sampled p data points from each subspace by $\mathbf{X}_i = \mathbf{U}_i \mathbf{Q}_i$, $1 \leq i \leq s$, where \mathbf{Q}_i is an $\tilde{r} \times p$ standard Gaussian random matrix. 20% samples were then randomly corrupted by adding Gaussian noise with zero mean and standard deviation $0.1\|\mathbf{x}\|$. Lin et al. empirically found that Relaxed LRR achieves the best clustering performance on this data set when $\lambda = 0.1$. So they tested all algorithms with $\lambda = 0.1$ in this experiment. To obtain ground truth solutions $(\mathbf{Z}_0, \mathbf{E}_0)$

Algorithm 8 Accelerated LADMAP for Relaxed LRR (7.180)

- 1: **Input:** Observation matrix \mathbf{D} and parameter $\lambda > 0$.
 - 2: **Initialize:** Set \mathbf{E}_0 , \mathbf{Z}_0 and Λ_0 to zero matrices, where \mathbf{Z}_0 is represented as $(\mathbf{U}_0, \Sigma_0, \mathbf{V}_0) \leftarrow (\mathbf{0}, \mathbf{0}, \mathbf{0})$. Set $\varepsilon_1 > 0$, $\varepsilon_2 > 0$, $\beta_{\max} \gg \beta_0 > 0$, $\eta_D > \|\mathbf{D}\|_2^2$, $r = 5$, and $k \leftarrow 0$.
 - 3: **while** not converge **do**
 - 4: Update $\mathbf{E}_{k+1} = \underset{\mathbf{E}}{\operatorname{argmin}} \lambda \|\mathbf{E}\|_{2,1} + \frac{\beta_k}{2} \|\mathbf{E} + (\mathbf{X}\mathbf{U}_k)\Sigma_k \mathbf{V}_k^T - \mathbf{X} + \Lambda_k/\beta_k\|_F^2$. This subproblem can be solved by using (7.182).
 - 5: Update the skinny SVD $(\mathbf{U}_{k+1}, \Sigma_{k+1}, \mathbf{V}_{k+1})$ of \mathbf{Z}_{k+1} :
 - 6: i. Compute the partial SVD $\tilde{\mathbf{U}}_r \tilde{\Sigma}_r \tilde{\mathbf{V}}_r^T$ of the *implicit* matrix \mathbf{N}_k , which is bi-diagonalized by the successive matrix-vector multiplication technique.
 - 7: ii. $\mathbf{U}_{k+1} = (\tilde{\mathbf{U}}_r)_{:,1:r'}$, $\Sigma_{k+1} = (\tilde{\Sigma}_r)_{1:r',1:r'} - (\beta_k \eta_D)^{-1} \mathbf{I}$, $\mathbf{V}_{k+1} = (\tilde{\mathbf{V}}_r)_{:,1:r'}$, where r' is the number of singular values in Σ_r that are greater than $(\beta_k \eta_D)^{-1}$.
 - 8: Update the predicted rank r :
 - 9:
 - 10: If $r' < r$, then $r = \min(r' + 1, n)$; otherwise, $r = \min(r' + \text{round}(0.05n), n)$.
 - 11: Update $\Lambda_{k+1} = \Lambda_k + \beta_k((\mathbf{X}\mathbf{U}_{k+1})\Sigma_{k+1}\mathbf{V}_{k+1}^T + \mathbf{E}_{k+1} - \mathbf{D})$.
 - 12: Update β_{k+1} by (7.171)-(7.172).
 - 13: $k \leftarrow k + 1$.
 - 14: **end while**
 - 15: **Output:** $(\mathbf{Z}_k, \mathbf{E}_k)$.
-

for measuring the relative errors in the solutions, they ran LADMAP 2000 iterations with $\beta_{\max} = 10^3$. This number of iterations is far more than necessary. So $(\mathbf{Z}_0, \mathbf{E}_0)$ can be regarded as the ground truth solution.

The comparison among different algorithms is shown in Table 7.3. We can see that the iteration numbers and the CPU times of both LADMAP and LADMAP(A) are much less than those of other methods, and LADMAP(A) is further much faster than LADMAP. Moreover, the advantage of LADMAP(A) is even greater when the ratio \tilde{r}/p is smaller. As \tilde{r}/p is roughly the ratio of the rank of \mathbf{Z}_0 to the size of \mathbf{Z}_0 , this testifies to the complexity estimations on LADMAP and LADMAP(A) for Relaxed LRR. Note that the iteration numbers of ADM seem to grow with the problem sizes, while those of LADMAPs do not. This can be attributed to the adoption of adaptive penalty. Finally, as APG actually solves an approximate problem of (7.180) by adding the squared constraint to the objective, its relative errors are larger and its clustering accuracy is lower than those of ADM and LADM based methods.

7.5 (Proximal) Linearized Alternating Direction Method with Parallel Splitting and Adaptive Penalty

(Taken from Chapter 4.1.5 of [83])

In LADM, it is assumed that there are only two blocks of variables, \mathbf{x} and \mathbf{y} , and the proximal operators of $f(\mathbf{x})$ and $g(\mathbf{y})$ are easily solvable. However, in reality these assumptions do not hold. For example, when there are several constraints on the variables, after introducing auxiliary variables to decouple the objective functions and the constraints, there will be much more than two blocks of variables. For some objective functions, such as the logistic loss function, they do not have easily solvable proximal operators. So in this subsection, we consider the following model problem [82, 90]:

$$\min_{\mathbf{x}_1, \dots, \mathbf{x}_n} \sum_{i=1}^n f_i(\mathbf{x}_i), \quad \text{s.t.} \quad \sum_{i=1}^n \mathcal{A}_i(\mathbf{x}_i) = \mathbf{b}, \quad (7.183)$$

where \mathbf{x}_i 's and \mathbf{b} are vectors or matrices, \mathcal{A}_i 's are linear mappings, and

$$f_i(\mathbf{x}) = g_i(\mathbf{x}) + h_i(\mathbf{x}). \quad (7.184)$$

Both g_i and h_i are convex and lower semi-continuous. Furthermore, g_i is L_i -smooth. h_i may be nonsmooth but is simple, in the sense that its proximal operator $\min_{\mathbf{x}} h_i(\mathbf{x}) + \frac{\alpha}{2} \|\mathbf{x} - \mathbf{a}\|_2^2$ can be solved cheaply, or even has a closed-form solution. In particular, either of g_i and h_i can be zero.

We have to emphasize that if $n \geq 3$, a naive generalization of the two-block (L)ADM may not converge [26]. So designing a new mechanism of updating the variables is necessary. It can be proven that if we change the serial update in (L)ADM with *parallel* update and choose some parameters appropriately, the convergence can still be guaranteed, even if linearization is used. This results in Proximal Linearized Alternating Direction Method of Multiplier with Parallel Splitting and Adaptive Penalty (PLADMPSAP) [82, 90]. When $g_i(\mathbf{x}) = 0, \forall i$, no linearization on g_i is needed. In this case, the method is called Linearized Alternating Direction Method of Multiplier with Parallel Splitting and Adaptive Penalty (LADMPSAP).

PLADMPSAP applies the same technique of linearization, but it linearizes both the component objective function g_i and the squared constraint, leading to the following

updates:

$$\begin{aligned}
 \mathbf{x}_i^{k+1} &= \underset{\mathbf{x}_i}{\operatorname{argmin}} \left(g_i(\mathbf{x}_i^k) + \langle \nabla g_i(\mathbf{x}_i^k), \mathbf{x}_i - \mathbf{x}_i^k \rangle \right) + h_i(\mathbf{x}_i) + \langle \boldsymbol{\lambda}^k, \mathcal{A}_i(\mathbf{x}_i^k) - \mathbf{b} \rangle \\
 &\quad + \left\langle \beta_k \mathcal{A}_i^* \left(\sum_{j=1}^n \mathcal{A}_j(\mathbf{x}_j^k) - \mathbf{b} \right), \mathbf{x}_i - \mathbf{x}_i^k \right\rangle + \frac{\tau_k^{(i)}}{2} \|\mathbf{x}_i - \mathbf{x}_i^k\|_2^2 \\
 &= \underset{\mathbf{x}_i}{\operatorname{argmin}} h_i(\mathbf{x}_i) + \left\langle \nabla g_i(\mathbf{x}_i^k) + \mathcal{A}_i^*(\boldsymbol{\lambda}^k) + \beta_k \mathcal{A}_i^* \left(\sum_{j=1}^n \mathcal{A}_j(\mathbf{x}_j^k) - \mathbf{b} \right), \mathbf{x}_i - \mathbf{x}_i^k \right\rangle \\
 &\quad + \frac{\tau_k^{(i)}}{2} \|\mathbf{x}_i - \mathbf{x}_i^k\|_2^2 \\
 &= \underset{\mathbf{x}_i}{\operatorname{argmin}} h_i(\mathbf{x}_i) + \frac{\tau_k^{(i)}}{2} \left\| \mathbf{x}_i - \mathbf{x}_i^k + \left[\nabla g_i(\mathbf{x}_i^k) + \mathcal{A}_i^*(\hat{\boldsymbol{\lambda}}^k) \right] / \tau_k^{(i)} \right\|_2^2,
 \end{aligned} \tag{7.185}$$

where

$$\hat{\boldsymbol{\lambda}}^k = \boldsymbol{\lambda}^k + \beta_k \left(\sum_{j=1}^n \mathcal{A}_j(\mathbf{x}_j^{k+1}) - \mathbf{b} \right), \tag{7.186}$$

and $\tau_k^{(i)} = T_i + \eta_i \beta_k$, in which $T_i \geq L_i$ and $\eta_i > n \|\mathcal{A}_i\|^2$. $\beta_k > 0$ is the penalty parameter in the augmented Lagrangian function. By assumption, subproblem (7.185) is easily solvable. Note that $\tau_k^{(i)}$ reflects the linearization in both g_i (the part T_i) and the squared constraint (the part $\eta_i \beta_k$).

The updates of Lagrange multiplier $\boldsymbol{\lambda}$ and the penalty β go as

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \beta_k \left(\sum_{i=1}^n \mathcal{A}_i(\mathbf{x}_i^{k+1}) - \mathbf{b} \right), \tag{7.187}$$

and $\beta_{k+1} = \min(\beta_{\max}, \rho \beta_k)$ with

$$\rho = \begin{cases} \rho_0, & \text{if } \max_i (\|\mathcal{A}_i\|^{-1} \|\nabla g_i(\mathbf{x}_i^{k+1}) - \nabla g_i(\mathbf{x}_i^k) - \tau_k^{(i)} (\mathbf{x}_i^{k+1} - \mathbf{x}_i^k)\|_2) / \|\mathbf{b}\|_2 < \varepsilon_2, \\ 1, & \text{otherwise,} \end{cases} \tag{7.188}$$

where $\rho_0 \geq 1$. Similar to the update rule (7.172) of LADMAP, the condition for assigning $\rho = \rho_0$ in (7.188) comes from the analysis on the stopping criteria (cf. Section 7.4).

The iteration terminates when the following two conditions are met:

$$\left\| \sum_{i=1}^n \mathcal{A}_i(\mathbf{x}_i^{k+1}) - \mathbf{b} \right\|_2 / \|\mathbf{b}\|_2 < \varepsilon_1, \tag{7.189}$$

$$\max \left(\left\{ \|\mathcal{A}_i\|^{-1} \left\| \nabla g_i(\mathbf{x}_i^{k+1}) - \nabla g_i(\mathbf{x}_i^k) - \tau_i^{(k)} (\mathbf{x}_i^{k+1} - \mathbf{x}_i^k) \right\|_2, i = 1, \dots, n \right\} \right) / \|\mathbf{b}\|_2 < \varepsilon_2. \tag{7.190}$$

Algorithm 9 (P)LADMP SAP for Solving (7.183) with f_i Satisfying (7.184).

- 1: **Initialize:** Set $\rho_0 > 1$, $\beta_{\max} \gg \beta_0 > 0$, $\boldsymbol{\lambda}^0$, $T_i \geq L_i$, $\eta_i > n\|\mathcal{A}_i\|^2$, \mathbf{x}_i^0 , $i = 1, \dots, n$.
 - 2: **while** (7.189) or (7.190) is not satisfied **do**
 - 3: Compute $\hat{\boldsymbol{\lambda}}^k$ as (7.186).
 - 4: Update \mathbf{x}_i 's *in parallel* by solving (7.185), $i = 1, \dots, n$.
 - 5: Update $\boldsymbol{\lambda}$ by (7.187) and β by $\beta_{k+1} = \min(\beta_{\max}, \rho\beta_k)$ with ρ defined in (7.188).
 - 6: $k \leftarrow k + 1$.
 - 7: **end while**
 - 8: **Output:** $(\mathbf{x}_1^k, \dots, \mathbf{x}_n^k)$.
-

These two conditions are also deduced from the KKT conditions. The complete algorithm is summarized in Algorithm 9.

We want to highlight that the updates of \mathbf{x}_i is *parallel* because $\hat{\boldsymbol{\lambda}}^k$ only utilizes the information from the previous round of iteration. This is different from LADMAP, which is for the case of two blocks of variables only.

The following theorem provides theoretical guarantee on PLADMP SAP.

Theorem 687 (Convergence of PLADMP SAP [82]). *If β_k is non-decreasing and upper bounded, $\tau_k^{(i)} = T_i + \eta_i \beta_k$, where $T_i \geq L_i$ and $\eta_i > n\|\mathcal{A}_i\|^2$, $i = 1, \dots, n$, then $\{(\{\mathbf{x}_i^k\}, \boldsymbol{\lambda}^k)\}$ generated by PLADMP SAP converge to a KKT point of problem (7.183).*

We further have the following convergence rate theorem for PLADMP SAP in an ergodic sense.

Theorem 688 (Convergence Rate of PLADMP SAP [82]). *Let $\{(\{\mathbf{x}_i^*\}, \boldsymbol{\lambda}^*)\}$ be any KKT point of problem (7.183). Define $\bar{\mathbf{x}}_i^K = \sum_{k=0}^K \gamma_k \mathbf{x}_i^{k+1}$, where $\gamma_k = \beta_k^{-1} / \sum_{j=0}^K \beta_j^{-1}$. Then the following inequality holds for $\bar{\mathbf{x}}_i^K$:*

$$\begin{aligned} & \sum_{i=1}^n (f_i(\bar{\mathbf{x}}_i^K) - f_i(\mathbf{x}_i^*) + \langle \mathcal{A}_i^*(\boldsymbol{\lambda}^*), \bar{\mathbf{x}}_i^K - \mathbf{x}_i^* \rangle) + \frac{\alpha\beta_0}{2} \left\| \sum_{i=1}^n \mathcal{A}_i(\bar{\mathbf{x}}_i^K) - \mathbf{b} \right\|_2^2 \\ & \leq C_0 / \left(2 \sum_{k=0}^K \beta_k^{-1} \right), \end{aligned} \quad (7.191)$$

where

$$\alpha^{-1} = (n+1) \max \left(1, \left\{ \frac{\|\mathcal{A}_i\|^2}{\eta_i - n\|\mathcal{A}_i\|^2}, i = 1, 2, \dots, n \right\} \right), \quad (7.192)$$

and

$$C_0 = \sum_{i=1}^n \beta_0^{-1} \tau_0^{(i)} \|\mathbf{x}_i^0 - \mathbf{x}_i^*\|_2^2 + \beta_0^{-2} \|\boldsymbol{\lambda}^0 - \boldsymbol{\lambda}^*\|_2^2. \quad (7.193)$$

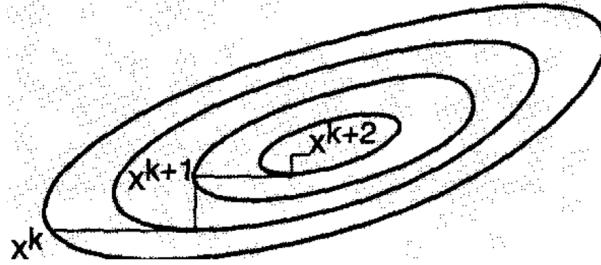


图 7.6: Illustration of the coordinate descent method.

Theorem 688 shows that $\bar{\mathbf{x}}^K$ is by $O\left(1/\sum_{k=0}^K \beta_k^{-1}\right)$ from being an optimal solution. It holds for both bounded and unbounded $\{\beta_k\}$. In the bounded case, $O\left(1/\sum_{k=0}^K \beta_k^{-1}\right)$ is simply $O(1/K)$. Theorem 688 also hints that $\sum_{k=0}^K \beta_k^{-1}$ should approach infinity to guarantee the convergence of PLADMPSAP. However, that $\bar{\mathbf{x}}^K$ converges to the optimal solution may not imply that the sequence $\{\mathbf{x}_k\}$ converges to the optimal solution. Neither is the convergence rate. Therefore, a non-ergodic convergence rate, i.e., measuring the optimality of \mathbf{x}_k directly, is desired. The result on $O(1/K)$ non-ergodic convergence rate, which is proven optimal for LADM with Nesterov's acceleration technique, can be found in [76, 90].

For more thorough discussions on PLADMPSAP, such as on problems with convex set constraints and when f_i 's all have bounded subgradients (such as norms), please see [82].

7.6 Coordinate descent and block coordinate descent

(Taken from Sections 1.8.1 and 2.7 of [12])

There are several nonderivative methods for minimizing functions. A particularly important one is the *coordinate descent method*. Here the cost is minimized along one coordinate direction at each iteration. The order in which coordinates are chosen may vary in the course of the algorithm. In the case where this order is cyclical, given \mathbf{x}^k , the i -th coordinate of \mathbf{x}^{k+1} is determined by

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}} f(x_1^{k+1}, x_2^{k+1}, \dots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \dots, x_n^k); \quad (7.194)$$

see Figure 7.6. The method can also be used for minimizing f subject to upper and lower bounds on the variables x_i . We will analyze the method within this more general context in the description of block coordinate descent.

An important advantage of the coordinate descent method is that it is well suited for *parallel computation*. In particular, suppose that there is a subset of coordinates $x_{i_1}, x_{i_2}, \dots, x_{i_m}$, which are not coupled through the cost function, that is, $f(\mathbf{x})$ can be written as $\sum_{r=1}^m f_{i_r}(\mathbf{x})$, where for each r , $f_{i_r}(\mathbf{x})$ does not depend on the coordinates x_{i_s} for all $s \neq r$. Then one can perform the m coordinate descent iterations

$$x_{i_r}^{k+1} = \underset{\xi}{\operatorname{argmin}} f_{i_r}(\mathbf{x}^k + \xi \mathbf{e}_{i_r}), \quad r = 1, \dots, m,$$

independently and in parallel. Thus, in problems with special structure where the set of coordinates can be partitioned into p subsets with the independence property just described, one can perform a full cycle of coordinate descent iterations in p (as opposed to n) parallel steps (assuming of course that a sufficient number of parallel processors is available).

The coordinate descent method generally has similar convergence properties to steepest descent. For continuously differentiable functions, it can be shown to generate sequences whose limit points are stationary, although the proof of this is sometimes complicated and requires some additional assumptions (see Proposition 689, which deals with a constrained version of coordinate descent and requires that the minimum of the objective function along each coordinate is uniquely attained.). There is also a great deal of analysis of coordinate descent in a context where its use is particularly favorable, namely in solving dual problems. Within this context, the unique attainment assumption of Proposition 689 is neither satisfied nor is it essential. The convergence rate of coordinate descent to nonsingular and singular local minima can be shown to be linear and sublinear, respectively, similar to steepest descent. Often, the choice between coordinate descent and steepest descent is dictated by the structure of the objective function. Both methods can be very slow, but for many practical contexts, they can be quite effective.

We now generalize unconstrained coordinate descent methods to solve the problem:

$$\begin{aligned} & \min_{\mathbf{x}} f(\mathbf{x}), \\ & \text{s.t. } \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{7.195}$$

where \mathcal{X} is a Cartesian product of closed convex sets $\mathcal{X}_1, \dots, \mathcal{X}_m$:

$$\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \dots \times \mathcal{X}_m. \tag{7.196}$$

We assume that \mathcal{X}_i is a closed convex subset of \mathbb{R}^{n_i} and $n = n_1 + \dots + n_m$. The vector \mathbf{x} is partitioned as

$$\mathbf{x} = (\mathbf{x}_1^T, \mathbf{x}_2^T, \dots, \mathbf{x}_m^T)^T,$$

where each \mathbf{x}_i belong to \mathbb{R}^{n_i} , so the constraint $\mathbf{x} \in \mathcal{X}$ is equivalent to

$$\mathbf{x}_i \in \mathcal{X}_i, \quad i = 1, \dots, m.$$

Let us assume that for every $\mathbf{x} \in \mathcal{X}$ and every $i = 1, \dots, m$, the optimization problem:

$$\begin{aligned} & \min_{\mathbf{x}} f(\mathbf{x}_1, \dots, \mathbf{x}_{i-1}, \boldsymbol{\xi}, \mathbf{x}_{i+1}, \dots, \mathbf{x}_m), \\ & \text{s.t. } \boldsymbol{\xi} \in \mathcal{X}_i, \end{aligned}$$

has at least one solution. The following algorithm, known as *block coordinate descent* or *nonlinear Gauss-Seidel* method, generates the next iterate $\mathbf{x}^{k+1} = (\mathbf{x}_1^{k+1}, \mathbf{x}_2^{k+1}, \dots, \mathbf{x}_m^{k+1})^T$, given the current iterate $\mathbf{x}^k = (\mathbf{x}_1^k, \mathbf{x}_2^k, \dots, \mathbf{x}_m^k)^T$, according to the iteration

$$\mathbf{x}_i^{k+1} = \underset{\boldsymbol{\xi} \in \mathcal{X}_i}{\operatorname{argmin}} f(\mathbf{x}_1^{k+1}, \dots, \mathbf{x}_{i-1}^{k+1}, \boldsymbol{\xi}, \mathbf{x}_{i+1}^k, \dots, \mathbf{x}_m^k), \quad i = 1, \dots, m. \quad (7.197)$$

Thus, at each iteration, the cost is minimized with respect to each of the “block coordinate” vector \mathbf{x}_i^k , taken in cyclic order. Naturally, the method makes practical sense if the minimization in (7.197) is fairly easily. This is frequently so when each \mathbf{x}_i is a scalar, but there are also other cases of interest, where \mathbf{x}_i is a multi-dimensional vector.

The following proposition gives the basic convergence result for the method. It turns out that it is necessary to make an assumption implying that the minimum in (7.197) is uniquely attained. The need for this assumption is not obvious but has been demonstrated by an example given by Powell. Exercise 692 provides a modified version of the algorithm, which does not require this assumption. Moreover, the differentiability is also necessary (show an counter-example).

Proposition 689 (Convergence of Block Coordinate Descent). *Suppose that f is continuously differentiable over the set \mathcal{X} of equation (7.196). Furthermore, suppose that for each i and $\mathbf{x} \in \mathcal{X}$, the minimum below*

$$\min_{\boldsymbol{\xi} \in \mathcal{X}_i} f(\mathbf{x}_1, \dots, \mathbf{x}_{i-1}, \boldsymbol{\xi}, \mathbf{x}_{i+1}, \dots, \mathbf{x}_m)$$

is uniquely attained. Let $\{\mathbf{x}^k\}$ be the sequence generated by the block coordinate descent method (7.197). Then every accumulate point of $\{\mathbf{x}^k\}$ is a stationary point.

Proof. Let

$$\mathbf{z}_i^k = (\mathbf{x}_1^{k+1,T}, \dots, \mathbf{x}_i^{k+1,T}, \mathbf{x}_{i+1}^{k,T}, \dots, \mathbf{x}_m^{k,T})^T.$$

Using the definition (7.197) of the method, we obtain

$$f(\mathbf{x}^k) \geq f(\mathbf{z}_1^k) \geq f(\mathbf{z}_2^k) \geq \dots \geq f(\mathbf{z}_{m-1}^k) \geq f(\mathbf{x}^{k+1}), \quad \forall k. \quad (7.198)$$

Let $\bar{\mathbf{x}} = (\bar{\mathbf{x}}_1^T, \dots, \bar{\mathbf{x}}_m^T)^T$ be a limit point of the sequence $\{\mathbf{x}^k\}$. Notice that $\bar{\mathbf{x}} \in \mathcal{X}$ because \mathcal{X} is closed. Equation (7.198) implies that the sequence $\{f(\mathbf{x}^k)\}$ converges to $f(\bar{\mathbf{x}})$. It now remains to show that $\bar{\mathbf{x}}$ minimizes f over \mathcal{X} .

Let $\{\mathbf{x}^{k_j} | j = 0, 1, \dots\}$ be a subsequence of $\{\mathbf{x}^k\}$ that converges to $\bar{\mathbf{x}}$. We first show that $\{\mathbf{x}_1^{k_j+1} - \mathbf{x}_1^{k_j}\}$ converges to zero as $j \rightarrow \infty$. Assume the contrary, or equivalently, that $\{\mathbf{z}_1^{k_j} - \mathbf{x}_1^{k_j}\}$ does not converge to zero. Let $\gamma^{k_j} = \|\mathbf{z}_1^{k_j} - \mathbf{x}_1^{k_j}\|$. By possibly restricting to a subsequence of $\{k_j\}$, we may assume that there exists some $\bar{\gamma} > 0$ such that $\gamma^{k_j} \geq \bar{\gamma}$ for all j . Let $\mathbf{s}_1^{k_j} = (\mathbf{z}_1^{k_j} - \mathbf{x}_1^{k_j})/\gamma^{k_j}$. Thus, $\mathbf{z}_1^{k_j} = \mathbf{x}_1^{k_j} + \gamma^{k_j} \mathbf{s}_1^{k_j}$, $\|\mathbf{s}_1^{k_j}\| = 1$, and $\mathbf{s}_1^{k_j}$ differs from zero only along the first block-component. Notice that $\mathbf{s}_1^{k_j}$ belongs to a compact set and therefore has a limit point $\bar{\mathbf{s}}_1$. By restricting to a further subsequence of $\{k_j\}$, we assume that $\mathbf{s}_1^{k_j}$ converges to $\bar{\mathbf{s}}_1$.

Let us fix some $\varepsilon \in [0, 1]$. Notice that $0 \leq \varepsilon \bar{\gamma} \leq \gamma^{k_j}$. Therefore, $\mathbf{x}_1^{k_j} + \varepsilon \bar{\gamma} \mathbf{s}_1^{k_j}$ lies on the segment joining $\mathbf{x}_1^{k_j}$ and $\mathbf{x}_1^{k_j} + \gamma^{k_j} \mathbf{s}_1^{k_j} = \mathbf{z}_1^{k_j}$, and belongs to \mathcal{X} because \mathcal{X} is convex. Using the fact that $\mathbf{z}_1^{k_j}$ minimizes f over all \mathbf{x} that differ from $\mathbf{x}_1^{k_j}$ along the first block-component, we obtain

$$f(\mathbf{z}_1^{k_j}) = f(\mathbf{x}_1^{k_j} + \gamma^{k_j} \mathbf{s}_1^{k_j}) \leq f(\mathbf{x}_1^{k_j} + \varepsilon \bar{\gamma} \mathbf{s}_1^{k_j}) \leq f(\mathbf{x}_1^{k_j}).$$

Since $f(\mathbf{x}^k)$ converges to $f(\bar{\mathbf{x}})$, Eq. (7.198) shows that $f(\mathbf{z}_1^k)$ also converges to $f(\bar{\mathbf{x}})$. We now take the limit as j tends to infinity, to obtain $f(\bar{\mathbf{x}}) \leq f(\bar{\mathbf{x}} + \varepsilon \bar{\gamma} \bar{\mathbf{s}}_1) \leq f(\bar{\mathbf{x}})$. We conclude that $f(\bar{\mathbf{x}}) = f(\bar{\mathbf{x}} + \varepsilon \bar{\gamma} \bar{\mathbf{s}}_1)$, for every $\varepsilon \in [0, 1]$. Since $\bar{\gamma} \bar{\mathbf{s}}_1 \neq \mathbf{0}$, this contradicts the assumption that f is uniquely minimized when viewed as a function of the first block-component. This contradiction establishes that $\mathbf{x}_1^{k_j+1} - \mathbf{x}_1^{k_j}$ converges to zero. In particular, $\mathbf{z}_1^{k_j}$ converges to $\bar{\mathbf{x}}$.

From the definition (7.197) of the algorithm, we have

$$f(\mathbf{z}_1^{k_j}) \leq f(\mathbf{x}_1, \mathbf{x}_2^{k_j}, \dots, \mathbf{x}_m^{k_j}), \quad \forall \mathbf{x}_1 \in \mathcal{X}_1.$$

Taking the limit as j tends to infinity, we obtain

$$f(\bar{\mathbf{x}}) \leq f(\mathbf{x}_1, \bar{\mathbf{x}}_2, \dots, \bar{\mathbf{x}}_m), \quad \forall \mathbf{x}_1 \in \mathcal{X}_1.$$

Using the conditions for optimality over a convex set (Proposition 2.1.2 in Section 2.1 of [12]), we conclude that

$$\langle \nabla_1 f(\bar{\mathbf{x}}), \mathbf{x}_1 - \bar{\mathbf{x}}_1 \rangle \geq 0, \quad \forall \mathbf{x}_1 \in \mathcal{X}_1,$$

where $\nabla_i f$ denotes the gradient of f with respect to the component \mathbf{x}_i .

Let us now consider the sequence $\{\mathbf{z}_1^{k_j}\}$. We have already shown that $\mathbf{z}_1^{k_j}$ converges to $\bar{\mathbf{x}}$. A verbatim repetition of the preceding argument shows that $\mathbf{x}_2^{k_j+1} - \mathbf{x}_2^{k_j}$ converges

to zero and $\langle \nabla_2 f(\bar{\mathbf{x}}), \mathbf{x}_2 - \bar{\mathbf{x}}_2 \rangle \geq 0$ for all $\mathbf{x}_2 \in \mathcal{X}_2$. Continuing inductively, we obtain $\langle \nabla_i f(\bar{\mathbf{x}}), \mathbf{x}_i - \bar{\mathbf{x}}_i \rangle \geq 0$ for all $\mathbf{x}_i \in \mathcal{X}_i$ and for every i . Adding these inequalities, and using the Cartesian product structure of the set \mathcal{X} , we conclude that $\langle \nabla f(\bar{\mathbf{x}}), \mathbf{x} - \bar{\mathbf{x}} \rangle \geq 0$ for all $\mathbf{x} \in \mathcal{X}$. \square

Block coordinate descent methods are often useful in contexts where the objective function and the constraints have a partially decomposable structure with respect to the problem's optimization variables. The following example illustrates the idea.

Example 690 (Hierarchical Decomposition). *Consider an optimization problem of the form*

$$\begin{aligned} \min_{\mathbf{x}, \{\mathbf{y}_i\}} \quad & \sum_{i=1}^m f_i(\mathbf{x}, \mathbf{y}_i), \\ \text{s.t. } \mathbf{x} \in \mathcal{X}, \mathbf{y}_i \in \mathcal{Y}_i, i = 1, \dots, m, \end{aligned}$$

where \mathcal{X} and \mathcal{Y}_i , $i = 1, \dots, m$, are closed, convex subsets of corresponding Euclidean spaces, and the functions f_i are continuously differentiable. This problem is associated with a paradigm of optimization of a system consisting of m subsystems, with the cost function f_i associated with the operations of the i th subsystem. Here \mathbf{y}_i is viewed as vectors of local decision variables that influences the cost of the i th subsystem only, and \mathbf{x} is viewed as a vector of global or coordinating decision variables that affects the operation of all the subsystems.

The coordinate descent method takes advantage of the decomposable structure and has the form

$$\begin{aligned} \mathbf{y}_i^{k+1} &= \underset{\mathbf{y}_i \in \mathcal{Y}_i}{\operatorname{argmin}} f_i(\mathbf{x}^k, \mathbf{y}_i), \quad i = 1, \dots, m, \\ \mathbf{x}^{k+1} &= \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmin}} \sum_{i=1}^m f_i(\mathbf{x}, \mathbf{y}_i^{k+1}). \end{aligned}$$

The method has a natural real-life interpretation: at each iteration, each subsystem optimizes its own cost, taking the global variables as fixed at their current values, and then the coordinator optimizes the overall cost for the current values of the local variables.

7.7 Exercises

(Taken from Section 2.7 of [12])

Exercise 691 (The Proximal Minimization Algorithm). *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a continuous differentiable function, let \mathcal{X} be a closed convex set, and let c be a positive scalar.*

(a) Show that the algorithm

$$\mathbf{x}^{k+1} = \operatorname{argmin}_{\mathbf{x} \in \mathcal{X}} \left\{ f(\mathbf{x}) + \frac{1}{2c} \|\mathbf{x} - \mathbf{x}^k\|^2 \right\}$$

is a special case of the block coordinate descent method applied to the problem

$$\begin{aligned} & \min_{\mathbf{x}, \mathbf{y}} f(\mathbf{x}) + \frac{1}{2c} \|\mathbf{x} - \mathbf{y}\|^2, \\ & \text{s.t. } \mathbf{x} \in \mathcal{X}, \mathbf{y} \in \mathbb{R}^n, \end{aligned}$$

which is equivalent to the problem of minimizing f over \mathcal{X} .

- Derive a convergence result based on Proposition 689 for the algorithm of part (a).
- Assume that f is convex. Show that if f has at least one minimizing point over \mathcal{X} , the entire sequence $\{\mathbf{x}^k\}$ converges to some such point. Hint: we have by definition

$$f(\mathbf{x}^{k+1}) + \frac{1}{2c} \|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 \leq f(\mathbf{x}) + \frac{1}{2c} \|\mathbf{x} - \mathbf{x}^k\|^2, \quad \forall \mathbf{x} \in \mathcal{X},$$

so that $\|\mathbf{x}^{k+1} - \mathbf{x}^k\| \leq \|\mathbf{x} - \mathbf{x}^k\|$ for all \mathbf{x} in the set

$$\mathcal{X}^k = \{\mathbf{x} \in \mathcal{X} | f(\mathbf{x}) \leq f(\mathbf{x}^{k+1})\}.$$

Hence \mathbf{x}^{k+1} is the unique projection of \mathbf{x}^k onto \mathcal{X}^k , and we have $\langle \mathbf{x}^{k+1} - \mathbf{x}^k, \mathbf{x} - \mathbf{x}^k \rangle \geq 0$ for all $\mathbf{x} \in \mathcal{X}^k$. Conclude that for every \mathbf{x}^* that minimizes f over \mathcal{X} , we have $\|\mathbf{x}^* - \mathbf{x}^{k+1}\| \leq \|\mathbf{x}^* - \mathbf{x}^k\|$.

Exercise 692. Consider the following variation of the method (7.197):

$$\mathbf{x}_i^{k+1} = \operatorname{argmin}_{\xi \in \mathcal{X}_i} f(\mathbf{x}_1^{k+1}, \dots, \mathbf{x}_{i-1}^{k+1}, \xi, \mathbf{x}_{i+1}^k, \dots, \mathbf{x}_m^k) + \frac{1}{2c} \|\xi - \mathbf{x}_i^k\|^2, \quad i = 1, \dots, m,$$

where c is a positive scalar. Assuming that f is convex, show that every accumulation point of the sequence $\{\mathbf{x}^k\}$ is a global minimum. Hint: Apply the result of Proposition 689 to the objective function

$$g(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) + \frac{1}{2c} \|\mathbf{x} - \mathbf{y}\|^2.$$

Exercise 693 (Parallel Projections Algorithm). We are given m closed convex sets $\mathcal{X}_1, \mathcal{X}_m$ in \mathbb{R}^n , and we want to find a point in their intersection. Consider the equivalent problem

$$\min_{\mathbf{x}, \{\mathbf{y}_i\}} \frac{1}{2} \sum_{i=1}^m \|\mathbf{x} - \mathbf{y}_i\|^2,$$

$$\text{s.t. } \mathbf{x} \in \mathbb{R}^n, \mathbf{y}_i \in \mathcal{X}_i, i = 1, \dots, m.$$

Derive a block coordinate descent algorithm involving projections on each of the sets \mathcal{X}_i that can be carried out independently for each other. State a convergence result for this algorithm.

表 7.1: Comparison between APG and ADM on the Relaxed RPCA problem, adapted from [81]. We present typical running times for randomly generated matrices. Corresponding to each triplet $\{m, \text{rank}(\mathbf{A}_0), \|\mathbf{E}_0\|_0\}$, the Relaxed RPCA problem was solved for the same data matrix \mathbf{D} using two different algorithms. For APG and ADM, the number of SVDs is equal to the number of iterations.

m	algorithm	$\frac{\ \mathbf{A}^* - \mathbf{A}_0\ _F}{\ \mathbf{A}^*\ _F}$	rank(\mathbf{A}^*)	$\ \mathbf{E}^*\ _0$	#SVD	time (s)
$\text{rank}(\mathbf{A}_0) = 0.05 m, \ \mathbf{E}_0\ _0 = 0.05 m^2$						
500	APG	1.12e-5	25	12,542	127	11.01
	ADM	5.21e-7	25	12,499	20	1.72
800	APG	9.84e-6	40	32,092	126	37.21
	ADM	3.29e-7	40	31,999	21	5.87
1,000	APG	8.79e-6	50	50,082	126	57.62
	ADM	2.67e-7	50	49,999	22	10.13
1,500	APG	7.16e-6	75	112,659	126	163.80
	ADM	1.86e-7	75	112,500	22	30.80
2,000	APG	6.27e-6	100	200,243	126	353.63
	ADM	9.54e-8	100	200,000	22	68.69
3,000	APG	5.20e-6	150	450,411	126	1,106.22
	ADM	1.49e-7	150	449,993	22	212.34
$\text{rank}(\mathbf{A}_0) = 0.05 m, \ \mathbf{E}_0\ _0 = 0.10 m^2$						
500	APG	1.41e-5	25	25,134	129	14.35
	ADM	9.31e-7	25	25,000	21	2.52
800	APG	1.12e-5	40	64,236	129	37.94
	ADM	4.87e-7	40	64,000	24	6.69
1,000	APG	9.97e-6	50	100,343	129	65.41
	ADM	3.78e-7	50	99,996	22	10.77
1,500	APG	8.18e-6	75	225,614	129	163.36
	ADM	2.79e-7	75	224,996	23	35.71
2,000	APG	7.11e-6	100	400,988	129	353.30
	ADM	3.31e-7	100	399,993	23	70.33
3,000	APG	5.79e-6	150	901,974	129	1,110.76
	ADM	2.27e-7	150	899,980	23	217.39

表 7.2: Comparison between APG and ADM on the Relaxed RPCA problem, adapted from [81]. Continued from Table 7.1 with different parameters of $\{m, \text{rank}(\mathbf{A}_0), \|\mathbf{E}_0\|_0\}$.

m	algorithm	$\frac{\ \mathbf{A}^* - \mathbf{A}_0\ _F}{\ \mathbf{A}^*\ _F}$	rank(\mathbf{A}^*)	$\ \mathbf{E}^*\ _0$	#SVD	time (s)
$\text{rank}(\mathbf{A}_0) = 0.10m, \ \mathbf{E}_0\ _0 = 0.05m^2$						
500	APG	9.36e-6	50	13,722	129	13.99
	ADM	6.05e-7	50	12,500	22	2.32
800	APG	7.45e-6	80	34,789	129	67.54
	ADM	3.08e-7	80	32,000	22	10.81
1,000	APG	6.64e-6	100	54,128	129	129.40
	ADM	2.61e-7	100	50,000	22	20.71
1,500	APG	5.43e-6	150	121,636	129	381.52
	ADM	1.76e-7	150	112,496	24	67.84
2,000	APG	4.77e-6	200	215,874	129	888.93
	ADM	2.49e-7	200	199,998	23	150.35
3,000	APG	3.98e-6	300	484,664	129	2,923.90
	ADM	1.30e-7	300	450,000	23	485.70
$\text{rank}(\mathbf{A}_0) = 0.10m, \ \mathbf{E}_0\ _0 = 0.10m^2$						
500	APG	9.78e-6	50	27,478	133	13.90
	ADM	7.64e-7	50	25,000	25	2.62
800	APG	8.66e-6	80	70,384	132	68.12
	ADM	4.77e-7	80	64,000	25	11.88
1,000	APG	7.75e-6	100	109,632	132	130.37
	ADM	3.73e-7	100	99,999	25	22.95
1,500	APG	6.31e-6	150	246,187	132	383.28
	ADM	5.42e-7	150	224,998	24	66.78
2,000	APG	5.49e-6	200	437,099	132	884.86
	ADM	4.27e-7	200	399,999	24	154.27
3,000	APG	4.50e-6	300	980,933	132	2,915.40
	ADM	3.39e-7	300	899,990	24	503.05

表 7.3: Comparison among APG, ADM, LADMAP, and LADMAP(A) on the synthetic data, adapted from [79]. For each quadruple (s, p, d, \tilde{r}) , the Relaxed LRR problem, with regularization parameter $\lambda = 0.1$, was solved for the same data using different algorithms. We present typical running time (in $\times 10^3$ seconds), iteration number, relative error (%) of output solution $(\mathbf{Z}^*, \mathbf{E}^*)$ and the clustering accuracy (%) of tested algorithms, respectively.

Size (s, p, d, \tilde{r})	Method	Time	#Iter.	$\frac{\ \mathbf{Z}^* - \mathbf{Z}_0\ }{\ \mathbf{Z}_0\ }$	$\frac{\ \mathbf{E}^* - \mathbf{E}_0\ }{\ \mathbf{E}_0\ }$	Acc.
$(10, 20, 200, 5)$	APG	0.0332	110	2.2079	1.5096	81.5
	ADM	0.0529	176	0.5491	0.5093	90.0
	LADMAP	0.0145	46	0.5480	0.5024	90.0
	LADMAP(A)	0.0010	46	0.5480	0.5024	90.0
$(15, 20, 300, 5)$	APG	0.0869	106	2.4824	1.0341	80.0
	ADM	0.1526	185	0.6519	0.4078	83.7
	LADMAP	0.0336	41	0.6518	0.4076	86.7
	LADMAP(A)	0.0015	41	0.6518	0.4076	86.7
$(20, 25, 500, 5)$	APG	1.8837	117	2.8905	2.4017	72.4
	ADM	3.7139	225	1.1191	1.0170	80.0
	LADMAP	0.7762	40	0.6379	0.4268	84.6
	LADMAP(A)	0.0053	40	0.6379	0.4268	84.6
$(30, 30, 900, 5)$	APG	6.1252	116	3.0667	0.9199	69.4
	ADM	11.7185	220	0.6865	0.4866	76.0
	LADMAP	2.3891	44	0.6864	0.4294	80.1
	LADMAP(A)	0.0058	44	0.6864	0.4294	80.1

Draft

第八章 Randomized Algorithms

(Taken from Chapter 14.3-14.5 of [119])

8.1 Stochastic Gradient Descent (SGD)

In stochastic gradient descent we do not require the update direction to be based exactly on the gradient. Instead, we allow the direction to be a random vector and only require that its *expected value* at each iteration will equal the gradient direction. Or, more generally, we require that the expected value of the random vector will be a subgradient of the function at the current vector.

An illustration of stochastic gradient descent versus gradient descent is given in Figure 8.1. As we will see in Section 8.3, in the context of learning problems, it is easy to find a random vector whose expectation is a subgradient of the risk function.

Stochastic Gradient Descent (SGD) for minimizing $f(\mathbf{w})$

parameters: Scalar $\eta > 0$, integer $T > 0$

parameters: $\mathbf{w}^{(1)} = \mathbf{0}$

for $t = 1, 2, \dots, T$

choose \mathbf{v}_t at random from a distribution such that $\mathbb{E}[\mathbf{v}_t | \mathbf{w}^{(t)}] \in \partial f(\mathbf{w}^{(t)})$

update $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \mathbf{v}_t$

output $\bar{\mathbf{w}} = \frac{1}{T} \sum_{t=1}^T \mathbf{w}^{(t)}$.

8.1.1 Analysis of SGD for Convex-Lipschitz-Bounded Functions

Recall the bound we achieved for the GD algorithm in Corollary 14.2 of [119]. For the stochastic case, in which only the expectation of \mathbf{v}_t is in $\partial f(\mathbf{w}^{(t)})$, we cannot directly apply

$$f(\mathbf{w}^{(t)}) - f(\mathbf{w}^*) \leq \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle. \quad (8.1)$$

However, since the expected value of \mathbf{v}_t is a subgradient of f at $\mathbf{w}^{(t)}$, we can still derive a similar bound on the *expected* output of stochastic gradient descent. This is formalized in the following theorem.

Theorem 694. Let $B, \rho > 0$. Let f be a convex function and let $\mathbf{w}^* \in \arg \min_{\mathbf{w}: \|\mathbf{w}\| \leq B} f(\mathbf{w})$. Assume that SGD is run for T iterations with $\eta = \sqrt{\frac{B^2}{\rho^2 T}}$. Assume also that for all t ,

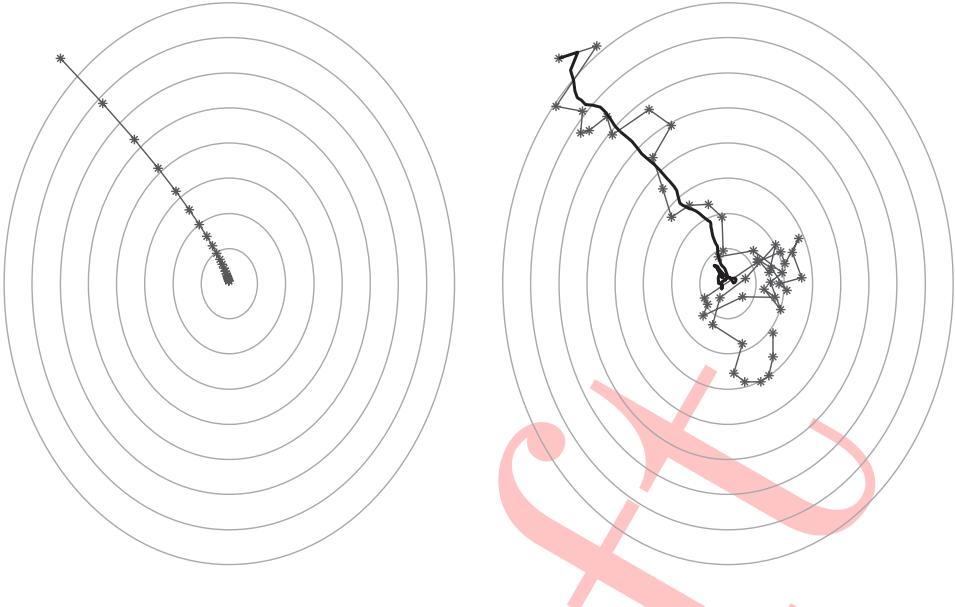


图 8.1: An illustration of the gradient descent algorithm (left) and the stochastic gradient descent algorithm (right). The function to be minimized is $1.25(x + 6)^2 + (y - 8)^2$. For the stochastic case, the solid line depicts the averaged value of \mathbf{w} .

