Optimization theory and application

优化理论与应用

a thesis

submitted to School of Mathematics & Physics

of Xi'an Jiaotong-Liverpool University

in partial fulfilment for the award of the degree of

BSc Applied Mathematics

By

Wenxin Liu 2033784

Supervisor: Dr. Zhiping Rao

May 7, 2024

Abstract

My Final year project is divided into two sections. The first section entails an investigation and analysis of optimization theory, mainly discussed some important contents like convex sets, convex functions, Lagrange dual function and Karush-Kuhn-Tucker (KKT) condition which will be widely used in numerical optimization methods. The second section focuses on the numerical optimization and some practical application of optimization, whereby the optimization methods like interior point method, Newton's method and conjugate gradient method are employed to tackle some traditional optimization problems like linear programming and quadratic programming.

本毕业项目分为两个部分。第一部分是优化理论的研究分析，主要呈现了基本的凸分析理论，包括凸集凸函数，以及重要的拉格朗日对偶函数和 KKT 条件，这些知识将被广泛运用于后续的数值优化算法中。第二部分重点介绍了数值优化方法理论研究和与之对应的实际应用，比如利用内点法、牛顿法和共轭梯度法等优化方法解决线性规划、二次规划等传统优化问题。

KEY WORDS: Optimization theory, Karush-Kuhn-Tucker (KKT) condition， Lagrange dual function, Newton's method, conjugate gradient method, interior point method

Acknowledgements

I will take this opportunity to thank my supervisor Dr. Zhiping Rao.

Throughout my third and fourth year of undergraduate studies, particularly during my graduation project, I have gained invaluable guidance, for which I am especially grateful to Dr. Zhiping Rao. I deeply appreciate the opportunity to study at Xi'an Jiaotong-Liverpool University (XJTLU) and to receive guidance from Dr. Zhiping Rao and other faculty members in the Department of Mathematics, such as Prof. Hui Zhang, Prof.Yichen Liu, Prof.Qiang Niu, and Prof.Alastair Darby.

Reflecting on my initial entry into XJTLU, I harbored some erroneous study habits and biases toward the institution. Coming from a provincial-level key reinforcement class in high school, I became overly confident, erroneously assuming that mathematics was merely an accumulation of techniques, reinforced through extensive practice of exercises. Therefore, even in my first year, I did not attend classes regularly, believing I could easily pass calculus and linear algebra based on my high school background. Upon entering my second year, I continued in this state, still skipping classes and relying solely on problem-solving to review subjects like mathematical analysis and advanced algebra before exams. However, I soon discovered that these courses were vastly different from the computational ones in my first year. They required a considerable amount of time to prove what I deemed obvious conclusions. Nevertheless, I persisted in my belief that problem-solving was the key to understanding, rather than taking the time to truly comprehend the proof process. Unfortunately, after my second year, all assignments ceased to provide answers, and even the course leader for mathematical analysis, Ali, concealed the lecture notes outside of class hours. This made me realize my academic predicament, with unknown concepts accumulating like a snowball, leading to poor performance in exams, dropping from scores in the 80s and 90s to barely passing with scores in the 70s.

During my second year, I maintained this chaotic state, showing no interest in studying mathematics. I began to question my decision to major in mathematics and contemplated how to graduate quickly, find a job, and get by. I was also reluctant to communicate with teachers because I hadn't learned anything, and the teachers required questions to be asked in English. I couldn't even remember the English names of the theorems, so I was afraid to ask questions. Then, in my third year, I met Dr. Zhiping Rao. I am grateful that in the first week of online classes, I did not fall into despair like before, but bravely attempted the Laplace homework and interacted with him during the live tutorial. I consider this a turning point, and I want to thank

him especially. I resolved my doubts and found Dr. Zhiping Rao to be a very approachable teacher, more like an older student than a teacher. I began to try to change myself, to prove that I was not inferior to others. I should not be clueless about everything despite having a good background in high school mathematics. I started sitting at the front of the class for the MTH212 course, organizing class notes every week, and actively seeking your guidance for questions I didn't understand after class. At that time, the power point he made was very helpful to me. Each theorem was followed by example problems. If the theorem was abstract and difficult to understand, I could refer to the example problems on the next page. In addition, the proof process was deliberately omitted from the power point, forcing me to take notes according to the teacher's thinking. Therefore, I always thought that his power point was one of the best I had ever seen, and it helped me absorb the knowledge well, especially as a student with an insufficient foundation. The interest in the MTH212 course also drove my study of other courses. I no longer relied on problem-solving to understand knowledge, but recorded notes for other courses and actively asked teachers for help when needed. I was no longer the slacker I used to be. I did well on the tests for the MTH212 course, indicating that I really understood the material. And most importantly, I found that communicating with teachers was a very happy thing. Of course, communicating with Dr. Zhiping Rao was the most enjoyable, because we could communicate in Chinese, and you were really easygoing. Even if I didn't understand, I could boldly ask for clarification, or ask if my understanding was correct.

Next is the MTH210 course in the following semester. I am grateful to have been assigned to Dr. Zhiping Rao's course again. However, the MTH210 course was indeed very difficult, and every week I would try to listen attentively but find myself unable to understand. I once thought about giving up, thinking that mathematics was really difficult. But after class, I still hesitated to ask Dr. Zhiping Rao. He patiently clarified the logic and train of thought for me again in Chinese. This kind of situation repeated itself in every class, including during office hours. After all, Partial Differential Equations (PDEs) were indeed very difficult. I remembered asking Dr. Zhiping Rao why he chose to study mathematics and continue with it. He said it was because he were good at math in the college entrance examination, so he chose to study mathematics, and then continued like this. He didn't read corresponding mathematics books out of interest during your studies. I think this is one of the most real answers I have ever heard, and it feels very warm. Dr. Zhiping Rao also encouraged me to continue studying mathematics, thinking that I had a good attitude and foundation. This is also one of the motivations for my perseverance in studying.

Perhaps teachers never care whether a student has previously studied poorly, after all, they have taught too many students. For teachers, what matters more is whether students have the right learning methods and a serious learning attitude. Those who are smart or hardworking will also continue to grow under the guidance and encouragement of teachers. Just like Prof.Yichen's saying: "If you don't understand, you have to spend time continuing to study." Prof. Hui Zhang would say: "When I was studying PDEs back then, I didn't understand and ran into the classroom crying." Aistis would say: "The questions I left for you were from those old books I found

in the library during my university days."

Looking back, I remembered the moment when I enthusiastically determined my topic and made a plan. I remember writing a word document, planning to submit a 3-page PDF report every week, and clearly listing the content I wanted to learn each month. Even during winter vacation, I did not choose to rest, but spent my time in the school computer room devouring convex optimization, and persisted in submitting reports every week. In order to learn more about optimization, I actively studied optimization courses and recommended books from different schools.

Now, with the graduation project almost completed, I have been thinking for a long time about how to express my gratitude. Finally, I decided to review the past two years of learning with Dr. Zhiping Rao. Here, I sincerely thank my mentor. He pulled back someone who was about to give up from the cliff. Under his guidance, I gradually determined my life goal, which is to study numerical optimization that interests me. I hope that my current efforts can support me in achieving my future dreams, and I hope that I will have enough ability to deserve the recommendation letter he wrote for me, and to become a student who makes him proud.

Finally, I would like to conclude with the words of Prof.Yichen Liu: "I hope that students who leave XJTLU can be comparable to those from Tsinghua University and Peking University." I was very moved when I heard him say this. I believe that Prof. Liu treats this place as his second home. He hopes that his efforts can better help his students (his younger brothers and sisters) to go out and embrace mathematics and the world.

Simultaneously, I must express gratitude to my fellow classmates. Yifan Li and Renjing Wang offer to review my report and give me suggestions to refine the note. Zhe Wang and I resolved the dilemmas encountered in the theory of duality, such as Farka's Lemma and Slater's Condition. Yuzhuo Jin help me considered some problems with prime-dual method and finished the Matlab code for interior method with linear programming. Wanqian Chen and Yucheng Shen always encourage me, and share their academic pursuits, thereby bolstering my determination to forge ahead.

Contents

List of Figures

Notations

Some specific sets

$\mathbf{R}$	Real numbers.
$\mathbf{R}^n$	Real n-vectors ($n \times 1$ matrices).
$\mathbf{R}^{m\times n}$	Real $m \times n$ matrices.
$\mathbf{R}_+, \mathbf{R}_{++}$	Nonnegative, positive real numbers.
$\mathbf{C}$	Complex numbers.
$\mathbf{C}^n$	Complex n-vectors.
$\mathbf{C}^{m\times n}$	Complex $m \times n$ matrices.
$\mathbf{Z}$	Integers.
$\mathbf{Z}_+$	Nonnegative integers.
$\mathbf{S}^n$	Symmetric $n \times n$ matrices.
$\mathbf{S}^n_+, \mathbf{S}^n_{++}$	Symmetric positive semidefinite, positive definite, $n \times n$ matrices.

Vectors and matrices

$\mathbf{1}$	Vector with all components one.
e_i	i th standard basis vector.
I	Identity matrix.
X^T	Transpose of matrix X.
X^H	Hermitian (complex conjugate) transpose of matrix X.
$\mathbf{tr}\, X$	Trace of matrix X.
$\lambda_i(X)$	i th largest eigenvalue of symmetric matrix X.
$\lambda_{\max}(X), \lambda_{\min}(X)$	Maximum, minimum eigenvalue of symmetric matrix X.
$\sigma_i(X)$	i th largest singular value of matrix X.
$\sigma_{\max}(X), \sigma_{\min}(X)$	Maximum, minimum singular value of matrix X.
$X^\dagger$	Moore-Penrose or pseudo-inverse of matrix X.
$x \perp y$	Vectors x and y are orthogonal: $x^T y = 0$.
$V^\perp$	Orthogonal complement of subspace V.
$\mathbf{diag}(x)$	Diagonal matrix with diagonal entries $x_1, \ldots, x_n$.
$\mathbf{diag}(X, Y, \ldots)$	Block diagonal matrix with diagonal blocks $X, Y, \ldots$
$\mathbf{rank}\, A$	Rank of matrix A.
$\mathit{Range}(A)$	Range of matrix A.

Norms and distances

$\|\cdot\|$	A norm.
$\|\cdot\|_*$	Dual of norm $\|\cdot\|$.
$\|x\|_2$	Euclidean (or ℓ_2-)norm of vector x.
$\|x\|_1$	ℓ_1-norm of vector x.
$\|x\|_\infty$	ℓ_∞-norm of vector x.
$\|X\|_2$	Spectral norm (maximum singular value) of matrix X
$B(c,r)$	Ball with center c and radius r.
$\mathsf{dist}(A,B)$	Distance between sets (or points) A and B.

Generalized inequalities

$x \preceq y$	Componentwise inequality between vectors x and y.
$x \prec y$	Strict componentwise inequality between vectors x and y
$X \preceq Y$	Matrix inequality between symmetric matrices X and Y .
$X \prec Y$	Strict matrix inequality between symmetric matrices X and Y .
$x \preceq Ky$	Generalized inequality induced by proper cone K.
$x \prec Ky$	Strict generalized inequality induced by proper cone K .
$x \preceq K^*y$	Dual generalized inequality.
$x \prec K^*y$	Dual strict generalized inequality.

Topology and convex analysis

$\mathsf{Card}\, C$	Cardinality of set C.
$\mathsf{Int}\, C$	Interior of set C.
$\mathsf{Relint}\, C$	Relative interior of $\mathsf{set}\, C$.
$\bar{C}$	Closure of set C.
∂C	Boundary of set C.
$\mathsf{conv}\, C$	Convex hull of set C.
$\mathsf{aff}\, C$	Affine hull of set C.
K^*	Dual cone associated with K.
I_C	Indicator function of set C.
S_C	Support function of set C.
f^*	Conjugate function of f.

Functions and derivatives

$f : A \to B$	f is a function on the set $\mathsf{dom}\, f \subseteq A$ into the set B.
$\mathsf{dom}\, f$	Domain of function f.
$\mathsf{epi}\, f$	Epigraph of function f.
∇f	Gradient of function f.
$\nabla^2 f$	Hessian of function f.

Chapter 1

Introduction

1.1 Motivation

Ever since I discovered my passion for mathematics and its application, I've consistently sought opportunities to dive deeper into its intricate world. During my undergraduate studies in Applied Mathematics, my passion for mathematics found its true calling. In my freshman year, I took on the rigorous Mathematical Modeling Competition (MCM). Along with my team, we tackled a challenging research project centered on microbial symbiosis and its environmental implications. The depth and complexity of the subject pushed us to immerse ourselves in rigorous self-study, exploring advanced mathematical models and algorithms. As we navigated the competition's challenges, we engaged in intense discussions, deepening our grasp of techniques such as time series prediction and logistic regression. Our collective dedication and the synergy of our team efforts led to commendable results in the competition.

My journey into mathematical modeling and application for optimization culminated when I led a project during our school's Operational Research festival. Our goal was to devise an optimal scheduling system for school sports meetings. Four weeks of intense research and collaboration earned us the prestigious first prize. At the core of our strategy was mixed integer programming. Guided by established Operational Research literature, we took inspiration from the 'traveling salesman' problem. We built two methods satisfying the real-world situation based on two different models. However, the dual-method approach led to two seemingly valid yet different solutions. This was our primary challenge. Neither solution was conclusively superior. Delving deeper and drawing upon my previous readings, I proposed we explore double optimization, a method introduced to us by our mentor. Yet, another hurdle emerged: our refined model lacked the saddle points crucial for solution stability. Drawing inspiration from the book Convex Optimization written by Stephen Boyd [Boyd and Vandenberghe (2004)], we studied the principle of generalized inequality and its proposition with a convex cone. I recognized the parallels in our dilemma. With a blend of determination and innovation, I steered our team towards merging our dual-method approach, effectively harnessing the strengths of both strategies. By employing advanced computer simulations, we unearthed an optimal balance and crafted a solution that was both robust and efficient. Reflecting on this endeavor, I not only honed my problem-solving

skills but also experienced firsthand the complexities of research. This project shaped my appreciation for the balance between theory and practicality, and it reaffirmed my passion for the field of mathematical modeling and optimization.

During last year's summer recess, I attended an academic summit on operational optimization at Zhejiang University. This symposium left an indelible impression on me. Professor Yuan Xiaoming from Hong Kong University shared his insights into current prevalent optimization issues, such as employing physical engines to enhance the solution of elastic body simulation, and utilizing the concept of optimal control in fluid dynamics to construct thermal convection models. These models, grounded profoundly in mathematical physics, intrigued me intensely. It dawned on me that our approach to resolving real-world issues is not in aimless trials or simply observing the outcome's quality. In essence, we create models to solve problems in anticipation of recapitulating these complex situations occurring in nature using the language of mathematical physics. I find myself mesmerized by the mathematical equations and symbols that manifest in the solution process, observing how they deduce exceptional results based on reason and logic. There is an unparalleled enjoyment found in such a journey of discovery.

As I look forward to further exploring the mathematical world of optimization, I choose optimization theory and application as my final year project. My goal is to get a deep understanding of some basic analysis for optimization and try some single applications with the method I am studying this year. I hope this period of studying experience will equip me with enough competence to pursue further study at KU Leuven, where I intend to finish my doctoral degree, particularly with Professor Stefan Vandewalle to explore sampling methods and robust optimization for partial differential equations. I am driven by the conviction that optimization is not just a study but a language that shapes my mathematical worldview and allows me to articulate complex concepts with clarity and precision.

1.2 Outlier

My Final year project is divided into two sections. The first section entails an investigation and analysis of optimization theory, mainly discussed some important contents like convex sets, convex functions, convex optimization problems, Lagrange dual function and Karush-Kuhn-Tucker (KKT) condition which will be widely used in numerical optimization methods. The second section focuses on the numerical optimization and some practical application of optimization, whereby the optimization methods like interior point method, Newton's method and conjugate gradient method are employed to tackle some traditional optimization problems like linear programming and quadratic programming.

Theory of optimization

In the theoretical part, there are four sections. The first section mainly involves the definition and properties of convex sets and convex functions, alongside numerous principles of convex analysis, i.e, operations on sets and functions that conserve convexity. Additionally, the report

also includes the definition and property of quasiconvex functions, which are more common in real-life problems.

In the second section, which is entitled Convex optimization problems, we meticulously examine optimization problems and elucidate a series of transformations that can be utilized to rephrase original problems. Furthermore, this section introduces certain prevalent subcategories of convex optimization problems, such as linear programming and quadratically constrained quadratic program (QCQP) problems. In the later sections and chapters, we will further discuss the problems listed in this section.

The third section provides a concise overview of vector optimization and multicriteria optimization. Within this section, the concept of Pareto optimality is introduced as a fundamental principle in multicriteria optimization, alongside the discussion of general inequalities. Additionally, basic propositions, such as the correlation between multicriteria optimization and vector optimization, are examined.

Section 4 focuses on duality theory, which constitutes a pivotal aspect of this report. It elaborates on fundamental concepts such as the Lagrange dual function, the classical Karush-Kuhn-Tucker (KKT) conditions for optimality, Slater's condition for strong duality, and alternative systems. These theoretical insights facilitate the establishment of connections between intricate original optimization problems and computationally simpler dual formulations. Consequently, this framework empowers us to effectively address the optimization challenges discussed in preceding chapters by means of problem transformation. Notably, the principles elucidated in this section serve as foundational elements underpinning various numerical optimization algorithms employed in practice.

Optimization Algorithms

In the algorithm part, there are four sections. The first section primarily encompasses the foundational context of numerical optimization, including the roles of minimized sequences and function convergence in optimization, the practical applications of second-order Taylor expansions in numerical optimization, and a brief introduction to algorithms for unconstrained minimization problems. This introduction covers the origins of descent methods, the application and convergence analysis of Newton's method, and the utilization and convergence analysis of the Conjugate Gradient method.

The second section gives a further discussion about equality constrained minimization problems where several methods like eliminating constraints and solving dual problems are provided to help solve this sort of problems. Additionally, this section primarily employs the second-order Taylor expansion approximation to solve these types of problems. In the approximation process, we utilize Newton's method as the core iterative idea and establish iterative equations based on the corresponding matrix generated by the KKT conditions. Furthermore, we investigate the complex situations of infeasible initial conditions and improve the original Newton's method by

proposing the concept of the residual function, thereby establishing a more stable convergent iterative relationship.

The third section mainly delves into methodologies for addressing inequality-constrained optimization problems. Within this segment, we commence by analyzing the relationship between this category of problems and all preceding minimized optimization problems. Employing an indicator function, we systematically transform all inequality-constrained problems into equality-constrained ones, subsequently employing previously mentioned algorithms to resolve them. In the process of transformation, mindful of numerical accuracy in computational endeavors, we opt for methodologies such as barrier methods to numerically approximate problem solutions. Additionally, this chapter scrutinizes the intricate scenarios of initial infeasibility and posits approaches such as the two-phase problem and the prime-dual method to tackle certain complexities arising from initial value issues.

The forth section represents a classic application and example where, through a comprehensive consolidation of preceding methodologies, an algorithm solver for linear programming problems is devised using an interior point method, with the standard Linear Programming (LP) problem as the focal point.

1.3 Literature Review

History development

Optimality is a fundamental principle that governs natural laws, biological behaviors, and social activities. It has been present since the earliest stages of human civilization, although before mathematics was well-established, optimization was done only through simulation. In the 19th century and even today, simulation is still used for optimization. For example, nowadays, people still use Hamiltonian Monte Carlo, and variational Markov Chain Monte Carlo simulations to solve optimization problems.
The history of optimization in mathematics can be divided into three periods. In the first period, only special techniques were available to optimize specific functions, such as quadratic functions of one variable:

$$y = ax^2 + bx + c.$$

These techniques were often used to solve problems related to constantly accelerating movement, like determining the highest point a stone can reach when thrown with a certain initial speed and angle. During this time, stone-throwing machines were important military weapons [Du et al. (2009)]. Today, computing maximum or minimum points of quadratic functions remains an important optimization technique, like solving least square problems.
During the second period, the huge development in optimization has happened since 1947. During that year, linear programming was discovered by G.B. Dantzig's and refined by John von Neumann, influenced by their World War II experience and the challenge to mechanize the planning process [Dantzig (1991)]. The development of computer technology allowed for the wide application of linear programming and the rapid growth of optimization. After that time,

the KKT condition, proposed in 1951, marked a new start of nonlinear programming after the pioneering contributions of Euler and Lagrange and sparked a lot of creation of new algorithms to help solve complex optimization problems. Since then, research mainly focused on gradient methods and Newton's method, which have been widely applied [Dantzig (1955)]. Starting from the DFP method proposed by Davidon, Fletcher and Powell, the 1960s was a period of active research on quasi-Newton methods, and good research was also done on conjugate gradient methods. During that time, unconstrained optimization reached its peak, leading to the prevalent approach of transforming constrained problems into unconstrained problems. This era saw a surge in the popularity of penalty and barrier methods, driven by the objective of minimizing a composite function that encapsulates both the original objective function and the impact of constraints [Wright (2005)].

The 1970s was a period of rapid development in optimization, and the sequence quadratic programming method and Lagrange multiplier method were the most important research results of this period [Andrei (2017)]. The rapid development of computers has enabled the research of nonlinear programming. Trust region methods, sparse quasi-Newton methods, methods for large-scale problems and parallel computing were studied in the 1980s [Steihaug (1983)]. In the 1990s, interior point methods, direct search methods and finite storage methods were used to solve nonlinear programming problems [Hillier and Lieberman (2015)].

Nowadays, optimization has merged with economics and has many branches, with researchers only able to specialize in a few. Several researchers in optimization have received Nobel Prizes in economics. Optimization has become an interdisciplinary area involving mathematics, computer science, industrial engineering, and management science.

Some Classical Problems in Optimization

In this section, we will introduce some classical problems in optimization which are remarked as the milestone during the development of optimization.

Least square problems

A least-square problem is an optimization problem with no constraints and an objective which is a sum of square terms of the form $a_i^T x - b_i$:

$$\text{minimize} \quad f_0(x) = \|Ax - b\|_2^2 = \sum_{i=1}^{k} \left(a_i^T x - b_i\right)^2 .$$

Here $A \in \mathbf{R}^{k \times n}$ (with $k \geq n$), a_i^T are the rows of A, and the vector $x \in \mathbf{R}^n$ is the optimization variable.

The solution of a least-square problem can be reduced to solving a set of linear equations,

$$\left(A^T A\right) x = A^T b,$$

so we have the analytical solution $x = \left(A^T A\right)^{-1} A^T b$. For least-squares problems we have good algorithms like Practical QR Algorithm and implicit Q Theorem for solving the problem to high accuracy, and solve it in a time approximately proportional to $n^2 k$, with a known constant [Golub and van Loan (2013)].

Linear programming

Another classical optimization problem is linear programming with its standard form:

$$\begin{aligned} &\text{Maximize} \quad z = a_1x_1 + c_2x_2 + \cdots\cdots + c_nx_n \\ &\text{s.t} \begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n \leq b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n \leq b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \cdots + a_{mn}x_n \leq b_m \\ x_1, x_2, x_3, \cdots, x_n \geqslant 0. \end{cases} \end{aligned}$$

There are a lot of methods to solve linear programming problems, one famous method is called simplex method based on the simple theorem for Optimal solution [Hillier and Lieberman (2015)], which is recalled as follows:

1. Optimal solution must be Corner point feasible (CPF) solutions.
2. Two CPF solutions are on the same edge created by $n-1$ constraints.
3. Two CPF solutions are adjacent, so for each step, to find the optimal solution, we just need to compare two adjacent CPF solutions and find a better one until no adjacent part is better than the current solution.

More details will be discussed in the following chapters.

Convex optimization Problems

A convex optimization problem follows as the form [Boyd and Vandenberghe (2004)].

$$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{aligned} \tag{1.1}$$

to describe the problem of finding an x that minimizes $f_0(x)$ among all x that satisfy the conditions $f_i(x) \leq 0, i = 1, \ldots, m$, and $h_i(x) = 0, i = 1, \ldots, p$.
where the functions $f_0, \ldots, f_m, h_1, \ldots, h_p : \mathbf{R}^n \to \mathbf{R}$ are convex, i.e., f, h satisfy

$$f_i(\alpha x + \beta y) \leq \alpha f_i(x) + \beta f_i(y), \quad i = 0, 1 \ldots, m,$$

for all $x, y \in \mathbf{R}^n$ and all $\alpha, \beta \in \mathbf{R}$ with $\alpha + \beta = 1, \alpha \geq 0, \beta \geq 0$.
The least-square problem and linear programming problem shown beforehand are both special cases of the general convex optimization problem.

Integer Programming

Integer programming is a special kind of linear programming with some variable being integers. For example, we require decision variables to be binary variables (or $0-1$ variables). We now show a classical example here:

$$\text{Maximize} \quad Z = 9x_1 + 5x_2 + 6x_3 + 4x_4,$$

subject to

$$\begin{aligned} 6x_1 + 3x_2 + 5x_3 + 2x_4 &\leq 10 \\ x_3 + x_4 &\leq 1 \\ -x_1 + x_3 &\leq 0 \\ -x_2 + x_4 &\leq 0 \\ x_j &\leq 1 \\ x_j &\geq 0 \end{aligned}$$

and

$$x_j \text{ is integer}, \qquad \text{for } j = 1, 2, 3, 4.$$

Equivalently, the last three lines of this model can be replaced by the single restriction x_j is binary, for $j = 1, 2, 3, 4$ [Hillier and Lieberman (2015)].
Integer programming or Mix integer programming (MIP) are commonly seen in real life problems like transportation assignment problems.

Nonlinear optimization problems

Nonlinear optimization refers to a problem where the objective or constraint functions are not linear and may not be convex. Unfortunately, there are no effective methods for solving these types of problems. Even seemingly simple problems with few variables can be very difficult and problems with hundreds of variables can be impossible to solve. As a result, different approaches are used to tackle nonlinear programming problems, but each approach requires some trade-offs. We will now show a famous nonlinear programming problem here [Boyd and Vandenberghe (2004)].
(SOCP) Second-order cone program has the form :

$$\begin{array}{ll} \text{minimize} & f^T x \\ \text{subject to} & \|A_i x + b_i\|_2 \leq c_i^T x + d_i, \quad i = 1, \ldots, m \\ & Fx = g, \end{array}$$

where $x \in \mathbf{R}^n$ is the optimization variable, $A_i \in \mathbf{R}^{n_i \times n}$, and $F \in \mathbf{R}^{p \times n}$. We call a constraint of the form

$$\|Ax + b\|_2 \leq c^T x + d,$$

where $A \in \mathrm{R}^{k \times n}$, a second-order cone constraint, since it is the same as requiring the affine function $(Ax + b, c^T x + d)$ to lie in the second-order cone in $\mathbf{R}^{k+1}$.

Some Classical optimization algorithms

Newton's method

Newton's method is a cornerstone across multiple domains, including numerical analysis, operations research, optimization, and control, with broad-ranging applications in computer science, industrial and financial research, and data mining. Its significance in optimization is immense, serving as the foundation for highly efficient techniques in both linear and nonlinear programming [Polyak (2007)]. The fundamental algorithm involves obtaining the zeros of the derivative

of the second-order Taylor expansion of the function using Newton's method, it has the form: Given $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ twice continuously differentiable, $x_0 \in \mathbb{R}^n$; at each iteration k,

$$\text{solve } \nabla^2 f(x_k)\,\Delta x_k = -\nabla f(x_k),$$
$$x_{k+1} = x_k + \Delta x_k,$$

where x_k is the the point at k iteration [Dennis Jr and Schnabel (1996)]. Notably, Newton's method underpins polynomial time interior point algorithms utilized in convex optimization, we will further discuss this in the following chapters.

Conjugate Gradient method

Conjugate gradient methods are distinguished by their minimal memory demands and robust convergence properties at both local and global levels. It was cited among the Top 10 algorithms of the 20th century by Computing in Science & Engineering Magazine. Hestenes and Stiefel (1952) introduced these techniques as an efficient approach to solving linear algebraic systems featuring positive definite matrices [Šolcová (2004)]. Later on, Fletcher and Reeves (1964) introduced the initial nonlinear conjugate gradient algorithm [Hager and Zhang (2006)]. Al-Baali's seminal contribution in 1985 solidified the global convergence of the FR algorithm in addressing broad-spectrum nonlinear challenges, particularly in scenarios featuring inexact line searches. In practical applications, Conjugate gradient method typically demands a greater number of iterations compared to quasi-Newton methods and Newton's methods. Nonetheless, despite this slower convergence, Conjugate gradient methods offer certain advantages due to their matrix-free nature and necessitate less computational memory, rendering them more suitable for large-scale problem domains [Andrei et al. (2020)]. We will further discuss this in the following chapters.

Interior point method

Contemporary interior methods maintain close ties with the "classical" barrier methods from the 1960s. The log-barrier function associated with (1.1) is

$$\begin{array}{ll} \text{minimize} & f_0(x) + \sum_{i=1}^{m} -(1/t)\log\left(-f_i(x)\right) \\ \text{subject to} & Ax = b, \end{array}$$

where $t > 0$ and t approximates to infinity.

We perform numerical optimization on the obtained barrier function using the established method like Newton's method, employing the dual gap as the criterion for achieving optimality. Additionally, upon reaching an optimal solution in each iteration, we further refine the barrier function approximation before repeating the optimization steps. This iterative process involves stepwise optimization of the barrier function, followed by optimization of the approximation to both the barrier function and the original function. Consequently, each optimization iteration occurs within the feasible domain, leading to the designation of this approach as an interior-point method [Conn et al. (1994)]. However, these approaches come with various limitations. For example, the computation process may encounter ill-conditioned matrices. Moreover, despite accurately calculating the Newton direction, primal barrier methods face inherent challenges in scaling the search direction during the initial iterations after reducing the barrier

parameter [Wright (2005)]. El-Bakry et al. (1996) proposed the prime-dual method for interior point method. In this method, problems are tackled using the modified Newton method, which produces search directions based on a primal-dual system specifically designed for interior methods.

Chapter 2

Theory of Optimization

In this chapter, we will mainly discuss the definition and properties of convex sets, convex functions, convex problems and dual theories, alongside numerous principles of convex analysis. The definitions and propositions presented are interesting and provide a strong foundation for practical numerical analysis for algorithms.

2.1 Basic definitions for optimization

The first section briefly introduces convex optimization, mainly discussing the definitions and properties of convex sets and functions.

Convex sets

Definition 2.1. *(Lines and segments) Suppose $x_1 \neq x_2$ are two points in R^n. Points in the form $y = \theta x_1 + (1-\theta)x_2$, where $\theta \in R$, form the line through x_1 and x_2. Specially, when $\theta \in [0,1]$, it forms the segment between x_1 and x_2.*

Definition 2.2. *(Affine sets) A set $C \subseteq R^n$ is affine if the line through any two distinct points in C lies in C.*

Definition 2.3. *(Affine combination) Assume $\theta_1, \theta_2, \ldots, \theta_k \in \mathrm{R}$, let new $\theta_k = 1 - \theta_1 - \theta_2 - \cdots - \theta_{k-1}$, we could define affine combination C as the form $C = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_k x_k$.*

Definition 2.4. *(Convex set) A set C is convex if the line segment between any two points in C lies in C, i.e., $\theta x_1 + (1-\theta)x_2 \in C$ where $\theta \in [0,1]$.*

Definition 2.5. *(Convex combination) Similarly to affine combination, we define convex combination as the form $\theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_k x_k$, where $\theta_1 + \theta_2 + \cdots + \theta_k = 1$ and $\theta_i \geq 0, \quad i = 1, 2, \ldots, k$.*

Theorem 2.6. *A Norm ball C with its center x_c and radius r given by $C = \{x \mid \|x - x_c\| \leq r\}$ is convex.*

Proof. Take $x_1, x_2 \in C$, we have $\|x_1 - x_c\| \leq r$, and $\|x_2 - x_c\| \leq r$,
so $\|\theta x_1 + (1-\theta)x_2 - x_c\| = \|\theta(x_1 - x_c) + (1-\theta)(x_2 - x_c)\| \leq \theta\|x_1 - x_c\| + (1-\theta)\|x_2 - x_c\| \leq r$. □

Proposition 2.7. *A Euclidean ball C in $\mathbf{R}^n$ has the form*

$$B(x_c, r) = \{x \mid \|x - x_c\|_2 \leq r\} = \left\{x \mid (x - x_c)^T (x - x_c) \leq r^2\right\},$$

where $r > 0$, and x_c is the center of the ball and the scalar r is its radius. We have C is convex. Similarly, given an ellipsoid D with its form

$$\varepsilon = \{x | (x - x_c^T) P^{-1} (x - x_c) \leq 1\},$$

where x_c is the center of the ellipsoid, P is symmetric and positive definite, D is also convex.

Proof. For Euclidean ball, the proof is quite similar to the proof of norm ball. We then mainly discuss the proof that ellipsoid is convex.
For Ellipsoids, it satisfies

$$\varepsilon = \{x | (x - x_c^T) P^{-1} (x - x_c) \leq 1\}.$$

For any $x_1, x_2 \in \varepsilon$, we have

$$(x_1 - x_c)^T P^{-1} (x_1 - x_c) \leq 1, \text{ and } (x_2 - x_c)^T P^{-1} (x_2 - x_c) \leq 1,$$

we need to prove $\theta x_1 + (1 - \theta) x_2 \in \epsilon$, i.e.

$$(\theta x_1 + (1 - \theta) x_2 - x_c)^T P^{-1} (\theta x_1 + (1 - \theta) x_2 - x_c) \leq 1.$$

For the left hand side, it is equal to

$$(\theta (x_1 - x_c) + (1 - \theta)(x_2 - x_c))^T P^{-1} (\theta (x_1 - x_c) + (1 - \theta)(x_2 - x_c)),$$

where we can deduce

$$\theta^2 (x_1 - x_c)^T P^{-1} (x_1 - x_c) + (1 - \theta)^2 (x_2 - x_c)^T P^{-1} (x_2 - x_c)$$

$$+ (1 - \theta)\theta (x_2 - x_c)^T P^{-1} (x_1 - x_c) + \theta (1 - \theta)(x_1 - x_c)^T P^{-1} (x_2 - x_c).$$

Notice that P is symmetric positive definite, so

$$P^{-1} = (P^{-1})^T,$$

and from this we obtain

$$(x_2 - x_c)^T P^{-1} (x_1 - x_c) = (x_1 - x_c)^T P^{-1} (x_2 - x_c)$$

we then get the left hand side

$$\leq \theta^2 + (1 - \theta)^2 + 2\theta (x_1 - x_c)^T P^{-1} (x_2 - x_c),$$

so it suffices to prove

$$\theta^2 + (1 - \theta)^2 + 2\theta (1 - \theta)(x_1 - x_c)^T P^{-1} (x_2 - x_c) \leq 1.$$

Notice that P is positive definite, so P^{-1} is also positive definite. We could then write

$$P^{-1} = U \Lambda U^T$$

where U is an orthogonal matrix. So we define

$$(P^{-1})^{\frac{1}{2}} = U\Lambda^{\frac{1}{2}}U^T$$

Thus

$$(x_1 - x_c)^T P^{-1}(x_2 - x_c) = (x_1 - x_c)^T P^{-\frac{1}{2}} P^{-\frac{1}{2}}(x_2 - x_c).$$

By Cauchy-Schwarz inequality, we have

$$\|(x_1 - x_c)^T P^{-\frac{1}{2}} P^{-\frac{1}{2}}(x_2 - x_c)\| \leq \|(x_1 - x_c)^T P^{-\frac{1}{2}}\| \|P^{-\frac{1}{2}}(x_2 - x_c)\|.$$

Since $(x_i - x_c)^T P^{-1}(x_i - x_c) = ||(x_i - x_c)^T P^{-\frac{1}{2}}||^2, i = 1, 2$, and we have both

$$(x_1 - x_c)^T P^{-1}(x_1 - x_c) \leq 1 \text{ and } (x_2 - x_c)^T P^{-1}(x_2 - x_c) \leq 1,$$

so finally for the left hand side we get

$$\theta^2 + (1-\theta)^2 + 2\theta(1-\theta) \leq 1.$$

□

Remark 2.8. *We can also prove this result by introducing $\|x\| = \sqrt{x^T P x}$ which is also a norm where P is symmetric and positive definite. Actually, this is called Mahalanobis norm.*

Definition 2.9. *(Convex cone) A set C is a convex cone if it is convex and it is a cone i.e. for any $x_1, x_2 \in C$ and $\theta_1, \theta_2 \geq 0$, we have $\theta_1 x_1 + \theta_2 x_2 \in C$.*

Proposition 2.10. *The set S_+^n is a convex cone, where S_+^n means the set of symmetric positive semi-definite matrix.*

Proof. If $\theta_1, \theta_2 \geq 0$ and $A, B \in S_+^n$, then $\theta_1 A + \theta_2 B \in S_+^n$ because we have

$$x^T(\theta_1 A + \theta_2 B)x = \theta_1 x^T A x + \theta_2 x^T B x \geq 0.$$

□

Definition 2.11. *(Polyhedron) A polyhedron $\mathcal{P}$ is a solution set of finite number of linear equalities and inequalities:*

$$\mathcal{P} = \{x \mid Ax \preceq b, Cx = d\},$$

where

$$A = \begin{bmatrix} a_1^T \\ \vdots \\ a_m^T \end{bmatrix}, \quad C = \begin{bmatrix} c_1^T \\ \vdots \\ c_p^T \end{bmatrix},$$

and the symbol $\preceq$ denotes vector inequality or componentwise inequality in $\mathbf{R}^m$.