$\|\mathbf{v}_t\| \leq \rho$ with probability 1. Then,

$$\mathbb{E}[f(\bar{\mathbf{w}})] - f(\mathbf{w}^*) \leq \frac{B\rho}{\sqrt{T}},$$

where $\bar{\mathbf{w}} = \frac{1}{T} \sum_{t=1}^T \mathbf{w}^{(t)}$. Therefore, for any $\epsilon > 0$, to achieve $\mathbb{E}[f(\bar{\mathbf{w}})] - f(\mathbf{w}^*) \leq \epsilon$, it suffices to run the SGD algorithm for a number of iterations that satisfies

$$T \geq \frac{B^2 \rho^2}{\epsilon^2}.$$

To prove Theorem 694, we introduce the following lemma first.

Lemma 695. Let $\mathbf{v}_1, \dots, \mathbf{v}_T$ be an arbitrary sequence of vectors. Any algorithm with an initialization $\mathbf{w}^{(1)} = \mathbf{0}$ and an update rule of the form

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \mathbf{v}_t \quad (8.2)$$

satisfies

$$\sum_{t=1}^T \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle \leq \frac{\|\mathbf{w}^*\|^2}{2\eta} + \frac{\eta}{2} \sum_{t=1}^T \|\mathbf{v}_t\|^2. \quad (8.3)$$

In particular, for every $B, \rho > 0$, if for all t we have that $\|\mathbf{v}_t\| \leq \rho$ and if we set $\eta = \frac{B}{\rho\sqrt{T}}$, then for every \mathbf{w}^* with $\|\mathbf{w}^*\| \leq B$ we have

$$\sum_{t=1}^T \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle \leq \frac{B\rho}{\sqrt{T}}$$

Proof. Using the algebraic manipulations (completing the square), we obtain

$$\begin{aligned}\langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle &= \frac{1}{\eta} \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \eta \mathbf{v}_t \rangle \\ &= \frac{1}{2\eta} (-\|\mathbf{w}^{(t)} - \mathbf{w}^* - \eta \mathbf{v}_t\|^2 + \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2 + \eta^2 \|\mathbf{v}_t\|^2) \\ &= \frac{1}{2\eta} (-\|\mathbf{w}^{(t+1)} - \mathbf{w}^*\|^2 + \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2) + \frac{\eta}{2} \|\mathbf{v}_t\|^2,\end{aligned}$$

where the last equality follows from the definition of the update rule. Summing the equality over t , we have

$$\sum_{t=1}^T \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle = \frac{1}{2\eta} \sum_{t=1}^T (-\|\mathbf{w}^{(t+1)} - \mathbf{w}^*\|^2 + \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2) + \frac{\eta}{2} \sum_{t=1}^T \|\mathbf{v}_t\|^2. \quad (8.4)$$

The first sum on the right-hand side is a telescopic sum that collapses to

$$-\|\mathbf{w}^{(T+1)} - \mathbf{w}^*\|^2 + \|\mathbf{w}^{(1)} - \mathbf{w}^*\|^2.$$

Plugging this in Equation (8.4), we have

$$\begin{aligned}\sum_{t=1}^T \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle &= \frac{1}{2\eta} (-\|\mathbf{w}^{(T+1)} - \mathbf{w}^*\|^2 + \|\mathbf{w}^{(1)} - \mathbf{w}^*\|^2) + \frac{\eta}{2} \sum_{t=1}^T \|\mathbf{v}_t\|^2 \\ &\leq \frac{1}{2\eta} \|\mathbf{w}^{(1)} - \mathbf{w}^*\|^2 + \frac{\eta}{2} \sum_{t=1}^T \|\mathbf{v}_t\|^2 \\ &= \frac{1}{2\eta} \|\mathbf{w}^*\|^2 + \frac{\eta}{2} \sum_{t=1}^T \|\mathbf{v}_t\|^2,\end{aligned}$$

where the last equality is due to the definition $\mathbf{w}^{(1)} = \mathbf{0}$. This proves the first part of the lemma (Equation (8.3)). The second part follows by upper bounding $\|\mathbf{w}^*\|$ by B , $\|\mathbf{v}_t\|$ by ρ , dividing by T , and plugging in the value of η . \square

Now we prove Theorem 694.

Proof. Let us introduce the notation $\mathbf{v}_{1:t}$ to denote the sequence $\mathbf{v}_1, \dots, \mathbf{v}_t$. Taking expectation of the following inequality

$$\begin{aligned}f(\bar{\mathbf{w}}) - f(\mathbf{w}^*) &= f\left(\frac{1}{T} \sum_{t=1}^T \mathbf{w}^{(t)}\right) - f(\mathbf{w}^*) \\ &\leq \frac{1}{T} \sum_{t=1}^T f(\mathbf{w}^{(t)}) - f(\mathbf{w}^*) \\ &= \frac{1}{T} \sum_{t=1}^T (f(\mathbf{w}^{(t)}) - f(\mathbf{w}^*)),\end{aligned} \quad (8.5)$$

we obtain

$$\mathbb{E}_{\mathbf{v}_{1:T}}[f(\bar{\mathbf{w}}) - f(\mathbf{w}^*)] \leq \mathbb{E}_{\mathbf{v}_{1:T}} \left[\frac{1}{T} \sum_{t=1}^T (f(\mathbf{w}^{(t)}) - f(\mathbf{w}^*)) \right].$$

Since Lemma 695 holds for any sequence $\mathbf{v}_1, \dots, \mathbf{v}_t$, it applies to SGD as well. By taking expectation of the bound in the lemma we have

$$\mathbb{E}_{\mathbf{v}_{1:T}} \left[\frac{1}{T} \sum_{t=1}^T \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle \right] \leq \frac{B\rho}{\sqrt{T}}. \quad (8.6)$$

It is left to show that

$$\mathbb{E}_{\mathbf{v}_{1:T}} \left[\frac{1}{T} \sum_{t=1}^T (f(\mathbf{w}^{(t)}) - f(\mathbf{w}^*)) \right] \leq \mathbb{E}_{\mathbf{v}_{1:T}} \left[\frac{1}{T} \sum_{t=1}^T \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle \right], \quad (8.7)$$

which we will hereby prove.

Using the linearity of the expectation we have

$$\mathbb{E}_{\mathbf{v}_{1:T}} \left[\frac{1}{T} \sum_{t=1}^T \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle \right] = \frac{1}{T} \sum_{t=1}^T \mathbb{E}_{\mathbf{v}_{1:T}} [\langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle].$$

Next, we recall the *law of total expectation*: For every two random variables α, β , and a function g , $\mathbb{E}_\alpha[g(\alpha)] = \mathbb{E}_\beta \mathbb{E}_\alpha[g(\alpha)|\beta]$. Setting $\alpha = \mathbf{v}_{1:t}$ and $\beta = \mathbf{v}_{1:t-1}$ we get that

$$\begin{aligned} \mathbb{E}_{\mathbf{v}_{1:T}} [\langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle] &= \mathbb{E}_{\mathbf{v}_{1:t}} [\langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle] \\ &= \mathbb{E}_{\mathbf{v}_{1:t-1}} \mathbb{E}_{\mathbf{v}_{1:t}} [\langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle | \mathbf{v}_{1:t-1}]. \end{aligned}$$

Once we know $\mathbf{v}_{1:t-1}$, the value of $\mathbf{w}^{(t)}$ is not random any more and therefore

$$\mathbb{E}_{\mathbf{v}_{1:t-1}} \mathbb{E}_{\mathbf{v}_{1:t}} [\langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle | \mathbf{v}_{1:t-1}] = \mathbb{E}_{\mathbf{v}_{1:t-1}} \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbb{E}_{\mathbf{v}_t} [\mathbf{v}_t | \mathbf{v}_{1:t-1}] \rangle.$$

Since $\mathbf{w}^{(t)}$ only depends on $\mathbf{v}_{1:t-1}$ and SGD requires that $\mathbb{E}_{\mathbf{v}_t} [\mathbf{v}_t | \mathbf{w}^{(t)}] \in \partial f(\mathbf{w}^{(t)})$ we obtain that $\mathbb{E}_{\mathbf{v}_t} [\mathbf{v}_t | \mathbf{v}_{1:t-1}] \in \partial f(\mathbf{w}^{(t)})$. Thus,

$$\mathbb{E}_{\mathbf{v}_{1:t-1}} \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbb{E}_{\mathbf{v}_t} [\mathbf{v}_t | \mathbf{v}_{1:t-1}] \rangle \geq \mathbb{E}_{\mathbf{v}_{1:t-1}} [f(\mathbf{w}^{(t)}) - f(\mathbf{w}^*)].$$

Overall, we have shown that

$$\begin{aligned} \mathbb{E}_{\mathbf{v}_{1:T}} [\langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle] &\geq \mathbb{E}_{\mathbf{v}_{1:t-1}} [f(\mathbf{w}^{(t)}) - f(\mathbf{w}^*)] \\ &= \mathbb{E}_{\mathbf{v}_{1:T}} [f(\mathbf{w}^{(t)}) - f(\mathbf{w}^*)]. \end{aligned}$$

Summing over, dividing by T , and using the linearity of expectation, we get that Equation (8.7) holds, which concludes our proof. \square

8.2 Variants

In this section we describe several variants of Stochastic Gradient Descent.

8.2.1 Adding a Projection Step

In the previous analyses of the GD and SGD algorithms, we required that the norm of \mathbf{w}^* will be at most B , which is equivalent to requiring that \mathbf{w}^* is in the set $\mathcal{H} = \{\mathbf{w} : \|\mathbf{w}\| \leq B\}$. In terms of learning, this means restricting ourselves to a B -bounded hypothesis class. Yet any step we take in the opposite direction of the gradient (or its expected direction) might result in stepping out of this bound, and there is even no guarantee that $\hat{\mathbf{w}}$ satisfies it. We show in the following how to overcome this problem while maintaining the same convergence rate.

The basic idea is to add a *projection step*; namely, we will now have a two-step update rule, where we first subtract a subgradient from the current value of \mathbf{w} and then project the resulting vector onto \mathcal{H} . Formally,

1. $\mathbf{w}^{(t+\frac{1}{2})} = \mathbf{w}^{(t)} - \eta \mathbf{v}_t$
2. $\mathbf{w}^{(t+1)} = \arg \min_{\mathbf{w} \in \mathcal{H}} \|\mathbf{w} - \mathbf{w}^{(t+\frac{1}{2})}\|.$

The projection step replaces the current value of \mathbf{w} by the vector in \mathcal{H} closest to it. Clearly, the projection step guarantees that $\mathbf{w}^{(t)} \in \mathcal{H}$ for all t . Since \mathcal{H} is convex this also implies that $\bar{\mathbf{w}} \in \mathcal{H}$ as required. We next show that the analysis of SGD with projections remains the same. This is based on the following lemma.

Lemma 696 (Projection Lemma). *Let \mathcal{H} be a closed convex set and let \mathbf{v} be the projection of \mathbf{w} onto \mathcal{H} , namely,*

$$\mathbf{v} = \arg \min_{\mathbf{x} \in \mathcal{H}} \|\mathbf{x} - \mathbf{w}\|^2.$$

Then, for every $\mathbf{u} \in \mathcal{H}$,

$$\|\mathbf{w} - \mathbf{u}\|^2 - \|\mathbf{v} - \mathbf{u}\|^2 \geq 0.$$

Proof. By the convexity of \mathcal{H} , for every $\alpha \in (0, 1)$ we have that $\mathbf{v} + \alpha(\mathbf{u} - \mathbf{v}) \in \mathcal{H}$. Therefore, from the optimality of \mathbf{v} we obtain

$$\begin{aligned} \|\mathbf{v} - \mathbf{w}\|^2 &\leq \|\mathbf{v} + \alpha(\mathbf{u} - \mathbf{v}) - \mathbf{w}\|^2 \\ &= \|\mathbf{v} - \mathbf{w}\|^2 + 2\alpha \langle \mathbf{v} - \mathbf{w}, \mathbf{u} - \mathbf{v} \rangle + \alpha^2 \|\mathbf{u} - \mathbf{v}\|^2. \end{aligned}$$

Rearranging, we obtain

$$2 \langle \mathbf{v} - \mathbf{w}, \mathbf{u} - \mathbf{v} \rangle \geq -\alpha \|\mathbf{u} - \mathbf{v}\|^2.$$

Taking the limit $\alpha \rightarrow 0$ we get that

$$\langle \mathbf{v} - \mathbf{w}, \mathbf{u} - \mathbf{v} \rangle \geq 0.$$

Therefore,

$$\begin{aligned}\|\mathbf{w} - \mathbf{u}\|^2 &= \|\mathbf{w} - \mathbf{v} + \mathbf{v} - \mathbf{u}\|^2 \\ &= \|\mathbf{w} - \mathbf{v}\|^2 + \|\mathbf{v} - \mathbf{u}\|^2 + 2\langle \mathbf{v} - \mathbf{w}, \mathbf{u} - \mathbf{v} \rangle \\ &\geq \|\mathbf{v} - \mathbf{u}\|^2.\end{aligned}$$

□

Equipped with the preceding lemma, we can easily adapt the analysis of SGD to the case in which we add projection steps on a closed and convex set. Simply note that for every t ,

$$\begin{aligned}&\|\mathbf{w}^{(t+1)} - \mathbf{w}^*\|^2 - \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2 \\ &= \|\mathbf{w}^{(t+1)} - \mathbf{w}^*\|^2 - \|\mathbf{w}^{(t+\frac{1}{2})} - \mathbf{w}^*\|^2 + \|\mathbf{w}^{(t+\frac{1}{2})} - \mathbf{w}^*\|^2 - \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2 \\ &\leq \|\mathbf{w}^{(t+\frac{1}{2})} - \mathbf{w}^*\|^2 - \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2.\end{aligned}$$

Therefore, Lemma 695 holds when we add projection steps and hence the rest of the analysis follows directly.

8.2.2 Variable Step Size

Another variant of SGD is decreasing the step size as a function of t . That is, rather than updating with a constant η , we use η_t . For instance, we can set $\eta_t = \frac{B}{\rho\sqrt{t}}$ and achieve a bound similar to Theorem 694. The idea is that when we are closer to the minimum of the function, we take our steps more carefully, so as not to “overshoot” the minimum.

8.2.3 Other Averaging Techniques

We have set the output vector to be $\hat{\mathbf{w}} = \frac{1}{T} \sum_{t=1}^T \mathbf{w}^{(t)}$. There are alternative approaches such as outputting $\mathbf{w}^{(t)}$ for some random $t \in [T]$, or outputting the average of $\mathbf{w}^{(t)}$ over the last αT iterations, for some $\alpha \in (0, 1)$. One can also take a weighted average of the last few iterates. These more sophisticated averaging schemes can improve the convergence speed in some situations, such as in the case of strongly convex functions defined in the following.

8.2.4 Strongly Convex Functions

In this section we show a variant of SGD that enjoys a faster convergence rate for problems in which the objective function is strongly convex. We rely on the following claim, which generalizes Lemma 13.5 of [119].

Claim 697. *If f is λ -strongly convex then for every \mathbf{w}, \mathbf{u} and $\mathbf{v} \in \partial f(\mathbf{w})$ we have*

$$\langle \mathbf{w} - \mathbf{u}, \mathbf{v} \rangle \geq f(\mathbf{w}) - f(\mathbf{u}) + \frac{\lambda}{2} \|\mathbf{w} - \mathbf{u}\|^2.$$

The proof is similar to the proof of Lemma 13.5 of [119] and is left as an exercise.

SGD for minimizing a λ -strongly convex function

Goal: Solve $\min_{\mathbf{w} \in \mathcal{H}} f(\mathbf{w})$

parameters: T

parameters: $\mathbf{w}^{(1)} = \mathbf{0}$

for $t = 1, 2, \dots, T$

Choose a random vector \mathbf{v}_t s.t. $\mathbb{E}[\mathbf{v}_t | \mathbf{w}^{(t)}] \in \partial f(\mathbf{w}^{(t)})$

Set $\eta_t = 1/(\lambda t)$

Set $\mathbf{w}^{(t+\frac{1}{2})} = \mathbf{w}^{(t)} - \eta_t \mathbf{v}_t$

Set $\mathbf{w}^{(t+1)} = \arg \min_{\mathbf{w} \in \mathcal{H}} \|\mathbf{w} - \mathbf{w}^{(t+\frac{1}{2})}\|^2$

output $\bar{\mathbf{w}} = \frac{1}{T} \sum_{t=1}^T \mathbf{w}^{(t)}$.

Theorem 698. *Assume that f is λ -strongly convex and that $\mathbb{E}[\|\mathbf{v}_t\|^2] \leq \rho^2$. Let $\mathbf{w}^* \in \arg \min_{\mathbf{w} \in \mathcal{H}} f(\mathbf{w})$ be an optimal solution. Then,*

$$\mathbb{E}[f(\hat{\mathbf{w}})] - f(\mathbf{w}^*) \leq \frac{\rho^2}{2\lambda T} (1 + \log(T)).$$

Proof. Let $\nabla^{(t)} = \mathbb{E}[\mathbf{v}_t | \mathbf{w}^{(t)}]$. Since f is strongly convex and $\nabla^{(t)}$ is in the subgradient set of f at $\mathbf{w}^{(t)}$ we have that

$$\langle \mathbf{w}^{(t)} - \mathbf{w}^*, \nabla^{(t)} \rangle \geq f(\mathbf{w}^{(t)}) - f(\mathbf{w}^*) + \frac{\lambda}{2} \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2. \quad (8.8)$$

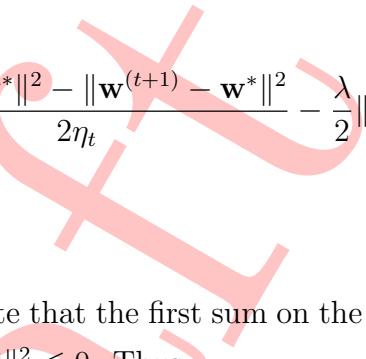
Next, we show that

$$\langle \mathbf{w}^{(t)} - \mathbf{w}^*, \nabla^{(t)} \rangle \leq \frac{\mathbb{E}[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2 - \|\mathbf{w}^{(t+1)} - \mathbf{w}^*\|^2]}{2\eta_t} + \frac{\eta_t}{2} \rho^2. \quad (8.9)$$

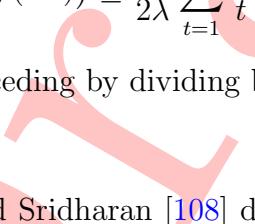
Since $\mathbf{w}^{(t+1)}$ is the projection of $\mathbf{w}^{(t+\frac{1}{2})}$ onto \mathcal{H} , and $\mathbf{w}^* \in \mathcal{H}$ we have that $\|\mathbf{w}^{(t+\frac{1}{2})} - \mathbf{w}^*\|^2 \geq \|\mathbf{w}^{(t+1)} - \mathbf{w}^*\|^2$. Therefore,

$$\begin{aligned}\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2 - \|\mathbf{w}^{(t+1)} - \mathbf{w}^*\|^2 &\geq \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2 - \|\mathbf{w}^{(t+\frac{1}{2})} - \mathbf{w}^*\|^2 \\ &= 2\eta_t \langle \mathbf{w}^{(t)} - \mathbf{w}^*, \mathbf{v}_t \rangle - \eta_t^2 \|\mathbf{v}_t\|^2.\end{aligned}$$

Taking expectation of both sides, rearranging, and using the assumption $\mathbb{E}[\|\mathbf{v}_t\|^2] \leq \rho^2$ yield Equation (8.9). Comparing Equation (8.8) and Equation (8.9) and summing over we obtain

$$\sum_{t=1}^T (\mathbb{E}[f(\mathbf{w}^{(t)})] - f(\mathbf{w}^*)) \leq \mathbb{E} \left[\sum_{t=1}^T \left(\frac{\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2 - \|\mathbf{w}^{(t+1)} - \mathbf{w}^*\|^2}{2\eta_t} - \frac{\lambda}{2} \|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2 \right) \right] + \frac{\rho^2}{2} \sum_{t=1}^T \eta_t.$$


Next, we use the definition $\eta_t = 1/(\lambda t)$ and note that the first sum on the right-hand side of the equation collapses to $-\lambda T \|\mathbf{w}^{(T+1)} - \mathbf{w}^*\|^2 \leq 0$. Thus,

$$\sum_{t=1}^T (\mathbb{E}[f(\mathbf{w}^{(t)})] - f(\mathbf{w}^*)) \leq \frac{\rho^2}{2\lambda} \sum_{t=1}^T \frac{1}{t} \leq \frac{\rho^2}{2\lambda} (1 + \log(T)).$$


The theorem follows from the preceding by dividing by T and using Jensen's inequality. \square

Remark Rakhlin, Shamir, and Sridharan [108] derived a convergence rate in which the $\log(T)$ term is eliminated for a variant of the algorithm in which we output the average of the last $T/2$ iterates, $\bar{\mathbf{w}} = \frac{2}{T} \sum_{t=T/2+1}^T \mathbf{w}^{(t)}$. Shamir and Zhang [120] have shown that Theorem 698 holds even if we output $\bar{\mathbf{w}} = \mathbf{w}^{(T)}$.

8.3 Learning with SGD

We have so far introduced and analyzed the SGD algorithm for general convex functions. Now we shall consider its applicability to learning tasks.

8.3.1 SGD for Risk Minimization

Recall that in learning we face the problem of minimizing the risk function

$$L_{\mathcal{D}}(\mathbf{w}) = \mathbb{E}_{z \sim \mathcal{D}} [\ell(\mathbf{w}, z)].$$

We have seen the method of empirical risk minimization, where we minimize the empirical risk, $L_S(\mathbf{w})$, as an estimate to minimizing $L_{\mathcal{D}}(\mathbf{w})$. SGD allows us to take a different

approach and minimize $L_{\mathcal{D}}(\mathbf{w})$ directly. Since we do not know \mathcal{D} , we cannot simply calculate $\nabla L_{\mathcal{D}}(\mathbf{w}^{(t)})$ and minimize it with the GD method. With SGD, however, all we need is to find an unbiased estimate of the gradient of $L_{\mathcal{D}}(\mathbf{w})$, that is, a random vector whose conditional expected value is $\nabla L_{\mathcal{D}}(\mathbf{w}^{(t)})$. We shall now see how such an estimate can be easily constructed.

For simplicity, let us first consider the case of differentiable loss functions. Hence the risk function $L_{\mathcal{D}}$ is also differentiable. The construction of the random vector \mathbf{v}_t will be as follows: First, sample $z \sim \mathcal{D}$. Then, define \mathbf{v}_t to be the gradient of the function $\ell(\mathbf{w}, z)$ with respect to \mathbf{w} , at the point $\mathbf{w}^{(t)}$. Then, by the linearity of the gradient we have

$$\mathbb{E}[\mathbf{v}_t | \mathbf{w}^{(t)}] = \mathbb{E}_{z \sim \mathcal{D}}[\nabla \ell(\mathbf{w}^{(t)}, z)] = \nabla \mathbb{E}_{z \sim \mathcal{D}}[\ell(\mathbf{w}^{(t)}, z)] = \nabla L_{\mathcal{D}}(\mathbf{w}^{(t)}). \quad (8.10)$$

The gradient of the loss function $\ell(\mathbf{w}, z)$ at $\mathbf{w}^{(t)}$ is therefore an unbiased estimate of the gradient of the risk function $L_{\mathcal{D}}(\mathbf{w}^{(t)})$ and is easily constructed by sampling a single fresh example $z \sim \mathcal{D}$ at each iteration t .

The same argument holds for nondifferentiable loss functions. We simply let \mathbf{v}_t be a subgradient of $\ell(\mathbf{w}, z)$ at $\mathbf{w}^{(t)}$. Then, for every \mathbf{u} we have

$$\ell(\mathbf{u}, z) - \ell(\mathbf{w}^{(t)}, z) \geq \langle \mathbf{u} - \mathbf{w}^{(t)}, \mathbf{v}_t \rangle.$$

Taking expectation on both sides with respect to $z \sim \mathcal{D}$ and conditioned on the value $\mathbf{w}^{(t)}$ we obtain

$$\begin{aligned} L_{\mathcal{D}}(\mathbf{u}) - L_{\mathcal{D}}(\mathbf{w}^{(t)}) &= \mathbb{E}[\ell(\mathbf{u}, z) - \ell(\mathbf{w}^{(t)}, z) | \mathbf{w}^{(t)}] \\ &\geq \mathbb{E}[\langle \mathbf{u} - \mathbf{w}^{(t)}, \mathbf{v}_t \rangle | \mathbf{w}^{(t)}] \\ &= \langle \mathbf{u} - \mathbf{w}^{(t)}, \mathbb{E}[\mathbf{v}_t | \mathbf{w}^{(t)}] \rangle. \end{aligned}$$

It follows that $\mathbb{E}[\mathbf{v}_t | \mathbf{w}^{(t)}]$ is a subgradient of $L_{\mathcal{D}}(\mathbf{w})$ at $\mathbf{w}^{(t)}$.

To summarize, the stochastic gradient descent framework for minimizing the risk is as follows. We shall now use our analysis of SGD to obtain a sample complexity analysis for learning convex-Lipschitz-bounded problems. Theorem 694 yields the following:

Corollary 699. *Consider a convex-Lipschitz-bounded learning problem with parameters ρ, B . Then, for every $\epsilon > 0$, if we run the SGD method for minimizing $L_{\mathcal{D}}(\mathbf{w})$ with a number of iterations (i.e., number of examples)*

$$T \geq \frac{B^2 \rho^2}{\epsilon^2}$$

and with $\eta = \sqrt{\frac{B^2}{\rho^2 T}}$, then the output of SGD satisfies

$$\mathbb{E}[L_{\mathcal{D}}(\bar{\mathbf{w}})] \leq \min_{\mathbf{w} \in \mathcal{H}} L_{\mathcal{D}}(\mathbf{w}) + \epsilon.$$

Stochastic Gradient Descent (SGD) for minimizing $L_{\mathcal{D}}(\mathbf{w})$

parameters: Scalar $\eta > 0$, integer $T > 0$

initialize: $\mathbf{w}^{(1)} = \mathbf{0}$

for $t = 1, 2, \dots, T$

sample $z \sim \mathcal{D}$

pick $\mathbf{v}_t \in \partial \ell(\mathbf{w}^{(t)}, z)$

update $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \mathbf{v}_t$

output $\bar{\mathbf{w}} = \frac{1}{T} \sum_{t=1}^T \mathbf{w}^{(t)}$.

It is interesting to note that the required sample complexity is of the same order of magnitude as the sample complexity guarantee we derived for regularized loss minimization. In fact, the sample complexity of SGD is even better than what we have derived for regularized loss minimization by a factor of 8.

8.3.2 Analyzing SGD for Convex-Smooth Learning Problems

In the previous chapter we saw that the regularized loss minimization rule also learns the class of convex-smooth-bounded learning problems. We now show that the SGD algorithm can be also used for such problems.

Theorem 700. Assume that for all z , the loss function $\ell(\cdot, z)$ is convex, β -smooth, and nonnegative. Then, if we run the SGD algorithm for minimizing $L_{\mathcal{D}}(\mathbf{w})$ we have that for every \mathbf{w}^* ,

$$\mathbb{E}[L_{\mathcal{D}}(\bar{\mathbf{w}})] \leq \frac{1}{1 - \eta\beta} \left(L_{\mathcal{D}}(\mathbf{w}^*) + \frac{\|\mathbf{w}^*\|^2}{2\eta T} \right).$$

Proof. Recall that if a function is β -smooth and nonnegative then it is self-bounded:

$$\|\nabla f(\mathbf{w})\|^2 \leq 2\beta f(\mathbf{w}).$$

To analyze SGD for convex-smooth problems, let us define z_1, \dots, z_T the random samples of the SGD algorithm, let $f_t(\cdot) = \ell(\cdot, z_t)$, and note that $\mathbf{v}_t = \nabla f_t(\mathbf{w}^{(t)})$. For all t , f_t is a convex function and therefore $f_t(\mathbf{w}^{(t)}) - f_t(\mathbf{w}^*) \leq \langle \mathbf{v}_t, \mathbf{w}^{(t)} - \mathbf{w}^* \rangle$. Summing over t and using Lemma 14.1 we obtain

$$\sum_{t=1}^T (f_t(\mathbf{w}^{(t)}) - f_t(\mathbf{w}^*)) \leq \sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{w}^{(t)} - \mathbf{w}^* \rangle \leq \frac{\|\mathbf{w}^*\|^2}{2\eta} + \frac{\eta}{2} \sum_{t=1}^T \|\mathbf{v}_t\|^2.$$

Combining the preceding with the self-boundedness of f_t yields

$$\sum_{t=1}^T (f_t(\mathbf{w}^{(t)}) - f_t(\mathbf{w}^*)) \leq \frac{\|\mathbf{w}^*\|^2}{2\eta} + \eta\beta \sum_{t=1}^T f_t(\mathbf{w}^{(t)}).$$

Dividing by T and rearranging, we obtain

$$\frac{1}{T} \sum_{t=1}^T (f_t \mathbf{w}^{(t)}) \leq \frac{1}{1 - \eta\beta} \left(\frac{1}{T} \sum_{t=1}^T (f_t \mathbf{w}^*) + \frac{\|\mathbf{w}^*\|^2}{2\eta T} \right).$$

Next, we take expectation of the two sides of the preceding equation with respect to z_1, \dots, z_T . Clearly, $\mathbb{E}[f_t(\mathbf{w}^*)] = L_{\mathcal{D}}(\mathbf{w}^*)$. In addition, using the same argument as in the proof of Theorem 694 we have that

$$\mathbb{E} \left[\frac{1}{T} \sum_{t=1}^T f_t(\mathbf{w}^{(t)}) \right] = \mathbb{E} \left[\frac{1}{T} \sum_{t=1}^T L_{\mathcal{D}}(\mathbf{w}^{(t)}) \right] \geq \mathbb{E}[L_{\mathcal{D}}(\bar{\mathbf{w}})].$$

Combining all we conclude our proof. \square

As a direct corollary we obtain:

Corollary 701. *Consider a convex-smooth-bounded learning problem with parameters β , B . Assume in addition that $\ell(\mathbf{0}, z) \leq 1$ for all $z \in Z$. For every $\epsilon > 0$, set $\eta = \frac{1}{\beta(1+3/\epsilon)}$. Then, running SGD with $T \geq 12B^2\beta/\epsilon^2$ yields*

$$\mathbb{E}[L_{\mathcal{D}}(\bar{\mathbf{w}})] \leq \min_{\mathbf{w} \in \mathcal{H}} L_{\mathcal{D}}(\mathbf{w}) + \epsilon.$$

8.3.3 SGD for Regularized Loss Minimization

We have shown that SGD enjoys the same worst-case sample complexity bound as regularized loss minimization. However, on some distributions, regularized loss minimization may yield a better solution. Therefore, in some cases we may want to solve the optimization problem associated with regularized loss minimization, namely,¹

$$\min_{\mathbf{w}} \left(\frac{\lambda}{2} \|\mathbf{w}\|^2 + L_S(\mathbf{w}) \right). \quad (8.11)$$

Since we are dealing with convex learning problems in which the loss function is convex, the preceding problem is also a convex optimization problem that can be solved using SGD as well, as we shall see in this section.

Define $f(\mathbf{w}) = \frac{\lambda}{2} \|\mathbf{w}\|^2 + L_S(\mathbf{w})$. Note that f is a λ -strongly convex function; therefore, we can apply the SGD variant given in Section 8.2.4 (with $\mathcal{H} = \mathbb{R}^d$). To apply this algorithm, we only need to find a way to construct an unbiased estimate of a subgradient of f at $\mathbf{w}^{(t)}$. This is easily done by noting that if we pick z uniformly at random from S , and choose $\mathbf{v}_t \in \partial \ell(\mathbf{w}^{(t)}, z)$ then the expected value of $\lambda \mathbf{w}^{(t)} + \mathbf{v}_t$ is a subgradient of f at $\mathbf{w}^{(t)}$.

¹We divided λ by 2 for convenience.

To analyze the resulting algorithm, we first rewrite the update rule (assuming that $\mathcal{H} = \mathbb{R}^d$ and therefore the projection step does not matter) as follows

$$\begin{aligned}
 \mathbf{w}^{(t+1)} &= \mathbf{w}^{(t)} - \frac{1}{\lambda t} (\lambda \mathbf{w}^{(t)} + \mathbf{v}_t) \\
 &= \left(1 - \frac{1}{t}\right) \mathbf{w}^{(t)} - \frac{1}{\lambda t} \mathbf{v}_t \\
 &= \frac{t-1}{t} \mathbf{w}^{(t)} - \frac{1}{\lambda t} \mathbf{v}_t \\
 &= \frac{t-1}{t} \left(\frac{t-2}{t-1} \mathbf{w}^{(t-1)} - \frac{1}{\lambda(t-1)} \mathbf{v}_{t-1} \right) - \frac{1}{\lambda t} \mathbf{v}_t \\
 &= -\frac{1}{\lambda t} \sum_{i=1}^t \mathbf{v}_i.
 \end{aligned} \tag{8.12}$$

If we assume that the loss function is ρ -Lipschitz, it follows that for all t we have $\|\mathbf{v}_t\| \leq \rho$ and therefore $\|\lambda \mathbf{w}^{(t)}\| \leq \rho$, which yields

$$\|\lambda \mathbf{w}^{(t)} + \mathbf{v}_t\| \leq 2\rho.$$

Theorem 698 therefore tells us that after performing T iterations we have that

$$\mathbb{E}[f(\bar{\mathbf{w}})] - f(\mathbf{w}^*) \leq \frac{4\rho^2}{\lambda T} (1 + \log(T)).$$

8.3.4 Exercises

Exercise 702. Prove Corollary 701.

Exercise 703 (Perceptron as a subgradient descent algorithm). Let $\mathcal{S} = ((\mathbf{x}_1, y_1), \dots, (\mathbf{x}_m, y_m)) \in (\mathbb{R}^d \times \{\pm 1\})^m$. Assume that there exists $\mathbf{w} \in \mathbb{R}^d$ such that for every $i \in \{1, \dots, m\}$ we have $y_i \langle \mathbf{w}, \mathbf{x}_i \rangle \geq 1$, and let \mathbf{w}^* be a vector that has the minimal norm among all vectors that satisfy the preceding requirement. Let $R = \max_i \|\mathbf{x}_i\|_2$. Define a function

$$f(\mathbf{w}) = \max_{i \in \{1, \dots, m\}} (1 - y_i \langle \mathbf{w}, \boldsymbol{\xi}_i \rangle).$$

- Show that $\min_{\mathbf{w}: \|\mathbf{w}\|_2 \leq \|\mathbf{w}^*\|_2} f(\mathbf{w}) = 0$ and show that any \mathbf{w} for which $f(\mathbf{w}) < 1$ separates the examples in \mathcal{S} .
- Show how to calculate a subgradient of f .
- Describe and analyze the subgradient descent algorithm for this case.

Exercise 704 (Variable step size (*)). Prove an analog of Theorem 694 for SGD with a variable step size $\eta_t = \frac{B}{\rho\sqrt{t}}$.

8.4 Sum of smooth and strongly convex functions

(Taken from Chapter 6.3 of [20])

Let us examine in more details the main example from Section 1.5. That is one is interested in the unconstrained minimization of

$$f(\mathbf{x}) = \frac{1}{m} \sum_{i=1}^m f_i(\mathbf{x}),$$

where f_1, \dots, f_m are β -smooth and convex functions, and f is α -strongly convex. Typically in machine learning α can be as small as $1/m$, while β is of order of a constant. In other words the condition number $\kappa = \beta/\alpha$ can be as large as $\Omega(m)$. Let us now compare the basic gradient descent, that is

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \frac{\eta}{m} \sum_{i=1}^m \nabla f_i(\mathbf{x}),$$

to SGD

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \nabla f_{i_t}(\mathbf{x}),$$

where i_t is drawn uniformly at random in $[m]$ (independently of everything else).

Theorem 705. *Let f be α -strongly convex and β -smooth on \mathcal{X} . Then projected gradient descent with $\eta = \frac{1}{\beta}$ satisfies for $t \geq 0$,*

$$\|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 \leq \exp\left(-\frac{t}{\kappa}\right) \|\mathbf{x}_1 - \mathbf{x}^*\|^2, \quad \text{where } \kappa = \beta/\alpha.$$

Theorem 706. *Let f be α -strongly convex, and assume that the stochastic oracle is such that $\mathbb{E}\|\tilde{g}(\mathbf{x})\|_*^2 \leq B^2$. Then SGD with $\eta_s = \frac{2}{\alpha(s+1)}$ satisfies*

$$f\left(\sum_{s=1}^t \frac{2s}{t(t+1)} \mathbf{x}_s\right) - f(\mathbf{x}^*) \leq \frac{2B^2}{\alpha(t+1)}.$$

Theorem 705 shows that gradient descent requires $\mathcal{O}(m\kappa \log(1/\varepsilon))$ gradient computations (which can be improved to $\mathcal{O}(m\sqrt{\kappa} \log(1/\varepsilon))$ with Nesterov's accelerated gradient descent), while Theorem 706 shows that SGD (with appropriate averaging) requires $\mathcal{O}(1/(\alpha\varepsilon))$ gradient computations. Thus one can obtain a low accuracy solution reasonably fast with SGD, but for high accuracy the basic gradient descent is more suitable. Can we get the best of both worlds? This question was answered positively in [114] with SAG (Stochastic Averaged Gradient) and in [118] with SDCA (Stochastic Dual Coordinate Ascent). These methods require only $\mathcal{O}((m + \kappa) \log(1/\varepsilon))$ gradient computations. We describe below the SVRG (Stochastic Variance Reduced Gradient descent) algorithm

from [65] which makes the main ideas of SAG and SDCA more transparent (see also [34] for more on the relation between these different methods). We also observe that a natural question is whether one can obtain a Nesterov's accelerated version of these algorithms that would need only $\mathcal{O}((m + \sqrt{m\kappa}) \log(1/\varepsilon))$, see [2, 117, 142] for recent works on this question.

To obtain a linear rate of convergence one needs to make “big steps”, that is the step-size should be of order of a constant. In SGD the stepsize is typically of order $1/\sqrt{t}$ because of the variance introduced by the stochastic oracle. The idea of SVRG is to “center” the output of the stochastic oracle in order to reduce the variance. Precisely instead of feeding $\nabla f_i(\mathbf{x})$ into the gradient descent one would use $\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}) + \nabla f(\mathbf{y})$ where \mathbf{y} is a centering sequence. This is a sensible idea since, when \mathbf{x} and \mathbf{y} are close to the optimum, one should have that $\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})$ will have a small variance, and of course $\nabla f(\mathbf{y})$ will also be small (note that $\nabla f_i(\mathbf{x})$ by itself is not necessarily small). This intuition is made formal with the following lemma.

Lemma 707. *Let f_1, \dots, f_m be β -smooth convex functions on \mathbb{R}^n , and i be a random variable uniformly distributed in $[m]$. Then*

$$\mathbb{E}\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{x}^*)\|_2^2 \leq 2\beta(f(\mathbf{x}) - f(\mathbf{x}^*)).$$

Proof. Let $g_i(\mathbf{x}) = f_i(\mathbf{x}) - f_i(\mathbf{x}^*) - \nabla f_i(\mathbf{x}^*)^\top(\mathbf{x} - \mathbf{x}^*)$. By convexity of f_i one has $g_i(\mathbf{x}) \geq 0$ for any \mathbf{x} and in particular using $(f\left(\mathbf{x} - \frac{1}{\beta}\nabla f(\mathbf{x})\right) - f(\mathbf{x}) \leq -\frac{1}{2\beta}\|\nabla f(\mathbf{x})\|^2)$ this yields $-g_i(\mathbf{x}) \leq -\frac{1}{2\beta}\|\nabla g_i(\mathbf{x})\|_2^2$ which can be equivalently written as

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{x}^*)\|_2^2 \leq 2\beta(f_i(\mathbf{x}) - f_i(\mathbf{x}^*) - \nabla f_i(\mathbf{x}^*)^\top(\mathbf{x} - \mathbf{x}^*)).$$

Taking expectation with respect to i and observing that $\mathbb{E}\nabla f_i(\mathbf{x}^*) = \nabla f(\mathbf{x}^*) = 0$ yields the claimed bound. □

On the other hand the computation of $\nabla f(\mathbf{y})$ is expensive (it requires m gradient computations), and thus the centering sequence should be updated more rarely than the main sequence. These ideas lead to the following epoch-based algorithm.

Let $\mathbf{y}^{(1)} \in \mathbb{R}^n$ be an arbitrary initial point. For $s = 1, 2, \dots$, let $\mathbf{x}_1^{(s)} = \mathbf{y}^{(s)}$. For $t = 1, \dots, k$ let

$$\mathbf{x}_{t+1}^{(s)} = \mathbf{x}_t^{(s)} - \eta \left(\nabla f_{i_t^{(s)}}(\mathbf{x}_t^{(s)}) - \nabla f_{i_t^{(s)}}(\mathbf{y}^{(s)}) + \nabla f(\mathbf{y}^{(s)}) \right),$$

where $i_t^{(s)}$ is drawn uniformly at random (and independently of everything else) in $[m]$.

Also let

$$y^{(s+1)} = \frac{1}{k} \sum_{t=1}^k \mathbf{x}_t^{(s)}.$$

Theorem 708. Let f_1, \dots, f_m be β -smooth convex functions on \mathbb{R}^n and f be α -strongly convex. Then SVRG with $\eta = \frac{1}{10\beta}$ and $k = 20\kappa$ satisfies

$$\mathbb{E}f(\mathbf{y}^{(s+1)}) - f(\mathbf{x}^*) \leq 0.9^s(f(\mathbf{y}^{(1)}) - f(\mathbf{x}^*)).$$

Proof. We fix a phase $s \geq 1$ and we denote by \mathbb{E} the expectation taken with respect to $i_1^{(s)}, \dots, i_k^{(s)}$. We show below that

$$\begin{aligned} & \mathbb{E}f(\mathbf{y}^{(s+1)}) - f(\mathbf{x}^*) \\ &= \mathbb{E}f\left(\frac{1}{k} \sum_{t=1}^k \mathbf{x}_t^{(s)}\right) - f(\mathbf{x}^*) \\ &\leq 0.9(f(\mathbf{y}^{(s)}) - f(\mathbf{x}^*)), \end{aligned}$$

which clearly implies the theorem. To simplify the notation in the following we drop the dependency on s , that is we want to show that

$$\mathbb{E}f\left(\frac{1}{k} \sum_{t=1}^k \mathbf{x}_t^{(s)}\right) - f(\mathbf{x}^*) \leq 0.9(f(\mathbf{y}) - f(\mathbf{x}^*)) \quad (8.13)$$

We start as for the proof of Theorem 705 (analysis of gradient descent for smooth and strongly convex functions) with

$$\|\mathbf{x}_{t+1} - \mathbf{x}^*\|_2^2 = \|\mathbf{x}_t - \mathbf{x}^*\|_2^2 - 2\eta \mathbf{v}_t^\top (\mathbf{x}_t - \mathbf{x}^*) + \eta^2 \|\mathbf{v}_t\|_2^2, \quad (8.14)$$

where

$$\mathbf{v}_t = \nabla f_{i_t}(\mathbf{x}_t) - \nabla f_{i_t}(\mathbf{y}) + \nabla f(\mathbf{y}).$$

Using Lemma 707, we upper bound $\mathbb{E}_{i_t} \|\mathbf{v}_t\|_2^2$ as follows (also recall that $\mathbb{E}\|\mathbf{X} - \mathbb{E}(\mathbf{X})\|_2^2 \leq \mathbb{E}\|\mathbf{X}\|_2^2$, and $\mathbb{E}_{i_t} \nabla f_{i_t}(\mathbf{x}^*) = 0$):

$$\begin{aligned} & \mathbb{E}_{i_t} \|\mathbf{v}_t\|_2^2 \\ & \leq 2\mathbb{E}_{i_t} \|\nabla f_{i_t}(\mathbf{x}_t) - \nabla f_{i_t}(\mathbf{x}^*)\|_2^2 \\ & \quad + 2\mathbb{E}_{i_t} \|\nabla f_{i_t}(\mathbf{y}) - \nabla f_{i_t}(\mathbf{x}^*) - \nabla f(\mathbf{y})\|_2^2 \\ & \leq 2\mathbb{E}_{i_t} \|\nabla f_{i_t}(\mathbf{x}_t) - \nabla f_{i_t}(\mathbf{x}^*)\|_2^2 \\ & \quad + 2\mathbb{E}_{i_t} \|\nabla f_{i_t}(\mathbf{y}) - \nabla f_{i_t}(\mathbf{x}^*)\|_2^2 \\ & \leq 4\beta(f(\mathbf{x}_t) - f(\mathbf{x}^*) + f(\mathbf{y}) - f(\mathbf{x}^*)). \end{aligned} \quad (8.15)$$

Also observe that

$$\mathbb{E}_{i_t} \mathbf{v}_t^\top (\mathbf{x}_t - \mathbf{x}^*) = \nabla f(\mathbf{x}_t)^\top (\mathbf{x}_t - \mathbf{x}^*) \geq f(\mathbf{x}_t) - f(\mathbf{x}^*),$$

and thus plugging this into (8.14) together with (8.15) one obtains

$$\begin{aligned} E_{i_t} \|\mathbf{x}_{t+1} - \mathbf{x}^*\|_2^2 &\leq \|\mathbf{x}_t - \mathbf{x}^*\|_2^2 - 2\eta(1 - 2\beta\eta)(f(\mathbf{x}_t) - f(\mathbf{x}^*)) \\ &\quad + 4\beta\eta^2(f(\mathbf{y}) - f(\mathbf{x}^*)). \end{aligned}$$

Summing the above inequality over $t = 1, \dots, k$ yields

$$\begin{aligned} E_{i_t} \|\mathbf{x}_{t+1} - \mathbf{x}^*\|_2^2 &\leq \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2 - 2\eta(1 - 2\beta\eta)\mathbb{E} \sum_{i=1}^k (f(\mathbf{x}_t) - f(\mathbf{x}^*)) \\ &\quad + 4\beta\eta^2 k(f(\mathbf{y}) - f(\mathbf{x}^*)). \end{aligned}$$

Noting that $\mathbf{x}_1 = \mathbf{y}$ and that by α -strong convexity one has $f(\mathbf{x}) - f(\mathbf{x}^*) \geq \frac{\alpha}{2}\|\mathbf{x} - \mathbf{x}^*\|_2^2$, one can rearrange the above display to obtain

$$\begin{aligned} &\mathbb{E} f\left(\frac{1}{k} \sum_{i=1}^k \mathbf{x}_t\right) - f(\mathbf{x}^*) \\ &\leq \left(\frac{1}{\alpha\eta(1 - 2\beta\eta)k} + \frac{2\beta\eta}{1 - 2\beta\eta}\right) (f(\mathbf{y}) - f(\mathbf{x}^*)). \end{aligned}$$

Using that $\eta = \frac{1}{10\beta}$ and $k = 20\kappa$ finally yields (8.13) which itself concludes the proof. \square

8.5 Random coordinate descent

(Taken from Chapter 6.4 of [20])

We assume throughout this section that f is a convex and differentiable function on \mathbb{R}^n , with a unique minimizer \mathbf{x}^* . We investigate one of the simplest possible scheme to optimize f , the random coordinate descent (RCD) method. In the following we denote $\nabla_i f(\mathbf{x}) = \frac{\partial f}{\partial x_i}(\mathbf{x})$. RCD is defined as follows, with an arbitrary initial point $\mathbf{x}_1 \in \mathbb{R}^n$,

$$\mathbf{x}_{s+1} = \mathbf{x}_s - \eta \nabla_{i_s} f(\mathbf{x}) \mathbf{e}_{i_s},$$

where i_s is drawn uniformly at random from $[n]$ (and independently of everything else). One can view RCD as SGD with the specific oracle $\tilde{g}(\mathbf{x}) = n \nabla_{\mathcal{I}} f(\mathbf{x}) \mathbf{e}_{\mathcal{I}}$ where \mathcal{I} is drawn uniformly at random from $[n]$. Clearly $\mathbb{E} \tilde{g}(\mathbf{x}) = \nabla f(\mathbf{x})$, and furthermore

$$\mathbb{E} \|\tilde{g}(\mathbf{x})\|_2^2 = \frac{1}{n} \sum_{i=1}^n \|n \nabla_i f(\mathbf{x}) \mathbf{e}_i\|_2^2 = n \|\nabla f(\mathbf{x})\|_2^2.$$

We first quote the convergence result for mirror descent.

Theorem 709. Let Φ be a mirror map 1-strongly convex on $\mathcal{X} \cap \mathcal{D}$ with respect to $\|\cdot\|$, and let $R^2 = \sup_{x \in \mathcal{X} \cap \mathcal{D}} \Phi(x) - \Phi(x_1)$. Let f be convex. Furthermore assume that the stochastic oracle is such that $\mathbb{E}\|\tilde{g}(x)\|_*^2 \leq B^2$. Then S-MD with $\eta = \frac{R}{B}\sqrt{\frac{2}{t}}$ satisfies

$$\mathbb{E}f\left(\frac{1}{t} \sum_{s=1}^t \mathbf{x}_s\right) - \min_{x \in \mathcal{X}} f(x) \leq RB\sqrt{\frac{2}{t}}.$$

Thus using Theorem 709 (with $\Phi(x) = \frac{1}{2}\|\mathbf{x}\|_2^2$, that is S-MD being SGD) one immediately obtains the following result.

Theorem 710. Let f be convex and L -Lipschitz on \mathbb{R}^n , then RCD with $\eta = \frac{R}{L}\sqrt{\frac{2}{nt}}$ satisfies

$$\mathbb{E}f\left(\frac{1}{t} \sum_{s=1}^t \mathbf{x}_s\right) - \min_{x \in \mathcal{X}} f(x) \leq RL\sqrt{\frac{2n}{t}}.$$

Somewhat unsurprisingly RCD requires n times more iterations than gradient descent to obtain the same accuracy. In the next section, we will see that this statement can be greatly improved by taking into account directional smoothness.

8.6 RCD for coordinate-smooth optimization

We assume now directional smoothness for f , that is there exists β_1, \dots, β_n such that for any $i \in [n]$, $\mathbf{x} \in \mathbb{R}^n$ and $u \in \mathbb{R}$,

$$|\nabla_i f(\mathbf{x} + u\mathbf{e}_i) - \nabla_i f(\mathbf{x})| \leq \beta_i |u|.$$

If f is twice differentiable then this is equivalent to $(\nabla^2 f(\mathbf{x}))_{i,i} \leq \beta_i$. In particular, since the maximal eigenvalue of a matrix is upper bounded by its trace, one can see that the directional smoothness implies that f is β -smooth with $\beta \leq \sum_{i=1}^n \beta_i$. We now study the following “aggressive” RCD, where the step-sizes are of order of the inverse smoothness:

$$\mathbf{x}_{s+1} = \mathbf{x}_s - \frac{1}{\beta_{i_s}} \nabla_{i_s} f(\mathbf{x}) \mathbf{e}_{i_s}.$$

Furthermore we study a more general sampling distribution than uniform, precisely for $\gamma \geq 0$ we assume that i_s is drawn (independently) from the distribution p_γ defined by

$$p_\gamma(i) = \frac{\beta_i^\gamma}{\sum_{j=1}^n \beta_j^\gamma}, i \in [n].$$

This algorithm was proposed in [103], and we denote it by RCD(γ). Observe that, up to a preprocessing step of complexity $\mathcal{O}(n)$, one can sample from p_γ in time $\mathcal{O}(\log(n))$.

The following rate of convergence is derived in [103], using the dual norms $\|\cdot\|_{[\gamma]}, \|\cdot\|_{[\gamma]}^*$ defined by

$$\|\mathbf{x}\|_{[\gamma]} = \sqrt{\sum_{i=1}^n \beta_i^\gamma \mathbf{x}_i^2}, \quad \|\mathbf{x}\|_{[\gamma]}^* = \sqrt{\sum_{i=1}^n \frac{1}{\beta_i^\gamma} \mathbf{x}_i^2}.$$

Theorem 711. Let f be convex and β -smooth on \mathbb{R}^n . Then gradient descent with $\eta = \frac{1}{\beta}$ satisfies

$$f(\mathbf{x}_t) - f(\mathbf{x}^*) \leq \frac{2\beta \|\mathbf{x}_1 - \mathbf{x}^*\|^2}{t-1}.$$

Theorem 712. Let f be convex and such that $u \in \mathbb{R} \mapsto f(\mathbf{x} + u\mathbf{e}_i)$ is β_i -smooth for any $i \in [n], \mathbf{x} \in \mathbb{R}^n$. Then $RCD(\gamma)$ satisfies for $t \geq 2$,

$$\mathbb{E} f(\mathbf{x}_t) - f(\mathbf{x}^*) \leq \frac{2R_{1-\gamma}^2(\mathbf{x}_1) \sum_{i=1}^n \beta_i^\gamma}{t-1},$$

where

$$R_{1-\gamma}(\mathbf{x}_1) = \sup_{\mathbf{x} \in \mathbb{R}^n : f(\mathbf{x}) \leq f(\mathbf{x}_1)} \|\mathbf{x} - \mathbf{x}^*\|_{[1-\gamma]}.$$

Recall from Theorem 711 that in this context the basic gradient descent attains a rate of $\beta \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2/t$, where $\beta \leq \sum_{i=1}^n \beta_i$. Thus we see that $RCD(1)$ greatly improves upon gradient descent for functions where β is of order of $\sum_{i=1}^n \beta_i$. Indeed in this case both methods attain the same accuracy after a fixed number of iterations, but the iterations of coordinate descent are potentially much cheaper than the iterations of gradient descent.

Proof. By applying $(f\left(\mathbf{x} - \frac{1}{\beta} \nabla f(\mathbf{x})\right) - f(\mathbf{x}) \leq -\frac{1}{2\beta} \|\nabla f(\mathbf{x})\|^2)$ to the β_i -smooth function $u \in \mathbb{R} \mapsto f(\mathbf{x} + u\mathbf{e}_i)$ one obtains

$$f\left(\mathbf{x} - \frac{1}{\beta_i} \nabla_i f(\mathbf{x}) \mathbf{e}_i\right) - f(\mathbf{x}) \leq -\frac{1}{2\beta_i} (\nabla_i f(\mathbf{x}))^2.$$

We use this as follows:

$$\begin{aligned} & \mathbb{E}_{i_s} f(\mathbf{x}_{s+1}) - f(\mathbf{x}_s) \\ &= \sum_{i=1}^n p_\gamma(i) \left(f\left(\mathbf{x}_s - \frac{1}{\beta_i} \nabla_i f(\mathbf{x}_s) \mathbf{e}_i\right) - f(\mathbf{x}_s) \right) \\ &\leq - \sum_{i=1}^n \frac{p_\gamma(i)}{2\beta_i} (\nabla_i f(\mathbf{x}_s))^2 \\ &= - \frac{1}{2 \sum_{i=1}^n \beta_i^\gamma} (\|\nabla f(\mathbf{x}_s)\|_{[1-\gamma]}^*)^2. \end{aligned}$$

Denote $\delta_s = \mathbb{E}f(\mathbf{x}_s) - f(\mathbf{x}^*)$. Observe that the above calculation can be used to show that $f(\mathbf{x}_{s+1}) \leq f(\mathbf{x}_s)$ and thus one has, by definition of $R_{1-\gamma}(\mathbf{x}_1)$,

$$\begin{aligned}\delta_s &\leq \nabla f(\mathbf{x}_s)^\top (\mathbf{x}_s - \mathbf{x}^*) \\ &\leq \|\mathbf{x}_s - \mathbf{x}^*\|_{[1-\gamma]} \|\nabla f(\mathbf{x}_s)\|_{[1-\gamma]}^* \\ &\leq R_{1-\gamma}(\mathbf{x}_1) \|\nabla f(\mathbf{x}_s)\|_{[1-\gamma]}^*.\end{aligned}$$

Thus putting together the above calculations one obtains

$$\delta_{s+1} \leq \delta_s - \frac{1}{2R_{1-\gamma}^2(\mathbf{x}_1) \sum_{i=1}^n \beta_i^\gamma} \delta_s^2.$$

The proof can be concluded with similar computations than for Theorem 711. □

We discussed above the specific case of $\gamma = 1$. Both $\gamma = 0$ and $\gamma = 1/2$ also have an interesting behavior, and we refer to [103] for more details. The latter paper also contains a discussion of high probability results and potential acceleration by Y. Nesterov. We also refer to [111] for a discussion of RCD in a distributed setting.

8.7 RCD for smooth and strongly convex optimization

If in addition to directional smoothness one also assumes strong convexity, then RCD attains in fact a linear rate.

Theorem 713. *Let $\gamma \geq 0$. Let f be α -strongly convex w.r.t. $\|\cdot\|_{[1-\gamma]}$, and such that $u \in \mathbb{R} \mapsto f(\mathbf{x} + u\mathbf{e}_i)$ is β_i -smooth for any $i \in [n], \mathbf{x} \in \mathbb{R}^n$. Let $\kappa_\gamma = \frac{\sum_{i=1}^n \beta_i^\gamma}{\alpha}$, then $\text{RCD}(\gamma)$ satisfies*

$$\mathbb{E}f(\mathbf{x}_{t+1}) - f(\mathbf{x}^*) \leq \left(1 - \frac{1}{\kappa_\gamma}\right)^t (f(\mathbf{x}_1) - f(\mathbf{x}^*)).$$

We use the following elementary lemma.

Lemma 714. *Let f be α -strongly convex w.r.t. $\|\cdot\|$ on \mathbb{R}^n , then*

$$f(\mathbf{x}) - f(\mathbf{x}^*) \leq \frac{1}{2\alpha} \|\nabla f(\mathbf{x})\|_*^2.$$

Proof. By strong convexity, Holder's inequality, and an elementary calculation,

$$\begin{aligned}f(\mathbf{x}) - f(\mathbf{y}) &\leq \nabla f(\mathbf{x})^\top (\mathbf{x} - \mathbf{y}) - \frac{\alpha}{2} \|\mathbf{x} - \mathbf{y}\|_2^2 \\ &\leq \|\nabla f(\mathbf{x})\|_* \|\mathbf{x} - \mathbf{y}\| - \frac{\alpha}{2} \|\mathbf{x} - \mathbf{y}\|_2^2 \\ &\leq \frac{1}{2\alpha} \|\nabla f(\mathbf{x})\|_*^2,\end{aligned}$$

which concludes the proof by taking $\mathbf{y} = \mathbf{x}^*$. □

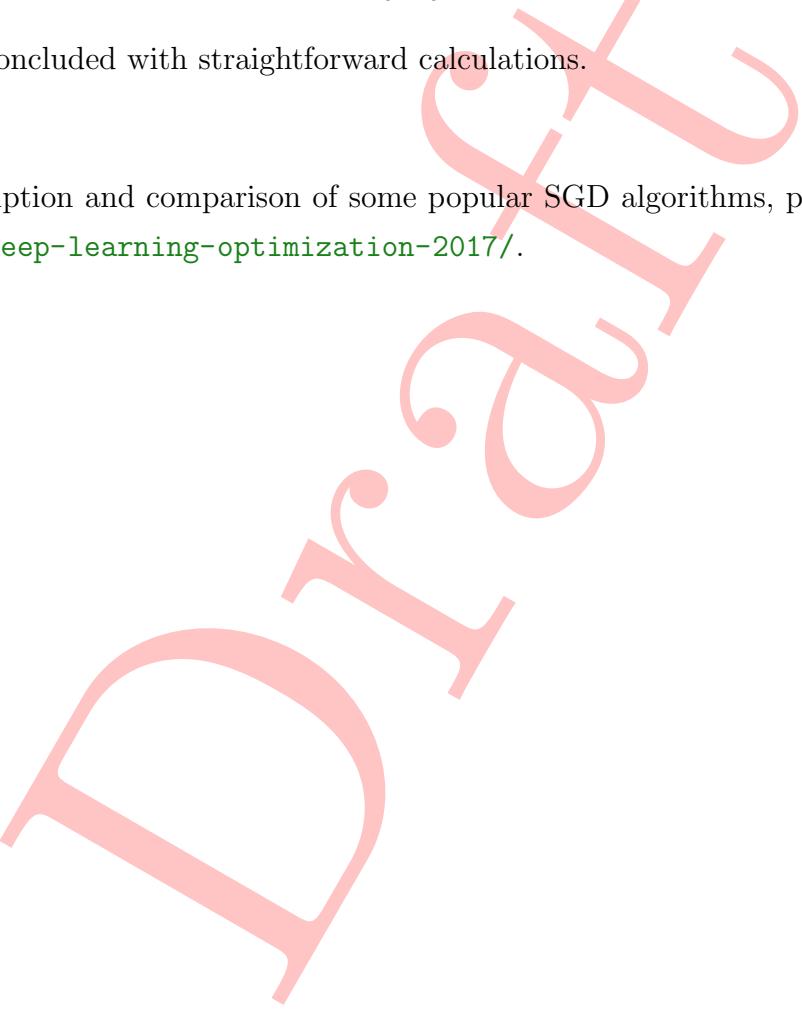
We can now prove Theorem 713.

Proof. In the proof of Theorem 712 we showed that

$$\delta_{s+1} \leq \delta_s - \frac{1}{2 \sum_{i=1}^n \beta_i^\gamma} (\|\nabla f(\mathbf{x}_s)\|_{[1-\gamma]}^*)^2.$$

On the other hand Lemma 714 shows that

$$(\|\nabla f(\mathbf{x}_s)\|_{[1-\gamma]}^*)^2 \geq 2\alpha\delta_s.$$

The proof is concluded with straightforward calculations. 

□

For description and comparison of some popular SGD algorithms, please see <http://ruder.io/deep-learning-optimization-2017/>.

第九章 Acceleration Techniques

(Taken from Chapter 3.7 of [20])

9.1 Nesterov's accelerated gradient descent

We describe here the original Nesterov's method which attains the optimal oracle complexity for smooth convex optimization. We give the details of the method both for the strongly convex and non-strongly convex case. We refer to [123] for a recent interpretation of the method in terms of differential equations, and to [145] for its relation to mirror descent.

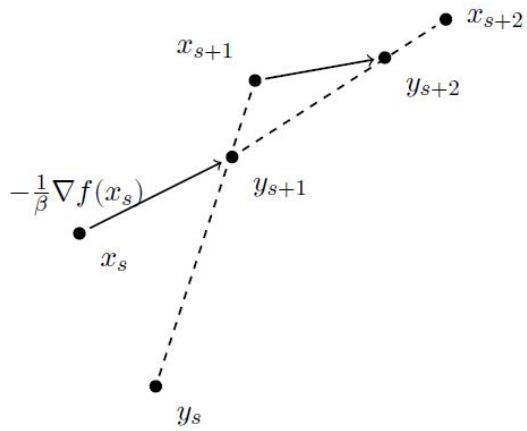


图 9.1: Illustration of Nesterov's accelerated gradient descent

9.1.1 The smooth and strongly convex case

Nesterov's accelerated gradient descent, illustrated in Fig. 9.1, can be described as follows: Start at an arbitrary initial point $\mathbf{x}_1 = \mathbf{y}_1$ and then iterate the following equations for $t \geq 1$,

$$\begin{aligned}\mathbf{y}_{t+1} &= \mathbf{x}_t - \frac{1}{\beta} \nabla f(\mathbf{x}_t), \\ \mathbf{x}_{t+1} &= \left(1 + \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right) \mathbf{y}_{t+1} - \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \mathbf{y}_t.\end{aligned}$$

Theorem 715. Let f be α -strongly convex and β -smooth, then Nesterov's accelerated gradient descent satisfies

$$f(\mathbf{y}_t) - f(\mathbf{x}^*) \leq \frac{\alpha + \beta}{2} \|\mathbf{x}_1 - \mathbf{x}^*\|^2 \exp\left(-\frac{t-1}{\sqrt{\kappa}}\right).$$

Proof. We define α -strongly convex quadratic functions $\Phi_s, s \geq 1$ by induction as follows:

$$\begin{aligned}\Phi_1(\mathbf{x}) &= f(\mathbf{x}_1) + \frac{\alpha}{2} \|\mathbf{x} - \mathbf{x}_1\|^2, \\ \Phi_{s+1}(\mathbf{x}) &= \left(1 - \frac{1}{\sqrt{\kappa}}\right) \Phi_s(\mathbf{x}) \\ &\quad + \frac{1}{\sqrt{\kappa}} \left(f(\mathbf{x}_s) + \nabla f(\mathbf{x}_s)^\top (\mathbf{x} - \mathbf{x}_s) + \frac{\alpha}{2} \|\mathbf{x} - \mathbf{x}_s\|^2\right).\end{aligned}\tag{9.1}$$

Intuitively Φ_s becomes a finer and finer approximation (from below) to f in the following sense:

$$\Phi_{s+1}(\mathbf{x}) \leq f(\mathbf{x}) + \left(1 - \frac{1}{\sqrt{\kappa}}\right)^s (\Phi_1(\mathbf{x}) - f(\mathbf{x})).\tag{9.2}$$

The above inequality can be proved immediately by induction, using the fact that by α -strong convexity one has

$$f(\mathbf{x}_s) + \nabla f(\mathbf{x}_s)^\top (\mathbf{x} - \mathbf{x}_s) + \frac{\alpha}{2} \|\mathbf{x} - \mathbf{x}_s\|^2 \leq f(\mathbf{x}).$$

Equation (9.2) by itself does not say much, for it to be useful one needs to understand how "far" below f is Φ_s . The following inequality answers this question:

$$f(\mathbf{y}_s) \leq \min_{\mathbf{x} \in \mathbb{R}^n} \Phi_s(\mathbf{x}).\tag{9.3}$$

The rest of the proof is devoted to showing that (9.3) holds true, but first let us see how to combine (9.2) and (9.3) to obtain the rate given by the theorem (we use that by β -smoothness one has $f(\mathbf{x}) - f(\mathbf{x}^*) \leq \frac{\beta}{2} \|\mathbf{x} - \mathbf{x}^*\|^2$):

$$\begin{aligned}f(\mathbf{y}_t) - f(\mathbf{x}^*) &\leq \Phi_t(\mathbf{x}^*) - f(\mathbf{x}^*) \\ &\leq \left(1 - \frac{1}{\sqrt{\kappa}}\right)^{t-1} (\Phi_1(\mathbf{x}^*) - f(\mathbf{x}^*)) \\ &\leq \frac{\alpha + \beta}{2} \|\mathbf{x}_1 - \mathbf{x}^*\|^2 \left(1 - \frac{1}{\sqrt{\kappa}}\right)^{t-1}.\end{aligned}$$

We now prove (9.3) by induction (note that it is true at $s = 1$ since $\mathbf{x}_1 = \mathbf{y}_1$). Let $\Phi_s^* = \min_{\mathbf{x} \in \mathbb{R}^n} \Phi_s(\mathbf{x})$. Using the definition of \mathbf{y}_{s+1} (and β -smoothness), convexity, and the

induction hypothesis, one gets

$$\begin{aligned}
 f(\mathbf{y}_{s+1}) &\leq f(\mathbf{x}_s) - \frac{1}{2\beta} \|\nabla f(\mathbf{x}_s)\|^2 \\
 &= \left(1 - \frac{1}{\sqrt{\kappa}}\right) f(\mathbf{y}_s) + \left(1 - \frac{1}{\sqrt{\kappa}}\right) (f(\mathbf{x}_s) - f(\mathbf{y}_s)) \\
 &\quad + \frac{1}{\sqrt{\kappa}} f(\mathbf{x}_s) - \frac{1}{2\beta} \|\nabla f(\mathbf{x}_s)\|^2 \\
 &\leq \left(1 - \frac{1}{\sqrt{\kappa}}\right) \Phi_s^* + \left(1 - \frac{1}{\sqrt{\kappa}}\right) \nabla f(\mathbf{x}_s)^\top (\mathbf{x}_s - \mathbf{y}_s) \\
 &\quad + \frac{1}{\sqrt{\kappa}} f(\mathbf{x}_s) - \frac{1}{2\beta} \|\nabla f(\mathbf{x}_s)\|^2.
 \end{aligned}$$

Thus we now have to show that

$$\begin{aligned}
 \Phi_{s+1}^* &\geq \left(1 - \frac{1}{\sqrt{\kappa}}\right) \Phi_s^* + \left(1 - \frac{1}{\sqrt{\kappa}}\right) \nabla f(\mathbf{x}_s)^\top (\mathbf{x}_s - \mathbf{y}_s) \\
 &\quad + \frac{1}{\sqrt{\kappa}} f(\mathbf{x}_s) - \frac{1}{2\beta} \|\nabla f(\mathbf{x}_s)\|^2.
 \end{aligned} \tag{9.4}$$

To prove this inequality we have to understand better the functions Φ_s . First note that $\nabla^2 \Phi_s(\mathbf{x}) = \alpha \mathbf{I}_n$ (immediate by induction) and thus Φ_s has to be of the following form:

$$\Phi_s(\mathbf{x}) = \Phi_s^* + \frac{\alpha}{2} \|\mathbf{x} - \mathbf{v}_s\|^2,$$

for some $\mathbf{v}_s \in \mathbb{R}^n$. Now observe that by differentiating (9.1) and using the above form of Φ_s one obtains

$$\nabla \Phi_{s+1}(\mathbf{x}) = \alpha \left(1 - \frac{1}{\sqrt{\kappa}}\right) (\mathbf{x} - \mathbf{v}_s) + \frac{1}{\sqrt{\kappa}} \nabla f(\mathbf{x}_s) + \frac{\alpha}{\sqrt{\kappa}} (\mathbf{x} - \mathbf{x}_s).$$

In particular Φ_{s+1} is by definition minimized at \mathbf{v}_{s+1} which can now be defined by induction using the above identity, precisely:

$$\mathbf{v}_{s+1} = \left(1 - \frac{1}{\sqrt{\kappa}}\right) \mathbf{v}_s + \frac{1}{\sqrt{\kappa}} \mathbf{x}_s - \frac{1}{\alpha \sqrt{\kappa}} \nabla f(\mathbf{x}_s). \tag{9.5}$$

Using the form of Φ_s and Φ_{s+1} , as well as the original definition (9.1) one gets the following identity by evaluating Φ_{s+1} at \mathbf{x}_s :

$$\begin{aligned}
 \Phi_{s+1}^* + \frac{\alpha}{2} \|\mathbf{x}_s - \mathbf{v}_{s+1}\|^2 \\
 &= \left(1 - \frac{1}{\sqrt{\kappa}}\right) \Phi_s^* + \frac{\alpha}{2} \left(1 - \frac{1}{\sqrt{\kappa}}\right) \|\mathbf{x}_s - \mathbf{v}_s\|^2 + \frac{1}{\sqrt{\kappa}} f(\mathbf{x}_s).
 \end{aligned} \tag{9.6}$$

Note that thanks to (9.5) one has

$$\begin{aligned}
 \|\mathbf{x}_s - \mathbf{v}_{s+1}\|^2 &= \left(1 - \frac{1}{\sqrt{\kappa}}\right)^2 \|\mathbf{x}_s - \mathbf{v}_s\|^2 + \frac{1}{\alpha^2 \kappa} \|\nabla f(\mathbf{x}_s)\|^2 \\
 &\quad - \frac{2}{\alpha \sqrt{\kappa}} \left(1 - \frac{1}{\sqrt{\kappa}}\right) \nabla f(\mathbf{x}_s)^\top (\mathbf{v}_s - \mathbf{x}_s),
 \end{aligned}$$

which combined with (9.6) yields

$$\begin{aligned}\Phi_{s+1}^* &= \left(1 - \frac{1}{\sqrt{\kappa}}\right)\Phi_s^* + \frac{1}{\sqrt{\kappa}}f(\mathbf{x}_s) + \frac{\alpha}{2\sqrt{\kappa}}\left(1 - \frac{1}{\sqrt{\kappa}}\right)\|\mathbf{x}_s - \mathbf{v}_s\|^2 \\ &\quad - \frac{1}{2\beta}\|\nabla f(\mathbf{x}_s)\|^2 + \frac{1}{\sqrt{\kappa}}\left(1 - \frac{1}{\sqrt{\kappa}}\right)\nabla f(\mathbf{x}_s)^\top(\mathbf{v}_s - \mathbf{x}_s).\end{aligned}$$

Finally we show by induction that $\mathbf{v}_s - \mathbf{x}_s = \sqrt{\kappa}(\mathbf{x}_s - \mathbf{y}_s)$, which concludes the proof of (9.4) and thus also concludes the proof of the theorem:

$$\begin{aligned}\mathbf{v}_{s+1} - \mathbf{x}_{s+1} &= \left(1 - \frac{1}{\sqrt{\kappa}}\right)\mathbf{v}_s + \frac{1}{\sqrt{\kappa}}\mathbf{x}_s - \frac{1}{\alpha\sqrt{\kappa}}\nabla f(\mathbf{x}_s) - \mathbf{x}_{s+1} \\ &= \sqrt{\kappa}\mathbf{x}_s - (\sqrt{\kappa} - 1)\mathbf{y}_s - \frac{\sqrt{\kappa}}{\beta}\nabla f(\mathbf{x}_s) - \mathbf{x}_{s+1} \\ &= \sqrt{\kappa}\mathbf{y}_{s+1} - (\sqrt{\kappa} - 1)\mathbf{y}_s - \mathbf{x}_{s+1} \\ &= \sqrt{\kappa}(\mathbf{x}_{s+1} - \mathbf{y}_{s+1}),\end{aligned}$$

where the first equality comes from (9.5), the second from the induction hypothesis, the third from the definition of \mathbf{y}_{s+1} and the last one from the definition of \mathbf{x}_{s+1} . □

9.1.2 The smooth case

In this section we show how to adapt Nesterov's accelerated gradient descent for the case $\alpha = 0$, using a time-varying combination of the elements in the primary sequence $\{\mathbf{y}_t\}$. First we define the following sequences:

$$\lambda_0 = 0, \lambda_t = \frac{1 + \sqrt{1 + 4\lambda_{t-1}^2}}{2}, \text{ and } \gamma_t = \frac{1 - \lambda_t}{\lambda_{t+1}}.$$

(Note that $\gamma_t \leq 0$.) Now the algorithm is simply defined by the following equations, with $\mathbf{x}_1 = \mathbf{y}_1$ an arbitrary initial point,

$$\begin{aligned}\mathbf{y}_{t+1} &= \mathbf{x}_t - \frac{1}{\beta}\nabla f(\mathbf{x}_t), \\ \mathbf{x}_{t+1} &= (1 - \gamma_t)\mathbf{y}_{t+1} + \gamma_t\mathbf{y}_t.\end{aligned}$$

Theorem 716. *Let f be a convex and β -smooth function, then Nesterov's accelerated gradient descent satisfies*

$$f(\mathbf{y}_t) - f(\mathbf{x}^*) \leq \frac{2\beta\|\mathbf{x}_1 - \mathbf{x}^*\|^2}{t^2}.$$

We follow here the proof of [9]. We also refer to [126] for a proof with simpler step-sizes.

Proof. Using the unconstrained version of Lemma:

Lemma 717. let $\mathbf{x}, \mathbf{y} \in \mathcal{X}$, $\mathbf{x}^+ = \prod_{\mathcal{X}} \left(\mathbf{x} - \frac{1}{\beta} \nabla f(\mathbf{x}) \right)$, and $g_{\mathcal{X}}(\mathbf{x}) = \beta(\mathbf{x} - \mathbf{x}^+)$. Then the following holds true:

$$f(\mathbf{x}^+) - f(\mathbf{y}) \leq g_{\mathcal{X}}(\mathbf{x})^\top (\mathbf{x} - \mathbf{y}) - \frac{1}{2\beta} \|g_{\mathcal{X}}(\mathbf{x})\|^2.$$

One obtains

$$\begin{aligned} & f(\mathbf{y}_{s+1}) - f(\mathbf{y}_s) \\ & \leq \nabla f(\mathbf{x}_s)^\top (\mathbf{x}_s - \mathbf{y}_s) - \frac{1}{2\beta} \|\nabla f(\mathbf{x}_s)\|^2 \\ & = \beta(\mathbf{x}_s - \mathbf{y}_{s+1})^\top (\mathbf{x}_s - \mathbf{y}_s) - \frac{\beta}{2} \|\mathbf{x}_s - \mathbf{y}_{s+1}\|^2. \end{aligned} \tag{9.7}$$

Similarly, we also get

$$f(\mathbf{y}_{s+1}) - f(\mathbf{x}^*) \leq \beta(\mathbf{x}_s - \mathbf{y}_{s+1})^\top (\mathbf{x}_s - \mathbf{x}^*) - \frac{\beta}{2} \|\mathbf{x}_s - \mathbf{y}_{s+1}\|^2. \tag{9.8}$$

Now multiplying (9.7) by $(\lambda_s - 1)$ and adding the result to (9.8), one obtains with $\delta_s = f(\mathbf{y}_s) - f(\mathbf{x}^*)$,

$$\begin{aligned} & \lambda_s \delta_{s+1} - (\lambda_s - 1) \delta_s \\ & \leq \beta(\mathbf{x}_s - \mathbf{y}_{s+1})^\top (\lambda_s \mathbf{x}_s - (\lambda_s - 1) \mathbf{y}_s - \mathbf{x}^*) - \frac{\beta}{2} \lambda_s \|\mathbf{x}_s - \mathbf{y}_{s+1}\|^2. \end{aligned}$$

Multiplying this inequality by λ_s and using that by definition $\lambda_{s-1}^2 = \lambda_s^2 - \lambda_s$, as well as the elementary identity $2\mathbf{a}^\top \mathbf{b} - \|\mathbf{a}\|^2 = \|\mathbf{b}\|^2 - \|\mathbf{b} - \mathbf{a}\|^2$, one obtains

$$\begin{aligned} & \lambda_s^2 \delta_{s+1} - \lambda_{s-1}^2 \delta_s \\ & \leq \frac{\beta}{2} (2\lambda_s(\mathbf{x}_s - \mathbf{y}_{s+1})^\top (\lambda_s \mathbf{x}_s - (\lambda_s - 1) \mathbf{y}_s - \mathbf{x}^*) \\ & \quad - \|\lambda_s(\mathbf{y}_{s+1} - \mathbf{x}_s)\|^2) \\ & = \frac{\beta}{2} (\|\lambda_s \mathbf{x}_s - (\lambda_s - 1) \mathbf{y}_s - \mathbf{x}^*\|^2 \\ & \quad - \|\lambda_s \mathbf{y}_{s+1} - (\lambda_s - 1) \mathbf{y}_s - \mathbf{x}^*\|^2). \end{aligned} \tag{9.9}$$

Next remark that, by definition, one has

$$\begin{aligned} & \mathbf{x}_{s+1} = \mathbf{y}_{s+1} + \gamma_s(\mathbf{y}_s - \mathbf{y}_{s+1}) \\ & \Leftrightarrow \lambda_{s+1} \mathbf{x}_{s+1} = \lambda_{s+1} \mathbf{y}_{s+1} + (1 - \lambda_s)(\mathbf{y}_s - \mathbf{y}_{s+1}) \\ & \Leftrightarrow \lambda_{s+1} \mathbf{x}_{s+1} - (\lambda_{s+1} - 1) \mathbf{y}_{s+1} = \lambda_s \mathbf{y}_{s+1} - (\lambda_s - 1) \mathbf{y}_s. \end{aligned} \tag{9.10}$$

Putting together (9.9) and (9.10) one gets with $\mathbf{u}_s = \lambda_s \mathbf{x}_s - (\lambda_s - 1) \mathbf{y}_s - \mathbf{x}^*$,

$$\lambda_s^2 \delta_{s+1} - \lambda_{s-1}^2 \delta_s^2 \leq \frac{\beta}{2} (\|\mathbf{u}_s\|^2 - \|\mathbf{u}_{s+1}\|^2).$$

Summing these inequalities from $s = 1$ to $s = t - 1$ one obtains:

$$\delta_t \leq \frac{\beta}{2\lambda_{t-1}^2} \|\mathbf{u}_1\|^2.$$

By induction it is easy to see that $\lambda_{t-1} \geq \frac{t}{2}$ which concludes the proof. □

9.1.2.1 Exercises

Exercise 718. Show that the γ_t in algorithm in Section 9.1.2 can also be written as $\gamma_k = \theta_k(1 - \theta_{k-1})/\theta_{k-1}$, where

$$\theta_0 = 1, \quad \text{and} \quad \theta_{k+1} = \left(\sqrt{\theta_k^4 + 4\theta_k^2} - \theta_k^2 \right) / 2.$$

第十章 Multiobjective Optimization

(Taken from Chapter 24 of [30].)

10.1 Introduction

When an optimization problem involves only one objective function, it is a single-objective optimization. Most engineering problems require the designer to optimize a number of conflicting objectives. The objectives are in conflict with each other if an improvement in one objective leads to deterioration in another. Multiobjective problems in which there is competition between objectives may have no single, unique optimal solution. Multiobjective optimization problems are also referred to as multicriteria or vector optimization problems. We can formulate a multiobjective optimization problem as follows: Find a decision variable that satisfies the given constraints and optimizes a vector function whose components are objective functions. Formally, the multiobjective optimization problem is stated as follows

$$\min_{\mathbf{x}} \mathbf{f}(\mathbf{x}) = \begin{pmatrix} f_1(x_1, x_2, \dots, x_n) \\ f_2(x_1, x_2, \dots, x_n) \\ \vdots \\ f_l(x_1, x_2, \dots, x_n) \end{pmatrix}$$

s.t. $\mathbf{x} \in \Omega.$

where $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^l$ and $\Omega \subseteq \mathbb{R}^n$. For example, the constraint set Ω can have the form

$$\Omega = \{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}, \quad (10.1)$$

where

$$\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m, \quad \mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^p, \quad m \leq n. \quad (10.2)$$

In general, we may have three different types of multiobjective optimization problems:

- Minimize all the objective functions.
- Maximize all the objective functions.
- Minimize some and maximize others.

However, as usual, any of these can be converted into an equivalent minimization problem.

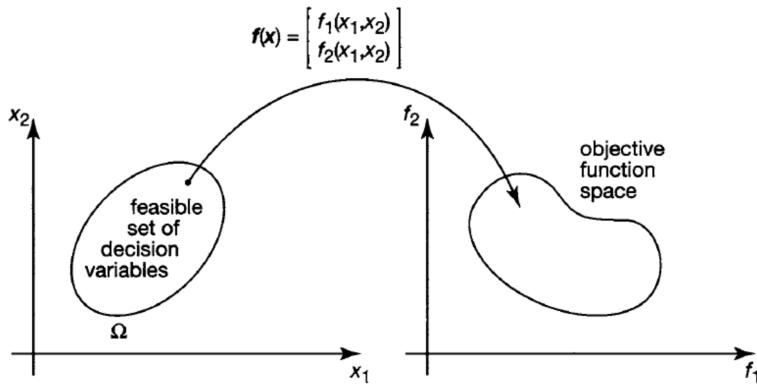


图 10.1: Two-dimensional illustration of a multiobjective vector function assigning to each decision variable a multiobjective vector function value.

10.2 Pareto Solutions

Note that multiobjective function assigns to each decision variable a multiobjective vector function value in the objective function space. A graphical illustration of this statement is illustrated in Figures 10.1 and 10.2. In Figure 10.1 the decision variable is a point $\mathbf{x} \in \mathbb{R}^2$ while the vector of objective functions is given by $\mathbf{f} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$. In Figure 10.2 the decision variable is a point $\mathbf{x} \in \mathbb{R}^2$ while the vector of objective functions is $\mathbf{f} : \mathbb{R}^2 \rightarrow \mathbb{R}^3$. In single-objective optimization problems our goal is to find a single solution, where we focus mainly on the decision variable space. On the other hand, in multiobjective problems we are usually more interested in the objective space. As pointed out by [99] (page 11), multiobjective problems are in a sense ill-defined because there is no natural ordering in the objective space. Miettinen [99] illustrates this statement with the following simple example. One can say that $[1, 1]^T$ is less than $[3, 3]^T$. But how do we compare $[1, 3]^T$ and $[3, 1]^T$? In general, in multiobjective optimization problems our goal is to find good compromises. Roughly speaking, in a multiobjective optimization problem, a solution is optimal if there exists no other solution, within the feasible set, that gives improved performance with regard to all the objectives. A formal definition of an optimal point for a multiobjective optimization problem was proposed by Francis Y. Edgeworth in 1881 and generalized by Vilfredo Pareto in 1896. It is customary now to refer to an optimal point of a multiobjective optimization problem as the Pareto minimizer, whose formal definition is given next.

Definition 719. Let $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^l$ and $\mathbf{x} \in \Omega$ be given. For the optimization problem

$$\min_{\mathbf{x}} \quad \mathbf{f}(\mathbf{x})$$

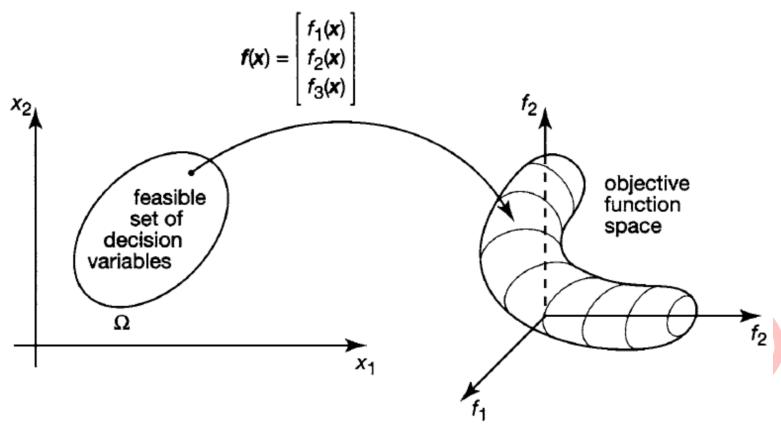


图 10.2: Three-dimensional illustration of a multiobjective vector function assigning to each decision variable a multiobjective vector function value.

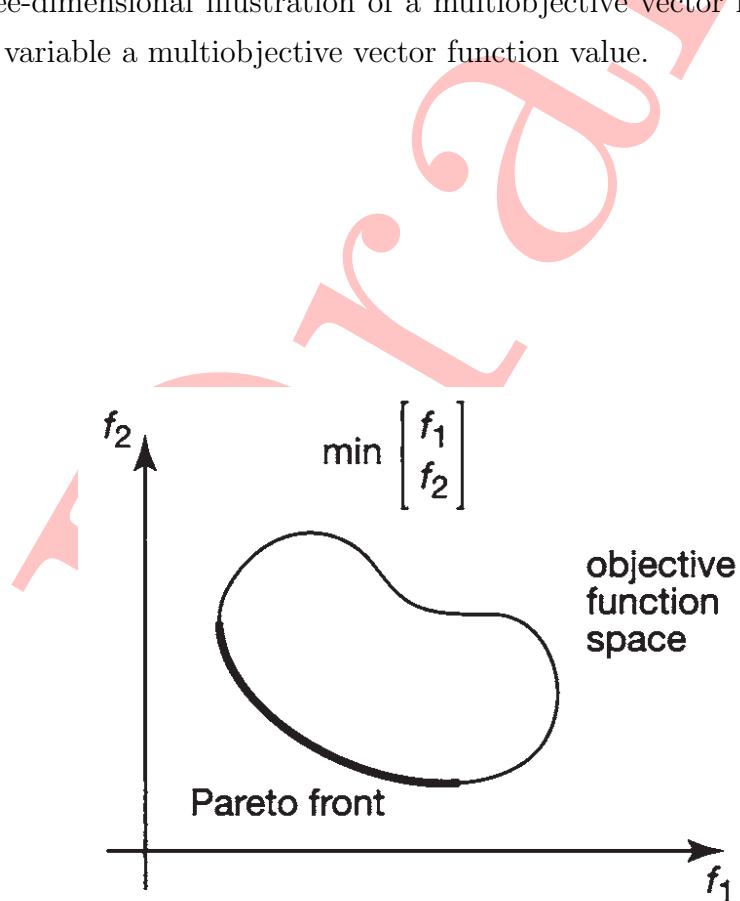


图 10.3: The Pareto front is marked with a heavy line.

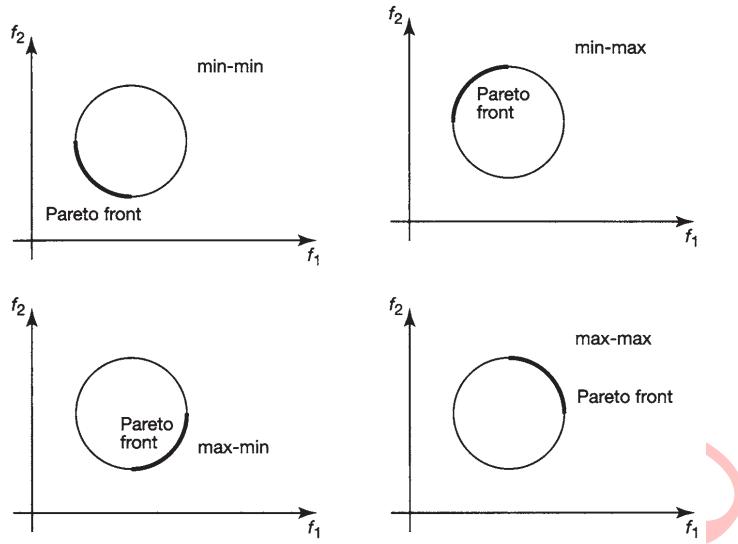


图 10.4: Pareto fronts for four possible cases of two-objective optimization.

$$s.t. \quad \mathbf{x} \in \Omega, \quad (10.3)$$

a point $\mathbf{x}^* \in \Omega$ is called a Pareto minimizer if there exists no $\mathbf{x} \in \Omega$ such that for $i = 1, 2, \dots, l$,

$$f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*), \quad (10.4)$$

and for at least one i ,

$$f_i(\mathbf{x}) < f_i(\mathbf{x}^*). \quad (10.5)$$

In other words, the point \mathbf{x}^* is a Pareto minimizer, or a nondominated solution, if there exists no other feasible decision variable \mathbf{x} that would decrease some objectives without causing simultaneous increase in at least one other variable.

The set of Pareto minimizers (optimizers) is called the Pareto front, as illustrated in Figure 10.3. Most multiobjective optimization algorithms use the concept of domination. A solution is said to be nondominated if it is Pareto optimal.

In Figure 10.4 we show different combinations of two-objective optimization and the corresponding Pareto fronts. In particular, in the upper left, we show the Pareto front for the case when we are minimizing both components of the objective function vector, which we represent by “min-min.” Similarly, “min-max” represents the case when we are minimizing the first objective function and maximizing the second; and so forth.

10.3 Computing the Pareto Front

When computing the Pareto front, two solutions are compared and the dominated solution is eliminated from the set of candidates of Pareto optimizers. Thus, the Pareto front consists of nondominated solutions.

To proceed, we need some notation. Let

$$\mathbf{x}^{*r} = [x_1^{*r}, x_2^{*r}, \dots, x_n^{*r}]^T \quad (10.6)$$

be the r -th candidate Pareto optimal solution, $r = 1, 2, \dots, R$, where R is the number of current candidate Pareto solutions. Let

$$(\mathbf{x}^{*r}) = [\mathbf{f}_1(\mathbf{x}^{*r}), \mathbf{f}_2(\mathbf{x}^{*r}), \dots, \mathbf{f}_l(\mathbf{x}^{*r})]^T \quad (10.7)$$

be the corresponding value of the objective function vector. For any new solution candidate \mathbf{x}^j , we evaluate the objective function vector $\mathbf{f}(\mathbf{x}^j)$. We then compare the new solution candidate with the existing Pareto solutions.

We need to consider three cases:

- \mathbf{x}^j dominates at least one candidate solution.
- \mathbf{x}^j does not dominate any existing candidate solutions.
- \mathbf{x}^j is dominated by a candidate solution.

If \mathbf{x}^j dominates at least one candidate solution, we delete the dominated solutions from the set and add the new solution \mathbf{x}^j to the set of candidates. In the second case, when the new candidate solution \mathbf{x}^j does not dominate any of the existing candidate Pareto solutions, add this new Pareto solution to the set of candidate Pareto solutions. Finally, in the third case, when the new candidate solution is dominated by at least one of the existing candidate Pareto solutions, we do not change the set of the existing candidate Pareto solutions.

Example 720. Consider the two-objective minimization problem whose data are as follows:

$\mathbf{x}^{(i)T}$	$(\mathbf{x}^{(i)})^T$
[5, 6]	[30, 45]
[4, 5]	[22, 29]
[3, 7]	[19, 53]
[6, 8]	[41, 75]
[1, 4]	[13, 45]
[6, 7]	[42, 55]
[2, 5]	[37, 46]
[3, 6]	[28, 37]
[2, 7]	[12, 51]
[4, 7]	[41, 67]

Suppose that we wish to find nondominated pairs for this problem. Recall that a point \mathbf{x}^* is a nondominated point if for all i and all \mathbf{x} ,

$$f_i(\mathbf{x}^*) \leq f_i(\mathbf{x}), \quad (10.8)$$

and at least for one component j of the objective vector, we have

$$f_j(\mathbf{x}^*) < f_j(\mathbf{x}). \quad (10.9)$$

To find the Pareto front, we start with the first pair as a candidate Pareto optimal solution and then compare the other pairs against this first pair, replacing the first pair as necessary. We then continue with the other pairs, building up a set of candidate Pareto solutions and modifying this set when appropriate. The result of the search gives the following Pareto optimal set:

$\mathbf{x}^{(i)T}$	$(\mathbf{x}^{(i)})^T$
[4, 5]	[22, 29]
[1, 4]	[13, 45]
[2, 7]	[12, 51]

We now discuss an algorithm for generating the Pareto front that implements the foregoing ideas. This algorithm is a minor modification of the algorithm of [105] (pp. 100-101). We use the following notation. Let J be the number of candidate solutions to be checked for optimality, while R is the number of current candidate Pareto solutions. Recall that ℓ is the number of objective functions, the dimension of the objective function

vector, and n is the dimension of the decision space, that is, the number of components of \mathbf{x} . The algorithm consists of eight steps.