Definition 2.12. *(Simplex) Given points $v_0, v_1, \ldots, v_k \in \mathbb{R}^n$ in general position, a simplex of dimension k (called a k-simplex) is the smallest convex set containing them, that is, the set*

$$\left\{ \lambda_0 v_0 + \lambda_1 v_1 + \cdots + \lambda_k v_k \mid \lambda_i \geqslant 0, \sum_{i=1}^{k} \lambda_i = 1 \right\}.$$

The points $v_0, v_1, \ldots, v_k$ are called the vertices of the simplex.

Proposition 2.13. *If C is a simplex, then it must be a polyhedron.*

Proof. Let C be a simplex and we choose $x \in C$ s.t. $x = \theta_0 v_0 + \theta_1 v_1 + \cdots + \theta_k v_k$ for some $\theta \succeq 0$ with $\mathbf{1}^T\theta = 1$.

$$x = (1 - \theta_1 - \cdots - \theta_k)v_0 + \theta_1 v_1 + \cdots + \theta_k v_k = v_0 + \theta_1(v_1 - v_0) + \cdots + \theta_k(v_k - v_0)$$

Define $y = (\theta_1, \ldots, \theta_k)$ and

$$B = \begin{bmatrix} v_1 - v_0 & \cdots & v_k - v_0 \end{bmatrix} \in \mathbf{R}^{n\times k},$$

B is of full rank as each part of vectors are linearly independent, and we say that $x \in C$ if and only if

$$x = v_0 + By$$

for some $y \succeq 0$ with $1^T y \leq 1$. Therefore there exists a nonsingular matrix $A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \in \mathbf{R}^{n\times n}$ such that

$$AB = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} B = \begin{bmatrix} I \\ 0 \end{bmatrix}.$$

so we derive

$$A_1 x = A_1 v_0 + y, \quad A_2 x = A_2 v_0.$$

This implies that $x \in C$ if and only if

$$A_2 x = A_2 v_0, \quad A_1 x \succeq A_1 v_0, \quad \mathbf{1}^T A_1 x \leq 1 + \mathbf{1}^T A_1 v_0,$$

which gives a set of linear equalities and inequalities in x, and this satisfies the definition of polyhedron. □

Proposition 2.14. *Convexity is preserved under intersection*

Proof. Assume $x_1, x_2 \in \bigcap_{i\in I} C_i$, where I is an index set. For both $x_1, x_2 \in C_i$, since any C_i is convex, $\theta x_1 + (1-\theta)x_2 \in C_i, i = 1, 2, 3, \ldots$, it also holds for $\bigcap_{i\in I} C_i$. □

Definition 2.15. *(Affine functions) A function f is affine if it is a sum of a linear function and a vector of constants, i.e., if it has the form $f(x) = Ax + b$, where $A \in R^{m\times n}$ and $b \in R^m$.*

Proposition 2.16. *If $S \subseteq R^n$ is convex, and $f : R^n \to R^m$ is an affine function. Then the image of $f(S)$ is convex. Similarly, if f is an affine function, then the inverse image $f^{-1}(S) = \{x | f(x) \in S, S \subseteq R^n \text{ is convex}\}$ is also convex.*

Proof. Take $x_1, x_2 \in S$, $\theta f(x_1) + (1-\theta)f(x_2) = \theta(Ax_1 + b) + (1-\theta)(Ax_2 + b) = A(\theta x_1 + (1-\theta)x_2) + b \in f(S)$ because S is convex.
For second part of proof, the argument is quite similar. □

Lemma 2.17. *Some basic operations on the convex set preserve convexity.*

1. *If $S \subseteq R^n$ is convex, $\alpha \in R$, then αS is convex*
2. *The sum of two convex sets is also convex. i.e., $S_1 + S_2 = \{x + y | x \in S_1, y \in S_2\}$ is convex if S_1, S_2 are convex.*

3. *The Cartesian product of two convex sets is convex, i.e.,* $S_1 \times S_2 = \{(x_1, x_2) | x_1 \in S_1, x_2 \in S_2\}$ *is convex if* S_1, S_2 *are convex.*

Definition 2.18. *(Perspective function) Give a function* $P : R^{n+1} \to R^n$*, with domain* $\mathbf{dom}\, P = R^n \times R_{++}$*, P is defined as* $P(z,t) = z/t$*.*

Proposition 2.19. *Given a perspective function P, if* $C \subseteq domP$ *is convex, then its image* $P(C)$ *is convex.*

Proof. Suppose that $x = (\tilde{x}, x_{n+1}), y = (\tilde{y}, y_{n+1}) \in \mathbf{R}^{n+1}$ with $x_{n+1} > 0$, $y_{n+1} > 0$. Then for $0 \le \theta \le 1$,

$$P(\theta x + (1-\theta)y) = \frac{\theta\tilde{x} + (1-\theta)\tilde{y}}{\theta x_{n+1} + (1-\theta)y_{n+1}} = \frac{\theta x_{n+1}}{\theta x_{n+1} + (1-\theta)y_{n+1}} \frac{\tilde{x}}{x_{n+1}} + \frac{(1-\theta)y_{n+1}}{\theta x_{n+1} + (1-\theta)y_{n+1}} \frac{\tilde{y}}{y_{n+1}}.$$

We set

$$\mu = \frac{\theta x_{n+1}}{\theta x_{n+1} + (1-\theta)y_{n+1}} \in [0,1].$$

Then we obtain $P(\theta x + (1-\theta)y) = \mu P(x) + (1-\mu)P(y) \in P(C)$.

□

Lemma 2.20. *Let P be the perspective function and C be a convex set in the image of perspective function, then* $P^{-1}(C)$ *is convex.*

Proof. If $C \subseteq \mathbf{R}^n$ is convex, then its preimage

$$P^{-1}(C) = \left\{(x,t) \in \mathbf{R}^{n+1} \mid x/t \in C, t > 0\right\},$$

is convex. To prove this, suppose there exists $(x,t) \in P^{-1}(C), (y,s) \in P^{-1}(C)$, with $0 \le \theta \le 1$. We need to demonstrate

$$\theta(x,t) + (1-\theta)(y,s) \in P^{-1}(C),$$

i.e.

$$\frac{\theta x + (1-\theta)y}{\theta t + (1-\theta)s} \in C.$$

This follows from

$$\frac{\theta x + (1-\theta)y}{\theta t + (1-\theta)s} = \mu(x/t) + (1-\mu)(y/s),$$

where

$$\mu = \frac{\theta t}{\theta t + (1-\theta)s} \in [0,1].$$

□

Definition 2.21. *(Linear-fractional functions) A linear-fractional function is formed by composing the perspective function with an affine function. Suppose* $g : \mathbf{R}^n \to \mathbf{R}^{m+1}$ *is affine, i.e.,*

$$g(x) = \begin{bmatrix} A \\ c^T \end{bmatrix} x + \begin{bmatrix} b \\ d \end{bmatrix},$$

where $A \in \mathbf{R}^{m\times n}, b \in \mathbf{R}^m, c \in \mathbf{R}^n$*, and* $d \in \mathbf{R}$*. The function* $f : \mathbf{R}^n \to \mathbf{R}^m$ *given by* $f = P \circ g$*, i.e.,*

$$f(x) = (Ax+b)/\left(c^T x + d\right), \quad \mathbf{dom}\, f = \left\{x \mid c^T x + d > 0\right\},$$

is called a linear-fractional (or projective) function.

Theorem 2.22. *(Strictly separating hyperplane theorem) Suppose C and D are nonempty disjoint closed convex sets, and at least one of them is bounded. Then there exist $a \neq 0$ and b such that $a^T x > b$ for all $x \in C$ and $a^T x < b$ for all $x \in D$.*

Proof. We will prove it in the framework of the Euclidean metric.
Since C and D are disjoint and at least one is compact (by Heine-Borel Theorem), assume C is compact, and D is closed. Notice that C and D are disjoint, i.e., $C \cap D = \emptyset$, so for any point $p \in C$, x can not also be the closure point for D, thus there exists $r > 0$, s.t $B(p,r) \cap D = \emptyset$, this means the distance between C and D, defined as

$$\mathbf{dist}(C,D) = \inf\{\|u - v\|_2 \mid u \in C, v \in D\}$$

is positive, and we can find two points $c \in C$ and $d \in D$ that satisfy $\|c - d\|_2 = \mathbf{dist}(C,D)$.
Define a hyperplane that is perpendicular to the line segment between c and d, and goes through the middle of this line segment.

$$a = c - d, \quad b = \frac{\|c\|_2^2 - \|d\|_2^2}{2}.$$

We will show that the affine function

$$f(x) = a^T x - b = (c-d)^T (x - (1/2)(c+d))$$

is nonnegative on C and nonpositive on D, i.e., that the hyperplane $\{x \mid a^T x = b\}$ separates C and D.
We will then show that f is nonnegative on C by contradiction. The proof that f is nonnegative on C is similar (or follows by swapping C and D and considering $-f$).
Suppose there is a point $u \in C$, which is different from c, so the distance between u and d should be larger than that between c and d. By contradiction, for u, we have:

$$f(u) = (c-d)^T (u - (1/2)(c+d)) < 0.$$

It is equivalent to express $f(u)$ as

$$f(u) = (c-d)^T (u - c + (1/2)(c-d)) = (c-d)^T (u-c) + (1/2)\|c-d\|_2^2.$$

This implies we must have $(c-d)^T (u-d) < 0$. We then derive that

$$\frac{d}{dt}\|c + t(u-d) - d\|_2^2 \Big|_{t=0} = 2(c-d)^T (u-d) < 0.$$

Therefore for some small $t > 0$, with $t \leq 1$, we have

$$\|c + t(u-d) - d\|_2 < \|c-d\|_2.$$

This shows that there exists a point along the line segment of u and d whose distance to c is smaller than d to c, which contradicts the previous condition.
Thus we conclude that f is nonnegative on C. □

Theorem 2.23. *(Separating hyperplane theorem): Suppose C and D are nonempty disjoint convex sets, i.e., $C \cap D = \emptyset$. Then there exist $a \neq 0$ and b such that $a^T x \leq b$ for all $x \in C$ and $a^T x \geq b$ for all $x \in D$.*

Proof. We will then divide the proof into three parts.

First part: We will prove separating hyperplane theorem is equivalent to prove that given a non-empty convex set E with $0 \notin E$, there exists a hyperplane which separates $\{0\}$ and E. To show necessary condition: Define $E = C - D = \{c - d | c \in C, d \in D\}$, by separating hyperplane theorem, we know that for any $x \in C$ and $y \in D$, $\exists a, b$ s.t $a^T x \leq b$ for all $x \in C$ and $a^T y \geq b$ for all $y \in D$, given any $z \in E$, $\exists x, y$ s.t $z = x - y$. So $a^T z = a^T(x - y) \geq b - b = 0$ and $a^T 0 = 0 \leq 0$.
For sufficiency: suppose for any $a \neq 0$ and $z \in E$, we have $a^T z \geq 0$, then choose any $x \in C$ and $y \in D$, there exists $z \in E$ s.t $x - y = z \in E$. So, $a^T z = a^T(x - y) \geq 0$, i.e., $a^T x \geq a^T y$, thus $\sup a^T y \leq \inf a^T x$. Choose $b \in \left[\sup_{y \in D} a^T y, \inf_{x \in c} a^T x\right]$, we get

$$a^T x \geqslant b \geqslant a^T y, \quad \forall x \in C, \quad y \in D,$$

thus finishing the proof.

Second part: We will then prove given a non-empty convex set E with $0 \notin E$ satisfying $\mathbf{dist}(E, 0) > 0$. We will prove it by proving $\mathbf{dist}(\bar{E}, 0) > 0$, otherwise 0 will be the closure point of E and contradicts the assumption. We then define a set with the variable r s.t $\overline{B(0,r)} \cap \bar{E} \neq \varnothing$. So we choose r be large enough and there must exists the point $z \in \overline{B(0,r)} \cap \bar{E}$ satisfies $\| 0 - z \| = \mathbf{dist}(\bar{E}, 0) = \mathbf{dist}(E, 0)$
We then prove it in the framework of the Euclidean metric. (Actually, the proof here is quite similar to the proof in the strict separating hyperplane theorem)
Define a hyperplane that is perpendicular to the line segment between 0 and z, and passes through its midpoint,

$$a = z, \quad b = \frac{\|z\|_2^2}{2}.$$

We will show that the affine function

$$f(x) = a^T x - b = (z - 0)^T (x - (1/2)(z + 0)) = z^T (x - (1/2)z)$$

is nonpositive on $\{0\}$ and nonnegative on E, i.e., that the hyperplane $\{x \mid a^T x = b\}$ separates $\{0\}$ and E.
We first show that f is nonnegative on E and prove it by contradiction. The proof that f is nonpositive on 0 is similar (or follows by swapping 0 and E and considering $-f$).
Suppose there is a point $u \in E$, this is different from point z, so the distance between u and 0 should be larger than 0 and z. By contradiction, for u, we have:

$$f(u) = (z - 0)^T (u - (1/2)(z + 0)) < 0.$$

It is equivalent to express $f(u)$ as

$$f(u) = z^T (u - z + (1/2)z) = z^T (u - z) + (1/2)\|z\|_2^2.$$

This implies $z^T(u - z) < 0$, and this result also implies

$$\frac{d}{dt}\|z + t(u - z) - 0\|_2^2 \Big|_{t=0} = 2z^T(u - z) < 0,$$

so for some small $t > 0$, with $t \leq 1$, we have

$$\|z + t(u - z)\|_2 < \|z - 0\|_2,$$

This shows that there exists a point along the line segment of u and z whose distance to 0 is smaller than z to 0, which contradicts the previous condition before. Thus f is nonnegative on E.

Third part: If $\mathbf{dist}(\bar{E}, 0) = 0$, this gives that 0 is the closure point of E. Because 0 and E are two disjoint sets, so $0 \in \partial E = \bar{E} \backslash E$ we then want to use the result from second part by creating a sequence of open ball with its center being 0 and radius being $r_n = \frac{1}{n}, \quad n = 1, 2, \ldots$, so we could find a sequence $z_n \in B(0, \frac{1}{n}) \backslash \bar{E}$, this also gives us that $\lim z_n = 0$, and $\mathbf{dist}(\bar{E}, z_n) > 0$. So for each z_n, there exists a_n, with $\| a_n \| = 1$ working as the normal vector s.t $a_n^T z \geq a_n^T z_n$, for any $z \in E$. Since $\lim z_n = 0$, we have $\lim a_n^T z_n = 0$, so $a_n^T z \geq 0$.

□

Lemma 2.24. *Let $C \subseteq \mathbb{R}^n$ be a closed convex set and $x \in \mathbb{R}^n$ a point not in C. Then x and C can be strictly separated by a hyperplane.*

Theorem 2.25. *(Converse separating hyperplane theorem) 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.*

Proof. It is obvious to get the separating hyperplane by the two disjoint sets using separating hyperplane theorem. For sufficiency, consider one open set C, and assume that for all $x \in C$, we have $a^T x + b \leq 0$ and all $y \in D$, $a^T y + b \geq 0$, we will show that equality will not hold in $a^T x + b \leq 0$. For some $x_0 \in C$ s.t. $a^T x_0 + b = 0$, for this interior point, $\exists \delta > 0$ s.t $B(x_0, \delta) \subseteq C$, we take $z = x_0 + \frac{\delta}{2} a, a^T z + b = a^T x_0 + \frac{\delta}{2} a^T a + b = \frac{\delta}{2} a^T a > 0$, this gives that two sets must be disjoint. □

Theorem 2.26. *(Supporting hyperplane theorem) for any nonempty convex set C, and at any $x_0 \in \partial C$, there exists a supporting hyperplane to C at x_0.*

Proof. This proof is quite similar to the third part of the proof in separating the hyperplane theorem by creating a series of open balls and sequences and using the limit to give the proof. □

Convex functions

Definition 2.27. *(Convex function) A function $f : R^n \to R$ is convex if $\mathbf{dom} f$ is a convex set and if for all $x, y \in \mathbf{dom} f$, and θ with $0 \leq \theta \leq 1$, we have $f(\theta x + (1 - \theta) y) \leq \theta f(x) + (1 - \theta) f(y)$.*

Definition 2.28. *(Extended-value extension) If f is convex we define its extended-value extension $\tilde{f} : \mathbf{R}^n \to \mathbf{R} \cup \{\infty\}$ by*

$$\tilde{f}(x) = \begin{cases} f(x) & x \in \mathbf{dom} f, \\ \infty & x \notin \mathbf{dom} f. \end{cases}$$

The extension $\tilde{f}$ is defined on all $\mathbf{R}^n$, and takes values in $\mathbf{R} \cup \{\infty\}$. We could say $\mathbf{dom} f = \{x \mid \tilde{f}(x) < \infty\}$.

Proposition 2.29. *Given θ with $0 \leq \theta \leq 1$,we also have $\tilde{f}(\theta x + (1-\theta)y) \leq \theta\tilde{f}(x) + (1-\theta)\tilde{f}(y)$.*

Proof. There are three cases.

1. $x, y \in \mathbf{dom}\, f$, then it must hold.
2. $x \in domf, y \notin domf$ or the inverse case, since the right hand side is ∞, so it still holds.
3. Both $x, y \notin domf$, then the right hand side is still ∞.

□

Proposition 2.30. *(First-order condition) f is convex if and only if* $\mathbf{dom}\, f$ *is convex and for $x, y \in \mathbf{dom}\, f, x \neq y$, we have*

$$f(y) \geq f(x) + \nabla f(x)^T(y-x). \tag{2.1}$$

Similiarly, for concave functions we also have: f is concave if and only if $\mathbf{dom}\, f$ *is convex and*

$$f(y) \leq f(x) + \nabla f(x)^T(y-x),$$

for all $x, y \in \mathbf{dom}\, f$.