Algorithm for Generating a Pareto Front

1. Generate an initial solution \mathbf{x}^1 and evaluate ${}^{*1} = (\mathbf{x}^1)$. This first solution generated is taken as a candidate Pareto solution. Set initial indices $R := 1$ and $j := 1$.
2. Set $j := j + 1$. If $j \leq J$, then generate solution \mathbf{x}^j and go to step 3. Otherwise, stop, because all the candidate solutions have already been considered.
3. Set $r := 1$ and $q := 0$ (q represents the number of eliminated solutions from the existing set of Pareto solutions).
4. If for all $i = 1, 2, \dots, \ell$,

$$f_i(\mathbf{x}^j) < f_i(\mathbf{x}^{*r}), \quad (10.10)$$

then set $q := q + 1$, ${}^{*R} := (\mathbf{x}^j)$, remember the solution that should be eliminated, and go to step 6.

5. If for all $i = 1, 2, \dots, \ell$,

$$f_i(\mathbf{x}^j) \geq f_i(\mathbf{x}^{*r}), \quad (10.11)$$

then go to step 2.

6. Set $r := r + 1$. If $r < R$, go to step 4.
7. If $q \neq 0$, remove from the candidate Pareto set the solutions that are eliminated in step 4, add solution \mathbf{x}^j as a new candidate Pareto solution, and go to step 2.
8. Set $R := R + 1$, ${}^{*R} := \mathbf{x}^j$, ${}^{*R} := (\mathbf{x}^j)$, and go to step 2.

Example 721. We apply the algorithm above to generate the Pareto front for the multi-objective optimization problem

$$\min_{x_1, x_2} \begin{pmatrix} -(x_1^2 + x_2) \\ x_1 + x_2^2 \end{pmatrix} \quad (10.12)$$

$$\text{s.t. } \begin{aligned} 2 \leq x_1 &\leq 6, \\ 5 \leq x_2 &\leq 9. \end{aligned} \quad (10.13)$$

We performed 100 iterations. At each iteration we randomly generated 50 feasible points. Then we applied the algorithm above to extract from this set of feasible points candidate Pareto optimal solutions. In Figure 10.5 we show Pareto optimal points obtained after 100 iterations of the algorithm. We also show level sets of the objective functions in the (x_1, x_2) -space. In Figure 10.6 we show the Pareto front in the objective function space after 100 iterations of the algorithm. The Pareto optimal points are marked with

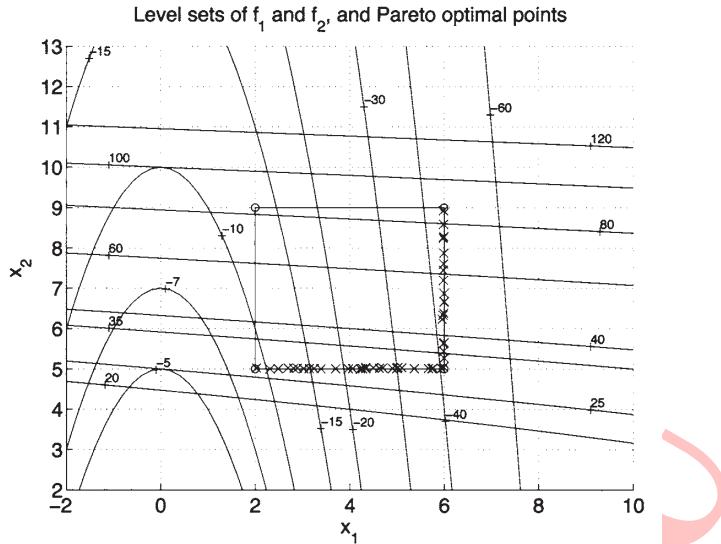


图 10.5: Pareto optimal points in the decision space along with the level sets of the objective functions f_1 and f_2 .

\times 's. The points marked with \cdot 's are the candidate solutions generated randomly at the beginning of the last iteration of the algorithm.

We have described a simple approach to computing the Pareto front. Alternative methods include those that apply genetic algorithms to solving multiobjective optimization problems, as discussed in [32, 33, 105].

10.4 From Multiobjective to Single-Objective Optimization

In some cases it is possible to deal with a multiobjective optimization problem by converting the problem into a single-objective optimization problem, so that standard optimization methods can be brought to bear. Here, we discuss four techniques to convert a multiobjective problem to a singleobjective problem. We assume throughout that an objective function vector $(\mathbf{x}) = [\mathbf{f}_1(\mathbf{x}), \dots, \mathbf{f}_\ell(\mathbf{x})]^T$ is given.

The first method is to form a single objective function by taking a linear combination, with positive coefficients, of the components of the objective function vector. Equivalently, we form a convex combination of the components of the objective function vector. In other words, we use

$$f(\mathbf{x}) = \mathbf{c}^T(\mathbf{x}) \quad (10.14)$$

as the (single) objective function, where \mathbf{c} is a vector of positive components. This method is also called the weighted-sum method, where the coefficients of the linear combination

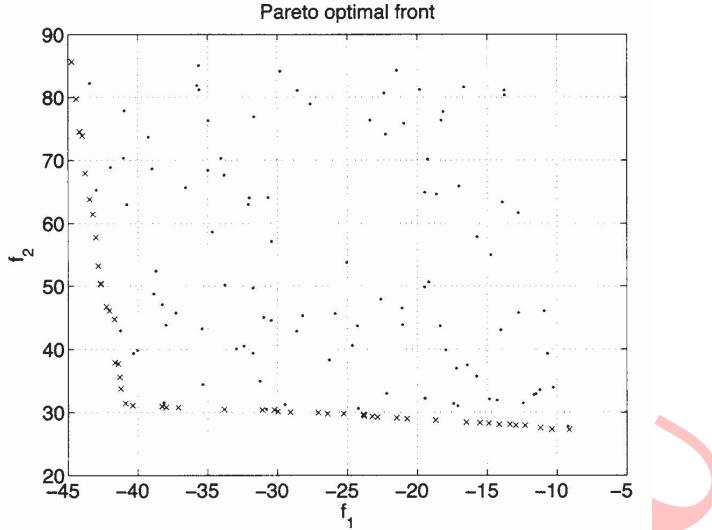


图 10.6: Pareto front for the problem of Example 24.2. Also marked are the objective vector values for the remaining candidate points generated in the last iteration.

(i.e., the components of \mathbf{c}) are called weights. These weights reflect the relative importance of the individual components in the objective vector. Of course, it might be difficult to determine suitable weight values.

A second method is to form a single objective function by taking the maximum of the components of the objective vector:

$$f(\mathbf{x}) = \max\{f_1(\mathbf{x}), \dots, f_\ell(\mathbf{x})\}. \quad (10.15)$$

In other words, we convert the multiobjective minimization problem into one of minimizing the maximum of the components. For this reason, it is also called the minimax method. Note that this method applies to situations where the components of the objective vector are comparable or compatible, in the sense that they are in the same units (e.g., they are all lengths measured in meters, or masses in kilograms). A limitation of this method is that the resulting single objective function might not be differentiable, thereby precluding the use of optimization methods that rely on differentiability (e.g., gradient algorithms). However, as we show in the following, a minimax problem with linear objective vector components and linear constraints can be reduced to a linear programming problem.

Example 722. Given vectors $\mathbf{v}_1, \dots, \mathbf{v}_p \in \mathbb{R}^n$ and scalars u_1, \dots, u_p , consider the minimax problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & \max\{\mathbf{v}_1^T \mathbf{x} + u_1, \dots, \mathbf{v}_p^T \mathbf{x} + u_p\} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b}, \end{aligned} \quad (10.16)$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$. Call this problem P1.

a. Consider the optimization problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & y \\ \text{s.t.} \quad & \mathbf{Ax} \leq \mathbf{b} \\ & y \geq \mathbf{v}_i^T \mathbf{x} + u_i, \quad i = 1, \dots, p, \end{aligned} \tag{10.17}$$

where the decision variable is the vector $[\mathbf{x}^T, y]^T$. Call this problem P2. Show that \mathbf{x}^* solves P1 if and only if $[\mathbf{x}^{*T}, y^*]^T$ with $y^* = \max\{\mathbf{v}_1^T \mathbf{x}^* + u_1, \dots, \mathbf{v}_p^T \mathbf{x}^* + u_p\}$ solves P2.

Hint: $y \geq \max\{a, b, c\}$ if and only if $y \geq a, y \geq b$, and $y \geq c$.

b. Use part a to derive a linear programming problem

$$\begin{aligned} \min_{\mathbf{z}} \quad & \hat{\mathbf{c}}^T \mathbf{z} \\ \text{s.t.} \quad & \hat{\mathbf{A}} \mathbf{z} \leq \hat{\mathbf{b}}, \end{aligned} \tag{10.18}$$

that is equivalent to P1 (by “equivalent” we mean that the solution to one gives us the solution to the other). Explain how a solution to the linear programming problem above gives a solution to P1.

Solution. a. First suppose that \mathbf{x}^* is optimal in P1. Let $y^* = \max\{\mathbf{v}_1^T \mathbf{x} + u_1, \dots, \mathbf{v}_p^T \mathbf{x} + u_p\}$. Then, $[\mathbf{x}^{*T}, y^*]^T$ is feasible in P2. Let $[\mathbf{x}^T, y]^T$ be any feasible point in P2. Then (by the hint)

$$y \geq \max\{\mathbf{v}_1^T \mathbf{x} + u_1, \dots, \mathbf{v}_p^T \mathbf{x} + u_p\}. \tag{10.19}$$

Moreover, \mathbf{x} is feasible in P1, and hence

$$\begin{aligned} y &\geq \max\{\mathbf{v}_1^T \mathbf{x} + u_1, \dots, \mathbf{v}_p^T \mathbf{x} + u_p\} \\ &\geq \max\{\mathbf{v}_1^T \mathbf{x}^* + u_1, \dots, \mathbf{v}_p^T \mathbf{x}^* + u_p\} \\ &= y^*. \end{aligned} \tag{10.20}$$

Hence, $[\mathbf{x}^{*T}, y^*]^T$ is optimal in the linear programming problem.

To prove the converse, suppose that \mathbf{x}^* is not optimal in P1. Then, there is some \mathbf{x}' that is feasible in P1 such that

$$\begin{aligned} y' &= \max\{\mathbf{v}_1^T \mathbf{x}' + u_1, \dots, \mathbf{v}_p^T \mathbf{x}' + u_p\} \\ &< \max\{\mathbf{v}_1^T \mathbf{x}^* + u_1, \dots, \mathbf{v}_p^T \mathbf{x}^* + u_p\} \\ &= y^*. \end{aligned} \tag{10.21}$$

But $[\mathbf{x}^T, y]^T$ is evidently feasible in P2, and has objective function value (y') that is lower than that of $[\mathbf{x}^{*T}, y^*]^T$. Hence, $[\mathbf{x}^{*T}, y^*]^T$ is not optimal in P2.

b. Define

$$\mathbf{z} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \hat{\mathbf{c}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \hat{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & 0 \\ \mathbf{v}_1^T & -1 \\ \vdots & \vdots \\ \mathbf{v}_p^T & -1 \end{bmatrix}, \quad \hat{\mathbf{b}} = \begin{bmatrix} \mathbf{b} \\ -u_1 \\ \vdots \\ -u_p \end{bmatrix}. \quad (10.22)$$

Then the equivalent problem can be written as

$$\begin{aligned} \min_{\mathbf{z}} \quad & \hat{\mathbf{c}}^T \mathbf{z} \\ \text{s.t.} \quad & \hat{\mathbf{A}} \mathbf{z} \leq \hat{\mathbf{b}}. \end{aligned} \quad (10.23)$$

By part a, if we obtain a solution to this linear programming problem, then the first n components form a solution to the original minimax problem.

□

A third method to convert a multiobjective problem to a single-objective problem, assuming that the components of the objective vector are nonnegative, is to form a single objective function by taking the p -norm of the objective vector:

$$f(\mathbf{x}) = \|(\mathbf{x})\|_p. \quad (10.24)$$

The minimax method can be viewed as a special case of this method, with $p = \infty$. The weighted-sum method with uniform weights can be viewed as this method with $p = 1$. To make the objective function differentiable in the case where p is finite (so that we can apply gradient methods, for example), we replace it by its p th power:

$$f(\mathbf{x}) = \|(\mathbf{x})\|_p^p = (f_1(\mathbf{x}))^p + \cdots + (f_\ell(\mathbf{x}))^p. \quad (10.25)$$

A fourth method is to minimize one of the components of the objective vector subject to constraints on the other components. For example, given , we solve

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_1(\mathbf{x}) \\ \text{s.t.} \quad & f_2(\mathbf{x}) \leq b_2, \\ & \vdots \\ & f_\ell(\mathbf{x}) \leq b_\ell, \end{aligned} \quad (10.26)$$

where b_2, \dots, b_ℓ are given constants that reflect satisfactory values for the objectives f_2, \dots, f_ℓ , respectively. Of course, this approach is suitable only in situations where these satisfactory values can be determined.

10.5 Uncertain Linear Programming Problems

In this section we show how multiobjective optimization methods can be used to solve linear programming problems with uncertain coefficients, including uncertain constraints and uncertain objective functions.

Uncertain Constraints

We first consider a generalization of linear programming to problems with uncertain constraints. Our exposition is based on a discussion of fuzzy linear programming by [129] (Chapter 30). We consider the following general linear programming problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq 0. \end{aligned} \tag{10.27}$$

We can represent the constraints in the form

$$(\mathbf{A}\mathbf{x})_i \leq \mathbf{b}_i, \quad i = 1, 2, \dots, m. \tag{10.28}$$

Suppose that the constraints' bounds are uncertain in the sense that they can vary within given tolerance values and can be represented as

$$(\mathbf{A}\mathbf{x})_i \leq \mathbf{b}_i + \theta t_i, \quad i = 1, 2, \dots, m. \tag{10.29}$$

where $\theta \in [0, 1]$ and $t_i > 0, i = 1, 2, \dots, m$.

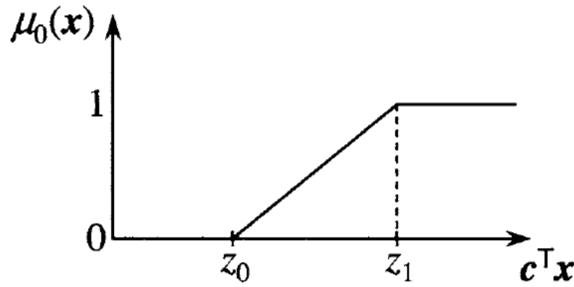
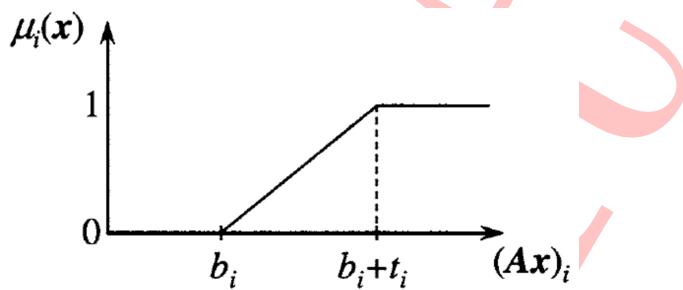
We now discuss a method to solve the problem above. First, solve the following two linear programming problems:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & (\mathbf{A}\mathbf{x})_i \leq \mathbf{b}_i, \quad i = 1, 2, \dots, m \\ & \mathbf{x} \geq 0. \end{aligned} \tag{10.30}$$

and

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & (\mathbf{A}\mathbf{x})_i \leq \mathbf{b}_i + t_i, \quad i = 1, 2, \dots, m \\ & \mathbf{x} \geq 0. \end{aligned} \tag{10.31}$$

Denote the solution to the two programs as $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(0)}$, respectively, and let $z_1 = \mathbf{c}^T \mathbf{x}^{(1)}$ and $z_0 = \mathbf{c}^T \mathbf{x}^{(0)}$. Using these definitions, we construct a function that characterizes the


 图 10.7: Plot of the function $\mu_0(x)$.

 图 10.8: Plot of the function $\mu_i(x)$.

“degree of the penalty” associated with the uncertain constraints in the linear programming problem

$$\mu_0(\mathbf{x}) = \begin{cases} 0, & \mathbf{c}^T \mathbf{x} < z_0, \\ \frac{\mathbf{c}^T \mathbf{x} - z_0}{z_1 - z_0}, & z_0 \leq \mathbf{c}^T \mathbf{x} \leq z_1, \\ 1, & \mathbf{c}^T \mathbf{x} > z_1. \end{cases} \quad (10.32)$$

A plot of this function is given in Figure 10.7. Note that when $\mathbf{c}^T \mathbf{x} \leq z_0$, then $\mu_0(\mathbf{x}) = 0$, which represents minimum degree of penalty. On the other hand, when $\mathbf{c}^T \mathbf{x} \geq z_1$, then $\mu_0(\mathbf{x}) = 1$, and we have a maximum degree of penalty. When $z_0 \leq \mathbf{c}^T \mathbf{x} \leq z_1$, the degree of penalty varies from 0 to 1.

Next, we introduce a function that describes the degree of penalty for violating the i th constraint:

$$\mu_i(\mathbf{x}) = \begin{cases} 0, & (\mathbf{A}\mathbf{x})_i - b_i < 0, \\ \frac{(\mathbf{A}\mathbf{x})_i - b_i}{t_i}, & 0 \leq (\mathbf{A}\mathbf{x})_i - b_i \leq t_i, \\ 1, & (\mathbf{A}\mathbf{x})_i - b_i > t_i. \end{cases} \quad (10.33)$$

A plot of this function is shown in Figure 10.8.

Using the definitions above we can reformulate the original linear programming problem as a multiobjective optimization problem, with the goal of minimizing the functions

that penalize constraint violations:

$$\min_{\mathbf{x}} \begin{bmatrix} \mu_0(\mathbf{x}) \\ \mu_1(\mathbf{x}) \\ \vdots \\ \mu_m(\mathbf{x}) \end{bmatrix} \quad (10.34)$$

$$s.t. \quad \mathbf{x} \geq \mathbf{0}. \quad (10.35)$$

We can employ the minimax method to solve the multiobjective optimization problem as a single-objective problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & \max\{\mu_0(\mathbf{x}), \mu_1(\mathbf{x}), \dots, \mu_m(\mathbf{x})\} \\ s.t. \quad & \mathbf{x} \geq \mathbf{0} \end{aligned} \quad (10.36)$$

As shown in Example 722, the problem above can be stated equivalently as

$$\begin{aligned} \min_{\mathbf{x}} \quad & \theta \\ s.t. \quad & \mu_0(\mathbf{x}) \leq \theta \\ & \mu_i(\mathbf{x}) \leq \theta, \quad i = 1, 2, \dots, m \\ & \theta \in [0, 1], \mathbf{x} \geq \mathbf{0}. \end{aligned} \quad (10.37)$$

Using now the definitions of μ_0 and $\mu_1, i = 1, \dots, m$, we restate the optimization problem above as

$$\begin{aligned} \min_{\mathbf{x}} \quad & \theta \\ s.t. \quad & \mathbf{c}^T \mathbf{x} \leq z_0 + \theta(z_1 - z_0) \\ & (\mathbf{Ax})_i \leq b_i + \theta t_i, \quad i = 1, 2, \dots, m \\ & \theta \in [0, 1], \mathbf{x} \geq \mathbf{0}. \end{aligned} \quad (10.38)$$

Example 723. Consider the following linear programming problem:

$$\begin{aligned} \min_{x_1, x_2} \quad & -\frac{1}{2}x_1 - x_2 \\ s.t. \quad & x_1 + x_2 \leq 5 \\ & x_2 \leq 3 \\ & x_1 \geq 0, x_2 \geq 0, \end{aligned} \quad (10.39)$$

where the tolerances are $t_1 = 2$ and $t_2 = 1$.

a. Solve the two linear programming problems to obtain $x^{(1)}$ and $x^{(0)}$ using the data above.

Then find z_1 and z_0 .

- b. Construct the equivalent optimization problem (involving θ) using the data above.
- c. Express the optimization problem as a linear programming problem in standard form.

Solution. a. We can solve these problems graphically to obtain

$$\mathbf{x}^{(1)} = [2, 3]^T, \quad \mathbf{x}^{(0)} = [3, 4]^T. \quad (10.40)$$

Hence,

$$z_1 = \mathbf{c}^T \mathbf{x}^{(1)} = -4, \quad z_0 = \mathbf{c}^T \mathbf{x}^{(0)} = -\frac{5}{2}. \quad (10.41)$$

b. The optimization problem has the form

$$\begin{aligned} & \min_{\mathbf{x}} \theta \\ & \text{s.t. } \mu_0(\mathbf{x}) \leq \theta \\ & \quad \mu_1(\mathbf{x}) \leq \theta \\ & \quad \mu_2(\mathbf{x}) \leq \theta \\ & \quad \theta \in [0, 1], \mathbf{x} \geq 0, \end{aligned} \quad (10.42)$$

where

$$\mu_0(\mathbf{x}) = \begin{cases} 0, & -\frac{1}{2}x_1 - x_2 < -\frac{5}{2}, \\ \frac{-\frac{1}{2}x_1 - x_2 + \frac{5}{2}}{3/2}, & -\frac{5}{2} \leq -\frac{1}{2}x_1 - x_2 \leq -4, \\ 1, & -\frac{1}{2}x_1 - x_2 > -4, \end{cases} \quad (10.43)$$

$$\mu_1(\mathbf{x}) = \begin{cases} 0, & x_1 + x_2 - 5 < 0, \\ \frac{x_1 + x_2 - 5}{2}, & 0 \leq x_1 + x_2 - 5 \leq 2, \\ 1, & x_1 + x_2 - 5 > 2, \end{cases} \quad (10.44)$$

$$\mu_2(\mathbf{x}) = \begin{cases} 0, & x_2 - 3 < 0, \\ x_2 - 3, & 0 \leq x_2 - 3 \leq 1, \\ 1, & x_2 - 3 > 1. \end{cases} \quad (10.45)$$

c. We have

$$\begin{aligned} & \min_{\mathbf{x}} \theta \\ & \text{s.t. } \mathbf{c}^T \mathbf{x} \leq z_0 + \theta(z_1 - z_0) \\ & \quad (\mathbf{A}\mathbf{x})_i \leq b_i + (1 - \theta)t_i, i = 1, 2 \\ & \quad \theta \in [0, 1], \mathbf{x} \geq \mathbf{0}. \end{aligned} \quad (10.46)$$

Using our data, we obtain

$$\begin{aligned} & \min_{x_1, x_2} \theta \\ s.t. \quad & \frac{1}{2}x_1 + x_2 \geq \frac{5}{2} - \frac{3}{2}\theta \\ & x_1 + x_2 \leq 5 + 2\theta \\ & x_2 \leq 3 + \theta \\ & \theta \in [0, 1], \mathbf{x} \geq \mathbf{0}. \end{aligned} \tag{10.47}$$

Write $x_3 = \theta$. Then, the above problem can be represented as

$$\begin{aligned} & \min_{\mathbf{x}} x_3 \\ s.t. \quad & x_1 + 2x_2 + 3x_3 \geq 11 \\ & x_1 + x_2 - 2x_3 \leq 5 \\ & x_2 - x_3 \leq 3 \\ & x_3 \leq 1, \\ & x_i \geq 0, \quad i = 1, 2, 3. \end{aligned} \tag{10.48}$$

The above linear program expressed in the form of a linear programming problem in standard form is

$$\begin{aligned} & \min_{\mathbf{x}} x_3 \\ s.t. \quad & x_1 + 2x_2 + 3x_3 - x_4 = 11 \\ & x_1 + x_2 - 2x_3 + x_5 = 5 \\ & x_2 - x_3 + x_6 = 3 \\ & x_3 + x_7 = 1, \\ & x_i \geq 0, \quad i = 1, 2, 3. \end{aligned} \tag{10.49}$$

□

Uncertain Objective Function Coefficients We now consider a linear programming problem with uncertain objective function coefficients. We assume that uncertainties of the objective coefficients are modeled by the following triangular function:

$$\mu(x; a, b, c) = \begin{cases} 0, & x < a, \\ (x - a)/(b - a), & a \leq x < b, \\ (c - x)/(c - b), & b \leq x < c. \\ 0, & x > c. \end{cases} \tag{10.50}$$

A plot of this function for $a = 1$, $b = 2$, and $c = 6$ is shown in Figure 10.9. In other words, the uncertain objective coefficients will be represented by the triangular functions

of the form given above. Following [129] (p. 386), we use the notation $\bar{c}_i = (c_i^-, c_i^0, c_i^+)$ to denote the uncertain coefficient c_i represented by the triangular function $\mu(x; c_i^-, c_i^0, c_i^+)$. Then the linear programming problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}, \end{aligned} \tag{10.51}$$

becomes

$$\begin{aligned} \min_{\mathbf{x}} \quad & \begin{bmatrix} (\mathbf{c}^-)^T \mathbf{x} \\ (\mathbf{c}^0)^T \mathbf{x} \\ (\mathbf{c}^+)^T \mathbf{x} \end{bmatrix} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{10.52}$$

where

$$\mathbf{c}^- = [c_1^-, \dots, c_n^-]^T, \mathbf{c}^0 = [c_1^0, \dots, c_n^0]^T, \mathbf{c}^+ = [c_1^+, \dots, c_n^+]^T. \tag{10.53}$$

This is a multiobjective optimization problem. Wang [129] suggests that instead of minimizing the three values $(\mathbf{c}^-)^T \mathbf{x}$, $(\mathbf{c}^0)^T \mathbf{x}$, and $(\mathbf{c}^+)^T \mathbf{x}$ simultaneously, the center, $(\mathbf{c}^0)^T \mathbf{x}$, be minimized; the left leg, $(\mathbf{c}^0 - \mathbf{c}^-)^T \mathbf{x}$, be maximized; and the right leg, $(\mathbf{c}^+ - \mathbf{c}^0)^T \mathbf{x}$, be minimized. This results in pushing the triangular functions to the left in the minimization process. Thus, the multiobjective optimization problem above can be changed to the following multiobjective optimization problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \begin{pmatrix} -(\mathbf{c}^0 - \mathbf{c}^-)^T \mathbf{x} \\ (\mathbf{c}^0)^T \mathbf{x} \\ (\mathbf{c}^+ - \mathbf{c}^0)^T \mathbf{x} \end{pmatrix} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned} \tag{10.54}$$

Uncertain Constraint Coefficients We may be faced with solving a linear programming problem with uncertain constraint coefficients. In this case the coefficients of the constraint matrix \mathbf{A} would be represented by triangular functions of the form given in the preceding section. That is, the coefficient a_{ij} of the constraint matrix \mathbf{A} would be modeled by the function $\bar{a}_{ij} = \mu(x; a_{ij}^-, a_{ij}^0, a_{ij}^+)$. Then, the linear programming problem

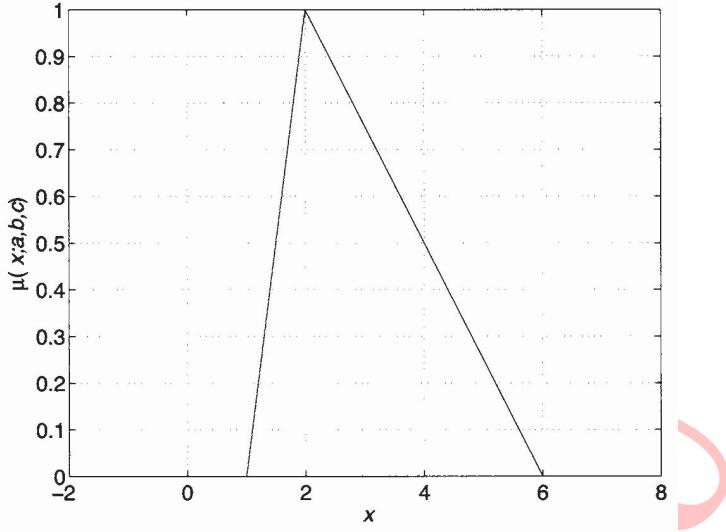


图 10.9: Plot of the triangular function $\mu(x; a, b, c)$ for $a = 1$, $b = 2$, and $c = 6$.

with uncertain constraint coefficients would take the form

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \begin{bmatrix} \mathbf{A}^- \mathbf{x} \\ \mathbf{A}^0 \mathbf{x} \\ \mathbf{A}^+ \mathbf{x} \end{bmatrix} \leq \begin{bmatrix} \mathbf{b} \\ \mathbf{b} \\ \mathbf{b} \end{bmatrix} \\ & \mathbf{x} \geq \mathbf{0}, \end{aligned} \tag{10.55}$$

where $\mathbf{A}^- = [a_{ij}^-]$, $\mathbf{A}^0 = [a_{ij}^0]$, and $\mathbf{A}^+ = [a_{ij}^+]$.

General Uncertainties Finally, we may be faced with solving an uncertain linear programming problem that is a combination of the basic uncertain linear programming problems discussed above. For example, suppose that we are asked to solve the following quite general uncertain linear programming problem¹:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \tilde{\mathbf{c}}^T \mathbf{x} \\ \text{s.t.} \quad & \tilde{\mathbf{A}} \mathbf{x} \leq \tilde{\mathbf{b}} \\ & \mathbf{x} \geq \mathbf{0}, \end{aligned} \tag{10.56}$$

where the tilde symbols refer to the uncertain data; that is, we have

$$\tilde{\mathbf{c}} = ((\mathbf{c}^-)^T, (\mathbf{c}^0)^T, (\mathbf{c}^+)^T)^T, \quad \tilde{\mathbf{A}} = (\mathbf{A}^-, \mathbf{A}^0, \mathbf{A}^+), \quad \tilde{\mathbf{b}} = (\mathbf{b}^-, \mathbf{b}^0, \mathbf{b}^+). \tag{10.57}$$

We can represent the uncertain linear programming problem above as a multiobjective

¹I think this example has some errors!

optimization problem of the form

$$\begin{aligned} \min_{\mathbf{x}} \quad & \begin{pmatrix} -(\mathbf{c}^0 - \mathbf{c}^-)^T \mathbf{x} \\ (\mathbf{c}^0)^T \mathbf{x} \\ (\mathbf{c}^+ - \mathbf{c}^0)^T \mathbf{x} \end{pmatrix} \\ \text{s.t.} \quad & \begin{pmatrix} \mathbf{A}^- \mathbf{x} \\ \mathbf{A}^0 \mathbf{x} \\ \mathbf{A}^+ \mathbf{x} \end{pmatrix} \leq \begin{pmatrix} \mathbf{b}^- \\ \mathbf{b}^0 \\ \mathbf{b}^+ \end{pmatrix} \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned} \tag{10.58}$$

10.6 EXERCISES

Exercise 724. Write a MATLAB program that implements the algorithm for generating a Pareto front, and test it on the problem in Example 720.

Exercise 725. Consider the multiobjective problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & (\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \Omega, \end{aligned} \tag{10.59}$$

where $\mathbf{x} : \mathbb{R}^n \rightarrow \mathbb{R}^\ell$.

a. Suppose that we solve the single-objective problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} \quad & \mathbf{x} \in \Omega, \end{aligned} \tag{10.60}$$

where $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{c} > \mathbf{0}$ (i.e., we use the weighted-sum approach). Show that if \mathbf{x}^* is a global minimizer for the single-objective problem above, then \mathbf{x}^* is a Pareto minimizer for the given multiobjective problem. Then show that it is not necessarily the case that if \mathbf{x}^* is a Pareto minimizer for the multiobjective problem, then there exists a $\mathbf{c} > \mathbf{0}$ such that \mathbf{x}^* is a global minimizer for the single-objective (weighted-sum) problem.

b. Assuming that for all $\mathbf{x} \in \Omega$, $(\mathbf{x}) \geq \mathbf{0}$, suppose that we solve the single-objective problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & (f_1(\mathbf{x}))^p + \cdots + (f_\ell(\mathbf{x}))^p \\ \text{s.t.} \quad & \mathbf{x} \in \Omega, \end{aligned} \tag{10.61}$$

where $p \in \mathbb{R}$, $p > 0$ (i.e., we use the minimum-norm approach). Show that if \mathbf{x}^* is a global minimizer for the single-objective problem above, then \mathbf{x}^* is a Pareto minimizer for the given multiobjective problem. Then show that it is not necessarily the case that if \mathbf{x}^* is a

Pareto minimizer for the multiobjective problem, then there exists a $p > 0$ such that \mathbf{x}^* is a global minimizer for the single-objective (minimum-norm) problem.

c. Suppose that we solve the single-objective problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & \max\{f_1(\mathbf{x}), \dots, f_\ell(\mathbf{x})\} \\ \text{s.t.} \quad & \mathbf{x} \in \Omega, \end{aligned} \tag{10.62}$$

(i.e., we use the minimax approach). Show that it is not necessarily the case that if \mathbf{x}^* is a Pareto minimizer for the given multiobjective problem, then \mathbf{x}^* is a global minimizer for the single-objective (minimax) problem. Then show that it also is not necessarily the case that if \mathbf{x}^* is a global minimizer for the single-objective problem, then \mathbf{x}^* is a Pareto minimizer for the multiobjective problem.

Exercise 726. Let $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^\ell$ be given. Consider the following multiobjective problem with equality constraints:

$$\begin{aligned} \min_{\mathbf{x}} \quad & (\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \Omega, \end{aligned} \tag{10.63}$$

Suppose that $\mathbf{f} \in \mathcal{C}^1$, all the components of \mathbf{f} are convex, and Ω is convex. Suppose that there exists \mathbf{x}^* and $\mathbf{c}^* > \mathbf{0}$ such that for any feasible direction \mathbf{d} at \mathbf{x}^* , we have

$$\mathbf{c}^{*T} D\mathbf{f}(\mathbf{x}^*) \mathbf{d} \geq 0. \tag{10.64}$$

Show that \mathbf{x}^* is a Pareto minimizer.

Exercise 727. Let $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^\ell$ and $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ be given. Consider the following multiobjective problem with equality constraints:

$$\begin{aligned} \min_{\mathbf{x}} \quad & (\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{x}) = \mathbf{0}, \end{aligned} \tag{10.65}$$

Suppose that $\mathbf{f}, \mathbf{h} \in \mathcal{C}^1$, all the components of \mathbf{f} are convex, and the constraint set is convex. Show that if there exists $\mathbf{x}^*, \mathbf{c}^* > \mathbf{0}$, and $\boldsymbol{\lambda}^*$ such that

$$\begin{aligned} \mathbf{c}^{*T} D(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} \mathbf{D}\mathbf{h}(\mathbf{x}^*) &= \mathbf{0}^T \\ \mathbf{h}(\mathbf{x}^*) &= \mathbf{0}, \end{aligned} \tag{10.66}$$

then \mathbf{x}^* is a Pareto minimizer. We can think of the above as a Lagrange condition for the constrained multiobjective function.

Exercise 728. Let $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^\ell$ and $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^p$ be given. Consider the following multiobjective problem with equality constraints:

$$\begin{aligned} & \min_{\mathbf{x}} \quad \mathbf{f}(\mathbf{x}) \\ & \text{s.t.} \quad \mathbf{g}(\mathbf{x}) \leq \mathbf{0}, \end{aligned} \tag{10.67}$$

Suppose that $\mathbf{f}, \mathbf{g} \in \mathcal{C}^1$, all the components of \mathbf{f} are convex, and the constraint set is convex. Show that if there exists $\mathbf{x}^*, \mathbf{c}^* > \mathbf{0}$, and $\boldsymbol{\mu}^*$ such that

$$\begin{aligned} & \boldsymbol{\mu}^* \geq \mathbf{0}, \\ & \mathbf{c}^{*T} D(\mathbf{x}^*) + \boldsymbol{\mu}^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T, \\ & \boldsymbol{\mu}^{*T} \mathbf{g}(\mathbf{x}^*) = \mathbf{0}, \\ & \mathbf{g}(\mathbf{x}^*) \leq \mathbf{0}, \end{aligned} \tag{10.68}$$

then \mathbf{x}^* is a Pareto minimizer. We can think of the above as a KKT condition for the constrained multiobjective function.

Exercise 729. Let $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^\ell$, $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, and $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^p$ be given. Consider the following multiobjective problem with equality constraints:

$$\begin{aligned} & \min_{\mathbf{x}} \quad \mathbf{f}(\mathbf{x}) \\ & \text{s.t.} \quad \mathbf{h}(\mathbf{x}) = \mathbf{0} \\ & \quad \mathbf{g}(\mathbf{x}) \leq \mathbf{0}, \end{aligned} \tag{10.69}$$

Suppose that $\mathbf{f}, \mathbf{g}, \mathbf{h} \in \mathcal{C}^1$, all the components of \mathbf{f} are convex, and the constraint set is convex. Show that if there exists $\mathbf{x}^*, \mathbf{c}^* > \mathbf{0}$, $\boldsymbol{\lambda}^*$, and $\boldsymbol{\mu}^*$ such that

$$\begin{aligned} & \boldsymbol{\mu}^* \geq \mathbf{0}, \\ & \mathbf{c}^{*T} D(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T} D\mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^{*T} D\mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T, \\ & \boldsymbol{\mu}^{*T} \mathbf{g}(\mathbf{x}^*) = \mathbf{0}, \\ & \mathbf{h}(\mathbf{x}^*) = \mathbf{0}, \\ & \mathbf{g}(\mathbf{x}^*) \leq \mathbf{0}, \end{aligned} \tag{10.70}$$

then \mathbf{x}^* is a Pareto minimizer.

Exercise 730. Let $f_1 : \mathbb{R}^n \rightarrow \mathbb{R}$ and $f_2 : \mathbb{R}^n \rightarrow \mathbb{R}$, $f_1, f_2 \in \mathcal{C}^1$. Consider the minimax problem

$$\min_{\mathbf{x}} \quad \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}. \tag{10.71}$$

Show that if \mathbf{x}^* is a local minimizer, then there exist $\mu_1^*, \mu_2^* \in \mathbb{R}$ such that

$$\mu_1^*, \mu_2^* \geq 0, \quad \mu_1^* \nabla f_1(\mathbf{x}^*) + \mu_2^* \nabla f_2(\mathbf{x}^*) = \mathbf{0}, \quad \mu_1^* + \mu_2^* = 1, \tag{10.72}$$

and $\mu_i^* = 0$ if $f_i(\mathbf{x}^*) < \max\{f_1(\mathbf{x}^*), f_2(\mathbf{x}^*)\}$.

(Taken from Chapter 14 of [109])

Exercise 731. Find $\mathbf{x} = (x_1, x_2)^T$ which minimizes

$$f_1(\mathbf{x}) = 2\rho h x_2 \sqrt{1 + x_1^2}, \quad f_2(\mathbf{x}) = \frac{Ph(1 + x_1^2)^{1.5} \sqrt{1 + x_1^4}}{2\sqrt{2}Ex_1^2 x_2},$$

subject to

$$\begin{aligned} g_1(\mathbf{x}) &= \frac{P(1 + x_1) \sqrt{1 + x_1^2}}{2\sqrt{2}x_1 x_2} - \sigma_0 \leq 0, \\ g_2(\mathbf{x}) &= \frac{P(x_1 - 1) \sqrt{1 + x_1^2}}{2\sqrt{2}x_1 x_2} - \sigma_0 \leq 0, \\ x_i &\geq x_i^{(l)}, \quad i = 1, 2, \end{aligned}$$

where $x_1 = x/h$, $x_2 = A/A_{ref}$, h the depth, E is Young's modulus, ρ the weight density, σ_0 the permissible stress, and $x_i^{(l)}$ the lower bound on x_i . Find the optimum solutions of the individual objective functions subject to the stated constraints using a graphical procedure. Data: $P = 10,000lb$, $\rho = 0.283lb/in^3$, $E = 30 \times 10^6$ psi, $h = 100$ in., $A_{ref} = 1in.^2$, $\sigma_0 = 20,000$ psi, $x_1^{(l)} = 0.1$, and $x_2^{(l)} = 1.0$.

Exercise 732. Solve the two-objective optimization problem stated in Exercise 731 using the weighting method with equal weights to the two objective functions. Use a graphical method of solution.

Exercise 733. Solve the two-objective optimization problem stated in Exercise 731 using the global criterion method with $p = 2$. Use a graphical method of solution.

Exercise 734. Formulate the two-objective optimization problem stated in Exercise 731 as a goal programming problem using the goals of 30 lb and 0.015 in. for the objectives f_1 and f_2 , respectively. Solve the problem using a graphical procedure.

Exercise 735. Consider the following two-objective optimization problem: Find $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5, x_6)^T$ to minimize

$$\begin{aligned} f_1(\mathbf{x}) &= -25(x_1 - 2)^2 - (x_2 - 2)^2 - (x_3 - 1)^2 - (x_4 - 4)^2 - (x_5 - 1)^2, \\ f_2(\mathbf{x}) &= x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 + x_6^2 \end{aligned}$$

subject to

$$\begin{aligned}
 -x_1 - x_2 + 2 &\leq 0, \\
 x_1 + x_2 - 6 &\leq 0, \\
 -x_1 + x_2 - 2 &\leq 0, \\
 x_1 - 3x_2 - 2 &\leq 0, \\
 (x_3 - 3)^2 + x_4^2 - 4 &\leq 0, \\
 -(x_5 - 3)^2 - x_6 + 4 &\leq 0, \\
 0 \leq x_i &\leq 10, \quad i = 1, 2, 6, \\
 1 \leq x_i &\leq 5, \quad i = 3, 5, \\
 0 \leq x_4 &\leq 6.
 \end{aligned}$$

Find the minima of the individual objective functions under the stated constraints using the MATLAB function `fmincon`.

Exercise 736. Find the solution of the two-objective optimization problem stated in Exercise 735 using the weighting function method with the weights $w_1 = w_2 = 1$. Use the MATLAB function `fmincon` for the solution.

Exercise 737. Find the solution of the two-objective optimization problem stated in Exercise 735 using the global criterion method with $p = 2$. Use the MATLAB function `fmincon` for the solution.

Exercise 738. Find the solution of the two-objective optimization problem stated in Exercise 735 using the bounded objective function method. Take the lower and upper bounds on f_2 as 80 and 120% of the optimum value f_2^* found in Exercise 735. Use the MATLAB function `fmincon` for the solution.

Exercise 739. Find the solution of the two-objective optimization problem stated in Exercise 735 using the goal attainment method. Use the MATLAB function `fgoalattain` for the solution. Use suitable goals for the objectives.

Exercise 740. Consider the following three-objective optimization problem: Find $\mathbf{x} = (x_1, x_2)^T$ to minimize

$$\begin{aligned}
 f_1(\mathbf{x}) &= 1.5 - x_1(1 - x_2), \\
 f_2(\mathbf{x}) &= 2.25 - x_1(1 - x_2^2), \\
 f_3(\mathbf{x}) &= 2.625 - x_1(1 - x_2^3)
 \end{aligned}$$

subject to

$$\begin{aligned} -x_1^2 - (x_2 - 0.5)^2 + 9 &\leq 0, \\ (x_1 - 1)^2 + (x_2 - 0.5)^2 - 6.25 &\leq 0, \\ -10 \leq x_i &\leq 10, \quad i = 1, 2. \end{aligned}$$

Find the minima of the individual objectives under the stated constraints using the MATLAB function fmincon.

Exercise 741. Find the solution of the 3-objective problem stated in Exercise 740 using the weighting function method with the weights $w_1 = w_2 = w_3 = 1$. Use the MATLAB function fmincon for the solution.

Exercise 742. Find the solution of the multiobjective problem stated in Exercise 740 using the goal attainment method. Use the MATLAB function fgoalattain for the solution. Use suitable goals for the objectives.

(Taken from Chapter 7 of [4])

Exercise 743. Find the convex Pareto front for the multiobjective optimization problem:

$$\min_x f_1 = x^2, \quad f_2 = (x - 2)^2,$$

where $x \in [-10^5, 10^5]$.

Exercise 744. Find the concave Pareto front for the multiobjective optimization problem.

$$\min_{\mathbf{x}} f_1 = x_1, \quad f_2 = g \left(1 - (f_1/g)^2 \right),$$

where $g = 1 + \frac{9}{n-1} \sum_{i=2}^n x_i$ and $x_i \in [0, 1]$. Take $n = 30$.

Exercise 745. Find the convex Pareto front for the multiobjective optimization problem.

$$\min_{\mathbf{x}} f_1 = x_1, \quad f_2 = g \left(1 - \sqrt{f_1/g} \right),$$

where $g = 1 + \frac{9}{n-1} \sum_{i=2}^n x_i$ and $x_i \in [0, 1]$. Take $n = 30$.

Exercise 746. Find the convex/concave Pareto front for the multiobjective optimization problem.

$$\min_{\mathbf{x}} f_1 = x_1, \quad f_2 = g \left(1 - (f_1/g)^4 - \sqrt[4]{f_1/g} \right),$$

where $g = 1 + \frac{9}{n-1} \sum_{i=2}^n x_i$ and $x_i \in [0, 1]$. Take $n = 30$.

Exercise 747. Find the concave Pareto front for the multiobjective optimization problem.

$$\min_{\mathbf{x}} f_1 = 1 - \exp \left(- \sum_{i=1}^n (x_i - 1/\sqrt{n})^2 \right), \quad f_2 = 1 - \exp \left(- \sum_{i=1}^n (x_i + 1/\sqrt{n})^2 \right),$$

where $x_i \in [-4, 4]$. Take $n = 2$.