Proof. Assume f is convex with $f : \mathbf{R}^n \to \mathbf{R}$. Let $x, y \in \mathbf{R}^n$ and define the function $g(t) = f(ty + (1-t)x)$, so $g'(t) = \nabla f(ty + (1-t)x)^T(y-x)$.
Because f is convex, so g is also convex and it should satisfies: $g(y) \geq g(x) + g'(x)(y-x)$,
let $y = 1$, $x = 0$, we then derive $g(1) \geq g(0) + g'(0)$.
Notice that $g(1) = f(y)$, $g(0) = f(x)$, and $g'(0) = \nabla f(x)^T(y-x)$.
so we get: $f(y) \geq f(x) + \nabla f(x)^T(y-x)$.
To show sufficiency, let $z = \theta x + (1-\theta)y$. So we obtain

$$f(x) \geq f(z) + \nabla f(z)^T(x-z), \quad f(y) \geq f(z) + \nabla f(z)^T(y-z).$$

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.
Second method: Now assume that (2.1) satisfies for any x and y, so if we choose $ty + (1-t)x \in \mathbf{dom}\, f$ and $\tilde{t}y + (1-\tilde{t})x \in \mathbf{dom}\, f$, we will obtain

$$f(ty + (1-t)x) \geq f(\tilde{t}y + (1-\tilde{t})x) + \nabla f(\tilde{t}y + (1-\tilde{t})x)^T(y-x)(t-\tilde{t}),$$

i.e., $g(t) \geq g(\tilde{t}) + g'(\tilde{t})(t-\tilde{t})$. We have seen that this implies that g is convex. □

Proposition 2.31. *(Second-order condition) Assume f is twice differentiable, then f is convex if and only if* $\mathbf{dom}\, f$ *is convex and its Hessian is positive semidefinite: for all $x \in \mathbf{dom}\, f, \nabla^2 f(x) \succeq 0$.*

Proof. We now prove it by first-order condition: we first consider n=1, let $x, y \in \mathbf{dom}\, f$ and $y > x$. We have $f(y) \geq f(x) + f'(x)(y-x)$ and $f(x) \geq f(y) + g'(y)(x-y)$. So we obtain

$$f'(x)(y-x) \leq f(y) - f(x) \leq f'(y)(y-x).$$

Dividing LHS and RHS by $(y-x)^2$ gives

$$\frac{f'(y) - f'(x)}{y-x} \geq 0, \quad \forall x, y, \quad x \neq y.$$

As we let $y \to x$, we obtain

$$f''(x) \geq 0, \quad \forall x \in \mathbf{dom}\, f.$$

For sufficiency, suppose $f''(x) \geq 0, \forall x \in \mathbf{dom}\, f$. By the mean value version of Taylor's theorem we have

$$f(y) = f(x) + f'(x)(y-x) + \frac{1}{2} f''(z)(y-x)^2, \text{ for some } z \in [x, y].$$
$$f(y) \geq f(x) + f'(x)(y-x).$$

Now to establish second order condition relationship in general dimension, we recall that convexity is equivalent to convexity along all lines; i.e., $f : \mathbb{R}^n \to \mathbb{R}$ is convex if $g(\alpha) = f(x_0 + \alpha v)$ is convex $\forall x_0 \in \mathbf{dom}(f)$ and $\forall v \in \mathbb{R}^n$. We just proved this happens iff

$$g''(\alpha) = v^T \nabla^2 f(x_0 + \alpha v)\, v \geq 0,$$

$\forall x_0 \in \mathbf{dom}\, f, \forall v \in \mathbb{R}^n$ and $\forall \alpha$ s.t. $x_0 + \alpha v \in \mathbf{dom}\, f$. Hence, f is convex iff $\nabla^2 f(x) \succeq 0$ for all $x \in \mathbf{dom}\, f$. □

Remark: $x \in \mathbf{dom}\, f$ and $\nabla^2 f(x) > 0$ then f is strictly convex while the reverse does not hold. eg: $f(x) = x^4$.

Example 2.1.1. *(Log-sum-exp) The function $f(x) = log(e^{x_1} + e^{x_2} + \cdots + e^{x_n})$ is convex on R^n*

Proof.

$$\frac{\partial f}{\partial x_i} = \frac{e^{x_i}}{e^{x_1} + \cdots + e^{x_n}},$$
$$\frac{\partial^2 f}{\partial {x_i}^2} = \frac{-e^{x_i} \cdot e^{x_i} + e^{x_i}\left(e^{x_i} + \cdots + e^{x_n}\right)}{\left(e^{x_1} + \cdots + e^{x_n}\right)^2},$$
$$\frac{\partial^2 f}{\partial x_i \partial x_j} = \frac{-e^{x_i} e^{x_j}}{\left(e^{x_1} + \cdots + e^{x_n}\right)^2}.$$

So the Hessian matrix is

$$\nabla^2 f(x) = \frac{1}{(e^{x_1} + \cdots + e^{x_n})^2} \left(\begin{bmatrix} e^{x_1}\left(e^{x_1} + \cdots + e^{x_n}\right) & 0 & \dots & 0 \\ 0 & e^{x_2}\left(e^{x_1} + \cdots + e^{x_n}\right) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{x_n}\left(e^{x_1} + \cdots + e^{x_n}\right) \end{bmatrix} - \begin{bmatrix} e^{x_1} \\ \vdots \\ e^{x_n} \end{bmatrix} \begin{bmatrix} e^{x_1} \\ \vdots \\ e^{x_n} \end{bmatrix}^T \right),$$

define $z = \left[e^{x_1}, \cdots, e^{x_n}\right]^T$, we get the Hessian of the log-sum-exp function is

$$\nabla^2 f(x) = \frac{1}{\left(\mathbf{1}^T z\right)^2} \left(\left(\mathbf{1}^T z\right) \mathbf{diag}(z) - z z^T \right).$$

To verify that $\nabla^2 f(x) \succeq 0$ we must show that for all v, $v^T \nabla^2 f(x) v \geq 0$, i.e.,

$$v^T \nabla^2 f(x) v = \frac{1}{\left(\mathbf{1}^T z\right)^2} \left(\left(\sum_{i=1}^n z_i \right) \left(\sum_{i=1}^n v_i^2 z_i \right) - \left(\sum_{i=1}^n v_i z_i \right)^2 \right) \geq 0. \tag{2.2}$$

If we define $a_i = v_i \sqrt{z_i}, b_i = \sqrt{z_i}$ and by Cauchy-Schwarz inequality $\left(a^T a\right)\left(b^T b\right) \geq \left(a^T b\right)^2$, we obtain that (2.2) always holds. □

Example 2.1.2. *(Geometric mean)* $f(x) = \left(\prod_{i=1}^n x_i\right)^{1/n}$ *is concave on* $\operatorname{dom} f = \mathbf{R}_{++}^n$.

Proof. Its Hessian $\nabla^2 f(x)$ is given by

$$\frac{\partial^2 f(x)}{\partial x_k^2} = -(n-1)\frac{\left(\prod_{i=1}^n x_i\right)^{1/n}}{n^2 x_k^2}, \quad \frac{\partial^2 f(x)}{\partial x_k \partial x_l} = \frac{\left(\prod_{i=1}^n x_i\right)^{1/n}}{n^2 x_k x_l} \quad \text{for } k \neq l,$$

and can be expressed as

$$\nabla^2 f(x) = -\frac{\prod_{i=1}^n x_i^{1/n}}{n^2} \left(n \operatorname{diag}\left(1/x_1^2, \ldots, 1/x_n^2\right) - qq^T \right),$$

where $q_i = 1/x_i$. We must show that $\nabla^2 f(x) \preceq 0$, i.e., that

$$v^T \nabla^2 f(x) v = -\frac{\prod_{i=1}^n x_i^{1/n}}{n^2} \left(n \sum_{i=1}^n v_i^2 / x_i^2 - \left(\sum_{i=1}^n v_i / x_i \right)^2 \right) \leq 0, \tag{2.3}$$

for all v. Again If we define $a_i = 1$ and $b_i = v_i / x_i$, and by Cauchy-Schwarz inequality $\left(a^T a\right)\left(b^T b\right) \geq \left(a^T b\right)^2$, we obtain that (2.3) always holds.

□

Example 2.1.3. *(Log-determinant) The function* $f(X) = \log\det X$ *is concave on* $\operatorname{dom} f = \mathbf{S}_{++}^n$.

Proof. We can verify concavity by considering an arbitrary line, given by $X = Z + tV$, where $Z, V \in \mathbf{S}^n$. We define $g(t) = f(Z + tV)$, and limit g to the range of t values where $Z + tV \succ 0$. We can safely assume that $t = 0$ falls within this range, i.e., $Z \succ 0$, we have

$$\begin{aligned} g(t) &= \log\det(Z + tV) \\ &= \log\det\left(Z^{1/2}\left(I + tZ^{-1/2} V Z^{-1/2}\right) Z^{1/2}\right) \\ &= \sum_{i=1}^n \log\left(1 + t\lambda_i\right) + \log\det Z \end{aligned}$$

Assume $\lambda_1, \cdots \lambda_n$ are the eigenvalues of $Z^{-\frac{1}{2}} V Z^{-\frac{1}{2}}$ and Q is the matrix whose columns are the related eigenvectors.

$$Z^{-\frac{1}{2}} V Z^{-\frac{1}{2}} = Q \Lambda Q^\top, \quad \Lambda = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix}.$$

Thus

$$\begin{aligned} \det\left(I + tz^{-\frac{1}{2}} V z^{\frac{1}{2}}\right) &= \det\left(QQ^\top + tQ\Lambda Q^\top\right) \\ &= \det\left(QQ^\top\right)\det(I + t\Lambda) \\ &= \det(I + t\Lambda). \end{aligned}$$

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 $g''(t) \leq 0$, this gives that f is concave. □

Definition 2.32. *(Sublevel sets) The α-sublevel set of a function $f : \mathbf{R}^n \to \mathbf{R}$ is defined as*

$$C_\alpha = \{x \in \operatorname{dom} f \mid f(x) \leq \alpha\}.$$

Proposition 2.33. *Sublevel sets of a convex function are convex, for any value of α.*

Proof. From the definition of convexity: if $x, y \in C_\alpha$, then $f(x) \leq \alpha$ and $f(y) \leq \alpha$, and so $f(\theta x + (1-\theta)y) \leq \alpha$ for $0 \leq \theta \leq 1$, and hence $\theta x + (1-\theta)y \in C_\alpha$. □

Remark 2.34. *It is interesting that if f is concave, then its α-superlevel set, given by $\{x \in \operatorname{dom} f \mid f(x) \geq \alpha\}$, is still a convex set. So the converse of proposition (2.33) is not true, i.e, a function can have all its sublevel sets convex, but not be a convex function.*

Definition 2.35. *(Epigraph and hypograph) The epigraph of a function $f : \mathbf{R}^n \to \mathbf{R}$ is defined as*

$$epi\ f = \{(x,t) \mid x \in \operatorname{dom} f, f(x) \leq t\},$$

which is a subset of $\mathbf{R}^{n+1}$. Similarly, a function is concave if and only if its hypograph, defined as

$$hypo\ f = \{(x,t) \mid t \leq f(x)\},$$

is a convex set.

Proposition 2.36. *A function f is convex if and only if its epigraph $\operatorname{epi} f$ is a convex set.*

Proof. for epigraph, we have $t \geq f(y) \geq f(x) + \Delta f(x)^T(y-x)$ so we get

$$(y,t) \in \operatorname{epi} f \Longrightarrow \begin{bmatrix} \nabla f(x) \\ -1 \end{bmatrix}^T \left(\begin{bmatrix} y \\ t \end{bmatrix} - \begin{bmatrix} x \\ f(x) \end{bmatrix} \right) \leq 0.$$

Notice that $\begin{bmatrix} \nabla f(x) \\ -1 \end{bmatrix}^T$ is the normal vector for the tangent line of the graph$(x, f(x))$, and when the multiple is less or equal to 0, then the angle is bigger than 90 degree. so it is convex. □

Definition 2.37. *(Jensen's inequality) From the definition of convex function (2.27), we have $f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta)f(y)$, we also call this as Jensen's inequality.*

Remark 2.38. *Jensen's inequality is easily extended to convex combinations of more than two points: If f is convex, $x_1, \ldots, x_k \in \operatorname{dom} f$, and $\theta_1, \ldots, \theta_k \geq 0$ with $\theta_1 + \cdots + \theta_k = 1$, then*

$$f(\theta_1 x_1 + \cdots + \theta_k x_k) \leq \theta_1 f(x_1) + \cdots + \theta_k f(x_k).$$

The inequality for convex functions, like with convex sets, applies to endless sums, integrals, and probable values. For example, if $p(x) \geq 0$ on $S \subseteq \operatorname{dom} f$, $\int_S p(x)dx = 1$, then

$$f\left(\int_S p(x)x\,dx\right) \leq \int_S f(x)p(x)dx,$$

provided the integrals exist.

Proposition 2.39. *(Hölder's inequality) for $p > 1, 1/p + 1/q = 1$, and $x, y \in \mathbf{R}^n$,*

$$\sum_{i=1}^{n} x_i y_i \leq \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p} \left(\sum_{i=1}^{n} |y_i|^q \right)^{1/q}.$$

Proof. We will first prove:

$$a^\theta b^{1-\theta} \leq \theta a + (1-\theta) b, \tag{2.4}$$

If b is zero, then it satisfies.

If b is not zero, suppose $b \geq a > 0$. Dividing both sides of the inequality by b and setting $t = a/b$, we are led to showing that $t^\lambda \leq \lambda t + (1-\lambda)$ with equality $t = 1$. But by elementary calculus, $t^\lambda - \lambda t$ as a function of t is strictly increasing for $t < 1$, so its maximum value $1 - \lambda$ occurs at $t = 1$.

Applying (2.4) 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

$$\left(\frac{|x_i|^p}{\sum_{j=1}^n |x_j|^p} \right)^{1/p} \left(\frac{|y_i|^q}{\sum_{j=1}^n |y_j|^q} \right)^{1/q} \leq \frac{|x_i|^p}{p \sum_{j=1}^n |x_j|^p} + \frac{|y_i|^q}{q \sum_{j=1}^n |y_j|^q}.$$

Then we do sum for both sides of the inequality, we obtain

$$\sum_{i=1} \left(\left(\frac{|x_i|^p}{\sum_{j=1}^n |x_j|^p} \right)^{1/p} \left(\frac{|y_i|^q}{\sum_{j=1}^n |y_j|^q} \right)^{1/q} \right) \leq \sum_{i=1} \left(\frac{|x_i|^p}{p \sum_{j=1}^n |x_j|^p} + \frac{|y_i|^q}{q \sum_{j=1}^n |y_j|^q} \right) = 1.$$

This is equivalent to

$$\sum_{i=1}^{n} x_i y_i \leq \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p} \left(\sum_{i=1}^{n} |y_i|^q \right)^{1/q}.$$

□

Proposition 2.40. *Nonnegative weighted sum could preserve convex, i.e, $f = w_1 f_1 + \cdots + w_m f_m$, where $w_1, w_2, \ldots,$ are nonnegative weight for each $f_i, i = 1, 2, 3, \ldots$ (The proof is similar to prove affine function is convex)*

Remark 2.41. *If $f(x, y)$ is convex in x for each $y \in \mathcal{A}$, and $w(y) \geq 0$ for each $y \in \mathcal{A}$, then the function g defined as*

$$g(x) = \int_{\mathcal{A}} w(y) f(x, y) dy$$

is convex in x (provided the integral exists). Additionally, we can prove it by associated epigraphs. For example, if $w \geq 0$ and f is convex, we have

$$\operatorname{epi}(wf) = \begin{bmatrix} I & 0 \\ 0 & w \end{bmatrix} \ epi\ f$$

Notice that the affine function could preserve convex.

Proposition 2.42. *If f_1 and f_2 are convex functions then their pointwise maximum f, defined by*

$$f(x) = \max \{ f_1(x), f_2(x) \},$$

with $\operatorname{dom} f = \operatorname{dom} f_1 \cap \operatorname{dom} f_2$, is also convex.

Proof. If $0 \leq \theta \leq 1$ and $x, y \in \operatorname{dom} f$, then

$$\begin{aligned} f(\theta x + (1-\theta)y) &= \max\{f_1(\theta x + (1-\theta)y), f_2(\theta x + (1-\theta)y)\} \\ &\leq \max\{\theta f_1(x) + (1-\theta)f_1(y), \theta f_2(x) + (1-\theta)f_2(y)\} \\ &\leq \theta \max\{f_1(x), f_2(x)\} + (1-\theta)\max\{f_1(y), f_2(y)\} \\ &= \theta f(x) + (1-\theta)f(y), \end{aligned}$$

□

Remark 2.43. *We can still use proposition* (2.42) *to show that if $f_1, \ldots, f_m$ are convex, then their pointwise maximum*

$$f(x) = \max\{f_1(x), \ldots, f_m(x)\}$$

is also convex by induction.

Proposition 2.44. *(Pointwise supremum property) Given an infinite set $\mathcal{A}$ of convex functions. If for each $y \in \mathcal{A}$, $f(x, y)$ is convex in x, then the function g, defined as*

$$g(x) = \sup_{y \in \mathcal{A}} f(x, y)$$

is convex in x with its domain

$$\operatorname{dom} g = \left\{ x \mid (x, y) \in \operatorname{dom} f \text{ for all } y \in \mathcal{A}, \sup_{y \in \mathcal{A}} f(x, y) < \infty \right\}.$$

Similarly, the pointwise infimum of a set of concave functions is a concave function.