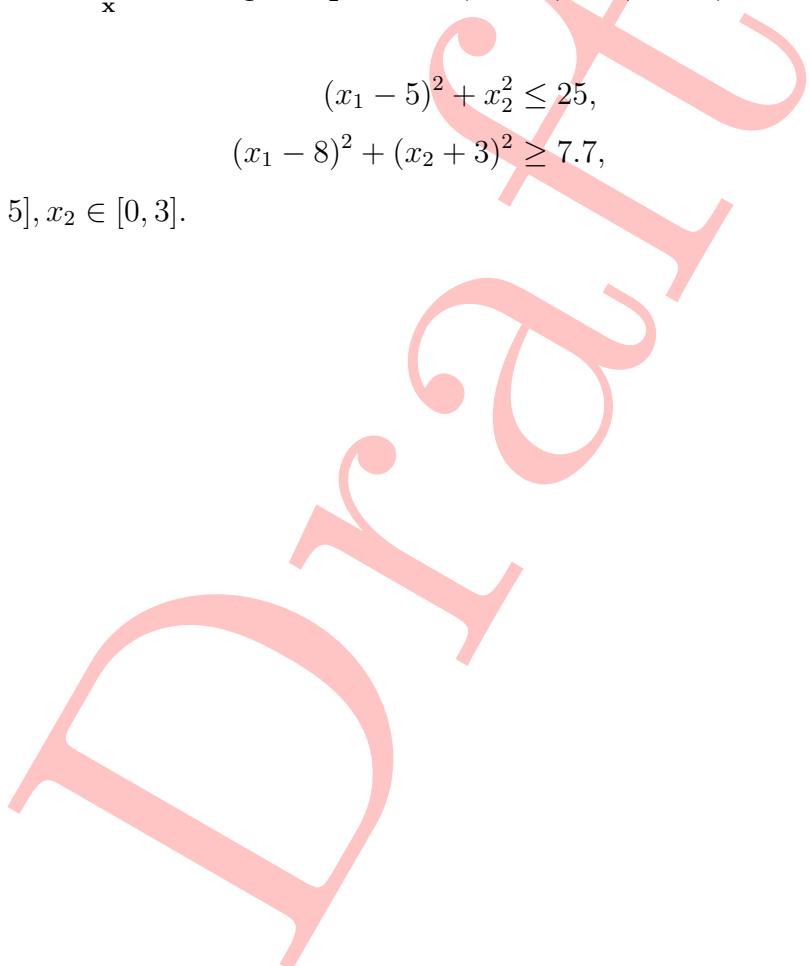
Exercise 748. Find the convex Pareto front for the constrained multiobjective optimization problem:

$$\min_{\mathbf{x}} f_1 = 4x_1^2 + 4x_2^2, \quad f_2 = (x_1 - 5)^2 + (x_2 - 5)^2,$$

subject to

$$(x_1 - 5)^2 + x_2^2 \leq 25, \\ (x_1 - 8)^2 + (x_2 + 3)^2 \geq 7.7,$$

where $x_1 \in [0, 5], x_2 \in [0, 3]$.



Draft

第十一章 Bilevel Optimization

11.1 Possible Transformations into a One-Level Problem

(Taken from Chapter 1.2 of [38])

Bilevel optimization problems are optimization problems where the feasible set is determined (in part) using the graph of the solution set mapping of a second parametric optimization problem. This problem is given as

$$\min_{\mathbf{y}} \{f(\mathbf{x}, \mathbf{y}) : g(\mathbf{x}, \mathbf{y}) \leq 0, \mathbf{y} \in \mathbf{T}\}, \quad (11.1)$$

where $f : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$, $g : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^p$, $\mathbf{T} \subseteq \mathbb{R}^m$ is a (closed) set.

Let $\mathbf{Y} : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$ denote the *feasible set mapping*:

$$\begin{aligned} \mathbf{Y}(\mathbf{x}) &:= \{\mathbf{y} : g(\mathbf{x}, \mathbf{y}) \leq 0\}, \\ \phi(\mathbf{x}) &:= \min_{\mathbf{y}} \{f(\mathbf{x}, \mathbf{y}) : g(\mathbf{x}, \mathbf{y}) \leq 0, \mathbf{y} \in \mathbf{T}\} \end{aligned}$$

the *optimal value function*, and $\Psi : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$ the *solution set mapping* of the problem (11.1) for a fixed value of \mathbf{x} :

$$\Psi(\mathbf{x}) := \{\mathbf{y} \in \mathbf{Y}(\mathbf{x}) \cap \mathbf{T} : f(\mathbf{x}, \mathbf{y}) \leq \varphi(\mathbf{x})\}.$$

Let

$$\mathbf{gph}\Psi := \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^m : \mathbf{y} \in \Psi(\mathbf{x})\}$$

be the graph of the mapping Ψ . Then, the bilevel optimization problem is given as

$$\min_{\mathbf{x}} \{F(\mathbf{x}, \mathbf{y}) : G(\mathbf{x}) \leq 0, (\mathbf{x}, \mathbf{y}) \in \mathbf{gph}\Psi, \mathbf{x} \in \mathbf{X}\}, \quad (11.2)$$

where $F : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$, $G : \mathbb{R}^n \rightarrow \mathbb{R}^q$, $\mathbf{X} \subset \mathbb{R}^n$ is a closed set.

Problems (11.1) and (11.2) can be interpreted as an hierarchical game of two decision makers (or players) which make their decisions according to an hierarchical order. The first player (which is called the *leader*) makes his selection first and communicates it to the second player (the so-called *follower*). Then, knowing the choice of the leader, the follower selects his *response* as an optimal solution of problem (11.1) and gives this back to the leader. Thus, the leader's task is to determine a best decision, i.e. a point $\hat{\mathbf{x}}$ which is feasible for the problem (11.2): $G(\hat{\mathbf{x}}) \leq 0$, $\hat{\mathbf{x}} \in \mathbf{X}$, minimizing together with the response $\hat{\mathbf{y}} \in \Psi(\hat{\mathbf{x}})$ the function $F(\mathbf{x}, \mathbf{y})$. Therefore, problem (11.1) is called the follower's problem and (11.2) the leader's problem. Problem (11.2) is the *bilevel optimization problem*.

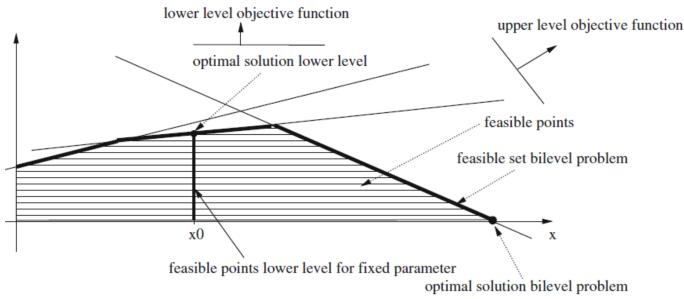


图 11.1: Illustration of the linear bilevel optimization problem

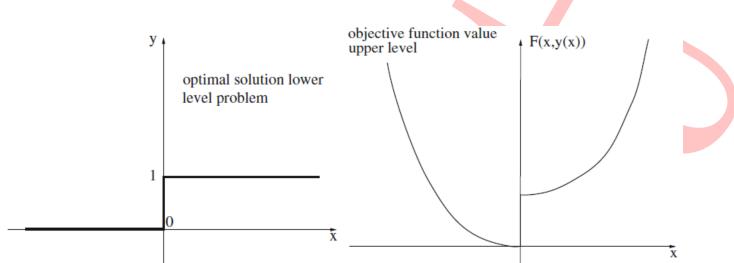


图 11.2: Mapping to be “minimized” in Example 750

Example 749. In case of a linear bilevel optimization problem with only one upper and one lower level variables, where all functions F, f, g_i are (affine) linear functions, the bilevel optimization problem is illustrated in Fig. 11.1. Here, $G(x) \equiv 0$ and the set $\{(x, y) : g(x, y) \leq 0\}$ of feasible points for all values of x is the hatched area. If x is fixed to x_0 the feasible set of the lower level problem (11.1) is the set of points (x_0, y) above x_0 . Now, if the lower level objective function $f(x, y) = -y$ is minimized on this set, the optimal solution of the lower level problem on the thick line is obtained. Then, if x is moved along the x -axis, the thick line as the set of feasible solutions of the upper level problem arises. In other words, the thick line equals the $\text{gph}\Psi$ of the solution set mapping of the lower level problem. This is the feasible set of the upper level (or bilevel) optimization problem. Then, minimizing the upper level objective function on this set, the (in this case unique) optimal solution of the bilevel optimization problem is obtained as indicated in Fig. 11.1.

It can be seen in Fig. 11.1 that the bilevel optimization problem is a nonconvex (since $\text{gph}\Psi$ is nonconvex) optimization problem. Hence, local optimal solutions and also stationary points can appear.

Example 750. Consider the problem

$$\min_{\mathbf{x}} \{x^2 + y : y \in \Psi(x)\},$$

where

$$\Psi(x) := \operatorname{argmin}_y \{-xy : 0 \leq y \leq 1\}.$$

Then, the graph of the mapping Ψ is given in the figure on the left hand side of Fig. 11.2 and the graph of the mapping $x \mapsto F(x, \Psi(x))$ of the upper level objective function is plotted in the figure on the right-hand side. Note, that this is not a function and that its minimum is unclear since its existence depends on the response $y \in \Psi(x)$ of the follower on the leader's selection at $x = 0$. If the solution $y = 0$ is taken for $x = 0$, an optimal solution of the bilevel optimization problem exists. This is the optimistic bilevel optimization problem introduced below. In all other cases, the minimum does not exist, the infimum function value of the upper level objective function is again zero but it is not attained. If $y = 1$ is taken, the so-called pessimistic bilevel optimization problem arises.

Hence, strictly speaking, the problem (11.2) is not well-posed in the case that the set $\Psi(\mathbf{x})$ is not a singleton for some \mathbf{x} , the mapping $\mathbf{x} \mapsto F(\mathbf{x}, \mathbf{y}(\mathbf{x}))$ is not a function. This is implied by an ambiguity in the computation of the upper level objective function value, which is rather an element in the set $\{F(\mathbf{x}, \mathbf{y}) : \mathbf{y} \in \Psi(\mathbf{x})\}$. We have used quotation marks in (11.2) to indicate this ambiguity. To overcome such an unpleasant situation, the leader has a number of possibilities:

1. The leader can assume that the follower is willing (and able) to cooperate. In this case, the leader can take that solution within the set $\Psi(\mathbf{x})$ which is a best one with respect to the upper level objective function. This leads then to the function

$$\varphi_o(\mathbf{x}) := \min\{F(\mathbf{x}, \mathbf{y}) : \mathbf{y} \in \Psi(\mathbf{x})\} \quad (11.3)$$

to be minimized on the set $\{\mathbf{x} : G(\mathbf{x}) \leq 0, \mathbf{x} \in \mathbf{X}\}$. This is the optimistic approach leading to the *optimistic bilevel optimization problem*. The function $\varphi_o(\mathbf{x})$ is called *optimistic solution function*. Roughly speaking, this problem is closely related to the problem

$$\min_{\mathbf{x}, \mathbf{y}} \{F(\mathbf{x}, \mathbf{y}) : G(\mathbf{x}) \leq 0, (\mathbf{x}, \mathbf{y}) \in \operatorname{gph} \Psi, \mathbf{x} \in \mathbf{X}\}. \quad (11.4)$$

If the point $\bar{\mathbf{x}}$ is a local minimum of the function $\varphi_o(\cdot)$ on the set

$$\{\mathbf{x} : G(\mathbf{x}) \leq 0, \mathbf{x} \in \mathbf{X}\}$$

and $\bar{\mathbf{y}} \in \Psi(\bar{\mathbf{x}})$, then the point $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is also a local minimum of problem (11.4). The opposite implication is in general not correct, as the following example shows:

Example 751. Consider the problem of minimizing the function $F(x, y) = x$ subject to $x \in [-1, 1]$ and $y \in \Psi(x) : \operatorname{argmin}_y\{xy : y \in [0, 1]\}$. Then,

$$y(x) \in \begin{cases} [0, 1] & \text{for } x = 0, \\ \{1\} & \text{for } x < 0, \\ \{0\} & \text{for } x > 0. \end{cases}$$

Hence, the point $(\bar{x}, \bar{y}) = (0, 0)$ is a local minimum of the problem

$$\min_{x,y}\{x : x \in [-1, 1], y \in \Psi(x)\}$$

since, for each feasible point (x, y) with $\|(x, y) - (\bar{x}, \bar{y})\| \leq 0.5$ we have $x \geq 0$. But, the point \bar{x} does not minimize the function $\varphi_o(x) = x$ on $[-1, 1]$ locally. For more information about the relation between both problems, the interested reader is referred to [35].

2. The leader has no possibility to influence the follower's selection neither he/she has an intuition about the follower's choice. In this case, the leader has to accept the follower's opportunity to take a worst solution with respect to the leader's objective function and he/she has to bound the damage resulting from such an unpleasant selection. This leads to the function

$$\varphi_p(\mathbf{x}) := \max\{F(\mathbf{x}, \mathbf{y}) : \mathbf{y} \in \Psi(\mathbf{x})\} \quad (11.5)$$

to be minimized on the set $\{\mathbf{x} : G(\mathbf{x}) \leq 0, \mathbf{x} \in \mathbf{X}\}$

$$\min\{\varphi_p(\mathbf{x}) : G(\mathbf{x}) \leq 0, \mathbf{x} \in \mathbf{X}\}. \quad (11.6)$$

This is the pessimistic approach resulting in the pessimistic bilevel optimization problem. The function $\varphi_p(\mathbf{x})$ is the pessimistic solution function. This problem is often much more complicated than the optimistic bilevel optimization problem, see [35].

In the literature there is also another pessimistic bilevel optimization problem. To describe this problem consider the bilevel optimization problem with connecting upper level constraints and an upper level objective function depending only on the upper level variable \mathbf{x} :

$$\min_{\mathbf{x}} \{F(\mathbf{x}) : G(\mathbf{x}, \mathbf{y}) \leq 0, \mathbf{y} \in \Psi(\mathbf{x})\}. \quad (11.7)$$

In this case, a point \mathbf{x} is feasible if there exists $\mathbf{y} \in \Psi(\mathbf{x})$ such that $G(\mathbf{x}, \mathbf{y}) \leq 0$, which can be written as

$$\min_{\mathbf{x}} \{F(\mathbf{x}) : G(\mathbf{x}, \mathbf{y}) \leq 0 \text{ for some } \mathbf{y} \in \Psi(\mathbf{x})\}.$$

Now, if the quantifier \exists is replaced by \forall we derive a second pessimistic bilevel optimization problem

$$\min_{\mathbf{x}} \{F(\mathbf{x}) : G(\mathbf{x}, \mathbf{y}) \leq 0 \text{ for all } \mathbf{y} \in \Psi(\mathbf{x})\}. \quad (11.8)$$

This problem has been investigated in [131]. The relations between (11.8) and (11.6) should be investigated in future.

3. The leader is able to predict a selection of the follower: $y(\mathbf{x}) \in \Psi(\mathbf{x})$ for all \mathbf{x} . If this function is inserted into the upper level objective function, this leads to the problem

$$\min_{\mathbf{x}} \{F(\mathbf{x}, y(\mathbf{x})) : G(\mathbf{x}) \leq 0, \mathbf{x} \in \mathbf{X}\}.$$

Such a function $y(\cdot)$ is called a selection function of the point-to-set mapping $\Psi(\cdot)$. Hence, we call this approach the *selection function approach*. One special case of this approach arises if the optimal solution of the lower level problem is unique for all values of \mathbf{x} . It is obvious that the optimistic and the pessimistic problems are special cases of the selection function approach. Even under restrictive assumptions (as in the case of linear bilevel optimization or if the follower's problem has a unique optimal solution for all \mathbf{x}), the function $y(\cdot)$ is in general not differentiable. Hence, the bilevel optimization problem is a nonsmooth optimization problem.

Definition 752. A point $\bar{\mathbf{z}} \in \mathbf{Z}$ is a local optimal solution of the optimization problem

$$\min_{\mathbf{z}} \{w(\mathbf{z}) : \mathbf{z} \in \mathbf{Z}\}$$

provided that there is a positive number $\varepsilon > 0$ such that

$$w(\mathbf{z}) \geq w(\bar{\mathbf{z}}), \forall \mathbf{z} \in \mathbf{Z} \text{ satisfying } \|\mathbf{z} - \bar{\mathbf{z}}\| \leq \varepsilon.$$

$\bar{\mathbf{z}}$ is a global optimal solution of this problem if ε can be taken arbitrarily large.

This well-known notion of a (local) optimal solution can be applied to the bilevel optimization problems and, using e.g. Weierstrab Theorem we obtain that problem (11.4) has a global optimal solution if the function F is continuous and the set $\mathbf{Z} := \{(\mathbf{x}, \mathbf{y}) : G(\mathbf{x}) \leq 0, (\mathbf{x}, \mathbf{y}) \in \text{gph}\Psi, \mathbf{x} \in \mathbf{X}\}$ is not empty and compact. If this set is not bounded but only a nonempty and closed set and the function F is continuous and coercive (i.e. $F(\mathbf{x}, \mathbf{y})$ tends to infinity for $\|(\mathbf{x}, \mathbf{y})\|$ tending to infinity) problem (11.4) has a global optimal solution, too. For closedness of the set \mathbf{Z} we need closedness of the graph of the solution set mapping of the lower level problem.

With respect to problem

$$\min\{\varphi_0(\mathbf{x}) : G(\mathbf{x}) \leq 0, \mathbf{x} \in \mathbf{X}\} \quad (11.9)$$

existence of an optimal solution is guaranteed if the function $\varphi_0(\cdot)$ is lower semicontinuous (which means that $\liminf_{\mathbf{x} \rightarrow \mathbf{x}^0} \varphi_0(\mathbf{x}) \geq \varphi_0(\mathbf{x}^0)$ for all \mathbf{x}^0) and the set \mathbf{Z} is not empty and compact by an obvious generalization of the Weierstrab Theorem. Again boundedness of this set can be replaced by coercivity. Lower semicontinuity of the function is again an implication of upper semicontinuity of the mapping $\mathbf{x} \mapsto \Psi(\mathbf{x})$, see for instance [7] in combination with Theorem 3.3 of [38]. It is easy to see that a function $w(\cdot)$ is lower semicontinuous if and only if its epigraph $\text{epi } w := \{(\mathbf{z}, \alpha) : w(\mathbf{z}) \leq \alpha\}$ is a closed set.

Example (750) showed already that an optimal solution of the problem (11.6) does often not exist. Its existence is guaranteed e.g. if the function $\varphi_p(\cdot)$ is lower semicontinuous and the set \mathbf{Z} is not empty and compact [93]. But, for lower semicontinuity of the function $\varphi_p(\cdot)$ lower semicontinuity of the solution set mapping $\mathbf{x} \mapsto \Psi(\mathbf{x})$ is needed which can often only be shown if the optimal solution of the lower level problem is unique (see [7]).

If an optimal solution of problem (11.6) does not exist we can aim to find a weak (global) optimum by replacing the epigraph of the objective function by its closure: Let $\bar{\varphi}_p$ be defined such that

$$\text{epi } \bar{\varphi}_p = \text{cl epi } \varphi_p.$$

Then, a local or global optimal solution of the problem

$$\min\{\bar{\varphi}_p(\mathbf{x}) : G(\mathbf{x}) \leq 0, \mathbf{x} \in \mathbf{X}\} \quad (11.10)$$

is called a (local or global) weak solution of the pessimistic bilevel optimization problem (11.6). Note that $\bar{\varphi}_p(\mathbf{x}^0) = \liminf_{\mathbf{x} \rightarrow \mathbf{x}^0} \varphi_p(\mathbf{x})$. The function $\bar{\varphi}_p(\cdot)$ is the largest lower semicontinuous function which is not larger than $\varphi_p(\cdot)$, see [44]. Hence, a weak global solution of problem (11.6) exists provided that $\mathbf{Z} \neq \emptyset$ is compact.

11.2 An Easy Bilevel Optimization Problem: Continuous Knapsack Problem in the Lower Level

(Taken from Chapter 1.3 of [38])

To illustrate the optimistic/pessimistic approaches to the bilevel optimization problem consider

$$\min_b \{\mathbf{d}^\top \mathbf{y} + fb : b_u \leq b \leq b_o, \dots, \mathbf{y} \in \Psi(\mathbf{x})\}, \quad (11.11)$$

where

$$\Psi(b) := \operatorname{argmin}_{\mathbf{y}} \{\mathbf{c}^\top \mathbf{y} : \mathbf{a}^\top \mathbf{y} \geq b, 0 \leq y_i \leq 1 \forall i = 1, \dots, n\}$$

and $\mathbf{a}, \mathbf{c}, \mathbf{d} \in \mathbb{R}_+^n$. Note that the upper level variable is b in this problem. Assume that the indices are ordered such that

$$\frac{c_i}{a_i} \leq \frac{c_{i+1}}{a_{i+1}}, i = 1, 2, \dots, n-1.$$

Then, for fixed $b \in \left\{ b : \sum_{i=1}^{k-1} a_i \leq b \leq \sum_{i=1}^k a_i \right\}$, the point

$$y_i = \begin{cases} 1, & i = 1, \dots, k-1 \\ \frac{b - \sum_{j=1}^{k-1} a_j}{a_k}, & i = k \\ 0, & i = k+1, \dots, n \end{cases} \quad (11.12)$$

is an optimal solution of the lower level problem. Its optimal function value in the lower level is

$$\varphi(b) = \sum_{i=1}^{k-1} c_i + \frac{c_k}{a_k} \left(b - \sum_{j=1}^{k-1} a_j \right),$$

which is an affine linear function of b . The function $b \mapsto \varphi(b)$ is convex. The optimal solution of the lower level problem is unique provided that $\frac{c_k}{a_k}$ is unique in the set $\left\{ \frac{c_i}{a_i} : i = 1, \dots, n \right\}$. Otherwise, the indices $i \in \left\{ j : \frac{c_i}{a_i} = \frac{c_k}{a_k} \right\}$ need to be ordered such that

$$d_t \leq d_{t+1} : t, t+1 \in \left\{ j : \frac{c_i}{a_i} = \frac{c_k}{a_k} \right\}$$

for the optimistic and

$$d_t \geq d_{t+1} : t, t+1 \in \left\{ j : \frac{c_i}{a_i} = \frac{c_k}{a_k} \right\}$$

for the pessimistic approaches. As illustration consider the following example:

Example 753. The lower level problem is

$$\begin{aligned} \min_{\mathbf{x}} & 10x_1 + 30x_2 + 8x_3 + 60x_4 + 4x_5 + 16x_6 + 32x_7 + 30x_8 \\ & + 120x_9 + 6x_{10} \\ & 5x_1 + 3x_2 + 2x_3 + 5x_4 + x_5 + 8x_6 + 4x_7 + 3x_8 \\ \text{s.t.} & + 6x_9 + 3x_{10} \geq b \\ & \forall i : 0 \leq y_i \leq 1, \end{aligned}$$

and the upper level objective function is

$$\begin{aligned} F(\mathbf{x}, f) = & 20x_1 + 15x_2 - 24x_3 + 20x_4 - 40x_5 \\ & + 80x_6 - 32x_7 - 60x_8 - 12x_9 - 60x_{10} + fb. \end{aligned}$$

This function is to be minimized subject to $\mathbf{x} \in \Psi(b)$ and b is in some closed interval $[b_u, b_o]$. Note that the upper level variable is b and the lower level one is \mathbf{x} in this example.

Using the above rules we obtain the the following sequence of the indices in the optimistic approach:

$i =$	10	1	6	5	3	7	8	2	4	9
$\frac{c_i}{a_i} =$	2	2	2	4	4	8	10	10	12	20
$\frac{d_i}{a_i} =$	-20	4	10	-40	-12	-8	-20	5	4	-2

Using the pessimistic approach we get

$i =$	6	1	10	3	5	7	2	8	4	9
$\frac{c_i}{a_i} =$	2	2	2	4	4	8	10	10	12	20
$\frac{d_i}{a_i} =$	10	4	-20	-40	-12	-8	-20	5	4	-2

Now, the functions $\varphi_o(b)$ and $\varphi_p(b)$ are plotted in Fig. 11.3. The upper function is the pessimistic value function, computed according to (11.5), the lower one the optimistic value function, cf. (11.3).

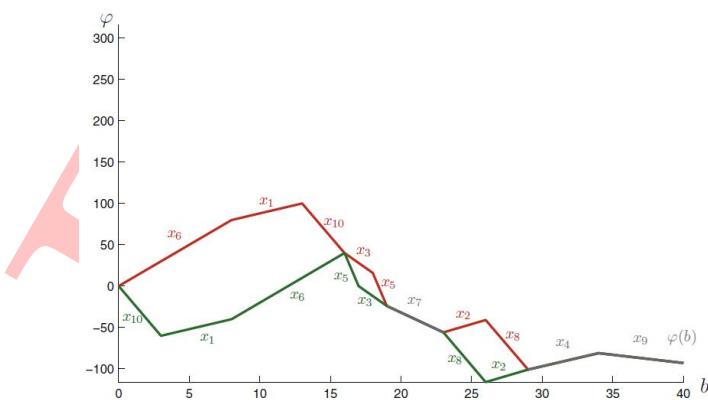


图 11.3: The optimistic and pessimistic objective value functions in Example 753, see [132]

Both functions are continuous, but not convex. Local optima of both functions can either be found at $b \in (b_u, b_o)$ or at points

$$b \in \left\{ \sum_{i \in I} a_i : I \subseteq \{1, 2, \dots, n\} \right\}.$$

Note that local optima can be found at vertices of the set

$$\left\{ (\mathbf{x}, b) : b_u \leq b \leq b_o, 0 \leq \mathbf{x}_i \leq 1, i = 1, \dots, n, \sum_{i=1}^n a_i x_i = b \right\}.$$

11.3 Reduction of Bilevel Programming to a Single Level Problem

(Taken from Chapter 3.1 of [38])

11.3.1 Different Approaches

The usually used approach to solve the bilevel optimization problem (11.1), (11.4) or to formulate (necessary or sufficient) optimality conditions, is to transform it into a one-level optimization problem. There are at least the following three possibilities to realize this:

Primal KKT transformation: The lower level problem can be replaced with its necessary (and sufficient) optimality conditions. This can only be done if the lower level problem is (for a fixed value of the parameter \mathbf{x}) convex. Otherwise, the feasible set of the bilevel optimization problem is enlarged by local optimal solutions and stationary points of the lower level problem. In this case, the global optimal solution of the bilevel problem is in general not a stationary solution for the resulting problem [101]. Let

$$Y(\mathbf{x}) := \{\mathbf{y} : g(\mathbf{x}, \mathbf{y}) \leq 0\}$$

denote the feasible set of the lower level problem and assume that $Y(\mathbf{x})$ is a convex set, $\mathbf{y} \mapsto f(\mathbf{x}, \mathbf{y})$ is a convex function, and $\mathbf{T} \subseteq \mathbb{R}^m$ is a convex set. Then, $\mathbf{y} \in \Psi(\mathbf{x})$ if and only if

$$0 \in \partial_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) + N_{Y(\mathbf{x}) \cap \mathbf{T}}(\mathbf{y}).$$

Here, $\partial_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$ is the subdifferential of the convex function $\mathbf{y} \mapsto f(\mathbf{x}, \mathbf{y})$ and $N_{Y(\mathbf{x}) \cap \mathbf{T}}(\mathbf{y})$ is the *normal cone* of convex analysis to the set $Y(\mathbf{x}) \cap \mathbf{T}$ at the point $\mathbf{y} \in Y(\mathbf{x}) \cap \mathbf{T}$:

$$N_{Y(\mathbf{x}) \cap \mathbf{T}}(\mathbf{y}) := \{\mathbf{d} \in \mathbb{R}^m : d^\top (\mathbf{z} - \mathbf{y}) \leq 0 \ \forall \mathbf{z} \in Y(\mathbf{x}) \cap \mathbf{T}\}.$$

Note that $N_{Y(\mathbf{x}) \cap \mathbf{T}}(\mathbf{y}) = \emptyset$ for $\mathbf{y} \notin Y(\mathbf{x}) \cap \mathbf{T}$. The problem (11.4) is then equivalent to

$$\min\{F(\mathbf{x}, \mathbf{y}) : G(\mathbf{x}) \leq 0, 0 \in \partial_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) + N_{Y(\mathbf{x}) \cap \mathbf{T}}(\mathbf{y}), \mathbf{x} \in \mathbf{X}\}, \quad (11.13)$$

see [37]. This is the *primal Karush-Kuhn-Tucker reformulation (primal KKT reformulation)*.

Problem (11.13) is fully equivalent to the bilevel optimization problem (11.1), (11.4) both if global and local optimal solutions of the problems are considered.

Classical KKT transformation: If $T = \mathbb{R}^m$, and $Y(\mathbf{x}) = \{\mathbf{y} : g(\mathbf{x}, \mathbf{y}) \leq 0\}$, the function $\mathbf{y} \mapsto g(\mathbf{x}, \mathbf{y})$ is convex for each fixed \mathbf{x} , and a regularity condition, as e.g.

Slater's condition: There exists $\hat{\mathbf{y}}$ with $g(\mathbf{x}, \hat{\mathbf{y}}) < 0$, is fulfilled, then $\mathbf{y} \in \Psi(\mathbf{x})$ if and only if the Karush-Kuhn-Tucker conditions (KKT conditions) are satisfied:

$$0 \in \partial_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) + \lambda^\top \partial_{\mathbf{y}} g(\mathbf{x}, \mathbf{y}), \lambda \geq 0, \lambda^\top g(\mathbf{x}, \mathbf{y}) = 0.$$

This leads to a second reformulation of the bilevel optimization problem:

$$\begin{aligned} & \min F(\mathbf{x}, \mathbf{y}) \\ & G(\mathbf{x}) \leq 0 \\ & 0 \in \partial_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) + \lambda^\top \partial_{\mathbf{y}} g(\mathbf{x}, \mathbf{y}) \\ & \lambda \geq 0, g(\mathbf{x}, \mathbf{y}) \leq 0, \lambda^\top g(\mathbf{x}, \mathbf{y}) = 0 \\ & \mathbf{x} \in \mathbf{X}. \end{aligned} \tag{11.14}$$

This is the reformulation most often used. Problem (11.14) is a (nonsmooth) mathematical program with complementarity constraints (or better with a generalized equation constraint). It can be called the classical KKT transformation. Using the relation

$$N_{Y(\mathbf{x})}(\mathbf{y}) = \{\mathbf{d} \in \mathbb{R}^m : \exists \lambda \geq 0, \lambda^\top g(\mathbf{x}, \mathbf{y}) = 0, \mathbf{d} = \lambda^\top \partial_{\mathbf{y}} g(\mathbf{x}, \mathbf{y})\}$$

which is valid if Slater's condition is satisfied at \mathbf{x} (see e.g.[40]), problem (11.13) reduces to (11.14).

The introduction of additional variables λ causes that problems (11.14) and (11.1), (11.4) are no longer fully equivalent. Clearly, each global optimal solution $(\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\lambda})$ of problem (11.14) is related to a global optimal solution $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ of (11.1), (11.4). Also, a local optimal solution $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ of (11.1), (11.4) is related to a local optimal solution $(\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\lambda})$ for each

$$\bar{\lambda} \in \Lambda(\bar{\mathbf{x}}, \bar{\mathbf{y}}) := \{\lambda \geq 0 : \lambda^\top g(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = 0, 0 \in \partial_{\mathbf{y}} f(\bar{\mathbf{x}}, \bar{\mathbf{y}}) + \lambda^\top \partial_{\mathbf{y}} g(\bar{\mathbf{x}}, \bar{\mathbf{y}})\}$$

provided that Slater's condition is satisfied. With respect to the opposite direction we have

Theorem 754. ([36]) Let the lower level problem (11.1) be a convex optimization problem and assume that Slater's condition is satisfied for all $\mathbf{x} \in \mathbf{X}$ with $\Psi(\mathbf{x}) \neq \emptyset$. A feasible point $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ of problem (11.4) is a local optimal solution of this problem iff $(\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\lambda})$ is a local optimal solution of problem (11.14) for each $\bar{\lambda} \in \Lambda(\bar{\mathbf{x}}, \bar{\mathbf{y}})$.

Proof. If $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is a local optimal solution of problem (11.4) then, since the Karush-Kuhn-Tucker conditions are sufficient and necessary optimality conditions and the objective function of (11.14) does not depend on the Lagrange multiplier the point $(\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\lambda})$ is a local optimum of (11.4) for all $\lambda \in \Lambda(\bar{\mathbf{x}}, \bar{\mathbf{y}})$.

Let $(\bar{\mathbf{x}}, \bar{\mathbf{y}}, \bar{\lambda})$ be a local optimal solution of (11.14) for all $\lambda \in \Lambda(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ and assume that $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is not a local optimal solution of (11.4). Then, there exists a sequence $\{(\mathbf{x}^k, \mathbf{y}^k)\}_{k=1}^\infty$ of feasible points for (11.4) converging to $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ such that $F(\mathbf{x}^k, \mathbf{y}^k) < F(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ for all k . Since the KKT conditions are necessary optimality conditions there exists a sequence $\{\lambda^k\}_{k=1}^\infty$ with $\lambda^k \in \Lambda(\mathbf{x}^k, \mathbf{y}^k)$. Obviously, $(\mathbf{x}^k, \mathbf{y}^k, \lambda^k)$ is feasible for (11.14) for all k . Since the mapping $(\mathbf{x}, \mathbf{y}) \mapsto \Lambda(\mathbf{x}, \mathbf{y})$ is upper semicontinuous [112], the sequence $\{\lambda^k\}$ has an accumulation point $\hat{\lambda} \in \Lambda(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ and $(\bar{\mathbf{x}}, \bar{\mathbf{y}}, \hat{\lambda})$ is feasible for (11.14). This violates the assumption. Hence, the theorem is proved. X

□

The following example shows that it is essential to consider all Lagrange multipliers:

Example 755. ([36]) Consider the linear lower level problem

$$\min_y \{-y : x + y \leq 1, -x + y \leq 1\} \quad (11.15)$$

having the unique optimal solution $y(x)$ and the set $\Lambda(x, y)$ of Lagrange multipliers

$$y(x) = \begin{cases} x + 1 & \text{if } x \leq 0 \\ -x + 1 & \text{if } x \geq 0 \end{cases}$$

$$\Lambda(x, y) = \begin{cases} \{(1, 0)\} & \text{if } x > 0 \\ \{(0, 1)\} & \text{if } x < 0 \\ \text{conv}\{(1, 0), (0, 1)\} & \text{if } x = 0 \end{cases}$$

where $\text{conv}(A)$ denotes the convex hull of the set A . This example is illustrated in Fig. 11.4.

The bilevel optimization problem

$$\min \{(x - 1)^2 + (y - 1)^2 : (x, y) \in \mathbf{gph} \Psi\} \quad (11.16)$$

has the unique optimal solution $(\bar{x}, \bar{y}) = (0.5, 0.5)$ and no local optimal solutions. The points $(\bar{x}, \bar{y}, \bar{\lambda}_1, \bar{\lambda}_2) = (0.5, 0.5, 1, 0)$ and $(x^0, y^0, \lambda_1^0, \lambda_2^0) = (0, 1, 0, 1)$ are global or local optimal solutions, resp., of problem (11.14). To see that the point $(x^0, y^0, \lambda_1^0, \lambda_2^0) = (0, 1, 0, 1)$ is locally optimal remark that in a not too large open neighborhood V of this point we have

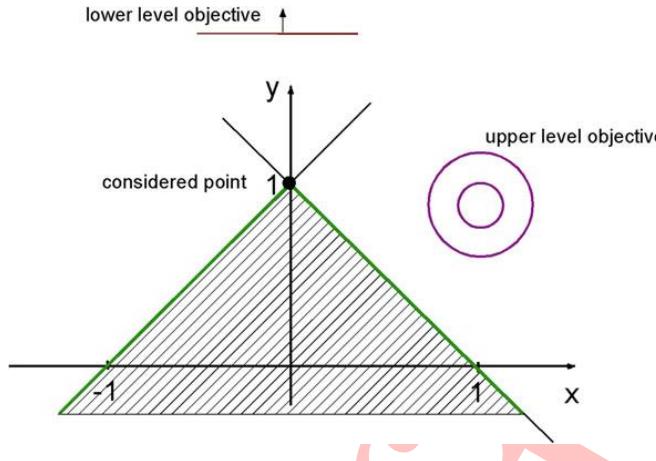


图 11.4: The considered stationary solution in Example 755

$\lambda_2 > 0$ implying that the second constraint in problem (11.15) is active, i.e. $y = x + 1$ which by feasibility implies $x \leq 0$. Substituting this into the upper level objective function gives

$$(x - 1)^2 + (y - 1)^2 = (x - 1)^2 + x^2 \geq 1$$

for each feasible point of the mathematical program with equilibrium constraints corresponding to the bilevel problem (11.16) in V . Together with $F(0, 1) = 1$ we derive that $(x^0, y^0, \lambda_1^0, \lambda_2^0)$ is indeed a local optimum of the mathematical program with equilibrium constraints corresponding to (11.16).

If the upper level objective function is replaced with a linear function it can also be shown that a local optimal solution of the MPEC corresponding to a linear bilevel optimization problem needs not to be related to a local optimum of the original problem.

The result in Theorem 754 is not correct if Slater's conditions is violated at some points, see [36].

Example 756. Consider the convex lower level problem

$$\min\{y_1 : y_1^2 - y_2 \leq x, y_1^2 + y_2 \leq 0\} \quad (11.17)$$

Then, for $x = 0$, Slater's condition is violated, $y = 0$ is the only feasible point for $x = 0$. The optimal solution for $x \geq 0$ is

$$y(x) = \begin{cases} (0, 0)^\top & \text{for } x = 0 \\ (-\sqrt{x/2}, -x/2) & \text{for } x > 0 \end{cases}$$

The Lagrange multiplier is $\lambda(x)$ with $\lambda_1(x) = \lambda_2(x) = \frac{1}{4\sqrt{x/2}}$ for $x > 0$. For $x = 0$, the problem is not regular and the Karush-Kuhn-Tucker conditions are not satisfied.

Let us consider the bilevel optimization problem

$$\min\{x : x \geq 0, y \in \Psi(x)\} \quad (11.18)$$

where $\Psi(x)$ is the solution set mapping of problem (11.17).

Then, the unique (global) optimal solution of the bilevel optimization problem (11.18) is $x = 0, y = 0$ and there do not exist local optimal solutions different from the global solution.

Consider the corresponding MPEC. Then, $(x, y(x), \lambda(x))$ is feasible for the MPEC for $x > 0$ and the objective function value converges to zero for $x \rightarrow 0$. However, an optimal solution of this problem does not exist, since for $x = 0$ the only optimal solution of the lower level problem is $y = 0$ and there does not exist a corresponding feasible solution of the MPEC.

Note that this example can be used to show that the MPEC need not to have any global optimal solution even if its feasible set is not empty and bounded.

The next result shows, that under a mild assumption the assertion in Theorem 754 is true iff (\bar{x}, \bar{y}) is globally optimal for problem (11.4).

Theorem 757. ([36]) Let $(\bar{x}, \bar{y}, \bar{\lambda})$ be a global optimal solution of problem (11.14), assume that the functions $y \mapsto f(x, y), y \mapsto g_i(x, y), i = 1, \dots, p$ are convex and that Slater's constraint qualification is satisfied for the lower level problem (11.1) for each $x \in X$. Then, (\bar{x}, \bar{y}) is a global optimal solution of the bilevel optimization problem.

The main part of the proof in [36] is based on independence of the optimal function value on the vector $\lambda \in \Lambda(x, y)$.

The following example is borrowed from [36]:

Example 758. Consider the lower level problem

$$\Psi_L(x) := \operatorname{argmin}_{y,z} \{-y - z : x + y \leq 1, -x + y \leq 1, 0 \leq z \leq 1\}$$

and the bilevel optimization problem

$$\min\{0.5x - y + 3z : (y, z) \in \Psi_L(x)\}.$$

Then, we derive $\Psi_L(x) = \{(y(x), z(x)) : z(x) = 1, x \in \mathbb{R}\}$ with

$$y(x) = \begin{cases} 1 - x, & \text{if } x \geq 0 \\ 1 + x, & \text{if } x \leq 0. \end{cases}$$

Substituting this solution into the upper level objective function we obtain

$$\begin{aligned} & 0.5x - y(x) + 3z(x) \\ = & \begin{cases} 0.5x - 1 + x + 3 = 2 + 1.5x \geq 2 & \text{if } x \geq 0 \\ 0.5x - 1 - x + 3 = 2 - 0.5x \geq 2 & \text{if } x \leq 0. \end{cases} \end{aligned}$$

Hence, $(\bar{x}, \bar{y}, \bar{z}) = (0, 1, 1)$ is the global (and unique local) optimal solution of the problem. At this point, three constraints in the lower level problem are active and the linear independence constraint qualification is violated.

Note that small smooth perturbations of the data of both the lower and the upper level problems will have no impact on this property. To see this, consider e.g. the case when the right-hand side of the first constraint $x + y \leq 1$ in the follower's problem is perturbed and the new lower level problem reads as follows:

$$\Psi_L(x) := \underset{y,z}{\operatorname{argmin}} \{-y - z : x + y \leq 1 + \alpha, -x + y \leq 1, 0 \leq z \leq 1\}.$$

Then, the optimal solution of the lower level problem becomes $z = 1$ and

$$y(x) = \begin{cases} 1 - x + \alpha & \text{if } x \geq 0.5\alpha \\ 1 + x & \text{if } x \leq 0.5\alpha. \end{cases}$$

Substituting this solution into the upper level objective function gives

$$0.5x - y(x) + 3z(x) = 0.5x - 1 + x - \alpha + 3 = 2 + 1.5x - \alpha \geq 2 - \alpha/4$$

if $x \geq 0.5\alpha$ and

$$0.5x - y(x) + 3z(x) = 0.5x - 1 - x + 3 = 2 - 0.5x \geq 2 - \alpha/4$$

if $x \leq 0.5\alpha$. The optimal objective function value is then $2 - \alpha/4$ with optimal solution $(x, y, z) = (\alpha/2, 1 + \alpha/2, 1)$ and again three constraints of the lower level problem are active. The other perturbations can be treated analogously.

Generic properties of bilevel optimization problems and mathematical programs with equilibrium constraints can be found in the papers [3, 66–68]. Due to its importance with respect to solution algorithms for bilevel optimization problems two main observations should be repeated:

1. Solution algorithms for mathematical programs with equilibrium constraints compute

stationary points as e.g. C-stationary or M-stationary points. Under additional assumptions these points are local minima of the MPEC. Due to Theorem 754 such a solution needs not to be related to a local optimum of the bilevel optimization problem if the Lagrangian multiplier of the lower level problem is not unique.

2. The last example shows that the linear independence constraint qualification is not generically satisfied in the lower level problem at the (global) optimal solution of the bilevel optimization problem. Hence, the Lagrange multiplier for the lower level problem needs not to be unique in general. This is in general not related to the MPEC-LICQ defined on page 68 of [38]. This is a bit surprising since (LICQ) is generically satisfied at optimal solutions of smooth optimization problems.

Optimal value transformation: Recall the definition of the optimal value function $\varphi(x)$. Then, a third reformulation is

$$\begin{aligned} & \min F(x, y) \\ & G(x) \leq 0 \\ & f(x, y) \leq \varphi(x) \\ & g(x, y) \leq 0, y \in \mathbf{T} \\ & x \in \mathbf{X}. \end{aligned} \tag{11.19}$$

This, again is a nonsmooth optimization problem since the optimal value function is in general not differentiable, even if all the constraint functions and the objective function in the lower level problem are smooth. We call this the *optimal value transformation*. Problem (11.19) is again fully equivalent to problem (11.4).

Draft

第十二章 Solving Sparse or Low-Rank Problems

(Taken from Chapter 4.2 of [83])

12.1 Nonconvex Algorithms

12.1.1 Generalized Singular Value Thresholding

Many nonconvex penalty functions have been proposed to enhance the sparse vector recovery while avoiding the discreteness disadvantages of the ℓ_0 norm. It is natural to apply these nonconvex penalty functions to singular values of a matrix, as a substitute of the rank function, to enhance low-rank matrix recovery. However, solving nonconvex low-rank minimization problems is much more challenging than solving nonconvex sparse minimization problems. So it is necessary to study low-rankness oriented optimization algorithms. Lu et al. [88, 91, 92] observed that all the existing nonconvex penalty functions are concave and monotonically increasing on $[0, \infty]$. So they considered the following low-rank model:

$$\min_{\mathbf{X}} F(\mathbf{X}) \triangleq \sum_{i=1}^{n(2)} g(\sigma_i(\mathbf{X})) + f(\mathbf{X}), \quad (12.1)$$

where $f(\mathbf{X})$ is L_f -smooth and g is a non-decreasing concave function on \mathbb{R}^+ , such as x^p ($0 < p < 1$). Table 12.1 gives examples of such g 's and their supergradients. Figure 12.1 gives their illustrations. Since $f(\mathbf{X})$ is L_f -smooth, it is natural to linearize $f(\mathbf{X})$ and minimize

$$Q(\mathbf{X}, \mathbf{Y}_k) = \sum_{i=1}^{n(2)} g(\sigma_i(\mathbf{X})) + \langle \nabla f(\mathbf{Y}_k), \mathbf{X} - \mathbf{Y}_k \rangle + \frac{L_f}{2} \|\mathbf{X} - \mathbf{Y}_k\|_F^2 \quad (12.2)$$

instead. Then in each iteration one only needs to solve subproblems in the following form:

$$\min_{\mathbf{X}} \sum_{i=1}^{n(2)} g(\sigma_i(\mathbf{X})) + \frac{1}{2} \|\mathbf{X} - \mathbf{B}\|_F^2. \quad (12.3)$$

Problem (12.3) is called Generalized Singular Value Thresholding (GSVT). Before solving it, we first present a result by Lu et al. [88] showing that the proximal operator $\text{Prox } g(\cdot)$ is monotone for any lower bounded function g .

Theorem 759 (Monotonicity of Proximal Operator [88]). *For any lower bounded function on \mathcal{C} , its proximal operator $\text{Prox } g(\cdot)$ is monotone, i.e., for any $x_i \in \mathcal{C}$ and $p_i^* \in \text{Prox } g(x_i)$, $i = 1, 2$, we have $(p_1^* - p_2^*)(x_1 - x_2) \geq 0$.*

表 12.1: Popular nonconvex surrogate functions of $\|\theta\|_0$ and their supergradients, adapted from [92].