Proof. We now use epigraphs to prove it, define $g(x) = \sup_{y \in \mathcal{A}} f(x, y)$ so its epigraph should be the intersection:

$$\operatorname{epi} g = \bigcap_{y \in \mathcal{A}} \operatorname{epi} f(\cdot, y).$$

Since the intersection of convex sets is still convex, so $\operatorname{epi} g$ is convex and this gives that g is also convex. □

Theorem 2.45. *(Composition of convex functions property) Given conditions on $h : \mathbf{R}^k \to \mathbf{R}$ and $g : \mathbf{R}^n \to \mathbf{R}^k$, their composition $f = h \circ g : \mathbf{R}^n \to \mathbf{R}$, defined by*

$$f(x) = h(g(x)), \quad \operatorname{dom} f = \{x \in \operatorname{dom} g \mid g(x) \in \operatorname{dom} h\}$$

could preserve the convexity or concavity if it satisfies:

1. *f is convex if h is convex and nondecreasing, and g is convex.*
2. *f is convex if h is convex and nonincreasing, and g is concave.*
3. *f is concave if h is concave and nondecreasing, and g is concave.*
4. *f is concave if h is concave and nonincreasing, and g is convex.*

Remark 2.46. *We could extend the domain by using extended-value extension.*

Proof. We will prove the second composition theorem (the proofs of the others is similar): f is convex if h is convex and nonincreasing, and g is concave.
Assume that $x, y \in \operatorname{dom} f$, we have $x, y \in \operatorname{dom} g$. Since g is concave, the domain is still convex i.e. $\theta x + (1-\theta)y \in \operatorname{dom} g$, with $0 \leq \theta \leq 1$. For the domain of h, we have that $g(x), g(y) \in \operatorname{dom} h$, and h is convex, so the $\theta g(x) + (1-\theta)g(y)$ is still in the domain of h so by using the property of concavity, we have

$$g(\theta x + (1-\theta)y) \geq \theta g(x) + (1-\theta)g(y).$$

Since h is nonincreasing. It gives that

$$h(g(\theta x + (1-\theta)y)) \leq h(\theta g(x) + (1-\theta)g(y)). \tag{2.5}$$

Additionally, $\theta x + (1-\theta)y \in \operatorname{dom} f$ and it is convex. From convexity of h, we have

$$h(\theta g(x) + (1-\theta)g(y)) \leq \theta h(g(x)) + (1-\theta)h(g(y)). \tag{2.6}$$

Combining (2.5), (2.6), we finally get that f is convex. □

Proposition 2.47. *(Minimization property of convex function) If f is convex in (x, y), and C is a convex nonempty set, then the function*

$$g(x) = \inf_{y \in C} f(x, y)$$

is convex in x.

Proof. For $x_1, x_2 \in \operatorname{dom} g$, let $\epsilon > 0$, then there exist $y_1, y_2 \in C$ such that $f(x_i, y_i) \leq g(x_i) + \epsilon$ for $i = 1, 2$. Now let $\theta \in [0, 1]$. We have

$$\begin{aligned} g(\theta x_1 + (1-\theta)x_2) &= \inf_{y \in C} f(\theta x_1 + (1-\theta)x_2, y) \\ &\leq f(\theta x_1 + (1-\theta)x_2, \theta y_1 + (1-\theta)y_2) \\ &\leq \theta f(x_1, y_1) + (1-\theta) f(x_2, y_2) \text{ (by using } f \text{ is convex)} \\ &\leq \theta g(x_1) + (1-\theta) g(x_2) + \epsilon. \end{aligned}$$

Since this inequality always holds for any $\epsilon > 0$, we obtain

$$g(\theta x_1 + (1-\theta)x_2) \leq \theta g(x_1) + (1-\theta) g(x_2).$$

□

Proposition 2.48. *(The perspective operation preserves convexity) If $f : \mathbf{R}^n \to \mathbf{R}$ is convex , then the perspective of f is the function $g : \mathbf{R}^{n+1} \to \mathbf{R}$ defined by*

$$g(x, t) = tf(x/t),$$

with domain

$$\operatorname{dom} g = \{(x, t) \mid x/t \in \operatorname{dom} f, t > 0\}.$$

is also convex.

Proof. We use epigraphs and the perspective mapping on $\mathbf{R}^{n+1}$ to make a short proof. For $t > 0$ we have

$$\begin{aligned}(x,t,s) \in \operatorname{epi} g &\iff tf(x/t) \le s \\ &\iff f(x/t) \le s/t \\ &\iff (x/t, s/t) \in \operatorname{epi} f.\end{aligned}$$

Therefore epi g is the inverse image of epi f under the perspective mapping that takes (u,v,w) to $(u,w)/v$. Since perspective mapping could preserve the convexity and epi$f$ is convex, so we get that epi g is convex, and this implies that the function g is convex. □

Definition 2.49. *(Conjugate Function)Let $f : \mathbf{R}^n \to \mathbf{R}$. The conjugate function $f^* : \mathbf{R}^n \to \mathbf{R}$, is defined as*

$$f^*(y) = \sup_{x \in \operatorname{dom} f} \left(y^T x - f(x)\right).$$

Proposition 2.50. *Conjugate function is convex.*

Proof. From the definition, The conjugate function $f^* : \mathbf{R}^n \to \mathbf{R}$, is

$$f^*(y) = \sup_{x \in \operatorname{dom} f} \left(y^T x - f(x)\right).$$

Because $y^T x$ is a linear function with respect to y, and by pointwise supremum property, $f^*(y)$ is convex □

Proposition 2.51. *(Fenchel's inequality) for all x, y, we have*

$$f(x) + f^*(y) \ge x^T y.$$

This is called Fenchel's inequality (or Young's inequality when f is differentiable).

For example: with $f(x) = (1/2)x^T Q x$, where $Q \in \mathbf{S}^n_{++}$, we obtain the inequality

$$x^T y \le (1/2)x^T Q x + (1/2)y^T Q^{-1} y.$$

Proposition 2.52. *Conjugate of the conjugate of a convex and closed function is the original function. That is, if f is convex, and f is closed, then $f^{**} = f$.*

Proof. We first prove $f^{**} \le f$
By definition, $f^{**}(x) = \sup_{y \in \operatorname{dom} f}(x^T y - f^*(y))$, by Fenchel's inequality, $f(x) + f^*(y) \ge x^T y$ i.e. $f(x) \ge x^T y - f^*(y)$ for all x, y, so $f^{**} \le f$.
we then prove $f^{**} \ge f$
By contradiction, suppose $(x, f^{**}(x)) \notin$ epi f; then by lemma (2.24), there is a strict separating hyperplane:

$$\begin{bmatrix} a \\ b \end{bmatrix}^T \begin{bmatrix} z - x \\ s - f^{**}(x) \end{bmatrix} \le c < 0 \quad \forall (z,s) \in \operatorname{epi} f$$

$$a^T(z - x) + b(s - f^{**}(x)) \le c < 0.$$

If $b > 0$, then choose s large enough, the above inequality will not hold. So $b \le 0$.
if $b < 0$, take $y = a/(-b)$ we get

$$y^T(z - x) - s + f^{**}(x)) \le \frac{c}{-b} < 0$$

and consider the maximization over $(z,s) \in \operatorname{epi} f$: given $s \geq f(z)$ and $y^T x \leq f^*(y)$, we get:

$$f^*(y) - y^T x + f^{**}(x) \leq c/(-b) < 0$$

this contradicts Fenchel's inequality.
If $b = 0$, we choose $\hat{y} \in \operatorname{dom} f^*$ and add small error terms of $(\hat{y}, -1)$ to (a,b) :
for $\epsilon > 0$ small enough, we obtain

$$\begin{bmatrix} a + \epsilon\hat{y} \\ -\epsilon \end{bmatrix}^T \begin{bmatrix} z - x \\ s - f^{**}(x) \end{bmatrix} \leq c + \epsilon \left(f^*(\hat{y}) - x^T\hat{y} + f^{**}(x) \right) < 0.$$

Then we can apply the arguments in the case of $b < 0$ to derive the desired result. □

Proposition 2.53. *The following properties of conjugate functions hold:*

1. *Separable sum*

$$\text{If } f(x_1, x_2) = g(x_1) + h(x_2), \text{ then } f^*(y_1, y_2) = g^*(y_1) + h^*(y_2).$$

2. *Scalar multiplication: (for $\alpha > 0$)*

$$\text{If } f(x) = \alpha g(x), \text{ then } f^*(y) = \alpha g^*(y/\alpha).$$

3. *Addition to affine function*

$$\text{If } f(x) = g(x) + a^T x + b, \text{ then } f^*(y) = g^*(y - a) - b.$$

4. *Infimal convolution*

$$\text{If } f(x) = \inf_{u+v=x}(g(u) + h(v)), \text{ then } f^*(y) = g^*(y) + h^*(y).$$

Definition 2.54. *(Quasiconvex function) A function $f : \mathbf{R}^n \to \mathbf{R}$ is called quasiconvex (or unimodal) if its domain and all its sublevel sets*

$$S_\alpha = \{x \in \operatorname{dom} f \mid f(x) \leq \alpha\},$$

for $\alpha \in \mathbf{R}$, are convex. Additionally, f is quasiconcave if $-f$ is quasiconvex, i.e., every superlevel set $\{x \mid f(x) \geq \alpha\}$ is convex.

Proposition 2.55. *A function f is quasiconvex if and only if $\operatorname{dom} f$ is convex and for any $x, y \in \operatorname{dom} f$ and $0 \leq \theta \leq 1$,*

$$f(\theta x + (1-\theta)y) \leq \max\{f(x), f(y)\},$$

i.e., the value of the function on a segment does not exceed the maximum of its values at the endpoints.

Proof. Assume f is quasiconvex, and we choose any $x, y \in dom f, x \neq y$, and we choose $\alpha = \max\{f(x), f(y)\}$, so this sublevel sets must contain both the point x, y. Since it is convex, it should contain the line segment $\theta x + (1-\theta)y$, so we get $f(\theta x + (1-\theta)y) \leq \max\{f(x), f(y)\}$. For sufficiency, choose any α to be our sublevel set, let $f(x), f(y) \leq \alpha$. We obtain when $0 \leq \theta \leq 1$,

$$f(\theta x + (1-\theta)y) \leq \max\{f(x), f(y)\} \leq \alpha.$$

Since it is convex and $\operatorname{dom} f$ is also convex, we say f is quasiconvex. □

Proposition 2.56. *(First-order condition) Suppose $f : \mathbf{R}^n \to \mathbf{R}$ is differentiable. Then f is quasiconvex if and only if* $\mathbf{dom}\, f$ *is convex and for all $x, y \in \mathbf{dom}\, f$*

$$f(y) \leq f(x) \Longrightarrow \nabla f(x)^T (y - x) \leq 0.$$

Proof. Assume f is quasiconvex with $f : \mathbf{R}^n \to \mathbf{R}$. Let $x, y \in \mathbf{R}^n$ and define the function $g(t) = f(ty + (1-t)x)$, so $g'(t) = \nabla f(ty + (1-t)x)^T (y - x)$.
Since f is quasiconvex, g is also quasiconvex and it should satisfy:
$\mathbf{dom}\, g$ is convex and for any $x, y \in \mathbf{dom}\, g$ and $0 \leq \theta \leq 1$,

$$g(\theta x + (1-\theta)y) \leq \mathbf{max}\{g(x), g(y)\}$$

Assume $g(y) \leq g(x)$, and let $y = 1, x = 0$, we then derive $g(1) = f(y)$, $g(0) = f(x)$, and for any point $z \in [0,1]$, since g is quasiconvex, we have $g(z) \leq g(0)$. Since g is differentiable, it gives that $g'(0) = \nabla f(x)^T (y - x) \leq 0$.
For sufficiency, given $f(y) \leq f(x)$ and let $\lambda(t) = tx + (1-t)y, t \in [0,1]$. Since x, y is a line segment, we can find the maximum of $f(x)$ i.e. $t_0 = argmax f(\lambda(t))$ given that the continuous image of compact set is compact. So we have$f(\lambda(t)) \leq f(\lambda(t_0))$, by letting $\lambda(t_0) = z$, we obtain

$$\nabla f(\lambda(t))^T (z - \lambda(t)) \leq 0.$$

Notice that $z - \lambda(t) = t_0 x + (1 - t_0)y - tx - (1-t)y = (t - t_0)(x - y)$, so we can write

$$f(x) - f(z) = \int_{t_0}^{1} \nabla f(\lambda(t))^T \cdot (x - y) dt = \int_{t_0}^{1} \nabla f(\lambda(t))^T \cdot \frac{z - \lambda(t)}{t_0 - t} dt \geq 0,$$

given that $t_0 - t$ here is less than 0 and $\nabla f(\lambda(t))^T (z - \lambda(t)) \leq 0$.
This shows that $f(tx + (1-t)y) \leq \mathbf{max}\{f(x), f(y)\} = f(x), t \in [0,1]$, and thus f is quasiconvex. □

Proposition 2.57. *(Second-order condition) If f is quasiconvex, then for all $x \in \mathbf{dom}\, f$, and all $y \in \mathbf{R}^n$, we have*

$$y^T \nabla f(x) = 0 \Longrightarrow y^T \nabla^2 f(x) y \geq 0.$$

Proof. We first consider this in dimension $n = 1$, i.e, $f : \mathbf{R} \to \mathbf{R}$.
We now show that if $f : \mathbf{R} \to \mathbf{R}$ is quasiconvex on an interval (a, b), then it must satisfy $y^T \nabla f(x) = 0 \Longrightarrow y^T \nabla^2 f(x) y \geq 0$. , i.e., if $f'(c) = 0$ for some $c \in (a, b)$, then $f''(c) \geq 0$ must hold. Otherwise, for small positive ϵ we have $f(c - \epsilon) < f(c)$ and $f(c + \epsilon) < f(c)$. this will gives 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.
For sufficiency, given $m \in (a, b)$ with $f'(m) = 0$, $f''(c) > 0$ always holds, so f is strictly increasing. Therefore f can cross the value 0 at most once. If $f' \neq 0$ on (a, b), then f is either nonincreasing or nondecreasing on (a, b), and therefore quasiconvex. If $f' = 0$ on (a, b), there is only one solution, say, $f'(m) = 0$, $m \in (a, b)$ Since $f''(m) > 0$, we must have that $f'(t) \leq 0$ for $a < t \leq m$, and $f'(t) \geq 0$ for $m \leq t < b$. The above two conditions both show that f must be quasiconvex. □

Proposition 2.58. *In the following cases, the quasiconvexity is preserved.*

1. *(Scalar) If $f(x)$ is quasiconvex and $w > 0$, then $g(x) = wf(x)$ is also quasiconvex.*
2. *(Composition) $g(x) = h(f(x))$ is quasiconvex if $f : R^n \to R$ is quasiconvex and $h : R \to R$ is nondecreasing.*
3. *(Minimization) If f is quasiconvex in (x, y), and C is a convex nonempty set, then the function*
$$g(x) = \inf_{y \in C} f(x, y)$$
is quasiconvex in x.
4. *(Pointwise Supremum) If $f(x)$ is quasiconvex, and A is a convex nonempty set, then $g(x) = \sup_{y \in \mathcal{A}} f(x, y)$ is also quasiconvex.*

Proof. 1. Let $S_a = \{x \mid g(x) \leq wa\}, x_1, x_2 \in S_a$, and $0 \leq \theta \leq 1$.

$$\begin{aligned} g(\theta x + (1-\theta)y) &= wf(\theta x + (1-\theta)y) \\ &\leq w\theta f(x) + w(1-\theta)f(y) \\ &\leq w\theta a + w(1-\theta)a = wa. \end{aligned}$$

2. Let $S_a = \{x \mid g(x) \leq a\}, x_1, x_2 \in S_a$, and $0 \leq \theta \leq 1$.

$$\begin{aligned} g(f(\theta x + (1-\theta)y)) &= h(f(\theta x + (1-\theta)y)) \\ &\leq h(\theta f(x) + (1-\theta)f(y)) \\ &\leq \theta h(f(x)) + (1-\theta)h(f(y)) \leq a. \end{aligned}$$

   Thus, S_a is convex and $g(x)$ is quasiconvex.

3. For $x_1, x_2 \in \operatorname{dom} g$, let $\epsilon > 0$. Since f is quasiconvex, then there exist $y_1, y_2 \in C$ and the maximum value a such that $f(x_i, y_i) \leq a + \epsilon$ for $i = 1, 2$. Now let $\theta \in [0, 1]$. We have

$$\begin{aligned} g(\theta x_1 + (1-\theta)x_2) &= \inf_{y \in C} f(\theta x_1 + (1-\theta)x_2, y) \\ &\leq f(\theta x_1 + (1-\theta)x_2, \theta y_1 + (1-\theta)y_2) \\ &\leq a + \epsilon. \end{aligned}$$

   Since it holds for any $\epsilon > 0$, we obtain $g(\theta x_1 + (1-\theta)x_2) \leq a$.

4. The proof is similar to 3.

□

Definition 2.59. *(Log-concave function) Given function $f : R^n \to R$ with $f(x) > 0$ for all $x \in \operatorname{dom} f$ and $\log f(x)$ is concave. For $x, y \in \operatorname{dom} f$ and $0 \leq \theta \leq 1$ then*

$$f(\theta x + (1-\theta)y) = f(x)^\theta f(y)^{1-\theta}.$$

Proposition 2.60. *(Twice differentiable log-convex/concave functions) Suppose f is twice differentiable, with $\operatorname{dom} f$ convex, then*

$$\nabla^2 \log f(x) = \frac{1}{f(x)} \nabla^2 f(x) - \frac{1}{f(x)^2} \nabla f(x) \nabla f(x)^T.$$

We conclude that f is log*-convex if and only if for all $x \in \operatorname{dom} f$,*

$$f(x)\nabla^2 f(x) \succeq \nabla f(x)\nabla f(x)^T,$$

and log*-concave if and only if for all $x \in \operatorname{dom} f$,*

$$f(x)\nabla^2 f(x) \preceq \nabla f(x)\nabla f(x)^T.$$

2.2 Convex optimization problems

In this section, we will introduce the basic definition for convex optimization problems and provide examples of classic optimization problems such as LP, QCQP, SOCP. We will also discuss an important algorithm for solving quasiconvex optimization problems.

Definition 2.61. *We use the notation*

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \textit{subject to} & f_i(x) \le 0, \quad i = 1, \dots, m \\ & h_i(x) = 0, \quad i = 1, \dots, p \end{array} \tag{2.7}$$

to describe the problem of finding an x that minimizes $f_0(x)$ among all x that satisfy the conditions $f_i(x) \le 0, i = 1, \dots, m$, and $h_i(x) = 0, i = 1, \dots, p$.

Here are some notations and definitions which will be used in later part.