Penalty	$g(\theta)(\theta \geq 0, \lambda > 0)$	Supergradient $\partial g(\theta)$
ℓ_p -norm	$\lambda\theta^p$	$\begin{cases} +\infty, & \text{if } \theta = 0, \\ \lambda p\theta^{p-1}, & \text{if } \theta > 0. \end{cases}$
SCAD [43]	$\begin{cases} \lambda\theta, & \text{if } \theta \leq \lambda, \\ \frac{-\theta^2+2\gamma\lambda\theta-\lambda^2}{2(\gamma-1)}, & \text{if } \lambda < \theta \leq \gamma\lambda, \\ \frac{\lambda^2(\gamma+1)}{2}, & \text{if } \theta > \gamma\lambda. \end{cases}$	$\begin{cases} \lambda, & \text{if } \theta \leq \lambda, \\ \frac{\gamma\lambda-\theta}{\gamma-1}, & \text{if } \lambda < \theta \leq \gamma\lambda, \\ 0, & \text{if } \theta > \gamma\lambda. \end{cases}$
Logarithm [51]	$\frac{\lambda}{\log(\gamma+1)} \log(\gamma\theta + 1)$	$\frac{\gamma\lambda}{(\gamma\theta+1) \log(\gamma+1)}$
MCP [140]	$\begin{cases} \lambda\theta - \frac{\theta^2}{2\gamma}, & \text{if } \theta < \gamma\lambda, \\ \frac{1}{2}\gamma\lambda^2, & \text{if } \theta \geq \gamma\lambda. \end{cases}$	$\begin{cases} \lambda - \frac{\theta}{\gamma}, & \text{if } \theta < \gamma\lambda, \\ 0, & \text{if } \theta \geq \gamma\lambda. \end{cases}$
Capped ℓ_1 [141]	$\begin{cases} \lambda\theta, & \text{if } \theta < \gamma, \\ \lambda\gamma, & \text{if } \theta \geq \gamma. \end{cases}$	$\begin{cases} \lambda, & \text{if } \theta < \gamma, \\ [0, \lambda], & \text{if } \theta = \gamma, \\ 0, & \text{if } \theta > \gamma. \end{cases}$
ETP [52]	$\frac{\lambda}{1-\exp(-\gamma)}(1 - \exp(-\gamma\theta))$	$\frac{\lambda\gamma}{1-\exp(-\gamma)} \exp(-\gamma\theta)$
Geman [54]	$\frac{\lambda\theta}{\theta+\gamma}$	$\frac{\lambda\gamma}{(\theta+\gamma)^2}$
Laplace [125]	$\lambda(1 - \exp(-\frac{\theta}{\gamma}))$	$\frac{\lambda}{\gamma} \exp(-\frac{\theta}{\gamma})$

Proof. The lower-boundedness assumption on g guarantees a finite solution to problem

$$\text{Prox } g(b) = \underset{x \in \mathcal{C}}{\operatorname{argmin}} g(x) + \frac{1}{2}(x - b)^2. \quad (12.4)$$

By the optimality of p_i^* , $i = 1, 2$, we have

$$g(p_2^*) + \frac{1}{2}(p_2^* - x_1)^2 \geq g(p_1^*) + \frac{1}{2}(p_1^* - x_1)^2, \quad \text{and} \quad (12.5)$$

$$g(p_1^*) + \frac{1}{2}(p_1^* - x_2)^2 \geq g(p_2^*) + \frac{1}{2}(p_2^* - x_2)^2. \quad (12.6)$$

Summing them together gives

$$(p_2^* - x_1)^2 + (p_1^* - x_2)^2 \geq (p_1^* - x_1)^2 + (p_2^* - x_2)^2, \quad (12.7)$$

which reduces to

$$(p_1^* - p_2^*)(x_1 - x_2) \geq 0.$$

□

Lu et al. [88] then provided a theorem to solve problem (12.3).

Theorem 760 (Solution to GSVT [88]). *Let $g : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ be a lower bounded function. Let $\mathbf{B} = \mathbf{U} \text{Diag}(\boldsymbol{\sigma}(\mathbf{B})) \mathbf{V}^T$ be the full SVD of $\mathbf{B} \in \mathbb{R}^{m \times n}$. Then an optimal solution to (12.3) is*

$$\mathbf{X}^* = \mathbf{U} \text{Diag}(\boldsymbol{\varrho}^*) \mathbf{V}^T, \quad (12.8)$$

where $\boldsymbol{\varrho}^*$ satisfies $\varrho_1^* \geq \varrho_2^* \geq \dots \geq \varrho_{n(2)}^*$, and for $i = 1, \dots, n(2)$,

$$\varrho_i^* \in \text{Prox } g(\sigma_i(\mathbf{B})) = \underset{\varrho_i \geq 0}{\operatorname{argmin}} g(\varrho_i) + \frac{1}{2}(\varrho_i - \sigma_i(\mathbf{B}))^2. \quad (12.9)$$

We first quote

Theorem 761 (von Neumann's Trace Inequality [61]). *For any matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m \times n}$ ($m \leq n$), $\text{tr}(\mathbf{A}^T \mathbf{B}) \leq \sum_{i=1}^m \sigma_i(\mathbf{A}) \sigma_i(\mathbf{B})$, where $\sigma_1(\mathbf{A}) \geq \sigma_2(\mathbf{A}) \geq \dots \geq 0$ and $\sigma_1(\mathbf{B}) \geq \sigma_2(\mathbf{B}) \geq \dots \geq 0$ are the singular values of \mathbf{A} and \mathbf{B} , respectively. The equality holds if and only if there exist column orthonormal matrices \mathbf{U} and \mathbf{V} such that $\mathbf{A} = \mathbf{U} \text{Diag}(\sigma(\mathbf{A})) \mathbf{V}^T$ and $\mathbf{B} = \mathbf{U} \text{Diag}(\sigma(\mathbf{B})) \mathbf{V}^T$ are the SVDs of \mathbf{A} and \mathbf{B} , simultaneously.*

Then we prove Theorem 760.

Proof. Denote $\sigma_1(\mathbf{X}) \geq \dots \geq \sigma_{n(2)}(\mathbf{X}) \geq 0$ as the singular values of \mathbf{X} . Problem (12.3) can be rewritten as

$$\min_{\boldsymbol{\varrho}: \varrho_1 \geq \dots \geq \varrho_{n(2)} \geq 0} \left\{ \min_{\boldsymbol{\sigma}(\mathbf{X}) = \boldsymbol{\varrho}} \sum_{i=1}^{n(2)} g(\varrho_i) + \frac{1}{2} \|\mathbf{X} - \mathbf{B}\|_F^2 \right\}. \quad (12.10)$$

By using the von Neumann's trace inequality (Theorem 761), we have

$$\begin{aligned} \|\mathbf{X} - \mathbf{B}\|_F^2 &= \text{tr}(\mathbf{X}^T \mathbf{X}) - 2\text{tr}(\mathbf{X}^T \mathbf{B}) + \text{tr}(\mathbf{B}^T \mathbf{B}) \\ &= \sum_{i=1}^{n(2)} \sigma_i^2(\mathbf{X}) - 2\text{tr}(\mathbf{X}^T \mathbf{B}) + \sum_{i=1}^{n(2)} \sigma_i^2(\mathbf{B}) \\ &\geq \sum_{i=1}^{n(2)} \sigma_i^2(\mathbf{X}) - 2 \sum_{i=1}^{n(2)} \sigma_i(\mathbf{X}) \sigma_i(\mathbf{B}) + \sum_{i=1}^{n(2)} \sigma_i^2(\mathbf{B}) \\ &= \sum_{i=1}^{n(2)} (\sigma_i(\mathbf{X}) - \sigma_i(\mathbf{B}))^2. \end{aligned}$$

Note that the above equality holds when \mathbf{X} admits the singular value decomposition $\mathbf{X} = \mathbf{U} \text{Diag}(\boldsymbol{\sigma}(\mathbf{X})) \mathbf{V}^T$, where \mathbf{U} and \mathbf{V} consist of the left and right singular vectors in the SVD of \mathbf{B} , respectively. In this case, problem (12.10) is reduced to

$$\min_{\boldsymbol{\varrho}: \varrho_1 \geq \dots \geq \varrho_{n(2)} \geq 0} \sum_{i=1}^{n(2)} \left(g(\varrho_i) + \frac{1}{2}(\varrho_i - \sigma_i(\mathbf{B}))^2 \right). \quad (12.11)$$

Since $\sigma_1(\mathbf{B}) \geq \sigma_2(\mathbf{B}) \geq \cdots \geq \sigma_{n(2)}(\mathbf{B})$, by Theorem 759 there exist $\varrho_i^* \in \text{Prox } g(\sigma_i(\mathbf{B}))$, such that $\varrho_1^* \geq \varrho_2^* \geq \cdots \geq \varrho_{n(2)}^*$. Such a choice of ϱ^* is optimal to (12.11), and thus (12.8)-(12.9) is optimal to (12.3). \square

From the above proof, we can see that the monotone property of $\text{Prox } g(\cdot)$ plays a key role in making problem (12.11) separable. Thus the solution (12.8)-(12.9) to (12.3) shares a similar formulation as the well-known Singular Value Thresholding (SVT) operator associated with the convex nuclear norm [22]. Note that for a convex g , $\text{Prox } g(\cdot)$ is always monotone. Indeed,

$$(p_1^* - p_2^*)(x_1 - x_2) \geq (p_1^* - p_2^*)^2 \geq 0, \quad \forall x_i \text{ and } p_i^* \in \text{Prox } g(x_i), i = 1, 2.$$

The above inequality can be obtained by the optimality of $\text{Prox } g(\cdot)$ and the convexity of g . Both the above inequality and Theorem 759 can be easily generalized to the multi-dimensional case.

There was some previous work claiming that the solution (12.8)-(12.9) is optimal to (12.3) for some special choices of nonconvex g . However, their results were not rigorous since they did not prove the monotone property of $\text{Prox } g(\cdot)$. Here we provide the first justification.

Note that it is possible that $\sigma_i(\mathbf{B}) = \sigma_j(\mathbf{B})$ for some $i < j$ in (12.9). Since $\text{Prox } g(\cdot)$ may not be unique when g is nonconvex, in this case we need to choose $\varrho_i^* \in \text{Prox } g(\sigma_i(\mathbf{B}))$ and $\varrho_j^* \in \text{Prox } g(\sigma_j(\mathbf{B}))$ such that $\varrho_i^* \geq \varrho_j^*$. This makes the major difference between GSVT and SVT.

Figure 12.2 illustrates the shrinkage effect of proximal operators of the convex ℓ_1 -norm and some functions in Table 12.1. Their shrinkage and thresholding effects are similar when b is relatively small. However, when b is relatively large, the proximal operator of the convex ℓ_1 -norm is biased, i.e., $\text{Prox } g(b) = b - 1$, implying that the ℓ_1 -norm may over-penalize. In contrast, the proximal operators of the nonconvex functions are closer to be unbiased, i.e., $\text{Prox } g(b) \approx b$. This also testifies to the necessity of using nonconvex penalties on the singular values to approximate the rank function.

12.1.2 Iteratively Reweighted Nuclear Norm Algorithm

Problem (12.1) can also be solved by the Iteratively Reweighted Nuclear Norm (IRNN) algorithm [92]. We assume that the penalty function g and the loss function f satisfy the following assumptions:

- A1** $g : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ is continuous, concave and monotonically increasing on $[0, \infty)$. It can be nonsmooth.

A2 $f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$ is an L_f -smooth function:

$$\|\nabla f(\mathbf{X}) - \nabla f(\mathbf{Y})\|_F \leq L_f \|\mathbf{X} - \mathbf{Y}\|_F, \quad \forall \mathbf{X}, \mathbf{Y} \in \mathbb{R}^{m \times n}. \quad (12.12)$$

$f(\mathbf{X})$ can be nonconvex.

A3 $F(\mathbf{X})$ is coercive: $F(\mathbf{X}) \rightarrow \infty$ if and only if $\|\mathbf{X}\|_F \rightarrow \infty$.

Table 12.1 gives examples of commonly used g 's and their supergradients and Figure 12.1 gives their illustrations. Since g is concave on $[0, \infty)$, we have

$$g(\sigma_i) \leq g(\sigma_i^k) + w_i^k(\sigma_i - \sigma_i^k), \quad (12.13)$$

where

$$w_i^k \in \partial g(\sigma_i^k). \quad (12.14)$$

As $\sigma_1^k \geq \sigma_2^k \geq \dots \geq \sigma_s^k \geq 0$, where $s = n_{(2)}$, by the increment of g and the anti-monotone property of supergradient, we have

$$0 \leq w_1^k \leq w_2^k \leq \dots \leq w_s^k. \quad (12.15)$$

This property will play an important role in our algorithm shown later. (12.13) motivates us to minimize its right hand side instead of $g(\sigma_i)$ directly, again in the spirit of Majorization Minimization [89, 134]. Thus we may solve the following surrogate problem instead:

$$\begin{aligned} \mathbf{X}_{k+1} &= \operatorname{argmin}_{\mathbf{X}} \sum_{i=1}^s g(\sigma_i^k) + w_i^k(\sigma_i - \sigma_i^k) + f(\mathbf{X}) \\ &= \operatorname{argmin}_{\mathbf{X}} \sum_{i=1}^s w_i^k \sigma_i + f(\mathbf{X}). \end{aligned} \quad (12.16)$$

Updating \mathbf{X}_{k+1} by solving the above weighted nuclear norm problem (12.16) appears to be an extension of the weighted ℓ_1 -norm problem in the IRL1 algorithm [25], which is a special DC programming algorithm. However, this is actually not true because in general the weighted nuclear norm in (12.16) is nonconvex (it is convex if and only if $w_1^k \geq w_2^k \geq \dots \geq w_s^k \geq 0$ [27]), while the weighted ℓ_1 -norm is always convex. The difference resides in that singular values of matrices are always sorted. So solving the nonconvex problem (12.16) is much more challenging than the convex weighted ℓ_1 -norm problem. In fact, it may not be easier than solving the original problem (12.1).

To proceed, we quote

Lemma 762 ([10, 12]). *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be an L -smooth function. Then for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$,*

$$f(\mathbf{x}) \leq f(\mathbf{y}) + \langle \mathbf{x} - \mathbf{y}, \nabla f(\mathbf{y}) \rangle + \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|^2. \quad (12.17)$$

Proof. We write

$$f(\mathbf{y}) - f(\mathbf{x}) = \int_0^1 \frac{df(\mathbf{x} + t(\mathbf{y} - \mathbf{x}))}{dt} dt = \int_0^1 [\nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x}))]^T (\mathbf{y} - \mathbf{x}) dt. \quad (12.18)$$

So

$$\begin{aligned} & f(\mathbf{y}) - f(\mathbf{x}) - [\nabla f(\mathbf{x})]^T (\mathbf{y} - \mathbf{x}) \\ &= \int_0^1 [\nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x})]^T (\mathbf{y} - \mathbf{x}) dt \\ &\leq \int_0^1 \|\nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x})\| \|\mathbf{y} - \mathbf{x}\| dt \\ &\leq \int_0^1 Lt \|\mathbf{y} - \mathbf{x}\|^2 dt = \frac{L}{2} \|\mathbf{y} - \mathbf{x}\|^2. \end{aligned} \quad (12.19)$$

□

To address the difficulty in solving (12.16), we further linearize $f(\mathbf{X})$ at \mathbf{X}_k and add a proximal term:

$$f(\mathbf{X}) \leq f(\mathbf{X}_k) + \langle \nabla f(\mathbf{X}_k), \mathbf{X} - \mathbf{X}_k \rangle + \frac{\mu}{2} \|\mathbf{X} - \mathbf{X}_k\|_F^2,$$

where any $\mu \geq L_f$ makes the above inequality valid due to Lemma 762. Then we update \mathbf{X}_{k+1} by solving

$$\begin{aligned} \mathbf{X}_{k+1} &= \underset{\mathbf{X}}{\operatorname{argmin}} \sum_{i=1}^s w_i^k \sigma_i + f(\mathbf{X}_k) + \langle \nabla f(\mathbf{X}_k), \mathbf{X} - \mathbf{X}_k \rangle + \frac{\mu}{2} \|\mathbf{X} - \mathbf{X}_k\|_F^2 \\ &= \underset{\mathbf{X}}{\operatorname{argmin}} \sum_{i=1}^s w_i^k \sigma_i + \frac{\mu}{2} \left\| \mathbf{X} - \left(\mathbf{X}_k - \frac{1}{\mu} \nabla f(\mathbf{X}_k) \right) \right\|_F^2 \end{aligned} \quad (12.20)$$

instead. Problem (12.20), being the proximal operator of the weighted nuclear norm, is still nonconvex, but actually it has a closed-form solution thanks to (12.15).

Lemma 763 ([27]). *For any $\lambda > 0$, $\mathbf{Y} \in \mathbb{R}^{m \times n}$ and $0 \leq w_1 \leq w_2 \leq \dots \leq w_s$, where $s = n_{(2)}$, a globally optimal solution to the following problem*

$$\min_{\mathbf{X}} \lambda \sum_{i=1}^s w_i \sigma_i(\mathbf{X}) + \frac{1}{2} \|\mathbf{X} - \mathbf{Y}\|_F^2, \quad (12.21)$$

is given by the weighted singular value thresholding:

$$\mathbf{X}^* = \mathbf{U} \mathcal{S}_{\lambda w}(\Sigma) \mathbf{V}^T, \quad (12.22)$$

where $\mathbf{Y} = \mathbf{U} \Sigma \mathbf{V}^T$ is the SVD of \mathbf{Y} and $\mathcal{S}_{\lambda w}(\Sigma) = \text{Diag}(\{\max(\Sigma_{ii} - \lambda w_i, 0)\})$.

Algorithm 10 Solving problem (12.1) by IRNN

1: **Input:** $\mu > L_f$, where L_f is a Lipschitz constant of $\nabla f(\mathbf{X})$.

2: **Initialize:** $k = 0$, \mathbf{X}_k , and w_i^k , $i = 1, \dots, s$.

3: **while** not converged **do**

4: Update \mathbf{X}_{k+1} by solving problem (12.20).

5: Update the weights w_i^{k+1} , $i = 1, \dots, s$, by

$$w_i^{k+1} \in \partial g(\sigma_i(\mathbf{X}_{k+1})). \quad (12.23)$$

6: $k \leftarrow k + 1$.

7: **end while**

8: **Output:** \mathbf{X}_k .

It is worth mentioning that for any g satisfying assumption **A3** and $\partial g(0) = \{\infty\}$, such as x^p ($0 < x < 1$), by the updating rule of \mathbf{X}_{k+1} in (12.20)-(12.22), we have that $\sigma_i^{k+1} = 0$ if $\sigma_i^k = 0$. So the rank of the sequence $\{\mathbf{X}_k\}$ is nonincreasing.

Iteratively updating w_i^k , $i = 1, \dots, s$, by (12.14) and \mathbf{X}_{k+1} by (12.20) leads to the proposed IRNN algorithm. The whole procedure of IRNN is shown in Algorithm 10. If the Lipschitz constant L_f is unknown or uncomputable, the backtracking rule can be used to estimate μ in each iteration [10].

12.1.2.1 Convergence Analysis

In this section, we give the convergence analysis on the IRNN algorithm, which is summarized in the following theorems.

Theorem 764 (Properties of IRNN [92]). *Assume that g and f in problem (12.1) satisfy the assumptions **A1-A3**. Then the sequence $\{\mathbf{X}_k\}$ generated by Algorithm 10 satisfies the following properties:*

(1) $F(\mathbf{X}_k)$ has sufficient descent. Indeed,

$$F(\mathbf{X}_k) - F(\mathbf{X}_{k+1}) \geq \frac{\mu - L_f}{2} \|\mathbf{X}_k - \mathbf{X}_{k+1}\|_F^2 \geq 0;$$

(2) $\lim_{k \rightarrow \infty} (\mathbf{X}_k - \mathbf{X}_{k+1}) = \mathbf{0}$;

(3) The sequence $\{\mathbf{X}_k\}$ is bounded.

Proof. First, since \mathbf{X}_{k+1} is a globally optimal solution to problem (12.20), we get

$$\begin{aligned} & \sum_{i=1}^s w_i^k \sigma_i^{k+1} + \langle \nabla f(\mathbf{X}_k), \mathbf{X}_{k+1} - \mathbf{X}_k \rangle + \frac{\mu}{2} \|\mathbf{X}_{k+1} - \mathbf{X}_k\|_F^2 \\ & \leq \sum_{i=1}^s w_i^k \sigma_i^k + \langle \nabla f(\mathbf{X}_k), \mathbf{X}_k - \mathbf{X}_k \rangle + \frac{\mu}{2} \|\mathbf{X}_k - \mathbf{X}_k\|_F^2. \end{aligned}$$

It can be rewritten as

$$\langle \nabla f(\mathbf{X}_k), \mathbf{X}_k - \mathbf{X}_{k+1} \rangle \geq - \sum_{i=1}^s w_i^k (\sigma_i^k - \sigma_i^{k+1}) + \frac{\mu}{2} \|\mathbf{X}_k - \mathbf{X}_{k+1}\|_F^2. \quad (12.24)$$

Second, since $f(\mathbf{X})$ is L_f -smooth, by using Lemma 762 we have

$$f(\mathbf{X}_k) - f(\mathbf{X}_{k+1}) \geq \langle \nabla f(\mathbf{X}_k), \mathbf{X}_k - \mathbf{X}_{k+1} \rangle - \frac{L_f}{2} \|\mathbf{X}_k - \mathbf{X}_{k+1}\|_F^2. \quad (12.25)$$

Third, since $w_i^k \in \partial g(\sigma_i^k)$, by the definition of the supergradient we have

$$g(\sigma_i^k) - g(\sigma_i^{k+1}) \geq w_i^k (\sigma_i^k - \sigma_i^{k+1}). \quad (12.26)$$

Summing (12.24), (12.25), and (12.26), for $i = 1, \dots, s$, together, we obtain

$$\begin{aligned} F(\mathbf{X}_k) - F(\mathbf{X}_{k+1}) &= \sum_{i=1}^s (g(\sigma_i^k) - g(\sigma_i^{k+1})) + f(\mathbf{X}_k) - f(\mathbf{X}_{k+1}) \\ &\geq \frac{\mu - L_f}{2} \|\mathbf{X}_{k+1} - \mathbf{X}_k\|_F^2 \geq 0. \end{aligned} \quad (12.27)$$

Thus $F(\mathbf{X}_k)$ has sufficient descent. Summing all the inequalities in (12.27) for $k \geq 1$, we get

$$F(\mathbf{X}_1) - F^* \geq \frac{\mu - L_f}{2} \sum_{k=1}^{\infty} \|\mathbf{X}_{k+1} - \mathbf{X}_k\|_F^2, \quad (12.28)$$

where F^* is the minimal objective function value. So

$$\sum_{k=1}^{\infty} \|\mathbf{X}_k - \mathbf{X}_{k+1}\|_F^2 \leq \frac{2(F(\mathbf{X}_1) - F^*)}{\mu - L_f}. \quad (12.29)$$

It implies that $\lim_{k \rightarrow \infty} (\mathbf{X}_k - \mathbf{X}_{k+1}) = \mathbf{0}$. The boundedness of $\{\mathbf{X}_k\}$ is a consequence of assumption **A3**. \square

Theorem 765 (Convergence of IRNN [92]). *Let $\{\mathbf{X}_k\}$ be the sequence generated by Algorithm 10. If the g in assumption **A1** is differentiable on $(0, +\infty)$, then any accumulation point \mathbf{X}^* of $\{\mathbf{X}_k\}$ is a stationary point of (12.1).*

Proof. The sequence $\{\mathbf{X}_k\}$ generated in Algorithm 10 is bounded as shown in Theorem 764. Thus there exists a matrix \mathbf{X}^* and a subsequence $\{\mathbf{X}_{k_j}\}$ such that $\lim_{j \rightarrow \infty} \mathbf{X}_{k_j} = \mathbf{X}^*$. From the fact that $\lim_{k \rightarrow \infty} (\mathbf{X}_k - \mathbf{X}_{k+1}) = \mathbf{0}$ in Theorem 764, we have $\lim_{j \rightarrow \infty} \mathbf{X}_{k_j+1} = \mathbf{X}^*$. Thus $\sigma_i(\mathbf{X}_{k_j+1}) \rightarrow \sigma_i(\mathbf{X}^*)$ for $i = 1, \dots, s$. By the choice of $w_i^{k_j} \in \partial g(\sigma_i(\mathbf{X}_{k_j}))$ and the upper semi-continuous property of the supergradient [31], any accumulation point w_i^* of $\{w_i^{k_j}\}$ is in $\partial g(\sigma_i(\mathbf{X}^*))$. Since g is nondecreasing on $[0, +\infty)$, we have that $w_i^{k_j} \geq 0$ and $w_i^* \geq 0$. If $\sigma_i(\mathbf{X}^*) > 0$, then $\partial g(\sigma_i(\mathbf{X}^*))$ is a singleton. If $\sigma_i(\mathbf{X}^*) = 0$, then w_i^* must be the right derivative of g at 0 due to $w_i^{k_j} \geq 0$. This is because g is concave, hence its right derivative exists. So w_i^* is also a singleton. Thus $\{w_i^{k_j}\}$ converges to w_i^* .

Define $h(\mathbf{X}, \mathbf{w}) = \sum_{i=1}^s w_i \sigma_i(\mathbf{X})$. Then $\lim_{j \rightarrow +\infty} h(\mathbf{X}_{k_j+1}, \mathbf{w}^{k_j}) = h(\mathbf{X}^*, \mathbf{w}^*)$. Since \mathbf{X}_{k_j+1} is optimal to problem (12.20), there exists $\mathbf{G}_{k_j+1} \in \partial h(\mathbf{X}_{k_j+1}, \mathbf{w}^{k_j})$ such that

$$\mathbf{G}_{k_j+1} + \nabla f(\mathbf{X}_{k_j}) + \mu(\mathbf{X}_{k_j+1} - \mathbf{X}_{k_j}) = \mathbf{0}. \quad (12.30)$$

Let $j \rightarrow \infty$ in (12.30). By the upper semi-continuous property of the supergradient [31] again there exists $\mathbf{G}^* \in \partial h(\mathbf{X}^*, \mathbf{w}^*)$, such that

$$\mathbf{0} = \mathbf{G}^* + \nabla f(\mathbf{X}^*) \in \partial F(\mathbf{X}^*). \quad (12.31)$$

Thus \mathbf{X}^* is a stationary point of (12.1). □

12.1.3 Truncated Nuclear Norm Minimization

Another function that approximates rank is the Truncated Nuclear Norm (TNN) [62]: $\|\mathbf{X}\|_r = \sum_{i=r+1}^{n(2)} \sigma_i(\mathbf{X})$. It does not involve the largest r singular values. So it is not a convex function. The intuition behind TNN is obvious. By minimizing TNN, the tailing singular values will be encouraged to be small, while the magnitudes of the first r singular values are unaffected. So a solution closer to a rank r matrix can be obtained. As $\|\mathbf{X}\|_r$ is non-convex, solving the truncated nuclear norm minimization problem is not very easy. However, we have the following theorem:

Theorem 766 (Variational Formulation of the Sum of Top Singular Values [62]). *For any given matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$, any nonnegative integer r ($r \leq n(2)$), and any matrices $\mathbf{A} \in \mathbb{R}^{r \times m}$ and $\mathbf{B} \in \mathbb{R}^{r \times n}$ satisfying $\mathbf{A}\mathbf{A}^T = \mathbf{I}_{r \times r}$ and $\mathbf{B}\mathbf{B}^T = \mathbf{I}_{r \times r}$, we have*

$$\text{tr}(\mathbf{AXB}^T) \leq \sum_{i=1}^r \sigma_i(\mathbf{X}). \quad (12.32)$$

Moreover, the above equality is achieved when $\mathbf{A} = \mathbf{U}^T$ and $\mathbf{B} = \mathbf{V}^T$, where $\mathbf{U}\Sigma\mathbf{V}^T$ is the skinny SVD of \mathbf{X} .

Proof. By von Neumann's trace inequality (Theorem 761), we have

$$\text{tr}(\mathbf{AXB}^T) = \text{tr}(\mathbf{XB}^T\mathbf{A}) \leq \sum_{i=1}^{n(2)} \sigma_i(\mathbf{X})\sigma_i(\mathbf{B}^T\mathbf{A}), \quad (12.33)$$

where $\sigma_1(\mathbf{X}) \geq \dots \geq \sigma_{n(2)}(\mathbf{X}) \geq 0$. As $\text{rank}(\mathbf{A}) = r$ and $\text{rank}(\mathbf{B}) = r$, so $\text{rank}(\mathbf{B}^T\mathbf{A}) \triangleq r' \leq r$. For $i \leq r'$, $\sigma_i(\mathbf{B}^T\mathbf{A}) > 0$ and $\sigma_i^2(\mathbf{B}^T\mathbf{A})$ is the i -th eigenvalue of $\mathbf{A}^T\mathbf{B}\mathbf{B}^T\mathbf{A} = \mathbf{A}^T\mathbf{A}$, which is also an eigenvalue of $\mathbf{AA}^T = \mathbf{I}$. So $\sigma_i(\mathbf{B}^T\mathbf{A}) = 1$, $i = 1, \dots, r'$, while the rest are all zeros. Then it follows that

$$\begin{aligned} \sum_{i=1}^{n(2)} \sigma_i(\mathbf{X})\sigma_i(\mathbf{B}^T\mathbf{A}) &= \sum_{i=1}^{r'} \sigma_i(\mathbf{X})\sigma_i(\mathbf{B}^T\mathbf{A}) + \sum_{i=r'+1}^{n(2)} \sigma_i(\mathbf{X})\sigma_i(\mathbf{B}^T\mathbf{A}) \\ &= \sum_{i=1}^{r'} \sigma_i(\mathbf{X}) \cdot 1 + \sum_{i=r'+1}^{n(2)} \sigma_i(\mathbf{X}) \cdot 0 \\ &= \sum_{i=1}^{r'} \sigma_i(\mathbf{X}). \end{aligned} \quad (12.34)$$

Also, notice that

$$\sum_{i=1}^{r'} \sigma_i(\mathbf{X}) \leq \sum_{i=1}^r \sigma_i(\mathbf{X}). \quad (12.35)$$

So we have

$$\text{tr}(\mathbf{AXB}^T) \leq \sum_{i=1}^r \sigma_i(\mathbf{X}). \quad (12.36)$$

Finally, it is easy to check that the above equality is achieved when $\mathbf{A} = \mathbf{U}^T$ and $\mathbf{B} = \mathbf{V}^T$. \square

Thus

$$\|\mathbf{X}\|_r = \|\mathbf{X}\|_* - \max_{\mathbf{AA}^T = \mathbf{I}, \mathbf{BB}^T = \mathbf{I}} \text{tr}(\mathbf{AXB}^T). \quad (12.37)$$

By replacing the nuclear norm with TNN, and solving the corresponding optimization problem, e.g., by alternating minimization between \mathbf{X} and (\mathbf{A}, \mathbf{B}) or by APG or ADM which uses the proximal operator shown below, encouraging results can be obtained by the proposed algorithm [62].

TNN can be generalized by applying different weights to different singular values, obtaining the Weighted Nuclear Norm (WNN) [58]: $\|\mathbf{X}\|_{\mathbf{w},*} = \sum_{i=1}^{\min(m,n)} w_i \sigma_i(\mathbf{X})$, which we have introduced in (12.16). However, in this case in general the proximal operator

$$\min_{\mathbf{X}} \|\mathbf{X}\|_{\mathbf{w},*} + \frac{\alpha}{2} \|\mathbf{X} - \mathbf{W}\|_F^2 \quad (12.38)$$

does not have a closed-form solution. Instead, a small-scale optimization w.r.t. the singular values needs to be solved numerically.

Theorem 767 (Proximal Operator of WNN [58]). *The optimal solution to (12.38) is given by $\mathbf{X} = \mathbf{U} \text{Diag}(\{\sigma_i(\mathbf{X})\}) \mathbf{V}^T$, where $\mathbf{U} \text{Diag}(\{\sigma_i(\mathbf{W})\}) \mathbf{V}^T$ is the SVD of \mathbf{W} , and*

$$\{\sigma_i(\mathbf{X})\} = \underset{\sigma_1 \geq \dots \geq \sigma_{n(2)} \geq 0}{\operatorname{argmin}} \sum_{i=1}^{n(2)} w_i \sigma_i + \frac{\alpha}{2} \sum_{i=1}^{n(2)} (\sigma_i - \sigma_i(\mathbf{W}))^2. \quad (12.39)$$

Proof. The proof is similar to that of Theorem 760. To see this,

$$\begin{aligned} \|\mathbf{X}\|_{\mathbf{w},*} + \frac{\alpha}{2} \|\mathbf{X} - \mathbf{W}\|_F^2 &= \sum_{i=1}^{n(2)} w_i \sigma_i + \frac{\alpha}{2} (\|\mathbf{X}\|_F^2 - 2\langle \mathbf{X}, \mathbf{W} \rangle + \|\mathbf{W}\|_F^2) \\ &= \sum_{i=1}^{n(2)} w_i \sigma_i + \frac{\alpha}{2} \left(\sum_{i=1}^{n(2)} \sigma_i^2 - 2\langle \mathbf{X}, \mathbf{W} \rangle + \sum_{i=1}^{n(2)} \sigma_i^2(\mathbf{W}) \right) \\ &\geq \sum_{i=1}^{n(2)} w_i \sigma_i + \frac{\alpha}{2} \left(\sum_{i=1}^{n(2)} \sigma_i^2 - 2 \sum_{i=1}^{n(2)} \sigma_i \sigma_i(\mathbf{W}) + \sum_{i=1}^{n(2)} \sigma_i^2(\mathbf{W}) \right) \\ &= \sum_{i=1}^{n(2)} w_i \sigma_i + \frac{\alpha}{2} \sum_{i=1}^{n(2)} (\sigma_i - \sigma_i(\mathbf{W}))^2 \quad (\sigma_1 \geq \dots \geq \sigma_{n(2)} \geq 0), \end{aligned} \quad (12.40)$$

where the inequality holds because of von Neumann's inequality (Theorem 761), and the equality can be achieved when \mathbf{X} and \mathbf{W} have the same singular vectors, as stated in the theorem. \square

Thus to solve subproblem (12.38), we only need to solve an equivalent problem (12.39). Indeed, we can use LADMAP [79] to solve (12.39) by reformulating (12.39) as:

$$\min_{\boldsymbol{\sigma}, \boldsymbol{\tau}} f(\boldsymbol{\sigma}), \quad \text{s.t.} \quad \mathcal{A}(\boldsymbol{\sigma}) + \mathcal{B}(\boldsymbol{\tau}) = \mathbf{0}, \quad \boldsymbol{\tau} \geq \mathbf{0}, \quad (12.41)$$

where

$$f(\boldsymbol{\sigma}) = \sum_{i=1}^{n(2)} w_i \sigma_i + \frac{\alpha}{2} \sum_{i=1}^{n(2)} (\sigma_i - \sigma_i(\mathbf{W}))^2, \quad (12.42)$$

$\mathcal{A}(\boldsymbol{\sigma}) = [\sigma_1 - \sigma_2, \dots, \sigma_{n(2)-1} - \sigma_{n(2)}, \sigma_{n(2)}]$, and $\mathcal{B}(\boldsymbol{\tau}) = [-\tau_1, \dots, -\tau_{n(2)}]$. Note that it is interesting that although problem (12.38) is nonconvex, problem (12.39) is convex.

When $w_1 \leq \dots \leq w_{n(2)}$, problem (12.39) has a closed-form solution given by Lemma 763.

12.1.4 Iteratively Reweighted Least Squares

For unconstrained problems which use the Schatten- p norm to approximate rank and the $\ell_{2,p}$ norm to approximate the $\ell_{2,0}$ norm, an effective way is Iteratively Reweighted Least Squares (IRLS) [87], i.e., approximating $\text{tr}((\mathbf{Z}\mathbf{Z}^T)^{p/2})$ with $\text{tr}((\mathbf{Z}_k\mathbf{Z}_k^T)^{(p/2)-1}(\mathbf{Z}\mathbf{Z}^T))$

and $\sum_{i=1}^n \|\mathbf{Z}_{:i}\|_2^q$ with $\sum_{i=1}^n \|\mathbf{Z}_{:i}^{(k)}\|_2^{q-2} \|\mathbf{Z}_{:i}\|_2^2$, where \mathbf{Z}_k is the value of low-rank matrix \mathbf{Z} at the k -th iteration and $\mathbf{Z}_{:i}^{(k)}$ is the i -th column of matrix \mathbf{Z} at the k -th iteration. So each time to update $\mathbf{Z} \in \mathbb{R}^{m \times n}$, a matrix equation needs to be solved.

To see this, let the smoothed optimization problem be

$$\begin{aligned} \min_{\mathbf{Z}} \mathcal{J}(\mathbf{Z}, \mu) &= \mathcal{L}(\mathbf{Z}) + \mathcal{F}(\mathbf{Z}) \\ &= \left\| \begin{bmatrix} \mathbf{Z} \\ \mu \mathbf{I} \end{bmatrix} \right\|_{S_p}^p + \lambda \left\| \begin{bmatrix} \mathbf{DZ} - \mathbf{D} \\ \mu \mathbf{1}^T \end{bmatrix} \right\|_{2,q}^q \\ &= \text{tr} (\mathbf{Z}^T \mathbf{Z} + \mu^2 \mathbf{I})^{\frac{p}{2}} + \lambda \sum_{i=1}^n (\|(\mathbf{DZ} - \mathbf{D})_{:i}\|_2^2 + \mu^2)^{\frac{q}{2}}. \end{aligned} \quad (12.43)$$

The derivative of $\mathcal{L}(\mathbf{Z})$ is

$$\frac{\partial \mathcal{L}}{\partial \mathbf{Z}} = p \mathbf{Z} (\mathbf{Z}^T \mathbf{Z} + \mu^2 \mathbf{I})^{\frac{p}{2}-1} \triangleq p \mathbf{Z} \mathbf{M}, \quad (12.44)$$

where $\mathbf{M} = (\mathbf{Z}^T \mathbf{Z} + \mu^2 \mathbf{I})^{\frac{p}{2}-1}$ is the weight matrix corresponding to $\mathcal{L}(\mathbf{Z})$. It is worth noting that \mathbf{M} can be computed without SVD [60] (but the *order* of complexity is still the same as that of SVD). For the derivative of \mathcal{F} , we have

$$\frac{\partial \mathcal{F}}{\partial \mathbf{Z}_{:i}} = \frac{q(\mathbf{D}^T \mathbf{DZ}_{:i} - \mathbf{D}^T \mathbf{D}_{:i})}{(\|(\mathbf{DZ} - \mathbf{D})_{:i}\|_2^2 + \mu^2)^{1-\frac{q}{2}}}. \quad (12.45)$$

Namely, $\partial \mathcal{F} / \partial \mathbf{Z} = q \mathbf{D}^T (\mathbf{DZ} - \mathbf{D}) \mathbf{N}$, where \mathbf{N} is the weight matrix corresponding to $\mathcal{F}(\mathbf{Z})$, which is a diagonal matrix with the i -th diagonal entry $\mathbf{N}_{ii} = (\|(\mathbf{DZ} - \mathbf{D})_{:i}\|_2^2 + \mu^2)^{q/2-1}$. So by requiring the derivative of \mathcal{J} w.r.t. \mathbf{Z} to be zero, we have

$$\frac{\partial \mathcal{J}}{\partial \mathbf{Z}} = p \mathbf{Z} \mathbf{M} + \lambda q \mathbf{D}^T (\mathbf{DZ} - \mathbf{D}) \mathbf{N} = \mathbf{0}, \quad (12.46)$$

or equivalently,

$$\lambda q \mathbf{D}^T \mathbf{DZ} + p \mathbf{Z} (\mathbf{M} \mathbf{N}^{-1}) = \lambda q \mathbf{D}^T \mathbf{D}. \quad (12.47)$$

(12.47) is known as the Sylvester equation, which can be solved by using `lyap()` in MATLAB.

It is noteworthy that both \mathbf{M} and \mathbf{N} depend only on \mathbf{Z} . Conversely, once \mathbf{M} and \mathbf{N} are fixed, \mathbf{Z} can be updated by solving the Sylvester equation (12.47). This fact motivates us to solve (12.44) by alternately updating \mathbf{Z} and (\mathbf{M}, \mathbf{N}) . The procedure is called Iteratively Reweighted Least Squares (IRLS), which is shown in Algorithm 11.

12.1.4.1 Convergence Analysis

Although minimizing non-convex optimization problems, when p or $q < 1$, the following two theorems guarantee the convergence of the IRLS algorithm:

Algorithm 11 Iteratively Reweighted Least Squares Algorithm for Problem (12.44)

- 1: **while** not converged **do**
- 2: Update \mathbf{Z}_{k+1} by solving the Sylvester equation

$$p\mathbf{Z}\mathbf{M}_k + \lambda q\mathbf{D}^T(\mathbf{D}\mathbf{Z} - \mathbf{D})\mathbf{N}_k = \mathbf{0}.$$

- 3: If $\|\mathbf{Z}_{k+1} - \mathbf{Z}_k\|_\infty \leq \varepsilon$, break.
- 4: Update \mathbf{M}_{k+1} and \mathbf{N}_{k+1} separately by

$$\mathbf{M}_{k+1} = (\mathbf{Z}_{k+1}^T \mathbf{Z}_{k+1} + \mu^2 \mathbf{I})^{\frac{p}{2}-1}, \quad (12.48)$$

$$(\mathbf{N}_{k+1})_{ij} = \begin{cases} ((\mathbf{D}\mathbf{Z}_{k+1} - \mathbf{D})_{:i} \|_2^2 + \mu^2)^{\frac{q}{2}-1}, & i = j, \\ 0, & i \neq j. \end{cases} \quad (12.49)$$

- 5: $k \leftarrow k + 1$.
 - 6: **end while**
 - 7: **Output:** \mathbf{Z}_k .
-

Theorem 768 (Properties of IRLS [87]). *The sequence $\{\mathbf{Z}_k\}$ generated in Algorithm 11 satisfies the following properties:*

1. $\mathcal{J}(\mathbf{Z}_k, \mu)$ is non-increasing, i.e., $\mathcal{J}(\mathbf{Z}_{k+1}, \mu) \leq \mathcal{J}(\mathbf{Z}_k, \mu)$;
2. The sequence $\{\mathbf{Z}_k\}$ is bounded;
3. $\lim_{k \rightarrow \infty} \|\mathbf{Z}_{k+1} - \mathbf{Z}_k\|_F = 0$.

Theorem 769 (Convergence of IRLS [87]). *Any accumulation point of the sequence $\{\mathbf{Z}_k\}$ generated by Algorithm 11 is a stationary point of problem (12.43). If $p \geq 1$ and $q \geq 1$, the stationary point is globally optimal.*

12.1.4.2 Experiments

When the regularization parameter μ is chosen appropriately, IRLS converges fast and results in an accurate solution. To solve Relaxed LRR (7.180), Lu et al. [87] proposed decreasing μ by $\mu_{k+1} = \mu_k / \rho$ with $\rho > 1$ and compared IRLS with various algorithms. The synthetic data were created as follows: 15 independent subspaces $\{\mathcal{S}_i\}_{i=1}^{15}$ were generated whose bases $\{\mathbf{U}_i\}_{i=1}^{15}$ were computed by $\mathbf{U}_{i+1} = \mathbf{T}\mathbf{U}_i$, $1 \leq i \leq 14$, where \mathbf{T} is a random rotation matrix. Thus each subspace has a dimension of 5 while the ambient dimension

表 12.2: Comparison of the speed of different algorithms under varying parameter settings, adapted from [87].

$\lambda = 0.1$			
Method	Minimum	Time	#Iter.
APG	111.481	129.6	312
ADM	37.572	77.2	187
LADMAP(A)	37.571	2.4	38
IRLS	37.571	26.5	105
$\lambda = 0.5$			
Method	Minimum	Time	#Iter.
APG	129.022	56.2	160
ADM	111.463	76.6	199
LADMAP(A)	111.463	123.6	391
IRLS	111.463	26.4	105
$\lambda = 1$			
Method	Minimum	Time	#Iter.
APG	147.171	44.0	109
ADM	124.586	105.7	257
LADMAP(A)	123.933	1081.4	1973
IRLS	123.933	24.9	105

is 200. Then 20 data points were sampled from each subspace by computing $\mathbf{U}_i \mathbf{Q}_i$, $1 \leq i \leq 15$, where \mathbf{Q}_i is a 5×20 standard Gaussian random matrix, while 20% samples were randomly chosen to be corrupted. Table 12.2 shows the statistics of each algorithm, such as the achieved minimum objective value at the last iteration, the computation time and the number of iterations. We can see that IRLS is always faster than APG and ADM. IRLS also outperforms LADMAP(A) when $\lambda = 0.5$ and 1. We also find that LADMAP(A) needs more iterations to converge when λ increases. This is because the rank of the solution grows with λ . When the rank is not low enough, partial SVD may not be faster than the full SVD [81]. Compared with LADMAP(A), IRLS is a better choice for small-sized or high-rank problems because it could avoid SVD.

12.1.5 Factorization Method

Another method for low-rank problems is to represent the expected low-rank matrix \mathbf{A} as $\mathbf{A} = \mathbf{XY}^T$, where \mathbf{X} and \mathbf{Y} both have r columns, so $\text{rank}(\mathbf{A}) \leq r$. For the Matrix Completion problem, the model is therefore formulated as [130]

$$\min_{\mathbf{X}, \mathbf{Y}, \mathbf{A}} \frac{1}{2} \|\mathbf{XY}^T - \mathbf{A}\|_F^2, \quad \text{s.t.} \quad \mathcal{P}_\Omega(\mathbf{A}) = \mathcal{P}_\Omega(\mathbf{D}). \quad (12.50)$$

While for the matrix recovery problem (cf. RPCA), Shen et al. [121] proposed the following non-convex optimization model:

~~$$\min_{\mathbf{X}, \mathbf{Y}, \mathbf{A}} \|\mathbf{A} - \mathbf{D}\|_1, \quad \text{s.t.} \quad \mathbf{A} = \mathbf{XY}^T. \quad (12.51)$$~~

The problem can be partially solved by the alternating minimization method. Take problem (12.50) as an example. Given the current iterates \mathbf{X}_k , \mathbf{Y}_k , and \mathbf{A}_k , the scheme updates three variables alternately by minimizing (12.50) w.r.t. each one while fixing the other two variables until they do not change [130]:

~~$$\begin{aligned} \mathbf{X}_{k+1} &\leftarrow \mathbf{A}_k(\mathbf{Y}_k^T)^\dagger = \underset{\mathbf{X}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{XY}_k^T - \mathbf{A}_k\|_F^2, \\ \mathbf{Y}_{k+1} &\leftarrow \mathbf{A}_k^T(\mathbf{X}_{k+1}^T)^\dagger = \underset{\mathbf{Y}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{X}_{k+1}\mathbf{Y}^T - \mathbf{A}_k\|_F^2, \\ \mathbf{A}_{k+1} &\leftarrow \mathbf{X}_{k+1}\mathbf{Y}_{k+1}^T + \mathcal{P}_\Omega(\mathbf{D} - \mathbf{X}_{k+1}\mathbf{Y}_{k+1}^T). \end{aligned} \quad (12.52)$$~~

The advantage of this kind of methods is its simplicity. However, we have to estimate the rank r of low-rank matrix a priori and the updates of \mathbf{X} and \mathbf{Y} may easily get stuck.

The alternating minimization on (12.51) can easily get stuck at any point, not necessarily critical points, due to the nonsmoothness of objective function. Xu et al. [134] proposed a Relaxed Majorization Minimization (MM) algorithm for the Robust Matrix Factorization (RMF) problem, which is more general than (12.51):

~~$$\min_{\mathbf{X} \in \mathcal{C}_x, \mathbf{Y} \in \mathcal{C}_y} \|\mathbf{W} \odot (\mathbf{XY}^T - \mathbf{D})\|_1 + R_x(\mathbf{X}) + R_y(\mathbf{Y}), \quad (12.53)$$~~

where $\mathbf{D} \in \mathbb{R}^{m \times n}$ is the observed matrix and $\mathbf{X} \in \mathbb{R}^{m \times r}$ and $\mathbf{Y} \in \mathbb{R}^{n \times r}$ are the unknown factor matrices. \mathbf{W} is the 0-1 binary mask with the same size as \mathbf{D} . The entry value 0 means that the corresponding entry in \mathbf{D} is missing, and 1 otherwise. The operator \odot is the Hadamard entry-wise product. $\mathcal{C}_x \subseteq \mathbb{R}^{m \times r}$ and $\mathcal{C}_y \subseteq \mathbb{R}^{n \times r}$ are some convex sets, e.g., nonnegative cones or balls in a certain norm. $R_x(\mathbf{X})$ and $R_y(\mathbf{Y})$ represent certain convex regularizations, e.g., ℓ_1 -norm, squared Frobenius norm, or elastic net. Variants of RMF, e.g., low-rank matrix recovery [69], Nonnegative Matrix Factorization (NMF) [75],

and dictionary learning [98, 144] can be obtained by combining different constraints and regularizations.

Suppose that $(\mathbf{X}_k, \mathbf{Y}_k)$ has been computed at the k -th iteration. Rather than updating (\mathbf{X}, \mathbf{Y}) directly, we aim at computing its increment over $(\mathbf{X}_k, \mathbf{Y}_k)$. To this end, we split (\mathbf{X}, \mathbf{Y}) as the sum of $(\mathbf{X}_k, \mathbf{Y}_k)$ and the unknown increment $(\Delta\mathbf{X}, \Delta\mathbf{Y})$:

$$(\mathbf{X}, \mathbf{Y}) = (\mathbf{X}_k, \mathbf{Y}_k) + (\Delta\mathbf{X}, \Delta\mathbf{Y}). \quad (12.54)$$

Then (12.53) can be rewritten as:

$$\begin{aligned} \min_{\Delta\mathbf{X} + \mathbf{X}_k \in \mathcal{C}_x, \Delta\mathbf{Y} + \mathbf{Y}_k \in \mathcal{C}_y} F_k(\Delta\mathbf{X}, \Delta\mathbf{Y}) &\triangleq \|\mathbf{W} \odot (\mathbf{D} - (\mathbf{X}_k + \Delta\mathbf{X})(\mathbf{Y}_k^T + \Delta\mathbf{Y})^T)\|_1 \\ &+ R_x(\mathbf{X}_k + \Delta\mathbf{X}) + R_y(\mathbf{Y}_k + \Delta\mathbf{Y}). \end{aligned} \quad (12.55)$$

Problem (12.55) is a problem on the increment $(\Delta\mathbf{X}, \Delta\mathbf{Y})$. However, it is not easier than the original problem (12.53). Inspired by MM, we try to approximate (12.55) with a convex surrogate. By the triangular inequality of norms, we can deduce:

$$\begin{aligned} F_k(\Delta\mathbf{X}, \Delta\mathbf{Y}) &\leq \|\mathbf{W} \odot (\mathbf{D} - \mathbf{X}_k \mathbf{Y}_k^T - \Delta\mathbf{X} \mathbf{Y}_k^T - \mathbf{X}_k \Delta\mathbf{Y}^T)\|_1 \\ &+ \|\mathbf{W} \odot (\Delta\mathbf{X} \Delta\mathbf{Y}^T)\|_1 + R_x(\mathbf{X}_k + \Delta\mathbf{X}) + R_y(\mathbf{Y}_k + \Delta\mathbf{Y}), \end{aligned} \quad (12.56)$$

where the term $\|\mathbf{W} \odot (\Delta\mathbf{X} \Delta\mathbf{Y}^T)\|_1$ can be further upper bounded by $\frac{\rho_x}{2} \|\Delta\mathbf{X}\|_F^2 + \frac{\rho_y}{2} \|\Delta\mathbf{Y}\|_F^2$, in which ρ_x and ρ_y are some positive constants. Denoting

$$\begin{aligned} \hat{G}_k(\Delta\mathbf{X}, \Delta\mathbf{Y}) &= \|\mathbf{W} \odot (\mathbf{D} - \mathbf{X}_k \mathbf{Y}_k^T - \Delta\mathbf{X} \mathbf{Y}_k^T - \mathbf{X}_k \Delta\mathbf{Y}^T)\|_1 \\ &+ R_x(\mathbf{X}_k + \Delta\mathbf{X}) + R_y(\mathbf{Y}_k + \Delta\mathbf{Y}), \end{aligned} \quad (12.57)$$

we have a strongly convex surrogate function of $F_k(\Delta\mathbf{X}, \Delta\mathbf{Y})$ as follows:

$$\begin{aligned} G_k(\Delta\mathbf{X}, \Delta\mathbf{Y}) &= \hat{G}_k(\Delta\mathbf{X}, \Delta\mathbf{Y}) + \frac{\rho_x}{2} \|\Delta\mathbf{X}\|_F^2 + \frac{\rho_y}{2} \|\Delta\mathbf{Y}\|_F^2, \\ \text{s.t. } \Delta\mathbf{X} + \mathbf{X}_k &\in \mathcal{C}_x, \Delta\mathbf{Y} + \mathbf{Y}_k \in \mathcal{C}_y. \end{aligned} \quad (12.58)$$

Minimizing $G_k(\Delta\mathbf{X}, \Delta\mathbf{Y})$ to find the optimal increment $(\Delta\mathbf{X}, \Delta\mathbf{Y})$ can be easily done by using LADMPSAP [82] (see Section 7.5, with $g_i = 0$, $i = 1, \dots, n$ in (7.184)), by introducing the auxiliary variable $\mathbf{E} = \mathbf{D} - \mathbf{X}_k \mathbf{Y}_k^T - \Delta\mathbf{X} \mathbf{Y}_k^T - \mathbf{X}_k \Delta\mathbf{Y}^T$.

Denote $\#\mathbf{W}_{(i,.)}$ and $\#\mathbf{W}_{(.j)}$ as the number of observed entries in the corresponding row and column of \mathbf{D} , respectively, and $\varepsilon > 0$ as any positive scalar. We have the following proposition.

Proposition 770 ([134]). *We have $\hat{G}_k(\Delta\mathbf{X}, \Delta\mathbf{Y}) + \frac{\bar{\rho}_x}{2} \|\Delta\mathbf{X}\|_F^2 + \frac{\bar{\rho}_y}{2} \|\Delta\mathbf{Y}\|_F^2 \geq F_k(\Delta\mathbf{X}, \Delta\mathbf{Y}) \geq \hat{G}_k(\Delta\mathbf{X}, \Delta\mathbf{Y}) - \frac{\bar{\rho}_x}{2} \|\Delta\mathbf{X}\|_F^2 - \frac{\bar{\rho}_y}{2} \|\Delta\mathbf{Y}\|_F^2$ holds for all possible $(\Delta\mathbf{X}, \Delta\mathbf{Y})$ and the equality holds if and only if $(\Delta\mathbf{X}, \Delta\mathbf{Y}) = (\mathbf{0}, \mathbf{0})$, where $\bar{\rho}_x = \max \{\#\mathbf{W}_{(i,.)}, i = 1, \dots, m\} + \varepsilon$ and $\bar{\rho}_y = \max \{\#\mathbf{W}_{(.j)}, j = 1, \dots, n\} + \varepsilon$.*

To help escape from the local minima near the initial points, a continuation-like trick is recommended. Namely ρ_x and ρ_y in (12.58) are initialized with relatively small values and then increase gradually, using the backtracking technique in [10] to ensure the locally majorant condition

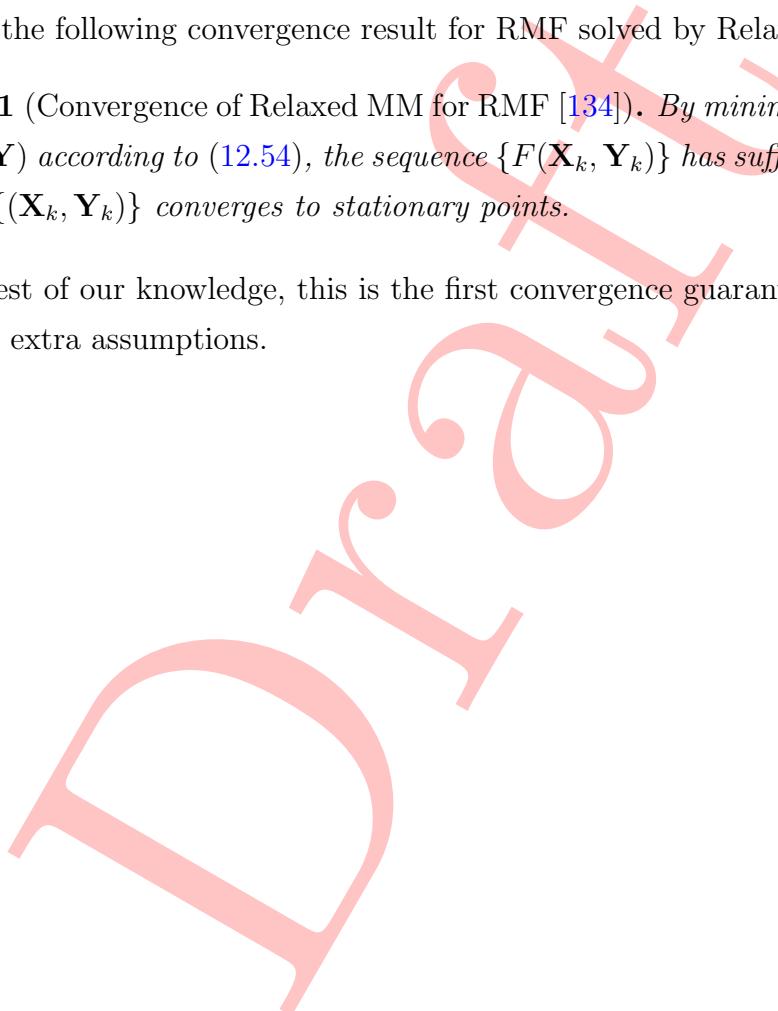
$$F_k(\Delta \mathbf{X}, \Delta \mathbf{Y}) \leq G_k(\Delta \mathbf{X}, \Delta \mathbf{Y}). \quad (12.59)$$

ρ_x and ρ_y eventually reach the upper bounds $\bar{\rho}_x$ and $\bar{\rho}_y$ in Proposition 770, respectively. Such an algorithm is called the locally majorant Relaxed MM for RMF.

We have the following convergence result for RMF solved by Relaxed MM.

Theorem 771 (Convergence of Relaxed MM for RMF [134]). *By minimizing (12.58) and updating (\mathbf{X}, \mathbf{Y}) according to (12.54), the sequence $\{F(\mathbf{X}_k, \mathbf{Y}_k)\}$ has sufficient descent and the sequence $\{(\mathbf{X}_k, \mathbf{Y}_k)\}$ converges to stationary points.*

To the best of our knowledge, this is the first convergence guarantee for variants of RMF without extra assumptions.



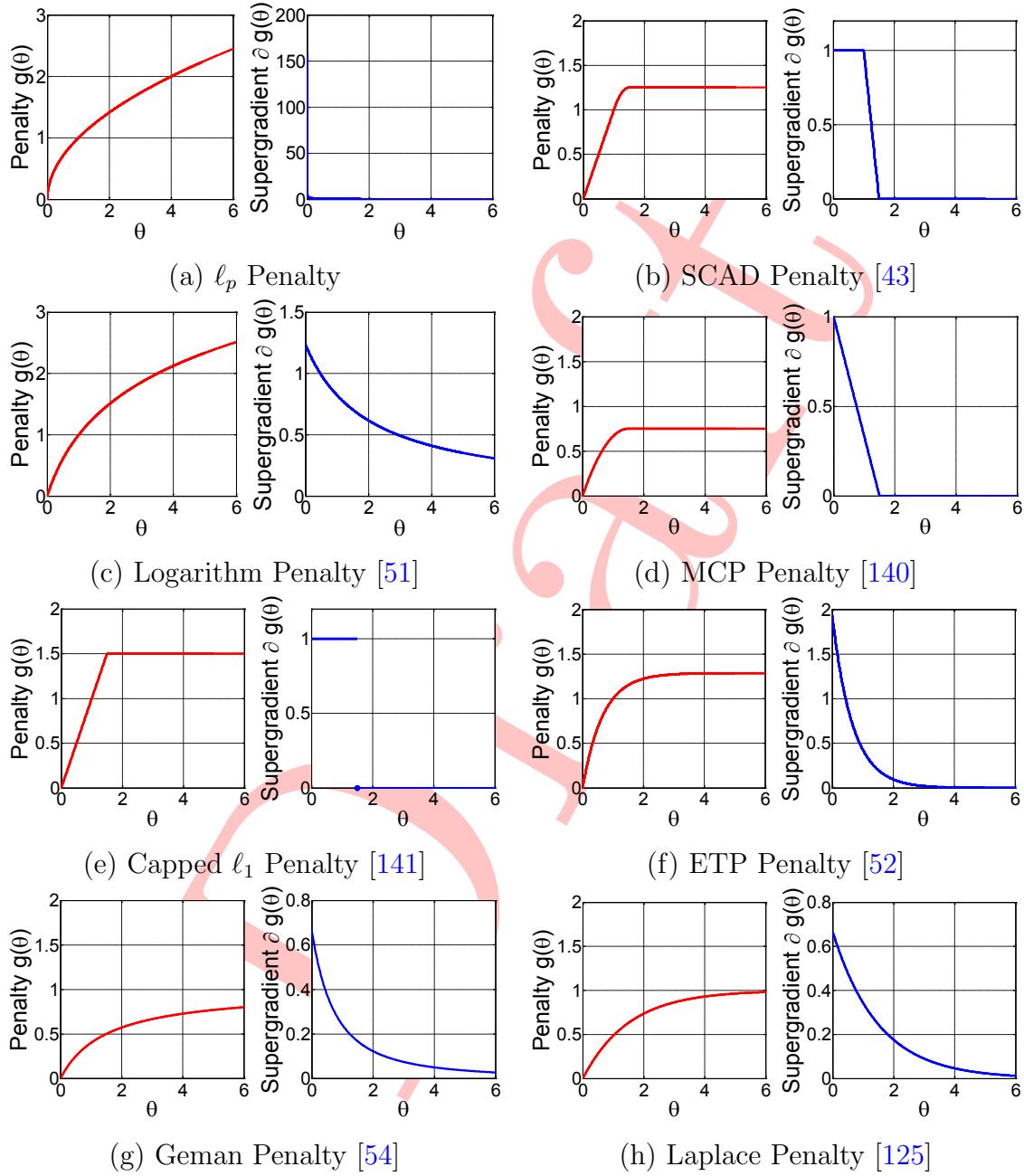


图 12.1: Illustration of the popular nonconvex surrogate functions $\|\theta\|_0$ (left) and their supergradients (right), adapted from [92]. For the ℓ_p penalty, $p = 0.5$. For all these penalties, $\lambda = 1$ and $\gamma = 1.5$.

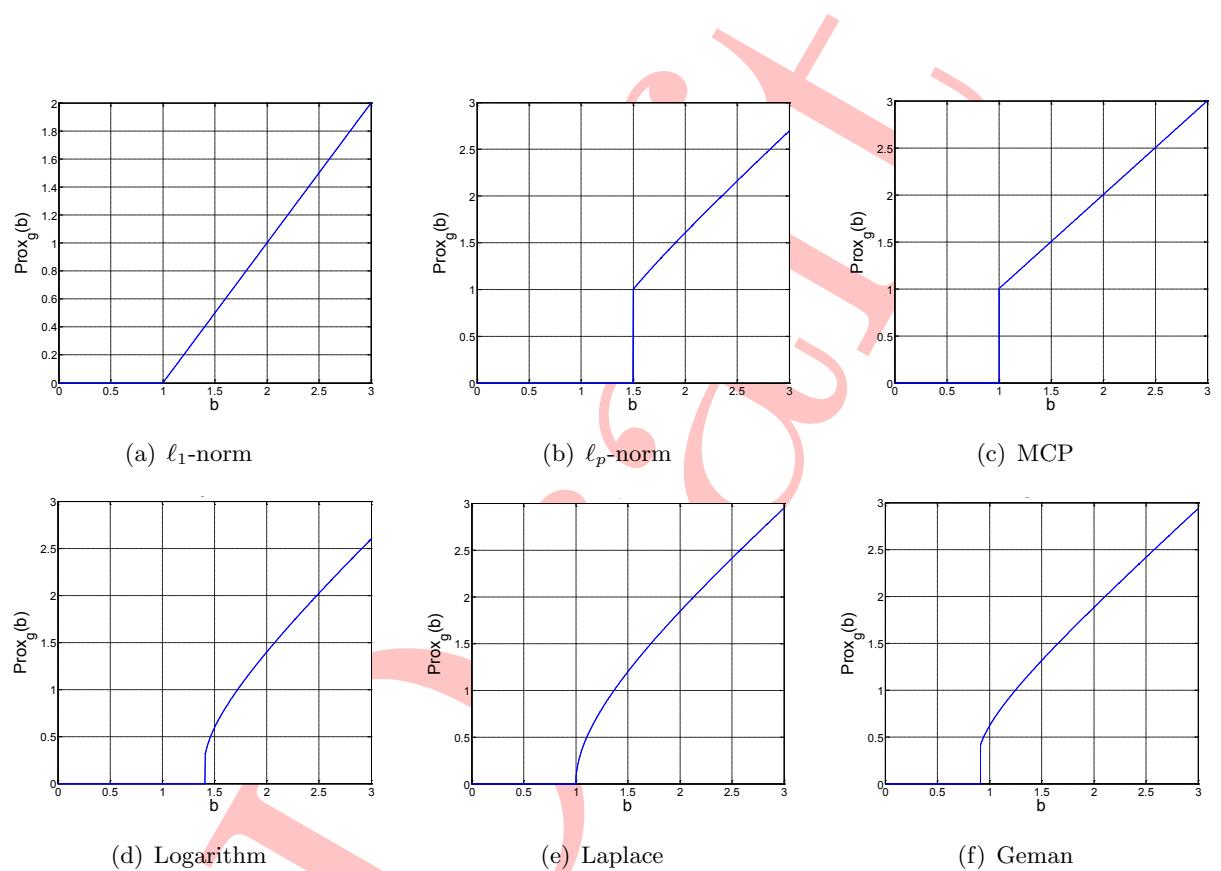


图 12.2: Plots of b v.s. $\text{Prox } g(b)$ for different choices of g : convex ℓ_1 -norm and popular nonconvex functions. The figure is adapted from [88].

Draft

第十三章 Kurdyka-Łojasiewicz Condition

(Taken from [135] and [77])

The Kurdyka-Łojasiewicz property was proposed by Łojasiewicz (1993), who proved that for a real analytic function f and $\phi(u) = u^{1-\gamma}$ with $\gamma \in [1/2, 1]$, $|f(\mathbf{x}) - f(\bar{\mathbf{x}})|^\gamma / \text{dist}(\mathbf{0}, \partial f(\mathbf{x}))$ is bounded around any critical point $\bar{\mathbf{x}}$. Kurdyka (1998) extended this property to a class of functions with the o -minimal structure. Bolte et al. (2007) extended to nonsmooth sub-analytic functions. Recently, the Kurdyka-Łojasiewicz property has been used to establish convergence analysis of proximal alternating minimization for nonconvex problems [5, 16, 135].

Definition 772. A function $\psi(\mathbf{x})$ satisfies the KL property at point $\bar{\mathbf{x}} \in \text{dom}(\partial\psi)$ if there exists $\theta \in [0, 1)$, such that

$$\frac{|\psi(\mathbf{x}) - \psi(\bar{\mathbf{x}})|^\theta}{\text{dist}(\mathbf{0}, \partial\psi(\mathbf{x}))} \quad (13.1)$$

is bounded around $\bar{\mathbf{x}}$ under the notational conventions: $0^0 = 1$, $\infty/\infty = 0/0 = 0$. In other words, in a certain neighborhood \mathcal{U} of $\bar{\mathbf{x}}$, there exists

$$\phi(s) = cs^{1-\theta}, \text{ for some } c > 0 \text{ and } \theta \in (0, 1), \quad (13.2)$$

such that the KL inequality holds

$$\phi'(|\psi(\mathbf{x}) - \psi(\bar{\mathbf{x}})|) \text{dist}(\mathbf{0}, \partial\psi(\mathbf{x})) \geq 1, \quad \forall \mathbf{x} \in \mathcal{U} \cap \text{dom}(\partial\psi) \text{ and } \psi(\mathbf{x}) \neq \psi(\bar{\mathbf{x}}), \quad (13.3)$$

where $\text{dom}(\partial\psi) \equiv \{\mathbf{x} | \partial\psi(\mathbf{x}) \neq \emptyset\}$ and $\text{dist}(\mathbf{0}, \partial\varphi(\mathbf{x})) \equiv \min\{\|\mathbf{y}\| : \mathbf{y} \in \partial\varphi(\mathbf{x})\}$. ϕ is called the desingularising function.

A more general definition of the KL property is not to specify the choice of ϕ as (13.2). Rather, it allows any $\phi : [0, \eta) \rightarrow \mathbb{R}^+$ satisfying

1. ϕ is concave and \mathcal{C}^1 on $(0, \eta)$;
2. ϕ is continuous at 0, $\phi(0) = 0$; and
3. $\phi'(\mathbf{x}) > 0$, $\forall \mathbf{x} \in (0, \eta)$.

When $\varphi(\mathbf{x})$ is a semi-algebraic function, the desingularising function $\phi(t)$ can be chosen to be the form of $\frac{C}{\theta}t^\theta$ with $\theta \in (0, 1]$ [16].

The KL inequality holds for semi-algebraic functions, real analytic functions, functions on the o -minimal structure, and sub-analytic functions [5, 16]. So KL property is general enough. Examples include:

Real analytic functions. A smooth function $\varphi(t)$ on \mathbb{R} is analytic if $\left(\frac{\varphi^k(t)}{k!}\right)^{1/k}$ is bounded for all k and on any compact set $\mathcal{D} \subset \mathbb{R}$. One can verify whether a real function $\psi(x)$ on \mathbb{R}^n is analytic by checking the analyticity of $\varphi(t) = \psi(\mathbf{x} + t\mathbf{y})$ for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. For example, any polynomial function is real analytic, such as $\|\mathbf{Ax} - \mathbf{b}\|^2$ and the first terms in the objectives of (1.9) and (1.10) of [135]. In addition, it is not difficult to verify that the nonconvex function $L_q(\mathbf{x}, \varepsilon, \lambda) = \sum_{i=1}^n (x_i^2 + \varepsilon^2)^{q/2} + \frac{1}{2\lambda} \|\mathbf{Ax} - \mathbf{b}\|^2$ with $0 < q < 1$ considered in [71] for sparse vector recovery is a real analytic function (the first term is the ε -smoothed ℓ_q -seminorm). The logistic loss function $\psi(t) = \log(1 + e^{-t})$ is also analytic. Therefore, all the above functions satisfy the KL inequality with $\theta \in [1/2, 1)$ in (13.1).

Locally strongly convex functions. A function $\psi(\mathbf{x})$ is strongly convex in a neighborhood \mathcal{D} with constant μ if

$$\psi(\mathbf{y}) \geq \psi(\mathbf{x}) + \langle g(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{x} - \mathbf{y}\|^2, \quad \forall g(\mathbf{x}) \in \partial\psi(\mathbf{x}), \mathbf{x}, \mathbf{y} \in \mathcal{D}.$$

According to the definition and using the Cauchy-Schwarz inequality, we have

$$\psi(\mathbf{y}) - \psi(\mathbf{x}) \geq \langle \gamma(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{x} - \mathbf{y}\|^2 \geq -\frac{1}{\mu} \|\gamma(\mathbf{x})\|^2, \quad \forall \gamma(\mathbf{x}) \in \partial\psi(\mathbf{x}),$$

where the last inequality is obtained by minimizing the middle term over \mathbf{y} . Hence, $\mu(\psi(\mathbf{x}) - \psi(\mathbf{y})) \leq (\text{dist}(\mathbf{0}, \partial\psi(\mathbf{x})))^2$, and ψ satisfies the KL inequality (13.3) at any point $\mathbf{y} \in \mathcal{D}$ with $\phi(s) = \frac{2}{\mu} \sqrt{s}$ and $\mathcal{U} = \mathcal{D} \cap \{\mathbf{x} : \psi(\mathbf{x}) \geq \psi(\mathbf{y})\}$. For example, the logistic loss function $\psi(t) = \log(1 + e^{-t})$ is strongly convex in any bounded set \mathcal{D} .

Semi-algebraic functions. A set $\mathcal{D} \subset \mathbb{R}^n$ is called semi-algebraic [15], if it can be represented as

$$\mathcal{D} = \bigcup_{i=1}^s \bigcap_{j=1}^t \{\mathbf{x} \in \mathbb{R}^n | p_{ij}(\mathbf{x}) = 0, q_{ij}(\mathbf{x}) > 0\},$$

where p_{ij} and q_{ij} are real polynomial functions. A function ψ is called semi-algebraic if its graph $\text{Graph}(\psi) \equiv \{(\mathbf{x}, \psi(\mathbf{x})) | \mathbf{x} \in \text{dom}(\psi)\}$ is a semi-algebraic set. Semi-algebraic functions are sub-analytic. So they satisfy the KL inequality.

Theorem 773 (Tarski-Seidenberg Principle). [5, 6, 16] *The image of a semi-algebraic set $\mathcal{A} \subset \mathbb{R}^{d+1}$ by the projection $\pi : \mathbb{R}^{d+1} \rightarrow \mathbb{R}^d$ is semi-algebraic.*

Corollary 774. $\sup\{g(\mathbf{u}, \mathbf{v}) : \mathbf{v} \in \mathcal{C}\}$, where g is semi-algebraic and \mathcal{C} is a semi-algebraic set is semi-algebraic.

We list some known elementary properties of semi-algebraic sets and functions below that can help identify semi-algebraic functions.

1. If a set \mathcal{D} is semi-algebraic, so is its closure $\text{cl}(\mathcal{D})$,
2. If \mathcal{D}_1 and \mathcal{D}_2 are both semi-algebraic, so are $\mathcal{D}_1 \cup \mathcal{D}_2$, $\mathcal{D}_1 \cap \mathcal{D}_2$, and $\mathbb{R}^n \setminus \mathcal{D}_1$.
3. Indicator functions of semi-algebraic sets are semi-algebraic.
4. Finite sums and products of semi-algebraic functions are semi-algebraic.
5. The composition of semi-algebraic functions is semi-algebraic.

From items 1 and 2, any polyhedral set is semi-algebraic, such as the nonnegative orthant $\mathbb{R}_+^n = \{\mathbf{x} \in \mathbb{R}^n : x_i \geq 0, \forall i\}$. Hence, the indicator function $\delta_{\mathbb{R}_+^n}$ is a semi-algebraic function. The absolute value function $\psi(t) = |t|$ is also semi-algebraic since its graph is $\text{cl}(\mathcal{D})$, where

$$\mathcal{D} = \{(t, s) : t + s = 0, -t > 0\} \cup \{(t, s) : t - s = 0, t > 0\}.$$

Hence, the ℓ_1 -norm $\|\mathbf{x}\|_1$ is semi-algebraic since it is the finite sum of absolute functions. In addition, the sup-norm $\|\mathbf{x}\|_\infty$ is semi-algebraic, which can be shown by observing

$$\text{Graph}(\|\mathbf{x}\|_\infty) = \{(\mathbf{x}, t) : t = \max_j |x_j|\} = \bigcup_i \{(\mathbf{x}, t) : |x_i| = t, |x_j| \leq t, \forall j \neq i\}.$$

Further, the Euclidean norm $\|\mathbf{x}\|$ is shown to be semi-algebraic in [15]. According to item 5, $\|\mathbf{Ax} - \mathbf{b}\|_1$, $\|\mathbf{Ax} - \mathbf{b}\|_\infty$, and $\|\mathbf{Ax} - \mathbf{b}\|$ are all semi-algebraic functions.

Sum of real analytic and semi-algebraic functions. Both real analytic and semi-algebraic functions are sub-analytic. If ψ_1 and ψ_2 are both sub-analytic and ψ_1 maps bounded sets to bounded sets, then $\psi_1 + \psi_2$ is also sub-analytic. Since real analytic functions map bounded set to bounded set, the sum of a real analytic function and a semi-algebraic function is sub-analytic, so the sum satisfies the KL property. For example, the sparse logistic regression function

$$\psi(\mathbf{x}, b) = \frac{1}{n} \sum_{i=1}^n \log(1 + \exp(-c_i(\mathbf{a}_i^T \mathbf{x} + b))) + \lambda \|\mathbf{x}\|_1$$

is sub-analytic and satisfies the KL inequality.

Actually, most convex functions encountered in finite dimensional applications satisfy the KL property [16]. The convex counterexample provided in [17] exhibits a wildly oscillatory collection of level sets, a phenomenon which seems highly unlikely to happen

with functions modeling real world problems. Moreover, an interesting and rather specific feature of convex functions is that their desingularizing function ϕ can be explicitly computed from rather common and simple properties [16].

Lemma 775 (Uniformized KL property). [16] Let Ω be a compact set and let $f : \mathbb{R}^n \rightarrow (-\infty, +\infty]$ be a proper and lower semicontinuous function. Assume that f is constant on Ω and satisfies the KL property at each point of Ω . Then there exists $\epsilon > 0$, $\eta > 0$ and $\varphi \in \Phi_\eta$, such that for all $\bar{\mathbf{u}}$ in Ω and all \mathbf{u} in the following intersection

$$\{\mathbf{u} \in \mathbb{R}^n : \text{dist}(\mathbf{u}, \Omega) < \epsilon\} \cap \{\mathbf{u} \in \mathbb{R}^n : f(\bar{\mathbf{u}}) < f(\mathbf{u}) < f(\bar{\mathbf{u}}) + \eta\}, \quad (13.4)$$

the following inequality holds

$$\varphi'(f(\mathbf{u}) - f(\bar{\mathbf{u}})) \text{dist}(0, \partial f(\mathbf{u})) > 1, \quad (13.5)$$

An important application of the KL property is that whenever we have:

$$\sum_{k=1}^{+\infty} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2 < +\infty, \quad (13.6)$$

we can immediately have

$$\sum_{k=1}^{+\infty} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| < +\infty. \quad (13.7)$$

Note that (13.6) can easily be obtained, e.g., by the sufficient descent property of an algorithm, but it cannot imply that the sequence $\{\mathbf{x}_k\}$ is convergent (and converge to a KKT point of the optimization problem). Rather, most likely we can only prove that any accumulation point of $\{\mathbf{x}_k\}$ is a KKT point of the optimization problem, which is a relatively weak convergence result. In contrast, (13.7) implies that the sequence $\{\mathbf{x}_k\}$ is a Cauchy sequence, thus is convergent and further we can often prove that the sequence converges to a KKT point of the optimization problem, which is much stronger than what (13.6) can imply. Furthermore, the convergence rate can also be analyzed if θ in (13.2) can be estimated. Below is an example of the usage of the KL property [77].

13.1 Monotone APG for Nonconvex Programs

We consider a general problem:

$$\min_{\mathbf{x} \in \mathbb{R}^n} F(\mathbf{x}) = f(\mathbf{x}) + g(\mathbf{x}), \quad (13.8)$$

We mainly consider nonconvex f and nonconvex nonsmooth g . In problem (13.8), we assume that f is a proper function with Lipschitz continuous gradients and g is proper and lower semicontinuous.

Algorithm 12 monotone APG with fixed stepsize

Initialize $\mathbf{z}_1 = \mathbf{x}_1 = \mathbf{x}_0$, $t_1 = 1$, $t_0 = 0$, $\alpha_y < \frac{1}{L}$, $\alpha_x < \frac{1}{L}$.

for $k = 1, 2, 3, \dots$ **do**

$$\mathbf{y}_k = \mathbf{x}_k + \frac{t_{k-1}}{t_k}(\mathbf{z}_k - \mathbf{x}_k) + \frac{t_{k-1} - 1}{t_k}(\mathbf{x}_k - \mathbf{x}_{k-1}), \quad (13.9)$$

$$\mathbf{z}_{k+1} = \text{prox}_{\alpha_y g}(\mathbf{y}_k - \alpha_y \nabla f(\mathbf{y}_k)), \quad (13.10)$$

$$\mathbf{v}_{k+1} = \text{prox}_{\alpha_x g}(\mathbf{x}_k - \alpha_x \nabla f(\mathbf{x}_k)), \quad (13.11)$$

$$t_{k+1} = \frac{\sqrt{4(t_k)^2 + 1} + 1}{2}, \quad (13.12)$$

$$\mathbf{x}_{k+1} = \begin{cases} \mathbf{z}_{k+1}, & \text{if } F(\mathbf{z}_{k+1}) \leq F(\mathbf{v}_{k+1}), \\ \mathbf{v}_{k+1}, & \text{otherwise.} \end{cases} \quad (13.13)$$

end for

13.2 Monotone APG

We summarize the monotone APG in Algorithm 12.

Theorem 776. Let f be a proper function with Lipschitz continuous gradients and g be proper and lower semicontinuous. For nonconvex f and nonconvex nonsmooth g , assume that (4.53) holds. Then $\{\mathbf{x}_k\}$ and $\{\mathbf{v}_k\}$ generated by Algorithm 12 are bounded. Let \mathbf{x}^* be any accumulation point of $\{\mathbf{x}_k\}$, we have $0 \in \partial F(\mathbf{x}^*)$.

Proof. (13.11) in Algorithm 12 can be seen as

$$\mathbf{v}_{k+1} = \underset{\mathbf{x}}{\operatorname{argmin}} \langle \nabla f(\mathbf{x}_k), \mathbf{x} - \mathbf{x}_k \rangle + \frac{1}{2\alpha_x} \|\mathbf{x} - \mathbf{x}_k\|^2 + g(\mathbf{x}). \quad (13.14)$$

So we have

$$\langle \nabla f(\mathbf{x}_k), \mathbf{v}_{k+1} - \mathbf{x}_k \rangle + \frac{1}{2\alpha_x} \|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2 + g(\mathbf{v}_{k+1}) \leq g(\mathbf{x}_k). \quad (13.15)$$

From the Lipschitz continuity of ∇f we have

$$F(\mathbf{v}_{k+1}) \leq g(\mathbf{v}_{k+1}) + f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{v}_{k+1} - \mathbf{x}_k \rangle + \frac{L}{2} \|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2 \quad (13.16)$$

$$\leq g(\mathbf{x}_k) - \langle \nabla f(\mathbf{x}_k), \mathbf{v}_{k+1} - \mathbf{x}_k \rangle - \frac{1}{2\alpha_x} \|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2 \quad (13.17)$$

$$+ f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{v}_{k+1} - \mathbf{x}_k \rangle + \frac{L}{2} \|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2 \quad (13.18)$$

$$= F(\mathbf{x}_k) - \left(\frac{1}{2\alpha_x} - \frac{L}{2} \right) \|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2. \quad (13.19)$$

If $F(\mathbf{z}_{k+1}) \leq F(\mathbf{v}_{k+1})$, then

$$\mathbf{x}_{k+1} = \mathbf{z}_{k+1}, F(\mathbf{x}_{k+1}) = F(\mathbf{z}_{k+1}) \leq F(\mathbf{v}_{k+1}). \quad (13.20)$$

If $F(\mathbf{z}_{k+1}) > F(\mathbf{v}_{k+1})$, then

$$\mathbf{x}_{k+1} = \mathbf{v}_{k+1}, F(\mathbf{x}_{k+1}) = F(\mathbf{v}_{k+1}). \quad (13.21)$$

From (13.19), (13.20) and (13.21) we have

$$F(\mathbf{x}_{k+1}) \leq F(\mathbf{v}_{k+1}) \leq F(\mathbf{x}_k). \quad (13.22)$$

So

$$F(\mathbf{x}_{k+1}) \leq F(\mathbf{x}_1), F(\mathbf{v}_{k+1}) \leq F(\mathbf{x}_1) \quad (13.23)$$

for all k . From the assumption we know that $\{\mathbf{x}_k\}$ and $\{\mathbf{v}_k\}$ are bounded. Thus $\{\mathbf{x}_k\}$ has accumulation points. As $F(\mathbf{x}_k)$ is nonincreasing, F has the same value at all the accumulation points. Let it be F^* . From (13.19) we have

$$\left(\frac{1}{2\alpha_x} - \frac{L}{2}\right) \|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2 \leq F(\mathbf{x}_k) - F(\mathbf{v}_{k+1}) \leq F(\mathbf{x}_k) - F(\mathbf{x}_{k+1}). \quad (13.24)$$

Summing over $k = 1, 2, \dots, \infty$, we have

$$\left(\frac{1}{2\alpha_x} - \frac{L}{2}\right) \sum_{k=1}^{\infty} \|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2 \leq F(\mathbf{x}_1) - F^* < \infty, \quad (13.25)$$

From $\alpha_x < \frac{1}{L}$ we have

$$\|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2 \rightarrow 0 \quad \text{as } k \rightarrow \infty. \quad (13.26)$$

From the optimality condition of (13.14) we have

$$0 \in \nabla f(\mathbf{x}_k) + \frac{1}{\alpha_x}(\mathbf{v}_{k+1} - \mathbf{x}_k) + \partial g(\mathbf{v}_{k+1}) \quad (13.27)$$

$$= \nabla f(\mathbf{v}_{k+1}) + \nabla f(\mathbf{x}_k) - \nabla f(\mathbf{v}_{k+1}) + \frac{1}{\alpha_x}(\mathbf{v}_{k+1} - \mathbf{x}_k) + \partial g(\mathbf{v}_{k+1}). \quad (13.28)$$

So we have

$$-\nabla f(\mathbf{x}_k) + \nabla f(\mathbf{v}_{k+1}) - \frac{1}{\alpha_x}(\mathbf{v}_{k+1} - \mathbf{x}_k) \in \partial F(\mathbf{v}_{k+1}), \quad (13.29)$$

and

$$\left\| \nabla f(\mathbf{x}_k) - \nabla f(\mathbf{v}_{k+1}) + \frac{1}{\alpha_x}(\mathbf{v}_{k+1} - \mathbf{x}_k) \right\| \leq \left(\frac{1}{\alpha_x} + L \right) \|\mathbf{v}_{k+1} - \mathbf{x}_k\| \rightarrow 0, \quad (13.30)$$

as $k \rightarrow \infty$.

Let \mathbf{x}^* be any accumulation point of $\{\mathbf{x}_k\}$, say $\{\mathbf{x}_{k_j}\} \rightarrow \mathbf{x}^*$ as $j \rightarrow \infty$. From (13.26) we have $\{\mathbf{v}_{k_j+1}\} \rightarrow \mathbf{x}^*$ as $j \rightarrow \infty$. From (13.14) we have

$$\langle \nabla f(\mathbf{x}_{k_j}), \mathbf{v}_{k_j+1} - \mathbf{x}_{k_j} \rangle + \frac{1}{2\alpha_x} \|\mathbf{v}_{k_j+1} - \mathbf{x}_{k_j}\|^2 + g(\mathbf{v}_{k_j+1}) \quad (13.31)$$

$$\leq \langle \nabla f(\mathbf{x}_{k_j}), \mathbf{x}^* - \mathbf{x}_{k_j} \rangle + \frac{1}{2\alpha_x} \|\mathbf{x}^* - \mathbf{x}_{k_j}\|^2 + g(\mathbf{x}^*). \quad (13.32)$$

So

$$\limsup_{j \rightarrow \infty} g(\mathbf{v}_{k_j+1}) \leq g(\mathbf{x}^*). \quad (13.33)$$

From the definition of lower semicontinuous of g we have

$$\liminf_{j \rightarrow \infty} g(\mathbf{v}_{k_j+1}) \geq g(\mathbf{x}^*). \quad (13.34)$$

So we have

$$\lim_{j \rightarrow \infty} g(\mathbf{v}_{k_j+1}) = g(\mathbf{x}^*). \quad (13.35)$$

Because f is continuously differentiable, we have

$$\lim_{j \rightarrow \infty} F(\mathbf{v}_{k_j+1}) = F(\mathbf{x}^*). \quad (13.36)$$

From $\{\mathbf{v}_{k_j+1}\} \rightarrow \mathbf{x}^*$, (13.36), (13.29), (13.30) and Proposition 265.2 we have

$$0 \in \partial F(\mathbf{x}^*). \quad (13.37)$$

□

Theorem 777. Assume that f and g are convex and ∇f is Lipschitz continuous. Then $\{\mathbf{x}_k\}$ generated by algorithm 12 satisfies

$$F(\mathbf{x}_{N+1}) - F(\mathbf{x}^*) \leq \frac{2}{\alpha_y(N+1)^2} \|\mathbf{x}_0 - \mathbf{x}^*\|^2. \quad (13.38)$$

where \mathbf{x}^* is a global minimizer of $F(\mathbf{x})$.

This theorem is quite similar to Theorem 5.1 in [10]. The proof is almost the same with [10]. We list the proof here only for the convenience of reader's reference.