Notation	*Definition*
Optimization variable	*is the variable x satisfying $x \in \mathbf{R}^n$*
Objective function	*is the function f_0 satisfying $f_0 : \mathbf{R}^n \to \mathbf{R}$.*
The inequality constraint functions	*are functions $f_i, i = 1, \dots, m$ satisfying $f_i : \mathbf{R}^n \to \mathbf{R}$.*
The equality constraint functions	*are functions $h_i, i = 1, \dots, p$ satisfying $h_i : \mathbf{R}^n \to \mathbf{R}$.*
The domain of optimization problems	*is set of points for which the objective and all constraint functions are defined $\mathcal{D} = \bigcap_{i=0}^{m} \operatorname{dom}, f_i \cap \bigcap_{i=1}^{p} \operatorname{dom} h_i$.*
Feasible point	*is the point if it satisfies the constraints $f_i(x) \le 0, i = 1, \dots, m$, and $h_i(x) = 0, i = 1, \dots, p$.*
The feasible set	*is the set of all feasible points.*
The optimal value of the problem	*is the optimal value$p^\star$satisfying $p^\star = \inf\{f_0(x) \mid f_i(x) \le 0, i = 1, \dots, m, h_i(x) = 0, i = 1, \dots, p\}$.*
Optimal point	*is the point $x^\star$ if it is feasible and $f_0(x^\star) = p^\star$.*
Optimal set	*is the set of all optimal points:$X_{\text{opt}} = \{x \mid f_i(x) \le 0, i = 1, \dots, m, h_i(x) = 0, i = 1, \dots, p, f_0(x) = p^\star\}$.*
ϵ-suboptimal	*a feasible point x is ϵ-suboptimal with $f_0(x) \le p^\star + \epsilon$.*
Local optimal	*A feasible point x is local optimal if there is an $R > 0$ such that $f_0(x) = \inf\{f_0(z) \mid f_i(z) \le 0, i = 1, \dots, m, h_i(z) = 0, i = 1, \dots, p, \|z - x\|_2 \le R\}$.*

Definition 2.62. *We say a problem with the form* (2.7) *also satisfies*

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \textit{subject to} & f_i(x) \le 0, \quad i = 1, \dots, m \\ & a_i^T x = b_i, \quad i = 1, \dots, p \end{array}$$

is convex optimization problem, where $f_0, \ldots, f_m$ are convex functions.

Remark 2.63. *If f_0 is quasiconvex, then it will be a quasiconvex optimization problem.*

Theorem 2.64. *For a convex optimization problem, any locally optimal point is also (globally) optimal.*

Proof. To see this, suppose that x is locally optimal for a convex optimization problem, assume x is feasible and

$$f_0(x) = \inf\{f_0(z) \mid z \text{ feasible}, \|z - x\|_2 \leq R\},$$

for some $R > 0$. Now suppose that x is not globally optimal, and there is another feasible y which is global optimal such that $f_0(y) < f_0(x)$, with $\|y - x\|_2 > R$. Consider the point z given by

$$z = (1-\theta)x + \theta y, \quad \theta = \frac{R}{2\|y - x\|_2}.$$

Then we have $\|z - x\|_2 = R/2 < R$, and by convexity of the feasible set, z is feasible. By convexity of f_0 we have

$$f_0(z) \leq (1-\theta)f_0(x) + \theta f_0(y) < f_0(x)$$

which contradicts the definition of local optimal. Hence for convex optimal problems, locally optimal is globally optimal. □

Definition 2.65. *A quasiconvex optimization problem is one of the form*

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \textit{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & a_i^T x = b_i, \quad i = 1, \ldots, p, \end{array}$$

where $f_1, \ldots, f_m$ are convex functions, but f_0 is quasiconvex.

Definition 2.66. *The standard form of linear programming is:*

$$\begin{array}{ll} \text{minimize} & c^T x \\ \textit{subject to} & Ax = b \\ & x \succeq 0. \end{array}$$

We now give a classical example of LP problems.

Example 2.2.1.

$$\begin{array}{ll} \text{minimize} & c^T x + d \\ \textit{subject to} & Gx + s = h \\ & Ax = b \\ & s \succeq 0. \end{array} \tag{2.8}$$

Remark 2.67. *We could use slack variable to make* (2.8) *a standard form:*

$$\begin{array}{ll} \text{minimize} & c^T x^+ - c^T x^- + d \\ \textit{subject to} & Gx^+ - Gx^- + s = h \\ & Ax^+ - Ax^- = b \\ & x^+ \succeq 0, \quad x^- \succeq 0, \quad s \succeq 0, \end{array}$$

Example 2.2.2. *The problem of minimizing a ratio of affine functions over a polyhedron is called a linear-fractional program:*

$$\begin{array}{ll} \textit{minimize} & f_0(x) \\ \textit{subject to} & Gx \preceq h \\ & Ax = b, \end{array}$$

where the objective function is given by

$$f_0(x) = \frac{e^T x + d}{e^T x + f}, \quad \operatorname{dom} f_0 = \left\{ x \mid e^T x + f > 0 \right\}.$$

The objective function is quasiconvex, but we can transform it to a linear program like

$$\begin{array}{ll} \textit{minimize} & c^T y + dz \\ \textit{subject to} & Gy - hz \preceq 0 \\ & Ay - bz = 0 \\ & e^T y + fz = 1 \\ & z \geq 0, \end{array}$$

with variables y, z.

Proof. To show the equivalence, we first note that if x is feasible in LP problem:

$$y = \frac{x}{e^T x + f}, \quad z = \frac{1}{e^T x + f}$$

$$\begin{cases} Gx \leqslant h & Gy - hz = \frac{Gx-h}{e^T x+f} \leqslant 0, \\ Ax = b & Ay - bz = \frac{Ax-b}{e^T x+f} = 0, \\ e^T x + f > 0 & e^T x + fz = 1, \\ & z \geqslant 0, \end{cases}$$

and the combination satisfy the function $f_0(x) = \frac{e^T x+d}{e^T x+f}$ Additionally, to show y, z are feasible in linear fractional program:
if $z > 0$, let $x = \frac{y}{z}$, then it solves the problem.
if $z = 0$, assume x_0 is a feasible point, so we get $x = x_0 + ty$ is also feasible for any $t \geq 0$.

$$\begin{aligned} & Gy \leqslant 0. Ay = 0, e^T y = 1, \\ & Gx = Gx_0 + tGy \leqslant h, \\ & Ax = Ax_0 + tAy = 1, \\ & e^T x + f = e^T x_0 + f + te^\top y > 0, \\ & f_0(x) = f_0(x_0 + ty) = \frac{c^T x_0 + c^T + y + d}{e^T x + e^T + y + f} \overset{t\to\infty}{=} C^T y. \end{aligned}$$

□

Definition 2.68. *(QP) The convex optimization problem is called a quadratic program (QP) if the objective function is (convex) quadratic, and the constraint functions are affine. A quadratic*

program can be expressed in the form

$$\begin{array}{ll} \textit{minimize} & (1/2)x^TPx + q^Tx + r \\ \textit{subject to} & Gx \preceq h \\ & Ax = b, \end{array}$$

where $P \in \mathbf{S}_+^n, G \in \mathbf{R}^{m\times n}$, *and* $A \in \mathbf{R}^{p\times n}$.

Definition 2.69. *(QCQP) The problem is called a quadratic constrained quadratic program if it satisfies the form:*

$$\begin{array}{ll} \text{minimize} & (1/2)x^TP_0x + q_0^Tx + r_0 \\ \textit{subject to} & (1/2)x^TP_ix + q_i^Tx + r_i \le 0, \quad i = 1,\dots,m \\ & Ax = b, \end{array}$$

where $P_i \in \mathbf{S}_+^n, i = 0, 1\dots, m$.

Definition 2.70. *(SOCP) Second-order cone program has the form :*

$$\begin{array}{ll} \text{minimize} & f^Tx \\ \textit{subject to} & \|A_ix + b_i\|_2 \le c_i^Tx + d_i, \quad i = 1,\dots,m \\ & Fx = g, \end{array}$$

where $x \in \mathbf{R}^n$ *is the optimization variable,* $A_i \in \mathbf{R}^{n_i\times n}$, *and* $F \in \mathbf{R}^{p\times n}$. *We call a constraint of the form*

$$\|Ax + b\|_2 \le c^Tx + d,$$

where $A \in \mathrm{R}^{k\times n}$, *a second-order cone constraint, since it is the same as requiring the affine function* $(Ax + b, c^Tx + d)$ *to lie in the second-order cone in* $\mathbf{R}^{k+1}$.

Definition 2.71. *(Robust linear constraint) Consider a linear program in inequality form:*

$$\begin{array}{ll} \textit{minimize} & c^Tx \\ \textit{subject to} & a_i^Tx \le b_i, \quad i = 1,\dots,m, \end{array}$$

Assume that c *and* b_i *are fixed, and that* a_i *are known to lie in given ellipsoids:*

$$a_i \in \mathcal{E}_i = \{\bar{a}_i + P_iu \mid \|u\|_2 \le 1\},$$

where $P_i \in \mathbf{R}^{n\times n}$. *(If* P_i *is singular we obtain 'flat' ellipsoids, of dimension rank* P_i*;* $P_i = 0$ *means that* a_i *is known perfectly.)*
We will require that the constraints be satisfied for all possible values of the parameters a_i, *which leads us to the robust linear program*

$$\begin{array}{ll} \textit{minimize} & c^Tx \\ \textit{subject to} & a_i^Tx \le b_i \textit{ for all } a_i \in \mathcal{E}_i, \quad i = 1,\dots,m. \end{array}$$

So, for the constraint, we change it into:

$$\begin{aligned} \sup\left\{a_i^Tx \mid a_i \in \mathcal{E}_i\right\} &= \bar{a}_i^Tx + \sup\left\{u^TP_i^Tx \mid \|u\|_2 \le 1\right\} \\ &= \bar{a}_i^Tx + \left\|P_i^Tx\right\|_2. \end{aligned}$$

Thus, the original problem becomes:

$$\bar{a}_i^T x + \left\| P_i^T x \right\|_2 \leq b_i,$$

which is evidently a second-order cone constraint. Hence the robust LP can be expressed as the SOCP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \textit{subject to} & \bar{a}_i^T x + \|P_i^T x\|_2 \leq b_i, \quad i = 1, \ldots, m. \end{array}$$

Numerical analysis for quasiconvex optimization problem

Given a quasiconvex optimization problem, we could build some relationship with convex feasibility problems by changing the object function and use bisection idea in numerical analysis to solve it.

To be more specific, since both quasiconvex function and convex function have the property that their sublevel sets are convex. We make equivalent relationship that

$$f_0(x) \leq t \Longleftrightarrow \phi_t(x) \leq 0,$$

since the original problem requires us to find the minimum of $f_0(x)$, it is euivalent to use bisection method to find the first x s.t $f_0(x) \leq t$ or we say find the first x satisfies $\phi_t(x) \leq 0$.

So the question becomes

$$\begin{array}{ll} \text{find} & x \\ \text{subject to} & \phi_t(x) \leq 0 \\ & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & Ax = b. \end{array}$$

We now provide bisection to solve a convex feasibility problem at each step. We assume that the problem is feasible, and start with an interval $[a, b]$ known to contain the optimal value $p^\star$. We then solve the convex feasibility problem at its midpoint $t = (a + b)/2$, to determine the optimal value's position in the interval, we divide it into two halves. We then update the interval based on which half the optimal value is in. This process is repeated until the interval becomes narrow enough.

Algorithm 1: Bisection method for quasiconvex optimization.

```
1: a ≤ p*, b ≥ p*, tolerance ε > 0.
2: while b − a ≥ ε do
3:     t := (a + b)/2
4:     Solve the convex feasibility problem
5:     if the convex problem is feasible then
6:         b := t
7:     else
8:         a := t
9:     end if
10: end while
```

2.3 Vector optimization

This section focuses on multivariable optimization. Unlike previous optimization problems that only had one objective function, multivariable optimization involves multiple objective functions. If there is no common optimal point for all the objective functions, finding a solution becomes more challenging. We now introduce a new idea to help us make the comparison.

Definition 2.72. *(Proper Cone K) A cone $K \subseteq \mathbf{R}^n$ is called a proper cone if it satisfies the following:*

1. *K is convex.*
2. *K is closed.*
3. *K is solid, which means it has nonempty interior.*
4. *K is pointed, which means that it contains no line (or equivalently, $x \in K, -x \in K \Longrightarrow x = 0$).*

We now use proper cone K to define a generalized inequality for vectors , which is a partial ordering on $\mathbf{R}^n$, we define it by

$$x \preceq_K y \Longleftrightarrow y - x \in K.$$

Example 2.3.1. *For nonnegative orthant $\mathbb{R}^n_+ = \{\mathbf{x} \in \mathbb{R}^n \mid x_i \geq 0, i = 1, \ldots, n\}$, we have:*

$$\mathbf{x} \succeq_{\mathbb{R}^n_+} \mathbf{y} \Leftrightarrow \mathbf{x} - \mathbf{y} \in \mathbb{R}^n_+ \text{ or } \mathbf{x} \geq \mathbf{y}.$$

Example 2.3.2. *For positive semidefinite cone $\mathbb{S}^n_+ = \{\mathbf{X} \in \mathbb{S}^n \mid \mathbf{X} \succeq 0\}$, we have:*

$$\mathbf{X} \succeq_{\mathbb{S}^n_+} \mathbf{Y} \Leftrightarrow \mathbf{X} - \mathbf{Y} \in \mathbb{S}^n_+ \text{ or } \mathbf{X} \succeq \mathbf{Y}.$$

Definition 2.73. *(Minimum element) We say that $x \in S$ is the minimum element of S (with respect to the generalized inequality $\preceq_K$) if for every $y \in S$ we have $x \preceq_K y$.*

Proposition 2.74. *Based on the definition, we get the property that:*

1. *A point $x \in S$ is the minimum element of S if and only if*

$$S \subseteq x + K.$$

Here $x + K$ denotes all the points that are comparable to x and greater than or equal to x (according to $\preceq_K$).

2. *A point $x \in S$ is a minimal element if and only if*

$$(x - K) \cap S = \{x\}.$$

Here $x - K$ denotes all the points that are comparable to x and less than or equal to x (according to $\preceq_K$); the only point in common with S is x.

Proof. 1. Choose any point $y \in S$, we obtain $x \preceq_K y \Longleftrightarrow y - x \in K \Longleftrightarrow y \in x + K$, so $S \subseteq x + K$. For sufficiency, since $S \subseteq x + K$, choose any point $y \in S$, we have $y \in K + x \Longleftrightarrow y - x \in K$.

2. Since x is the minimum point in S, so all the points in S is equal or larger than x, and $x - K$ denotes all the points that are comparable to x and less than or equal to x (according to $\preceq_K$), so the intersection is the only point x.

□

Definition 2.75. *Suppose $K \subseteq \mathbf{R}^n$ is a proper cone with associated generalized inequality $\preceq_K$. A function $f : \mathbf{R}^n \to \mathbf{R}$ is called K-nondecreasing if*

$$x \preceq_K y \Longrightarrow f(x) \leq f(y),$$

and K-increasing if

$$x \preceq_K y, x \neq y \Longrightarrow f(x) < f(y).$$

Proposition 2.76. *A differentiable function f, with convex domain, is K-nondecreasing if and only if*

$$\nabla f(x) \succeq_{K^*} 0$$

for all $x \in \mathbf{dom}\, f$. And if

$$\nabla f(x) \succ_{K^*} 0$$

for all $x \in \mathbf{dom}\, f$, then f is K-increasing.

Proof. Assume that f satisfies $\nabla f(x) \succeq_{K^*} 0$ for all x, but is not K-nondecreasing, i.e., there exist x, y with $x \preceq_K y$ and $f(y) < f(x)$. By differentiability of f there exists a $t \in [0,1]$ with

$$\frac{d}{dt} f(x + t(y - x)) = \nabla f(x + t(y - x))^T (y - x) < 0.$$

Since $y - x \in K$ this means

$$\nabla f(x + t(y - x)) \notin K^*,$$

which contradicts our assumption.
Similarly we could prove the condition for K-increasing. □

Definition 2.77. *Suppose $K \subseteq \mathbf{R}^m$ is a proper cone with associated generalized inequality $\preceq_K$. We say $f : \mathbf{R}^n \to \mathbf{R}^m$ is K-convex if for all x, y, and $0 \leq \theta \leq 1$,*

$$f(\theta x + (1 - \theta)y) \preceq_K \quad \theta f(x) + (1 - \theta) f(y).$$

Definition 2.78. *(Vector optimization problem) We define the vector optimization problem if the problem has the form:*

$$\begin{array}{ll} \textit{minimize (with respect to } K) & f_0(x) \\ \textit{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p, \end{array}$$

with $x \in \mathbf{R}^n$ is the optimization variable, $K \subseteq \mathbf{R}^q$ is a proper cone, $f_0 : \mathbf{R}^n \to \mathbf{R}^q$ is the objective function, $f_i : \mathbf{R}^n \to \mathbf{R}$ are the inequality constraint functions, and $h_i : \mathbf{R}^n \to \mathbf{R}$ are the equality constraint functions.

Proposition 2.79. *Consider the set of objective values of feasible points,*

$$\mathcal{O} = \{f_0(x) \mid \exists x \in \mathcal{D}, f_i(x) \leq 0, i = 1, \ldots, m, h_i(x) = 0, i = 1, \ldots, p\} \subseteq \mathbf{R}^q,$$

which is called the set of achievable objective values. A point $x^\star$ is optimal if and only if it is feasible and

$$\mathcal{O} \subseteq f_0(x^\star) + K.$$

Proof. The proof is almost the same as the proof for the minimal element property. □

Definition 2.80. *(Pareto optimal) A point x is Pareto optimal if and only if it is feasible and*

$$(f_0(x) - K) \cap \mathcal{O} = \{f_0(x)\}.$$

Proposition 2.81. *A vector optimization problem can have many Pareto optimal values (and points). The set of Pareto optimal values, denoted $\mathcal{P}$, satisfies*

$$\mathcal{P} \subseteq \mathcal{O} \cap \partial\mathcal{O}.$$

Proof. Choose any x is Pareto optimal, $(f_0(x) - K) \cap \mathcal{O} = \{f_0(x)\}$, we need to show $f_0(x) \in \partial\mathcal{O}$. For any $\epsilon > 0$, the open ball $B(f_0(x), \epsilon) \cap \mathcal{O} \neq \emptyset$. So we then need to show $f_0(x)$ is not an interior point. Since the set $(f_0(x) - K)$ is a closed cone, we define a closed ball $\bar{B}(f_0(x), k)$ with its center at $f_0(x)$. Then define $D = \bar{B}(f_0(x), k) \cup (f_0(x) - K)$, D is closed and bounded, then $f_0(x) \in D$ is feasible and it is also the accumulation point, so we have $B(f_0(x), \epsilon) \cap ((f_0(x) - K) \backslash f_0(x)) \neq \emptyset$, so $f_0(x) \in \partial\mathcal{O}$, so $\mathcal{P} \subseteq \mathcal{O} \cap \partial\mathcal{O}$. □

Definition 2.82. *(Muticriterion optimization)When a vector optimization problem involves the cone $K = \mathbf{R}^q_+$, it is called a multicriterion or multi-objective optimization problem. And we change the vector optimization problem form into*

$$\begin{array}{ll} \text{minimize} & F_j(x) \\ \textit{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p, \end{array}$$

for $j = 1, \ldots, q$, where $F_j(x)$ represent jth objective of the problem.

Proposition 2.83. *If in vector optimization problem, the object function is convex, then the scalarization is the same as original problem.*

Proof. We first prove the for any x is Pareto optimal in scalarization, x will also be Pareto optimal in original problem. We prove it by contradiction. The scalarization leads to the following problems:

$$\begin{array}{ll} \text{minimize} & \lambda^T f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{array}$$

with $0 \preceq_K \lambda$.

If x is Pareto optimal for scalarization but is not Pareto optimal for original problem, then there exist a feasible y, satisfying $f_0(y) \preceq_K f_0(x)$, and $f_0(x) \neq f_0(y)$. Since $f_0(x) - f_0(y) \succeq_K 0$ and is nonzero, we have $\lambda^T (f_0(x) - f_0(y)) > 0$, i.e., $\lambda^T f_0(x) > \lambda^T f_0(y)$. This contradicts the

assumption that x is optimal for the scalar problem.
Second, we prove x is Pareto optimal in original problem, x will also be Pareto optimal in scalarization problem. By proposition (2.40) , we know that nonnegative weighted sum could preserve convex. So for each fixed vector λ, since it is convex, we could find the according optimal x for the scalar problem. If this x is not the Pareto optimal for original problem, there must exist $y \in B(x, \epsilon)$ s.t $f_0(y)$ is better than $f_0(x)$. □

Remark 2.84. *From the proof, we can see that by scalarization, we can always find subsets of Pareto optimals. And we find all Pareto optimals, if and only if function is convex.*

2.4 Dual property

In this section, we will further discuss the method to solve optimization problems by dual property. Duality theory is one of the most important part to optimization. We will present the classical Karush-Kuhn-Tucker (KKT) conditions for optimality, Slater's condition for strong duality, alternative system and Farka's lemma.

Definition 2.85. *Given an optimization problem in the standard form:*

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \textit{subject to} & f_i(x) \le 0, \quad i = 1, \dots, m \\ & h_i(x) = 0, \quad i = 1, \dots, p \end{array}$$

with variable $x \in \mathbf{R}^n$. We assume its domain $\mathcal{D} = \bigcap_{i=0}^m \operatorname{dom} f_i \cap \bigcap_{i=1}^p \operatorname{dom} h_i$ is nonempty, and denote the optimal value by $p^\star$. The basic idea in Lagrangian duality is to take the constraints into account by augmenting the objective function with a weighted sum of the constraint functions. We define the Lagrangian $L : \mathbf{R}^n \times \mathbf{R}^m \times \mathbf{R}^p \to \mathbf{R}$ as

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x) \tag{2.9}$$

with $\operatorname{dom} L = \mathcal{D} \times \mathbf{R}^m \times \mathbf{R}^p$. We refer to λ_i as the Lagrange multiplier associated with the i th inequality constraint $f_i(x) \le 0$; 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

Definition 2.86. *We define the Lagrange dual function (or just dual function) $g : \mathbf{R}^m \times \mathbf{R}^p \to \mathbf{R}$ as the minimum value of the Lagrangian over x : for $\lambda \in \mathbf{R}^m, \nu \in \mathbf{R}^p$,*

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x) \right). \tag{2.10}$$

Notice that Lagrange dual function is always concave since it is the infimum of linear functions.

Proposition 2.87. *The dual function yields lower bounds on the optimal value $p^\star$: For any $\lambda \succeq 0$ and any ν we have*

$$g(\lambda, \nu) \le p^\star.$$

Proof. Suppose $\tilde{x}$ is a feasible point, i.e., $f_i(\tilde{x}) \leq 0$ and $h_i(\tilde{x}) = 0$, and $\lambda \succeq 0$. 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, therefore

$$L(\tilde{x}, \lambda, \nu) = f_0(\tilde{x}) + \sum_{i=1}^{m} \lambda_i f_i(\tilde{x}) + \sum_{i=1}^{p} \nu_i h_i(\tilde{x}) \leq f_0(\tilde{x}). \tag{2.11}$$

Hence

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) \leq L(\tilde{x}, \lambda, \nu) \leq f_0(\tilde{x}).$$

□

Remark 2.88. *Recall the definition of conjugate function* (2.49)*, it is easy to find that conjugate function and Lagrange dual function are closely related. It's natural to use conjugate function to express Lagrange dual functions.*

Example 2.4.1. *Consider an optimization problem with linear inequality and equality constraints,*

$$\begin{array}{ll} \textit{minimize} & f_0(x) \\ \textit{subject to} & Ax \preceq b \\ & Cx = d. \end{array}$$

Using the conjugate of f_0 we can write the dual function for the problem as

$$\begin{aligned} g(\lambda, \nu) &= \inf_x \left(f_0(x) + \lambda^T (Ax - b) + \nu^T (Cx - d) \right) \\ &= -b^T \lambda - d^T \nu + \inf_x \left(f_0(x) + \left(A^T \lambda + C^T \nu \right)^T x \right) \\ &= -b^T \lambda - d^T \nu - f_0^* \left(-A^T \lambda - C^T \nu \right). \end{aligned}$$

The domain of g follows from the domain of f_0^* :

$$\operatorname{dom} g = \left\{ (\lambda, \nu) \mid -A^T \lambda - C^T \nu \in \operatorname{dom} f_0^* \right\}$$

Example 2.4.2. *We now consider the problem*

$$\begin{array}{ll} \textit{minimize} & \|x\| \\ \textit{subject to} & Ax = b, \end{array}$$

where $\|\cdot\|$ is any norm. We first prove the conjugate of $f_0 = \|\cdot\|$ is

$$f_0^*(y) = \begin{cases} 0 & \|y\|_* \leq 1 \\ \infty & \textit{otherwise} \end{cases}$$

and then use this to express the Lagrange dual function.
By definition, $f^* = \sup_x (y^T x - ||x||)$, when $\|y\|_* \leq 1$: By definition of dual norm, we have $y^T x \leq \|x\| \|y\|_* \leq \|x\|$ for all x. So $\sup_x (y^T x - ||x||) = 0$ when $x = 0$.
When $\|y\|_* \geq 1$: there exists an $\tilde{x}$ with $\|\tilde{x}\| \leq 1$ and $y^T \tilde{x} = \|y\|_* > 1$, hence $\|\tilde{x}\| - \|y\|_* < 0$. We then consider $x = t\tilde{x}$ with $t > 0$:

$$y^T x - \|x\| = t \left(\|y\|_* - \|\tilde{x}\| \right) \to \infty \quad \textit{as } t \to \infty.$$

We then derive Lagrangian is $\|x\| + v^T(Ax - b)$. *Thus dual function is now*

$$g(v) = \inf_{x\in\mathcal{D}} \|x\| + v^T(Ax - b) = -b^T v + \sup_{x\in\mathcal{D}}(-\|x\| - (A^T v)^T x)$$
$$= -b^T v - f_0^*\left(-A^T v\right) = \begin{cases} -b^T v & \|A^T v\|_* \le 1 \\ -\infty & otherwise. \end{cases}$$

Definition 2.89. *(Lagrange dual problem) Remember previously we find the Lagrange dual function for original optimization problem. We then find the maximum value of this Lagrange dual function with its only constraint* $\lambda \succeq 0$. *We say if the problem is Lagrange dual problem if it has the form*

$$\begin{array}{ll} maximize & g(\lambda, v) \\ subject\ to & \lambda \succeq 0. \end{array}$$

Example 2.4.3. *We now use a simple example with the original problem:*

$$\begin{array}{ll} \text{minimize} & c^T x \\ subject\ to & Ax \preceq b. \end{array}$$

The Lagrangian is

$$L(x, \lambda) = c^T x + \lambda^T(Ax - b) = -b^T\lambda + \left(A^T\lambda + c\right)^T x,$$

so the dual function is

$$g(\lambda) = \inf_x L(x, \lambda) = -b^T\lambda + \inf_x \left(A^T\lambda + c\right)^T x.$$

The infimum of a linear function is $-\infty$, *except in the special case when it is identically zero, so the dual function is*

$$g(\lambda) = \begin{cases} -b^T\lambda & A^T\lambda + c = 0 \\ -\infty & otherwise. \end{cases}$$

The dual variable λ *is dual feasible if* $\lambda \succeq 0$ *and* $A^T\lambda + c = 0$. *The Lagrange dual problem is to maximize* g *over all* $\lambda \succeq 0$*:*

$$\begin{array}{ll} maximize & -b^T\lambda \\ subject\ to & A^T\lambda + c = 0 \\ & \lambda \succeq 0, \end{array} \tag{2.12}$$

Remark 2.90. *For LP problems, the dual of the dual problem is the original problem. We will then show it below.*

(3.30) is an equivalent problem to

$$\begin{array}{ll} \text{minimize} & b^T\lambda \\ \text{subject to} & A^T\lambda + c = 0 \\ & -\lambda \preceq 0, \end{array}$$

The Lagrangian is

$$L(\lambda, p, v) = b^T\lambda - p^T\lambda + v^T(A^T\lambda + c) = b^T\lambda - p^T\lambda + (Av)^T\lambda + cv^T,$$

with $p \geq 0$. So the dual function is

$$g(p,\nu) = \inf_{\lambda} L(\lambda, p, \nu) = c^T\nu + \inf_{\lambda}(A\nu + b - p)^T\lambda.$$

The infimum of a linear function is $-\infty$, except in the special case when it is identically zero, then the dual function is

$$g(p,\nu) = \begin{cases} c^T\nu & A\nu + b - p = 0 \\ -\infty & \text{otherwise} \end{cases}$$

The dual variable λ is dual feasible if $p \succeq 0$ and $A\nu + b - p = 0$. The Lagrange dual problem is to maximize g over all $p \succeq 0$:

$$\begin{array}{ll} \text{maximize} & c^T\nu \\ \text{subject to} & A\nu + b - p = 0 \\ & p \succeq 0. \end{array}$$

It is equivalent to:

$$\begin{array}{ll} \text{minimize} & -c^T\nu \\ \text{subject to} & A\nu \succeq -b. \end{array}$$

If we take $x = -\nu$, we get the same problem in the form:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b, \end{array}$$

which is equivalent to the original problem.

Proposition 2.91. *(Weak duality) If we denote the maximum value of a dual problem by d' and the minimum value of the original problem by p'. Then $d' \leq p'$.*

Proof. It is same as the proposition (2.87). □

Definition 2.92. *(Strong duality) If $d' = p'$, we say the strong duality holds.*

Definition 2.93. *(Relative Interior Domain) The relative Interior Domain is:* $\operatorname{relint} D = \{x \in D | B(x,r) \cap \operatorname{aff} D \subseteq D, \exists r > 0\}$.

Definition 2.94. *(Slater's condition) There exists an $x \in \operatorname{relint} D$ such that*

$$f_i(x) < 0, \quad i = 1, \ldots, m, \quad Ax = b.$$

Proposition 2.95. *If Slater's condition holds (and the problem is convex), then we will have strong duality. Additionally, if the first k constraint functions $f_1, \ldots, f_k$ are affine, then strong duality holds provided the following weaker condition holds: there exists an $x \in \operatorname{relint} D$ with*

$$f_i(x) \leq 0, \quad i = 1, \ldots, k, \quad f_i(x) < 0, \quad i = k+1, \ldots, m, \quad Ax = b.$$

Or we say, Slater's condition could be refined by affine function.

Remark 2.96. *If LP is feasible, then it always holds Slater's condition, and thus always satisfying strong duality (More details will be shown in proposition*(2.116)*).*

We now show a strange example that if both original problem and its dual problem are infeasible, then there will be no strong duality for LP.

Example 2.4.4. *Consider the example*

$$\begin{array}{ll} \text{minimize} & x \\ \textit{subject to} & \begin{bmatrix} 0 \\ 1 \end{bmatrix} x \preceq \begin{bmatrix} -1 \\ 1 \end{bmatrix}. \end{array}$$

we will have $p^\star = \infty$ and for its dual problem:

$$\begin{array}{ll} \text{maximum} & -b^T\lambda \\ \textit{subject to} & I + A^T\lambda = 0 \\ & \lambda \succeq 0, \end{array}$$

where now the optimal solution is $d^\star = -\infty$.

Example 2.4.5. *We now consider another famous example about entropy maximization problem:*

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^n x_i \log x_i \\ \textit{subject to} & Ax \preceq b \\ & 1^T x = 1, \end{array}$$

with domain $\mathcal{D} = \mathbf{R}^n_+$.
The Lagrangian is

$$L(x, \lambda, \nu) = \sum_{i=1}^n x_i \log x_i + \lambda^T(Ax - b) + \nu^T(x - 1),$$

so the dual function is

$$g(\lambda, \nu) = \inf_x L(x, \lambda, \nu) = -b^T\lambda - \nu + \inf_x \left(\sum_{i=1}^n x_i \log x_i + \lambda^T A x + \nu^T x \right).$$

We build its relationship with conjugate function. Notice the conjugate function for $f_0(x) = \sum_{i=1}^n x_i \log x_i$ is $f^*(y) = \sum_{i=1}^n e^{y_i - 1}$. We will now use the result in example (2.4.1). So we could write it as

$$g(\lambda, \nu) = -b^T\lambda - \nu - e^{-\nu - 1} \sum_{i=1}^n e^{-a_i^T \lambda}.$$

The dual problem is

$$\begin{array}{ll} \textit{maximize} & -b^T\lambda - \nu - e^{-\nu-1} \sum_{i=1}^n e^{-a_i^T\lambda} \\ \textit{subject to} & \lambda \succeq 0, \end{array}$$

with variables $\lambda \in \mathbf{R}^m, \nu \in \mathbf{R}$. Additionally, we can simplify this dual problem, for fixed λ, by letting $g'(\nu) = 0$, we get

$$\nu = \log \sum_{i=1}^n e^{-a_i^T\lambda} - 1.$$

Substituting this optimal value of ν into the dual problem gives

$$\begin{array}{ll} \textit{maximize} & -b^T\lambda - \log \left(\sum_{i=1}^n e^{-a_i^T\lambda} \right) \\ \textit{subject to} & \lambda \succeq 0, \end{array}$$

The weak Slater's condition tells us that the strong duality holds.

Example 2.4.6. *One rare occasions strong duality can be obtained for a nonconvex problem. As an important example, we consider the problem of minimizing a nonconvex quadratic function over the unit ball,*

$$\begin{array}{ll} \textit{minimize} & x^T A x + 2b^T x \\ \textit{subject to} & x^T x \leq 1, \end{array}$$

where $A \in \mathbf{S}^n, A \not\succeq 0$, and $b \in \mathbf{R}^n$. Since $A \not\succeq 0$, this is not a convex problem. The Lagrangian is

$$L(x, \lambda) = x^T A x + 2b^T x + \lambda \left(x^T x - 1\right) = x^T (A + \lambda I) x + 2b^T x - \lambda.$$

We have

$$\begin{aligned} L(x + \Delta x, \lambda) &= (x + \Delta x)^T (A + \lambda I)(x + \Delta x) + 2b^T (x + \Delta x) - \lambda \\ &= x^T (A + \lambda I) x + x^T (A + \lambda I) \Delta x + \Delta x^T (A + \lambda I) x + \Delta x^T (A + \lambda I) \Delta x + 2b^T x + 2b^T \Delta x \\ &= L(x, \lambda) + x^T ((A + \lambda I) + (A + \lambda I)^T) \Delta x + 2b^T \Delta x + \mathcal{O}(\Delta x^2) \\ &= L(x, \lambda) + 2x^T (A + \lambda I) \Delta x + 2b^T \Delta x + \mathcal{O}(\Delta x^2) \\ &= L(x, \lambda) + \nabla L(x, \lambda)^T \Delta x. \end{aligned}$$

For simplicity, we define $P = (A + \lambda I)$. If $P \not\succeq 0$, the Lagrange function L is unbounded below: choose y with $y^T P y < 0$ and $x = ty$, so

$$L(x, \lambda) = t^2 \left(y^T P y\right) + 2t \left(b^T y\right) - \lambda \to -\infty \quad \textit{if } t \to \pm\infty.$$

If $P \geq 0$, decompose q as $b = Pu + v$ with $u = P^\dagger q$ and $v = (I - PP^\dagger) b$. Pu is projection of q on $\mathcal{R}(P), v$ is projection on nullspace of P.

- *If $v \neq 0$ (i.e., $b \notin \mathcal{R}(P)$), the Lagrange function L is unbounded below: for $x = -tv$, so*

$$L(x, \lambda) = t^2 \left(v^T P v\right) - 2t \left(b^T v\right) - \lambda = -2t\|v\|^2 + r \to -\infty \quad \textit{if } t \to \infty.$$

- *If $v = 0, x^* = -u$ is optimal since L is convex and $\nabla L(x, \lambda) = 2Px^* + 2b = 0$, so*

$$L(x^*, \lambda) = -b^T P^\dagger b - \lambda.$$

Thus the dual function is given by

$$g(\lambda) = \begin{cases} -b^T (A + \lambda I)^\dagger b - \lambda & A + \lambda I \succeq 0, \quad b \in \mathit{Range}(A + \lambda I) \\ -\infty & \textit{otherwise}, \end{cases}$$

where $(A + \lambda I)^\dagger$ is the pseudo-inverse of $A + \lambda I$. The Lagrange dual problem is thus

$$\begin{array}{ll} \textit{maximize} & -b^T (A + \lambda I)^\dagger b - \lambda \\ \textit{subject to} & A + \lambda I \succeq 0, \quad b \in \mathit{Range}(A + \lambda I), \end{array}$$

with variable $\lambda \in \mathbf{R}$. Since A is symmetric, we could write

$$A + \lambda I = QDQ^T + \lambda QQ^T = [q_1, q_2, \ldots, q_n] \begin{bmatrix} \lambda_1 + \lambda & & & \\ & \lambda_2 + \lambda & & \\ & & \ddots & \\ & & & \lambda_n + \lambda \end{bmatrix} \begin{bmatrix} q_1^T \\ \vdots \\ q_n^T \end{bmatrix}.$$

where λ_i and Q are the eigenvalues and corresponding (orthonormal) eigenvectors matrix of A. To derive pseudo-inverse, we only need to take the reciprocal of all non-zero elements for the diagonal matrix :

$$(A+\lambda I)^{-1} = [q_1, q_2, \ldots, q_n] \begin{bmatrix} \frac{1}{\lambda_1+\lambda} & & & \\ & \frac{1}{\lambda_2+\lambda} & & \\ & & \ddots & \\ & & & \frac{1}{\lambda_n+\lambda} \end{bmatrix} \begin{bmatrix} q_1^T \\ \vdots \\ q_n^T \end{bmatrix}.$$

So we can write the problem in the form:

$$\begin{array}{ll} \textit{maximize} & -\sum_{i=1}^{n} \left(q_i^T b\right)^2 / \left(\lambda_i + \lambda\right) - \lambda \\ \textit{subject to} & \lambda \geq -\lambda_{\min}(A), \end{array}$$

We interpret $\left(q_i^T b\right)^2 / 0$ as 0 if $q_i^T b = 0$ and as ∞ otherwise.