Proof. (13.10) in Algorithm 12 can be seen as

$$\mathbf{z}_{k+1} = \operatorname{argmin}_{\mathbf{x}} \langle \nabla f(\mathbf{y}_k), \mathbf{x} - \mathbf{y}_k \rangle + \frac{1}{2\alpha_y} \|\mathbf{x} - \mathbf{y}_k\|^2 + g(\mathbf{x}). \quad (13.39)$$

From the optimality condition, we have

$$0 \in \nabla f(\mathbf{y}_k) + \frac{1}{\alpha_y} (\mathbf{z}_{k+1} - \mathbf{y}_k) + \partial g(\mathbf{z}_{k+1}). \quad (13.40)$$

From the convexity of g we have

$$g(\mathbf{x}) - g(\mathbf{z}_{k+1}) \geq \left\langle -\nabla f(\mathbf{y}_k) - \frac{1}{\alpha_y} (\mathbf{z}_{k+1} - \mathbf{y}_k), \mathbf{x} - \mathbf{z}_{k+1} \right\rangle, \forall \mathbf{x}. \quad (13.41)$$

From the Lipschitz continuity of ∇f and convexity of f we have

$$F(\mathbf{z}_{k+1}) \leq g(\mathbf{z}_{k+1}) + f(\mathbf{y}_k) + \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{y}_k \rangle + \frac{L}{2} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2 \quad (13.42)$$

$$= g(\mathbf{z}_{k+1}) + f(\mathbf{y}_k) + \langle \nabla f(\mathbf{y}_k), \mathbf{x} - \mathbf{y}_k \rangle + \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{x} \rangle \quad (13.43)$$

$$+ \frac{L}{2} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2 \quad (13.44)$$

$$\leq g(\mathbf{z}_{k+1}) + f(\mathbf{x}) + \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{x} \rangle + \frac{L}{2} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2 \quad (13.45)$$

$$\leq g(\mathbf{x}) + \left\langle \nabla f(\mathbf{y}_k) + \frac{1}{\alpha_y} (\mathbf{z}_{k+1} - \mathbf{y}_k), \mathbf{x} - \mathbf{z}_{k+1} \right\rangle \quad (13.46)$$

$$+ f(\mathbf{x}) + \langle \nabla f(\mathbf{y}_k), \mathbf{z}_{k+1} - \mathbf{x} \rangle + \frac{L}{2} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2 \quad (13.47)$$

$$= F(\mathbf{x}) + \frac{1}{\alpha_y} \langle \mathbf{z}_{k+1} - \mathbf{y}_k, \mathbf{x} - \mathbf{z}_{k+1} \rangle + \frac{L}{2} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2 \quad (13.48)$$

$$= F(\mathbf{x}) + \frac{1}{\alpha_y} \langle \mathbf{z}_{k+1} - \mathbf{y}_k, \mathbf{x} - \mathbf{y}_k + \mathbf{y}_k - \mathbf{z}_{k+1} \rangle + \frac{L}{2} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2 \quad (13.49)$$

$$= F(\mathbf{x}) + \frac{1}{\alpha_y} \langle \mathbf{z}_{k+1} - \mathbf{y}_k, \mathbf{x} - \mathbf{y}_k \rangle - \left(\frac{1}{\alpha_y} - \frac{L}{2} \right) \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2 \quad (13.50)$$

$$\leq F(\mathbf{x}) + \frac{1}{\alpha_y} \langle \mathbf{z}_{k+1} - \mathbf{y}_k, \mathbf{x} - \mathbf{y}_k \rangle - \frac{1}{2\alpha_y} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2. \quad (13.51)$$

Let $\mathbf{x} = \mathbf{x}_k$ and \mathbf{x}^* , we have

$$F(\mathbf{z}_{k+1}) - F(\mathbf{x}_k) \leq \frac{1}{\alpha_y} \langle \mathbf{z}_{k+1} - \mathbf{y}_k, \mathbf{x}_k - \mathbf{y}_k \rangle - \frac{1}{2\alpha_y} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2, \quad (13.52)$$

$$F(\mathbf{z}_{k+1}) - F(\mathbf{x}^*) \leq \frac{1}{\alpha_y} \langle \mathbf{z}_{k+1} - \mathbf{y}_k, \mathbf{x}^* - \mathbf{y}_k \rangle - \frac{1}{2\alpha_y} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2. \quad (13.53)$$

Multiplying (13.52) by $t_k - 1$ and adding (13.53) we have

$$t_k F(\mathbf{z}_{k+1}) - (t_k - 1) F(\mathbf{x}_k) - F(\mathbf{x}^*) \quad (13.54)$$

$$\leq \frac{1}{\alpha_y} \langle \mathbf{z}_{k+1} - \mathbf{y}_k, (t_k - 1)(\mathbf{x}_k - \mathbf{y}_k) + \mathbf{x}^* - \mathbf{y}_k \rangle - \frac{t_k}{2\alpha_y} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2. \quad (13.55)$$

So we have

$$t_k (F(\mathbf{z}_{k+1}) - F(\mathbf{x}^*)) - (t_k - 1) (F(\mathbf{x}_k) - F(\mathbf{x}^*)) \quad (13.56)$$

$$\leq \frac{1}{\alpha_y} \langle \mathbf{z}_{k+1} - \mathbf{y}_k, (t_k - 1)(\mathbf{x}_k - \mathbf{y}_k) + \mathbf{x}^* - \mathbf{y}_k \rangle - \frac{t_k}{2\alpha_y} \|\mathbf{z}_{k+1} - \mathbf{y}_k\|^2. \quad (13.57)$$

Multiplying both sides by t_k and using $(t_k)^2 - t_k = (t_{k-1})^2$ from (13.12) we have

$$(t_k)^2 (F(\mathbf{z}_{k+1}) - F(\mathbf{x}^*)) - (t_{k-1})^2 (F(\mathbf{x}_k) - F(\mathbf{x}^*)) \quad (13.58)$$

$$\leq \frac{1}{\alpha_y} \langle t_k(\mathbf{z}_{k+1} - \mathbf{y}_k), (t_k - 1)(\mathbf{x}_k - \mathbf{y}_k) + \mathbf{x}^* - \mathbf{y}_k \rangle - \frac{1}{2\alpha_y} \|t_k(\mathbf{z}_{k+1} - \mathbf{y}_k)\|^2 \quad (13.59)$$

$$= \frac{1}{\alpha_y} \langle t_k(\mathbf{z}_{k+1} - \mathbf{y}_k), (t_k - 1)\mathbf{x}_k - t_k\mathbf{y}_k + \mathbf{x}^* \rangle - \frac{1}{2\alpha_y} \|t_k(\mathbf{z}_{k+1} - \mathbf{y}_k)\|^2 \quad (13.60)$$

$$= \frac{1}{2\alpha_y} (\|(t_k - 1)\mathbf{x}_k - t_k\mathbf{y}_k + \mathbf{x}^*\|^2 - \|(t_k - 1)\mathbf{x}_k - t_k\mathbf{z}_{k+1} + \mathbf{x}^*\|^2). \quad (13.61)$$

Define

$$\mathbf{u}_{k+1} = t_k\mathbf{z}_{k+1} - (t_k - 1)\mathbf{x}_k - \mathbf{x}^*. \quad (13.62)$$

Let

$$\mathbf{u}_k = t_{k-1}\mathbf{z}_k - (t_{k-1} - 1)\mathbf{x}_{k-1} - \mathbf{x}^* = t_k\mathbf{y}_k - (t_k - 1)\mathbf{x}_k - \mathbf{x}^*. \quad (13.63)$$

We have

$$\mathbf{y}_k = \frac{t_{k-1}\mathbf{z}_k - (t_{k-1} - 1)\mathbf{x}_{k-1} + (t_k - 1)\mathbf{x}_k}{t_k} \quad (13.64)$$

$$= \mathbf{x}_k + \frac{t_{k-1}}{t_k}(\mathbf{z}_k - \mathbf{x}_k) + \frac{t_{k-1} - 1}{t_k}(\mathbf{x}_k - \mathbf{x}_{k-1}), \quad (13.65)$$

which is the same with (13.9) in Algorithm 12. So we have

$$(t_k)^2 (F(\mathbf{z}_{k+1}) - F(\mathbf{x}^*)) - (t_{k-1})^2 (F(\mathbf{x}_k) - F(\mathbf{x}^*)) \quad (13.66)$$

$$\leq \frac{1}{2\alpha_y} (\|\mathbf{u}_k\|^2 - \|\mathbf{u}_{k+1}\|^2). \quad (13.67)$$

If $F(\mathbf{z}_{k+1}) \leq F(\mathbf{v}_{k+1})$, then $\mathbf{x}_{k+1} = \mathbf{z}_{k+1}$. So

$$(t_k)^2 (F(\mathbf{x}_{k+1}) - F(\mathbf{x}^*)) - (t_{k-1})^2 (F(\mathbf{x}_k) - F(\mathbf{x}^*)) \quad (13.68)$$

$$= (t_k)^2 (F(\mathbf{z}_{k+1}) - F(\mathbf{x}^*)) - (t_{k-1})^2 (F(\mathbf{x}_k) - F(\mathbf{x}^*)) \quad (13.69)$$

$$\leq \frac{1}{2\alpha_y} (\|\mathbf{u}_k\|^2 - \|\mathbf{u}_{k+1}\|^2). \quad (13.70)$$

If $F(\mathbf{z}_{k+1}) > F(\mathbf{v}_{k+1})$, then $\mathbf{x}_{k+1} = \mathbf{v}_{k+1}$. So

$$(t_k)^2 (F(\mathbf{x}_{k+1}) - F(\mathbf{x}^*)) - (t_{k-1})^2 (F(\mathbf{x}_k) - F(\mathbf{x}^*)) \quad (13.71)$$

$$\leq (t_k)^2 (F(\mathbf{z}_{k+1}) - F(\mathbf{x}^*)) - (t_{k-1})^2 (F(\mathbf{x}_k) - F(\mathbf{x}^*)) \quad (13.72)$$

$$\leq \frac{1}{2\alpha_y} (\|\mathbf{u}_k\|^2 - \|\mathbf{u}_{k+1}\|^2). \quad (13.73)$$

Summing over $k = 1, \dots, N$, we have

$$(t_N)^2 (F(\mathbf{x}_{N+1}) - F(\mathbf{x}^*)) \quad (13.74)$$

$$= (t_N)^2 (F(\mathbf{x}_{N+1}) - F(\mathbf{x}^*)) - (t^0)^2 (F(\mathbf{x}_1) - F(\mathbf{x}^*)) \quad (13.75)$$

$$\leq \frac{1}{2\alpha_y} (\|\mathbf{u}_1\|^2 - \|\mathbf{u}_{N+1}\|^2) \quad (13.76)$$

$$\leq \frac{1}{2\alpha_y} \|\mathbf{u}_1\|^2 \quad (13.77)$$

$$= \frac{1}{2\alpha_y} \|\mathbf{x}_0 - \mathbf{x}^*\|^2. \quad (13.78)$$

From (13.12) we can easily have that $t_k \geq \frac{k+1}{2}$. So we have

$$F(\mathbf{x}_{N+1}) - F(\mathbf{x}^*) \leq \frac{2}{\alpha_y(N+1)^2} \|\mathbf{x}_0 - \mathbf{x}^*\|^2. \quad (13.79)$$

□

Theorem 778. Let f be a proper function with Lipschitz continuous gradients and g be proper and lower semicontinuous. For nonconvex f and nonconvex nonsmooth g , assume that (4.53) holds. If we further assume that f and g satisfy the KL property, and the desingularising function has the form of $\varphi(t) = \frac{C}{\theta} t^\theta$ for some $C > 0, \theta \in (0, 1]$, then

1. If $\theta = 1$, then there exists k_1 such that $F(\mathbf{x}_k) = F^*$ for all $k > k_1$ and the algorithm terminates in finite steps.

2. If $\theta \in [\frac{1}{2}, 1)$, then there exists k_2 such that for all $k > k_2$,

$$F(\mathbf{x}_k) - F^* \leq \left(\frac{d_1 C^2}{1 + d_1 C^2} \right)^{k-k_2} r_{k_2}. \quad (13.80)$$

3. If $\theta \in (0, \frac{1}{2})$, then there exists k_3 such that for all $k > k_3$,

$$F(\mathbf{x}_k) - F^* \leq \left(\frac{C}{(k - k_3)d_2(1 - 2\theta)} \right)^{\frac{1}{1-2\theta}}, \quad (13.81)$$

where F^* is the same function value at all the accumulation points of $\{\mathbf{x}_k\}$, $r_k = F(\mathbf{v}_k) - F^*$, $d_1 = \left(\frac{1}{\alpha_x} + L \right)^2 / \left(\frac{1}{2\alpha_x} - \frac{L}{2} \right)$, $d_2 = \min \left\{ \frac{1}{2d_1 C}, \frac{C}{1-2\theta} \left(2^{\frac{2\theta-1}{2\theta-2}} - 1 \right) r_0^{2\theta-1} \right\}$

This theorem is similar to Theorem 4 in [50] and the proof is almost the same as [50]. We will discuss the difference later.

Proof. From (13.19) and (13.22) we have

$$F(\mathbf{v}_{k+1}) \leq F(\mathbf{x}_k) - \left(\frac{1}{2\alpha_x} - \frac{L}{2} \right) \|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2 \quad (13.82)$$

$$\leq F(\mathbf{v}_k) - \left(\frac{1}{2\alpha_x} - \frac{L}{2} \right) \|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2. \quad (13.83)$$

From (13.30) we have

$$\text{dist}(0, \partial F(\mathbf{v}_{k+1})) \leq \left(\frac{1}{\alpha_x} + L \right) \|\mathbf{v}_{k+1} - \mathbf{x}_k\|. \quad (13.84)$$

From (13.26) we know that $\{\mathbf{x}_k\}$ and $\{\mathbf{v}_k\}$ have the same accumulation points. Let Ω be the set that contains all the accumulation points of $\{\mathbf{x}_k\}$ (also $\{\mathbf{v}_k\}$). Because $F(\mathbf{v}_k)$ is nonincreasing, F has the same value at all the accumulation points in Ω . Let it be F^* . So we have

$$F(\mathbf{v}_k) \geq F^*, F(\mathbf{v}_k) \rightarrow F^*. \quad (13.85)$$

If there exists \bar{k} such that $F(\mathbf{v}^{\bar{k}}) = F^*$, then $F(\mathbf{v}^{\bar{k}}) = F(\mathbf{v}^{\bar{k}+1}) = \dots = F^*$. So $\|\mathbf{v}^{\bar{k}+1} - \mathbf{x}^{\bar{k}}\| = \|\mathbf{v}^{\bar{k}+2} - \mathbf{x}^{\bar{k}+1}\| = \dots = 0$. The conclusion holds. If $F(\mathbf{v}_k) > F^*$ for all k , then from $F(\mathbf{v}_k) \rightarrow F^*$ we know that there exists \hat{k}_1 such that $F(\mathbf{v}_k) < F^* + \eta$ whenever $k > \hat{k}_1$. On the other hand, because $\text{dist}(\mathbf{v}_k, \Omega) \rightarrow 0$, there exists \hat{k}_2 such that $\text{dist}(\mathbf{v}_k, \Omega) < \varepsilon$ whenever $k > \hat{k}_2$. Let $k > k_0 = \max\{\hat{k}_1, \hat{k}_2\}$, we have

$$\mathbf{v}_k \in \{\mathbf{v}, \text{dist}(\mathbf{v}, \Omega) \leq \varepsilon\} \cap [F^* < F(\mathbf{v}) < F^* + \eta]. \quad (13.86)$$

From the uniform KL property in Lemma 775, there exists a concave function φ such that

$$\varphi'(F(\mathbf{v}_k) - F^*) \text{dist}(0, \partial F(\mathbf{v}_k)) \geq 1. \quad (13.87)$$

Define $r_k = F(\mathbf{v}_k) - F^*$. We suppose that $r_k > 0$ for all k . Otherwise $F(\mathbf{v}_k) = F(\mathbf{v}_{k+1}) = \dots = F^*$ and the algorithm terminates in finite steps. By supposing this (13.87) holds.

From (13.84), (13.87) and (13.83) we have

$$1 \leq [\varphi'(F(\mathbf{v}_k) - F^*) \text{dist}(0, \partial F(\mathbf{v}_k))]^2 \quad (13.88)$$

$$\leq [\varphi'(r_k)]^2 \left(\frac{1}{\alpha_x} + L \right)^2 \|\mathbf{v}_k - \mathbf{x}_{k-1}\|^2 \quad (13.89)$$

$$\leq [\varphi'(r_k)]^2 \left(\frac{1}{\alpha_x} + L \right)^2 \frac{F(\mathbf{v}_{k-1}) - F(\mathbf{v}_k)}{\left(\frac{1}{2\alpha_x} - \frac{L}{2} \right)} \quad (13.90)$$

$$= d_1 [\varphi'(r_k)]^2 (r_{k-1} - r_k), \quad (13.91)$$

for all $k > k_0$, where $d_1 = \left(\frac{1}{\alpha_x} + L\right)^2 / \left(\frac{1}{2\alpha_x} - \frac{L}{2}\right)$. Because φ has the form of $\varphi(t) = \frac{C}{\theta}t^\theta$, we have $\varphi'(t) = Ct^{\theta-1}$. So (13.91) becomes

$$1 \leq d_1 C^2 r_k^{2\theta-2} (r_{k-1} - r_k). \quad (13.92)$$

1. Case $\theta = 1$.

In this case, (13.92) becomes

$$1 \leq d_1 C^2 (r_k - r_{k+1}). \quad (13.93)$$

Because $r_k \rightarrow 0$ and $d_1 > 0$, $C > 0$, this is a contradiction. So there exists k_1 such that $r_k = 0$ for all $k > k_1$. The algorithm terminates in finite steps.

2. Case $\theta \in [\frac{1}{2}, 1)$.

In this case, $0 < 2 - 2\theta \leq 1$. As $r_k \rightarrow 0$, there exists \hat{k}_3 such that $r_k^{2-2\theta} \geq r_k$ for all $k > \hat{k}_3$. (13.92) becomes

$$r_k \leq d_1 C^2 (r_{k-1} - r_k). \quad (13.94)$$

So we have

$$r_k \leq \frac{d_1 C^2}{1 + d_1 C^2} r_{k-1}, \quad (13.95)$$

for all $k_2 > \max\{k_0, \hat{k}_3\}$ and

$$r_k \leq \left(\frac{d_1 C^2}{1 + d_1 C^2} \right)^{k-k_2} r_{k_2}. \quad (13.96)$$

So we have

$$F(\mathbf{x}_k) - F^* \leq F(\mathbf{v}_k) - F^* = r_k \leq \left(\frac{d_1 C^2}{1 + d_1 C^2} \right)^{k-k_2} r_{k_2}. \quad (13.97)$$

3. Case $\theta \in (0, \frac{1}{2})$.

In this case, $2\theta - 2 \in (-2, -1)$, $2\theta - 1 \in (-1, 0)$. As $r_{k-1} > r_k$, we have $r_{k-1}^{2\theta-2} < r_k^{2\theta-2}$ and $r_0^{2\theta-1} < \dots < r_{k-1}^{2\theta-1} < r_k^{2\theta-1}$

Define $\phi(t) = \frac{C}{1-2\theta} t^{2\theta-1}$, then $\phi'(t) = -Ct^{2\theta-2}$.

If $r_k^{2\theta-2} \leq 2r_{k-1}^{2\theta-2}$, then

$$\phi(r_k) - \phi(r_{k-1}) = \int_{r_{k-1}}^{r_k} \phi'(t) dt = C \int_{r_k}^{r_{k-1}} t^{2\theta-2} dt \quad (13.98)$$

$$\geq C(r_{k-1} - r_k) r_{k-1}^{2\theta-2} \geq \frac{C}{2} (r_{k-1} - r_k) r_k^{2\theta-2} \quad (13.99)$$

$$\geq \frac{1}{2d_1 C}. \quad (13.100)$$

for all $k > k_0$.

If $r_k^{2\theta-2} \geq 2r_{k-1}^{2\theta-2}$, then $r_k^{2\theta-1} \geq 2^{\frac{2\theta-1}{2\theta-2}} r_{k-1}^{2\theta-1}$.

$$\phi(r_k) - \phi(r_{k-1}) = \frac{C}{1-2\theta}(r_k^{2\theta-1} - r_{k-1}^{2\theta-1}) \quad (13.101)$$

$$\geq \frac{C}{1-2\theta}(2^{\frac{2\theta-1}{2\theta-2}} - 1)r_{k-1}^{2\theta-1} \quad (13.102)$$

$$= qr_{k-1}^{2\theta-1} \geq qr_0^{2\theta-1}. \quad (13.103)$$

where $q = \frac{C}{1-2\theta}(2^{\frac{2\theta-1}{2\theta-2}} - 1)$. Let $d_2 = \min\{\frac{1}{2d_1C}, qr_0^{2\theta-1}\}$, we have

$$\phi(r_k) - \phi(r_{k-1}) \geq d_2, \quad (13.104)$$

for all $k > k_0$ and

$$\phi(r_k) \geq \phi(r_k) - \phi(r_{k_0}) \geq \sum_{i=k_0+1}^k \phi(r_i) - \phi(r_{i-1}) \geq (k - k_0)d_2. \quad (13.105)$$

So we have

$$r_k^{2\theta-1} \geq \frac{(k - k_0)d_2(1 - 2\theta)}{C}, \quad (13.106)$$

and

$$r_k \leq \left(\frac{C}{(k - k_0)d_2(1 - 2\theta)} \right)^{\frac{1}{1-2\theta}}. \quad (13.107)$$

Let $k_3 = k_0$ we have

$$F(\mathbf{x}_k) - F^* \leq F(\mathbf{v}_k) - F^* = r_k \leq \left(\frac{C}{(k - k_3)d_2(1 - 2\theta)} \right)^{\frac{1}{1-2\theta}}, \quad (13.108)$$

which completes the proof. \square

Difference from the conditions in [50]:

[50] considered general descent method with the conditions:

$$F(\mathbf{x}_{k+1}) \leq F(\mathbf{x}_k) - \alpha \|\mathbf{x}_{k+1} - \mathbf{x}_k\|, \quad (13.109)$$

and

$$\|\partial F(\mathbf{x}_{k+1})\| \leq \beta \|\mathbf{x}_{k+1} - \mathbf{x}_k\| \quad (13.110)$$

Proximal gradient method is a typical example satisfying these conditions. However, to make the proximal gradient method both accelerate and converge, we introduce the

intermediate variables \mathbf{y}_k , \mathbf{v}_k and \mathbf{z}_k . This makes our algorithm more complex and the conditions satisfied by our algorithm becomes

$$F(\mathbf{x}_{k+1}) \leq F(\mathbf{x}_k) - \alpha \|\mathbf{v}_{k+1} - \mathbf{x}_k\|, F(\mathbf{v}_{k+1}) \leq F(\mathbf{v}_k) - \alpha \|\mathbf{v}_{k+1} - \mathbf{x}_k\| \quad (13.111)$$

and

$$\|\partial F(\mathbf{v}_{k+1})\| \leq \beta \|\mathbf{v}_{k+1} - \mathbf{x}_k\| \quad (13.112)$$

The intermediate variable \mathbf{v} makes the main difference. As a result, under the conditions of (13.109) and (13.110), a useful conclusion of finite length of $\{\mathbf{x}\}$: $\sum_{i=k}^{\infty} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| < \infty$ can be achieved and $\{\mathbf{x}_k\}$ is a converged sequence. Accordingly, the convergence rate for $\|\mathbf{x}_k - \mathbf{x}^*\|$ can be obtained. By contrast, our algorithm can only get $\sum_{i=1}^{\infty} \|\mathbf{v}_{k+1} - \mathbf{x}_k\| < \infty$. Neither the convergence rate for $\|\mathbf{x}_k - \mathbf{x}^*\|$ nor $\{\mathbf{x}_k\}$ is a converged sequence can be obtained.

13.2.1 Exercises

Exercise 779. Prove that real polynomial functions, logistic loss function $\log(1 + e^{-t})$, $\|\mathbf{x}\|_p$ ($p \geq 0$), $\|\mathbf{x}\|_\infty$, and indicator functions of the positive semidefinite (PSD) cone, the Stiefel manifolds and the set of constant rank matrices all satisfy the KL property.

Exercise 780. Prove that the nuclear norm is a semi-algebraic function.

Exercise 781. Prove that the function $f(x) = \text{dist}(x, \mathcal{S})^2$ is semi-algebraic whenever \mathcal{S} is a nonempty semi-algebraic subset of \mathbb{R}^d .

Exercise 782. Prove that if an algorithm results in a sequence with sufficient descent:

$$f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k) \leq -\alpha \|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2, \quad \text{where } \alpha > 0, \quad (13.113)$$

then the sequence has a finite sum of squared distance (13.6).

第十四章 Optimization with Orthogonality Constraints

14.1 Examples of Applying Orthogonality Constraint

14.1.1 Discriminant Analysis and Manifold Learning

14.1.1.1 Discriminant Analysis

14.1.1.2 Manifold Learning

14.1.1.3 Solution by Eigenvalue Decomposition

14.1.2 Finding a Selection Matrix

If we want to find a selection matrix \mathbf{S} , we may impose the following constraints for approximation:

$$\mathbf{S} \geq 0, \quad \mathbf{S}\mathbf{S}^T = \mathbf{I}.$$

Draft

第十五章 Problem Reformulation Techniques

See also Section 6.6.6.

See also July 9, 2018, “机器学习中的优化方法”, WeChat and email for the equivalence between nuclear norm minimization and SDP.

Convexification. 1. by duality. 2. by adding convex functions. 3. by transformation, e.g.,

$$\min_{\xi} \sum_{k=1}^5 c_k(-\log_2(\gamma k \xi_k)), \quad s.t. \quad \sum_{k=1}^5 \xi_k^2 = 1,$$

can be convexified by

$$\min_{\xi} \sum_{k=1}^5 c_k(-\log_2(\gamma k \sqrt{\xi_k})), \quad s.t. \quad \sum_{k=1}^5 \xi_k = 1.$$

See paper 2920 of NIPS 2018.

See paper 6517 of NIPS 2018 for SDP and parabolic relaxation.

See, e.g., the book “Semidefinite optimization and convex algebraic geometry”) studying systematic approaches to convex relaxations of polynomial problems.

Draft

参 考 文 献

- [1] 邹应. 拓扑学习题集. 武汉大学出版社, 2001.
- [2] Alekh Agarwal and Léon Bottou. A lower bound for the optimization of finite sums. In *International Conference on Machine Learning*, pages 78–86, 2015.
- [3] Naman Agarwal, Zeyuan Allen-Zhu, Brian Bullins, Elad Hazan, and Tengyu Ma. Finding approximate local minima faster than gradient descent. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1195–1199. ACM, 2017.
- [4] G.B. Allende and G. Still. Solving bilevel programs with the KKT-approach. *Mathematical Programming*, 138:309–332, 2013.
- [5] Rajesh Kumar Arora. *Optimization: Algorithms and Applications*. CRC Press, 2015.
- [6] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Łojasiewicz inequality. *Mathematics of Operations Research*, 35:438–457, 2010.
- [7] H. Attouch, J. Bolte, and B. F. Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods. *Mathematical Programming*, 137:91–129, 2013.
- [8] B. Bank, J. Guddat, D. Klatte, B. Kummer, and K. Tammer. *Non-linear Parametric Optimization*. Birkhäuser Verlag, Basel, 1983.
- [9] Mokhtar S. Bazaraa, Hanif D. Sherali, and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. John Wiley & Sons, Inc., 3 edition, 2006.
- [10] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, pages 183–202, 2009.
- [11] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

- [12] D. P. Bertsekas. Necessary and sufficient conditions for a penalty method to be exact. *Mathematical Programming*, 9(1):87–99, 1975.
- [13] Dimitri P Bertsekas. *Nonlinear Programming*. Athena Scientific, 2 edition, 1999.
- [14] Dimitri P. Bertsekas. *Convex Optimization Theory*. Athena Scientific, 2009.
- [15] Dimitri P. Bertsekas, Angelia Nedic, and Asuman E. Ozdaglar. *Convex Analysis and Optimization*. Athena Scientific, 2003.
- [16] Dimitri P. Bertsekas and John N. Tsitsiklis, editors. *Parallel and Distributed Computation: Numerical Methods*. Prentice-Hall, 1989.
- [17] J. Bochnak, M. Coste, and M.-F. Roy. *Real Algebraic Geometry*. Springer-Verlag, Berlin, 1998.
- [18] J. Bolte, S. Sabach, and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494, 2014.
- [19] Jérôme Bolte, Aris Daniilidis, Olivier Ley, and Laurent Mazet. Characterizations of Łojasiewicz inequalities: subgradient flows, talweg, convexity. *Transactions of the American Mathematical Society*, 362:3319–3363, 2010.
- [20] Jérôme Bolte, Trong Phong Nguyen, Juan Peypouquet, and Bruce W. Suter. From error bounds to the complexity of first-order descent methods for convex functions. *Mathematical Programming*, 165(2):471–507, 2017.
- [21] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [22] C. G. Broyden. Quasi-Newton methods. *Optimization Methods in Electronics and Communications (K. W. Cattermole and J. J. O'Reilly, eds.)*, vol. 1 of *Mathematical Topics in Telecommunications*, (New York), Wiley, pages 105–110, 1984.
- [23] Sébastien Bubeck. *Convex Optimization: Algorithms and Complexity*. Now Publishers Inc., 2015.
- [24] Charles L. Byrne. *A First Course in Optimization*. CRC Press, 2015.
- [25] J. Cai, E.J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

- [26] E. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.
- [27] E.J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM*, 58(3):1–37, 2011.
- [28] E.J. Candès, Michael B Wakin, and Stephen P Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier Analysis and Applications*, 14(5-6):877–905, 2008.
- [29] Caihua Chen, Bingsheng He, Yinyu Ye, and Xiaoming Yuan. The direct extension of ADMM for multi-block convex minimization problems is not necessarily convergent. *Mathematical Programming*, 155(1-2):57–79, 2016.
- [30] Kun Chen, Hongbo Dong, and Kung-Sik Chan. Reduced rank regression via adaptive nuclear norm penalization. *Biometrika*, 100(4):901–920, 2013.
- [31] Tianyi Chen, Georgios B. Giannakis, Tao Sun, and Wotao Yin. LAG: Lazily aggregated gradient for communication-efficient distributed learning. *arXiv preprint, arXiv:1805.09965*, 2018.
- [32] E. K. P. Chong and B. E. Brewington. Distributed communications resource management for tracking and surveillance networks. In *Proceedings of the Conference on Signal and Data Processing of Small Targets 2005 (SPIE Vol. 5913), part of the SPIE Symposium on Optics & Photonics, San Diego, California*, pages 280–291, 2005.
- [33] E. K. P. Chong and B. E. Brewington. Decentralized rate control for tracking and surveillance networks. *Ad Hoc Networks, special issue on Recent Advances in Wireless Sensor Networks*, 5(6):910–928, 2007.
- [34] Edwin K. P. Chong and Stanislaw H. Żak. *An Introduction to Optimization*. John Wiley & Sons, Inc., 4 edition, 2013.
- [35] Frank H Clarke. Nonsmooth analysis and optimization. In *International Congress of Mathematicians*, pages 847–853, 1983.
- [36] C. A. Coello Coello, D. A. Van Veldhuizen, and G. B. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York: Kluwer Academic/Plenum Publishers, 2002.

- [37] K. Deb. *Multi-objective Optimization Using Evolutionary Algorithms*. Chichester, England: Wiley, 2001.
- [38] A. Defazio, F. Bach, and S. Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, 2014.
- [39] S. Dempe. *Foundations of Bilevel Programming*. Kluwer Academic Publishers, Dordrecht, 2002.
- [40] S. Dempe and J. Dutta. Is bilevel programming a special case of a mathematical program with complementarity constraints? *Mathematical Programming*, 131:37–48, 2012.
- [41] S. Dempe and A.B. Zemkoho. The generalized mangasarian-fromowitz constraint qualification and optimality conditions for bilevel programs. *Journal of Optimization Theory and Applications*, 148(1):46–68, 2011.
- [42] Stephan Dempe, Vyacheslav Kalashnikov, Gerardo A. Pérez-Valdés, and Nataliya Kalashnykova. *Bilevel Programming Problems: Theory, Algorithms and Applications to Energy Networks*. Springer, 2015.
- [43] V. F. Dem'yanov and L. V. Vasil'ev. *Nondifferentiable Optimization*. New York: Optimization Software, Inc., Publications Division, 1985.
- [44] A. Dhara and J. Dutta. *Optimality Conditions in Convex Optimization, A Finite-dimensional View*. CRC Press, Boca Raton, 2012.
- [45] P. Domingos. A few useful things to know about machine learning. *Communications of the ACM*, 55(10):78–87, 2012.
- [46] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(July):2121–2159, 2011.
- [47] John Duchi and Yoram Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10(December):2899–2934, 2009.
- [48] Essan Elhamifar and René Vidal. Sparse subspace clustering. *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

- [49] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.
- [50] Cong Fang, Feng Cheng, and Zhouchen Lin. Faster and non-ergodic $O(1/k)$ stochastic alternating direction method of multipliers. In *Advances in Neural Information Processing Systems*, 2017.
- [51] Cong Fang and Zhouchen Lin. Parallel asynchronous stochastic variance reduction for nonconvex optimization. In *AAAI Conference on Artificial Intelligence*, 2017.
- [52] D. Fanghäel. *Zwei-Ebenen-Optimierung mit diskreter unterer Ebene und stetiger oberer Ebene (Ph.D. Thesis)*. TU Bergakademie Freiberg, 2006.
- [53] M. Fazel. *Matrix Rank Minimization with Applications*. Ph.D. Thesis, Stanford University, 2002.
- [54] R. Fletcher. *Practical Methods of Optimization*. Chichester: Wiley, 2 edition, 1987.
- [55] C. A. Floudas and P. M. Pardalos (Eds.). *Encyclopedia of Optimization*. Springer, 2 edition, 2009.
- [56] Wilhelm Forst and Dieter Hoffmann. *Optimization: Theory and Practice*. Springer, 2010.
- [57] S. Foucart and H. Rauhut. *A Mathematical Introduction to Compressive Sensing*. Springer, 2013.
- [58] Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1-2):95–110, 1956.
- [59] P. Frankel, G. Garrigos, and J. Peypouquet. Splitting methods with variable metric for Kurdyka-Łojasiewicz functions and general convergence rates. *Journal of Optimization Theory and Applications*, 165:874–900, 2014.
- [60] J. Friedman. Fast sparse regression and classification. *International Journal of Forecasting*, 28(3):722–738, 2012.
- [61] Cuixia Gao, Naiyan Wang, Qi Yu, and Zhihua Zhang. A feasible nonconvex relaxation approach to feature selection. In *AAAI Conference on Artificial Intelligence*, 2011.

- [62] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points – online stochastic gradient for tensor decomposition. In *Conference on Learning Theory*, 2015.
- [63] I. M. Gel’fand. *Lectures on Linear Algebra*. New York: Interscience Publishers, 1961.
- [64] D. Geman and C. Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE Transactions on Image Processing*, 4(7):932–946, 1995.
- [65] G. Golub and C. Van Loan. *Matrix computations*. Johns Hopkins University Press, 1996.
- [66] Marco Gori. *Machine Learning: A Constraint-Based Approach*. Morgan Kaufmann Publishers, 2018.
- [67] Igor Griva, Stephen G. Nash, and Ariela Sofer. *Linear and Nonlinear Optimization*. Society for Industrial and Applied Mathematics, 2 edition, 2009.
- [68] S. Gu, L. Zhang, W. Zuo, and X. Feng. Weighted nuclear norm minimization with application to image denoising. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2862–2869, 2014.
- [69] Robert Hannah and Wotao Yin. On unbounded delays in asynchronous parallel fixed-point algorithms. *Journal of Scientific Computing*, 76(1):299–326, 2018.
- [70] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2001.
- [71] Bingsheng He, Min Tao, and Xiaoming Yuan. Alternating direction method with Gaussian back substitution for separable convex programming. *SIAM Journal on Optimization*, 22(2):313–340, 2012.
- [72] N. J. Higham. *Functions of Matrices: Theory and Computation*. Society for Industrial and Applied Mathematics, 2008.
- [73] M. Hong, Z.Q. Luo, and M. Razaviyayn. Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems. *SIAM Journal on Optimization*, 26(1):337–364, 2016.
- [74] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, 2013.

- [75] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He. Fast and accurate matrix completion via truncated nuclear norm regularization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(9):2117–2130, 2013.
- [76] Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning*, pages 427–435, 2013.
- [77] Martin Jaggi, Marek Sulovsk, et al. A simple algorithm for nuclear norm regularized problems. In *International Conference on Machine Learning*, pages 471–478, 2010.
- [78] D. Jakovetic, J. Xavier, and J. Moura. Fast distributed gradient methods. *IEEE Transactions on Automatic Control*, 59:1131–1146, 2014.
- [79] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pages 1724–1732, 2017.
- [80] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, 2013.
- [81] H.Th. Jongen, J.-J. Rückmann, and V. Shikhman. MPCC: critical point theory. *SIAM Journal of Optimization*, 20:473–484, 2009.
- [82] H.Th. Jongen and V. Shikhman. Bilevel optimization. *Mathematical Programming*, 136:65–90, 2012.
- [83] H.Th. Jongen, V. Shikhman, and S. Steffensen. Characterization of strong stability for c-stationary points in MPCC. *Mathematical Programming*, 132:295–308, 2012.
- [84] Kenji Kawaguchi. Deep learning without poor local minima. In *Advances in Neural Information Processing Systems*, 2016.
- [85] Q. Ke and T. Kanade. Robust ℓ_1 -norm factorization in the presence of outliers and missing data by alternative convex programming. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 739–746, 2005.
- [86] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.
- [87] L. Kolev. Iterative algorithm for the minimum fuel and minimum amplitude problems for linear discrete systems. *International Journal of Control*, 21(5):779–784, 1975.

- [88] M.-J. Lai and J. Wang. An unconstrained ℓ_q minimization with $0 < q \leq 1$ for sparse solution of underdetermined linear systems. *SIAM Journal on Optimization*, 21:82–101, 2011.
- [89] M.J. Lai and W. Yin. Augmented ℓ_1 and nuclear-norm models with a globally linearly convergent algorithm. *SIAM Journal on Imaging Sciences*, pp. 183-202, 6(2):1059–1091, 2013.
- [90] Guanghui Lan, Soomin Lee, and Yi Zhou. Communication-efficient algorithms for decentralized and stochastic optimization. *arXiv preprint, arXiv:1701.03961*, 2017.
- [91] S. Lang. *Calculus of Several Variables*. New York: Springer-Verlag, 3 edition, 1987.
- [92] Kenneth Lange. *Optimization*. Springer, 2 edition, 2013.
- [93] R. M. Larsen. <http://soi.stanford.edu/~rmunk/propack/>, 2004.
- [94] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, pages 556–562, 2001.
- [95] Huan Li and Zhouchen Lin. Accelerated proximal gradient methods for nonconvex programming. In *Advances in Neural Information Processing Systems*, pages 612–620, 2015.
- [96] Huan Li and Zhouchen Lin. Accelerated alternating direction method of multipliers: an optimal $O(1/K)$ nonergodic analysis. *arXiv preprint, arXiv:1608.06366*, 2017.
- [97] T. Lin, S. Ma, and S. Zhang. Global convergence of unmodified 3-block ADMM for a class of convex minimization problems. *Preprint*, 2015.
- [98] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penalty for low-rank representation. In *Advances in Neural Information Processing Systems*, pages 612–620, 2011.
- [99] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penalty for low-rank representation. In *Advances in Neural Information Processing Systems*, 2011.
- [100] Zhouchen Lin, Minming Chen, and Yi Ma. The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint, arXiv:1009.5055*, 2010.

- [101] Zhouchen Lin, Risheng Liu, and Huan Li. Linearized alternating direction method with parallel splitting and adaptive penalty for separable convex programs in machine learning. *Machine Learning*, 99(2):287–325, 2015.
- [102] Zhouchen Lin and Hongyang Zhang. *Low-Rank Models in Visual Analysis: Theories, Algorithms, and Applications*. Academic Press, 2017.
- [103] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In *International Conference on Machine Learning*, pages 663–670, 2010.
- [104] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 35(1):171–184, 2013.
- [105] Guangcan Liu, Zhouchen Lin, and Yong Yu. Robust subspace segmentation by low-rank representation. *International Conference on Machine Learning*, 2010.
- [106] C. Lu, Z. Lin, and S. Yan. Smoothed low rank and sparse matrix recovery by iteratively reweighted least squared minimization. *IEEE Transactions on Image Processing*, 24(2):646–654, 2015.
- [107] C. Lu, C. Zhu, C. Xu, S. Yan, and Z. Lin. Generalized singular value thresholding. In *AAAI Conference on Artificial Intelligence*, pages 1805–1811, 2015.
- [108] Canyi Lu, Jiashi Feng, Shuicheng Yan, , and Zhouchen Lin. A unified alternating direction method of multipliers by majorization minimization. *submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.
- [109] Canyi Lu, Huan Li, Zhouchen Lin, and Shuicheng Yan. Fast proximal linearized alternating direction method of multiplier with parallel splitting. In *AAAI Conference on Artificial Intelligence*, pages 739–745, 2016.
- [110] Canyi Lu, Jinhui Tang, Shuicheng Yan, and Zhouchen Lin. Generalized nonconvex nonsmooth low-rank minimization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4130–4137, 2014.
- [111] Canyi Lu, Jinhui Tang, Shuicheng Yan, and Zhouchen Lin. Nonconvex nonsmooth low-rank minimization via iteratively reweighted nuclear norm. *IEEE Transactions on Image Processing*, 25(2):829–839, 2016.
- [112] R. Lucchetti, F. Mignanego, and G. Pieri. Existence theorem of equilibrium points in Stackelberg games with constraints. *Optimization*, 18:857–866, 1987.

- [113] D. G. Luenberger and Y. Ye. *Linear and Nonlinear Programming*. New York, NY: Springer Science + Business Media, 3 edition, 2008.
- [114] Jan R. Magnus and Heinz Neudecker. *Matrix Differential Calculus with Applications in Statistics and Econometrics*. John Wiley & Sons, Inc., 3 edition, 2007.
- [115] Michael W. Mahoney. Randomized algorithms for matrices and data. *Foundations and Trends in Machine Learning*, 3(2):123–224, 2011.
- [116] Julien Mairal. Optimization with first-order surrogate functions. In *International Conference on Machine Learning*, pages 783–791, 2013.
- [117] Julien Mairal. Incremental majorization-minimization optimization with application to large-scale machine learning. *SIAM Journal on Optimization*, 25(2):829–855, 2015.
- [118] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11:19–60, 2010.
- [119] K. M. Miettinen. *Nonlinear Multiobjective Optimization*. Norwell, MA: Kluwer Academic Publishers, 1998.
- [120] S. A. Miller and E. K. P. Chong. Flow-rate control for managing communications in tracking and surveillance networks. In *Proceedings of the Conference on Signal and Data Processing of Small Targets 2007 (SPIE Vol. 6699), part of the SPIE Symposium on Optics & Photonics, San Diego, California*, 2007.
- [121] J.A. Mirrlees. The theory of moral hazard and unobservable behaviour: part I. *Review on Economic Study*, 66:3–21, 1999.
- [122] Tomoya Murata and Taiji Suzuki. Doubly accelerated stochastic variance reduced dual averaging method for regularized empirical risk minimization. In *Advances in Neural Information Processing Systems*, pages 608–617, 2017.
- [123] S. G. Nash and A. Sofer. *Linear and Nonlinear Programming*. New York: McGraw-Hill Book Co., 1996.
- [124] A. Nedic and A. Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009.

- [125] Y. Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Mathematics Doklady*, 27(2):372–376, 1983.
- [126] Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22:341–362, 2012.
- [127] Yurii Nesterov, editor. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer, 2004.
- [128] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2 edition, 2006.
- [129] A. Osyczka. *Evolutionary Algorithms for Single and Multicriteria Design Optimization*. Heidelberg, Germany: Physica-Verlag, 2002.
- [130] Hua Ouyang, Niao He, Long Q. Tran, and Alexander Gray. Stochastic alternating direction method of multipliers. In *International Conference on Machine Learning*, 2013.
- [131] Y. Ouyang, Y. Chen, G. Lan, and E. Pasiliao. An accelerated linearized alternating direction method of multipliers. *SIAM Journal on Imaging Sciences*, 8(1):644–681, 2015.
- [132] Zhimin Peng, Yangyang Xu, Ming Yan, and Wotao Yin. Arock: An algorithmic framework for asynchronous parallel coordinate updates. *SIAM Journal of Scientific Computing*, 38(5):A2851–A2879, 2016.
- [133] Kaare Brandt Petersen and Michael Syskind Pedersen. *The Matrix Cookbook*. <http://orion.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf>, 2012.
- [134] M. J. D. Powell. Convergence properties of algorithms for nonlinear optimization. *SIAM Review*, 28(4):487–500, 1986.
- [135] A. Rakhlin, O. Shamir, and K. Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. In *International Conference on Machine Learning*, 2012.
- [136] Singiresu S. Rao. *Engineering Optimization: Theory and Practice*. John Wiley & Sons, Inc., 4 edition, 2009.
- [137] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of Adam and beyond. In *International Conference on Learning Representation*, 2018.

- [138] Sashank J. Reddi, Suvrit Sra, Barnabas Poczos, and Alex Smola. Stochastic Frank-Wolfe methods for nonconvex optimization. In *54th Annual Allerton Conference on Communication, Control, and Computing*, 2016.
- [139] X. Ren and Z. Lin. Linearized alternating direction method with adaptive penalty and warm starts for fast solving transform invariant low-rank textures. *International Journal of Computer Vision*, 104(1):1–14, 2013.
- [140] P. Richtárik and M. Takáć. Parallel coordinate descent methods for big data optimization. *Mathematical Programming*, 156(1-2):433–484, 2016.
- [141] S.M. Robinson. Generalized equations and their solutions, part II: applications to nonlinear programming. *Mathematical Programming Study*, 19:200–221, 1982.
- [142] R.T. Rockafellar and R. Wets, editors. *Variational Analysis*. Springer, 1998.
- [143] N. Le Roux, M. Schmidt, and F. Bach. A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets. In *Advances in Neural Information Processing Systems*, 2012.
- [144] Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *International Conference on Machine Learning*, 2017.
- [145] B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, 2002.
- [146] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.
- [147] S. Shalev-Shwartz and T. Zhang. Accelerated mini-batch stochastic dual coordinate ascent. In *Advances in Neural Information Processing Systems*, 2013.
- [148] S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14:567–599, 2013.
- [149] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.
- [150] O. Shamir and T. Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *International Conference on Machine Learning*, 2013.

参 考 文 献

- [151] Fanhua Shang, Yuanyuan Liu, James Cheng, Zhi-Quan Luo, and Zhouchen Lin. Bilinear factor matrix norm minimization for robust pca: Algorithms and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.
- [152] Yuan Shen, Zaiwen Wen, and Yin Zhang. Augmented Lagrangian alternating direction method for matrix separation based on low-rank factorization. *Optimization Methods and Software*, 29(2):239–263, 2014.
- [153] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.
- [154] Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. EXTRA: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.
- [155] W. Su, S. Boyd, and E. Candés. A differential equation for modeling Nesterov’s accelerated gradient method: Theory and insights. In *Advances in Neural Information Processing Systems*, 2014.
- [156] Ju Sun, Qing Qu, and John Wright. Complete dictionary recovery using nonconvex optimization. In *International Conference on Machine Learning*, pages 2351–2360, 2015.
- [157] Ruoyu Sun, Zhi-Quan Luo, and Yinyu Ye. On the expected convergence of randomly permuted ADMM. In *arxiv:1503.06387*, 2015.
- [158] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.
- [159] J. Trzasko and A. Manduca. Highly undersampled magnetic resonance image reconstruction via homotopic-minimization. *IEEE Transactions on Medical Imaging*, 28(1):106–121, 2009.
- [160] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. In *?????*, 2008.
- [161] Luiz Velho, Paulo Cezar Pinto Carvalho, Luiz Henrique De Figueiredo, and Jonas Gomes. *Mathematical Optimization in Computer Graphics and Vision*. Morgan Kaufmann Publishers, 2008.
- [162] René Vidal. Subspace clustering. *IEEE Signal Processing Magazine*, 28(3):52–68, 2011.

- [163] L.-X. Wang. *A Course in Fuzzy Systems and Control*. Upper Saddle River, NJ: Prentice Hall, 1999.
- [164] Yu Wang, Wotao Yin, and Jinshan Zeng. Global convergence of ADMM in non-convex nonsmooth optimization. *arXiv preprint, arXiv:1511.06324*, 2018.
- [165] Z. Wen, W. Yin, and Y. Zhang. Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. *Mathematical Programming Computation*, 4(4):333–361, 2012.
- [166] W. Wiesemann, A. Tsoukalas, P.-M. Kleniati, and B. Rustem. Pessimistic bi-level optimisation. *SIAM Journal on Optimization*, 23:353–380, 2013.
- [167] R. Winter. *Zwei-Ebenen-Optimierung mit einem stetigen Knapsack-Problem in der unteren Ebene: Optimistischer und pessimistischer Zugang (Technical Report)*. TU Bergakademie Freiberg, 2010.
- [168] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transaction on Evolutionary Computation*, 1(1):67–82, 1997.
- [169] Stephen J. Wright. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34, 2015.
- [170] Lin Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11(October):2543–2596, 2010.
- [171] Chen Xu, Zhouchen Lin, and Hongbin Zha. A unified convex surrogate for the schatten- p norm. In *AAAI Conference on Artificial Intelligence*, 2017.
- [172] Chen Xu, Zhouchen Lin, Zhenyu Zhao, and Hongbin Zha. Relaxed majorization-minimization for non-smooth and non-convex optimization. In *AAAI Conference on Artificial Intelligence*, pages 812–818, 2016.
- [173] Yangyang Xu and Wotao Yin. A block coordinate descent method for regularized multi-convex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences*, 6(3):1758–1789, 2013.
- [174] Allen Yang, Arvind Ganesh, Shankar Sastry, and Yi Ma. Fast l1-minimization algorithms and an application in robust face recognition: A review. Technical Report UCB/EECS-2010-13, EECS Department, University of California, Berkeley, Feb 2010.

参 考 文 献

- [175] Xin-She Yang. *Engineering Optimization: An Introduction with Metaheuristic Applications*. John Wiley & Sons, Inc., 2010.
- [176] Stella X. Yu and Jianbo Shi. Multiclass spectral clustering. *International Conference on Computer Vision*, 2003.
- [177] W. I. Zangwill. *Nonlinear Programming: A Unified Approach*. Englewood Cliffs, NJ: Prentice-Hall, 1969.
- [178] C.-H Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.
- [179] Tong Zhang. Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, 11:1081–1107, 2010.
- [180] Y. Zhang and L. Xiao. Stochastic primal-dual coordinate method for regularized empirical risk minimization. *Journal of Machine Learning Research*, 18(84):1–42, 2017.
- [181] Yuchen Zhang, John C. Duchi, and Martin J. Wainwright. Communication-efficient algorithms for statistical optimization. *Journal of Machine Learning Research*, 14(1):3321–3363, 2013.
- [182] Yuchen Zhang, Percy Liang, and Moses Charikar. A hitting time analysis of stochastic gradient Langevin dynamics. In *Conference on Learning Theory*, 2017.
- [183] Z. Zhang, A. Ganesh, X. Liang, and Y. Ma. TILT: Transform invariant low-rank textures. *International Journal of Computer Vision*, 99(1):1–24, 2012.
- [184] Shuxin Zheng, Qi Meng, Taifeng Wang, Wei Chen, Nenghai Yu, Zhi-Ming Ma, and Tie-Yan Liu. Asynchronous stochastic gradient descent with delay compensation for distributed deep learning. In *International Conference on Machine Learning*, 2017.
- [185] Pan Zhou, Chao Zhang, and Zhouchen Lin. Bilevel model based discriminative dictionary learning for recognition. *IEEE Transactions on Image Processing*, pages 1173–1187, 2016.
- [186] Zeyuan Allen Zhu and Lorenzo Orecchia. Linear coupling: An ultimate unification of gradient and mirror descent. In *Innovations in Theoretical Computer Science Conference*, pages 3:1–3:22, 2017.

- [187] Wangmeng Zuo, Deyu Meng, Lei Zhang, Xiangchu Feng, and David Zhang. A generalized iterated shrinkage algorithm for non-convex sparse coding. In *International Conference on Computer Vision*, pages 217–224, 2013.