Geometry interpretation

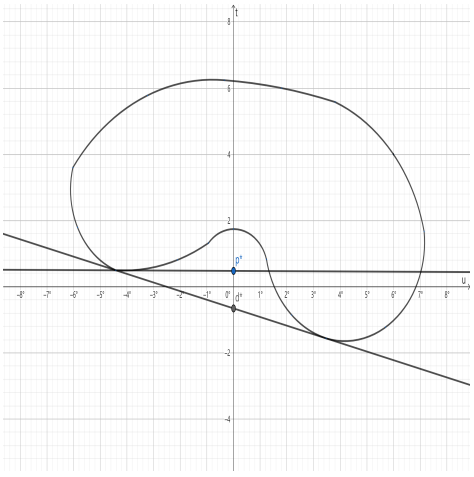
Consider the domain for optimal problem and we now give the graph satisfying all the domain.

$$\mathcal{G} = \left\{ \left(f_1(x), \ldots, f_m(x), h_1(x), \ldots, h_p(x), f_0(x)\right) \in \mathbf{R}^m \times \mathbf{R}^p \times \mathbf{R} \mid x \in \mathcal{D} \right\}.$$

The optimal value $p^\star$ for this can be expressed in terms of $\mathcal{G}$ satisfying the constraint as

$$p^\star = \inf\{t \mid (u, v, t) \in \mathcal{G}, u \preceq 0, v = 0\}.$$

For example, see the graph below, we get the optimal solution p^* by drawing the parallel line to u axis. And we also say that this line is the supporting hyperplane to this special constraint of $\mathcal{G}$ with $u \preceq 0, v = 0$.



For dual function at (λ, ν), we minimize the affine function

$$(\lambda, \nu, 1)^T (u, v, t) = \sum_{i=1}^{m} \lambda_i u_i + \sum_{i=1}^{p} \nu_i v_i + t$$

over $(u, v, t) \in \mathcal{G}$, i.e., we have

$$g(\lambda, \nu) = \inf \left\{ (\lambda, \nu, 1)^T (u, v, t) \mid (u, v, t) \in \mathcal{G} \right\}.$$

If the infimum is finite, then the inequality

$$(\lambda, \nu, 1)^T (u, v, t) \geq g(\lambda, \nu)$$

defines a supporting hyperplane to $\mathcal{G}$.
Now suppose $\lambda \succeq 0$. Then, obviously, $t \geq (\lambda, \nu, 1)^T (u, v, t)$ if $u \preceq 0$ and $v = 0$. Therefore

$$\begin{aligned} p^\star &= \inf\{t \mid (u, v, t) \in \mathcal{G}, u \preceq 0, v = 0\} \\ &= \inf\left\{(\lambda, \nu, 1)^T (u, v, t) \mid (u, v, t) \in \mathcal{G}, u \preceq 0, v = 0\right\} \\ &\geq \inf\left\{(\lambda, \nu, 1)^T (u, v, t) \mid (u, v, t) \in \mathcal{G}\right\} \\ &= g(\lambda, \nu). \end{aligned}$$

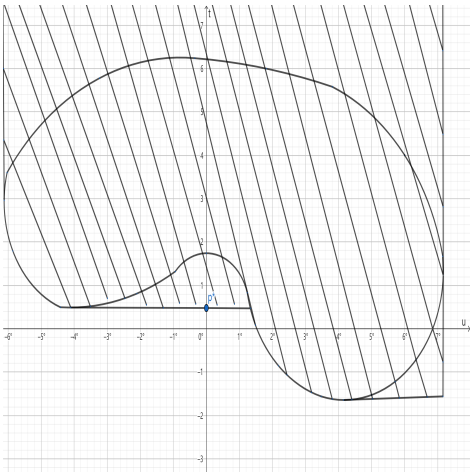
Definition 2.97. *(Epigraph for $\mathcal{G}$) We define the set $\mathcal{A} \subseteq \mathbf{R}^m \times \mathbf{R}^p \times \mathbf{R}$ as*

$$\mathcal{A} = \mathcal{G} + \left(\mathbf{R}_+^m \times \{0\} \times \mathbf{R}_+\right),$$

or, more explicitly,

$$\begin{aligned} \mathcal{A} = \{(u, v, t) \mid \exists x \in \mathcal{D}, f_i(x) \leq u_i, i = 1, \ldots, m \\ h_i(x) = v_i, i = 1, \ldots, p, f_0(x) \leq t\}. \end{aligned}$$

We can also say that this epigraph is defined by using the general inequality by the convex cone. For example, the epigraph for $\mathcal{G}$ is now shown below.



Multi-criterion interpretation
We take $\tilde{\lambda} = (\lambda, 1)$ for the $F(x) = (f_1(x), \ldots, f_m(x), f_0(x))$ Thus, in scalarization we minimize the function

$$\tilde{\lambda}^T F(x) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x),$$

which is exactly the Lagrangian form.
Recall that we discuss the multi-criterion optimal problems in vector optimization section, the epigraph set here $\mathcal{A}$ is exactly the same as

$$\mathcal{A} = \left\{t \in \mathbf{R}^{m+1} \mid \exists x \in \mathcal{D}, f_i(x) \leq t_i, i = 0, \ldots, m\right\}.$$

So we say that in processing Lagrange dual problem steps, the first step to get the infinity of the Lagrange dual function is exactly the same as find Pareto optimal points and the second step is to choose the best λ will give us just some Pareto optimal points based on some λ combinations.

Remark 2.98. *Recall that not all the Pareto optimal points can be obtained by scalarization, details are shown before in* (2.83) .

Saddle point interpretation

To introduce this interpretation, we first introduce a famous inequality called Min-Max inequality.

Definition 2.99. *(Min-max inequality) We call the inequality as max-min inequality if it satisfies*

$$\sup_{z\in Z}\inf_{w\in W} f(w,z) \le \inf_{w\in W}\sup_{z\in Z} f(w,z)$$

for any $f : \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}$ *(and any* $W \subseteq \mathbf{R}^n$ *and* $Z \subseteq \mathbf{R}^m$ *).*

Proof. It is obvious that $\inf_{w\in W} f \le \sup_{z\in Z} f$, and for any $z \in Z$, this inequality holds. So we then write it as

$$\sup_{z\in Z}\inf_{w\in W} f \le \sup_{z\in Z} f.$$

Choose infimum of w for both sides of this inequality, we will obtain

$$\inf_{w\in W}\sup_{z\in Z}\inf_{w\in W} f \le \inf_{w\in W}\sup_{z\in Z} f. \tag{2.13}$$

Notice that the left hand side of (2.13) is equal to $\sup_{z\in Z}\inf_{w\in W} f$, so we get this inequality. □

Recall that in the standard form of the problem, we have the constraints $f_i(x) \le 0$, and we first write Lagrangian as (2.9), so if we consider $\sup_{\lambda\succeq 0} L(x,\lambda)$, we have

$$\begin{aligned}\sup_{\lambda\succeq 0} L(x,\lambda) &= \sup_{\lambda\succeq 0}\left(f_0(x) + \sum_{i=1}^{m}\lambda_i f_i(x)\right)\\ &= \begin{cases} f_0(x) & f_i(x)\le 0, \quad i=1,\dots,m\\ \infty & \text{otherwise.}\end{cases}\end{aligned}$$

Thus we say the optimal point is

$$p^\star = \inf_x \sup_{\lambda\succeq 0} L(x,\lambda).$$

Considering its dual function (2.10), we also obtain

$$d^\star = \sup_{\lambda\succeq 0}\inf_x L(x,\lambda).$$

Thus, we can describe weak duality as the weak min-max inequality

$$\sup_{\lambda\succeq 0}\inf_x L(x,\lambda) \le \inf_x \sup_{\lambda\succeq 0} L(x,\lambda)$$

and strong duality as the equality

$$\sup_{\lambda\succeq 0}\inf_x L(x,\lambda) = \inf_x \sup_{\lambda\succeq 0} L(x,\lambda).$$

Strong duality means that the order of the minimization over x and the maximization over $\lambda \succeq 0$ can be switched without changing the outcome.

Proposition 2.100. *If Slater's condition holds (and the problem is convex), then we will have strong duality.*

We now use the supporting hyperplane to prove it.

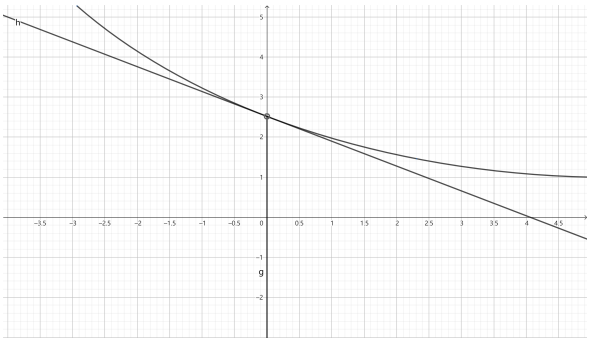
Proof. We first use the graph to give the idea to form the proof. Let $\mathcal{A}$ be the upper epigraph denoted as

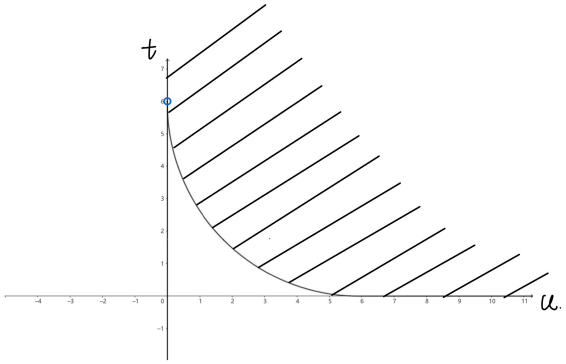
$$\mathcal{A} = \{(u, v, t) \mid \exists x \in \mathcal{D}, f_i(x) \leq u_i, i = 1, \ldots, m \\ h_i(x) = v_i, i = 1, \ldots, p, f_0(x) \leq t\}.$$

and $\mathcal{B}$ denoted as

$$\mathcal{B} = \{(0, 0, s) \in \mathbf{R}^m \times \mathbf{R}^p \times \mathbf{R} \mid s < p^\star\}.$$

If $\mathcal{A}$ is convex and there exist point satisfying $f_i(x) < 0$, then we must find a hyperplane to separate $\mathcal{A}$ and $\mathcal{B}$ and this hyperplane passes the optimal point p^*. And if $\mathcal{A}$ does not hold Slater's condition, it is still possible to find such kind of hyperplane, (as long as part of $\mathcal{A}$ is not an arc tangent to t axis, thus the hyperplane is vertical to u axis). This also shows that Slater's condition is a necessary condition for strong duality. These two conditions are shown in the two pictures below:





Consider the optimal problem, with $f_0, \ldots, f_m$ convex, and assume Slater's condition holds: There exists $\tilde{x} \in \mathsf{relint}\,\mathcal{D}$ with $f_i(\tilde{x}) < 0, i = 1, \ldots, m$, and $A\tilde{x} = b$. In order to simplify the proof, we make two additional assumptions: first that $\mathcal{D}$ has nonempty interior (hence, we have the same dimension of interior and relative interior, i.e, $\mathsf{relint}\,\mathcal{D} = \mathsf{int}\,\mathcal{D}$) and second, that $\mathsf{rank}\,A = p$(If A is not of row full rank, say, $\mathsf{rankA} = l < p$, we choose sub-matrix of A called A' such that $\mathsf{rankA}' = l$, and solve the problem with this A' matrix). We assume that $p^\star$ is finite. (Since there is a feasible point, we can only have $p^\star = -\infty$ or $p^\star$ finite; if $p^\star = -\infty$, then

$d^\star = -\infty$ by weak duality).

Since problem is convex, we derive the epigraph A is convex, we will then first prove the sets $\mathcal{A}$ and $\mathcal{B}$ do not intersect by using contradiction. Suppose $(u,v,t) \in \mathcal{A} \cap \mathcal{B}$, since $(u,v,t) \in \mathcal{B}$ we have $u = 0, v = 0$, and $t < p^\star$. Since $(u,v,t) \in \mathcal{A}$, there exists an x with $f_i(x) \leq 0, i = 1,\ldots,m, Ax - b = 0$, and $f_0(x) \leq t < p^\star$, which is impossible since $p^\star$ is the optimal value of the primal problem.

According to separating hyperplane theorem, we derive that there exists $(\tilde{\lambda}, \tilde{v}, \mu) \neq 0$ and α such that

$$(u,v,t) \in \mathcal{A} \Longrightarrow \tilde{\lambda}^T u + \tilde{v}^T v + \mu t \geq \alpha,$$

and

$$(u,v,t) \in \mathcal{B} \Longrightarrow \tilde{\lambda}^T u + \tilde{v}^T v + \mu t \leq \alpha.$$

This concludes that $\tilde{\lambda} \succeq 0$ and $\mu \geq 0$, (otherwise if we choose u, t very large, then the larger inequality will not hold). And we take $(0,0,t)$ in $\mathcal{B}$ to get that $\mu t \leq \alpha$ for all $t < p^\star$, thus, $\mu p^\star \leq \alpha$. So in general, we conclude that for any $x \in \mathcal{D}$

$$\sum_{i=1}^{m} \tilde{\lambda}_i f_i(x) + \tilde{v}^T (Ax - b) + \mu f_0(x) \geq \mu p^\star.$$

Assume that $\mu > 0$. We divide both sides of the inequality by μ and obtain

$$L(x, \tilde{\lambda}/\mu, \tilde{v}/\mu) \geq p^\star$$

for all $x \in \mathcal{D}$, from which it follows, by minimizing over x, that $g(\lambda, \nu) \geq p^\star$, where we define

$$\lambda = \tilde{\lambda}/\mu, \quad \nu = \tilde{v}/\mu.$$

By weak duality we have $g(\lambda, \nu) \leq p^\star$, so we get $g(\lambda, \nu) = p^\star$. This shows that strong duality holds when $\mu > 0$.

Now consider the case $\mu = 0$. This means that for all $x \in \mathcal{D}$,

$$\sum_{i=1}^{m} \bar{\lambda}_i f_i(x) + \tilde{v}^T (Ax - b) \geq 0.$$

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 $\tilde{\lambda} = 0$. Notice that the normal vector $(\tilde{\lambda}, \tilde{v}, \mu) \neq 0$ and $\tilde{\lambda} = 0, \mu = 0$, we conclude that $\tilde{v} \neq 0$. Then this requires that for all $x \in \mathcal{D}$, $\tilde{v}^T (Ax - b) \geq 0$. Additionally, for the special points $\tilde{x}$ satisfying $\tilde{x} \in \operatorname{int} \mathcal{D}$, it follows $\tilde{v}^T (A\tilde{x} - b) = 0$. Since $\operatorname{relint} \mathcal{D} = \operatorname{int} \mathcal{D}$, we must have points in $\mathcal{D}$ with $\tilde{v}^T (Ax - b) < 0$ unless $A^T \tilde{v} = 0$. This, of course, contradicts our assumption that $\operatorname{rank} A = p$. Thus we get that $\mu \neq 0$,(this also corresponds to what we have explained by using the graph of hyperplane). □

Numerical analysis to check optimization

Recall in previous part we derive the relationship for primal problem and its dual problem, we have $p^* \geq g(\lambda, \nu)$. The equality holds if it satisfies strong duality. Actually, if both primal and dual problem's optimal is hard to find, we can use this property, say, comparing how much the current results for two problems are close to each other, to find an approximate optimal point for the problem. We now give a new definition to demonstrate it.

Definition 2.101. *(Duality gap) We define the gap between prime and dual objectives, $f_0(x) - g(\lambda, \nu)$ as duality gap associated with the prime feasible point x and dual feasible point (λ, ν). So we have $p^* \in [g(\lambda, \nu), f_0(x)]$ and $d^* \in [g(\lambda, \nu), f_0(x)]$.*

Additionally, consider the definition for suboptimal we provided before,we could use this comparison in optimization algorithms to provide nonheuristic stopping criteria. Suppose an algorithm produces a sequence of primal feasible $x^{(k)}$ and dual feasible $\left(\lambda^{(k)}, \nu^{(k)}\right)$, for $k = 1, 2, \ldots$, and $\epsilon_{\text{abs}} > 0$ is a given required absolute accuracy. Then the stopping criterion

$$f_0\left(x^{(k)}\right) - g\left(\lambda^{(k)}, \nu^{(k)}\right) \leq \epsilon_{\text{abs}}$$

guarantees that when the algorithm terminates, $x^{(k)}$ is ϵ_{abs}-suboptimal. Remember in numerical analysis, we have relative error

$$\epsilon_{\text{rel}} = \frac{f_0\left(x^{(k)}\right) - p^\star}{|p^\star|}.$$

We give two practical options for an approximated value of p^* by considering the dual gap:

$$\text{If } g\left(\lambda^{(k)}, \nu^{(k)}\right) > 0, \text{ we take the criterion, } \frac{f_0\left(x^{(k)}\right) - g\left(\lambda^{(k)}, \nu^{(k)}\right)}{g\left(\lambda^{(k)}, \nu^{(k)}\right)} \leq \epsilon_{\text{rel}},$$

or

$$\text{If } f_0\left(x^{(k)}\right) < 0, \text{ we take the criterion, } \frac{f_0\left(x^{(k)}\right) - g\left(\lambda^{(k)}, \nu^{(k)}\right)}{-f_0\left(x^{(k)}\right)} \leq \epsilon_{\text{rel}}.$$

Given an prime problem satisfying dual property, its optimal points satisfy :

$$\begin{aligned} f_0(x^\star) &= g(\lambda^\star, \nu^\star) \\ &= \inf_x \left(f_0(x) + \sum_{i=1}^m \lambda_i^\star f_i(x) + \sum_{i=1}^p \nu_i^\star h_i(x) \right) \\ &\leq f_0(x^\star) + \sum_{i=1}^m \lambda_i^\star f_i(x^\star) + \sum_{i=1}^p \nu_i^\star h_i(x^\star) \\ &\leq f_0(x^\star). \end{aligned}$$

Since we have $f_i(x) \leq 0$ and $\lambda \geq 0$. The last inequality implies $\sum_{i=1}^m \lambda_i^\star f_i(x^\star) = 0$, we call this Complementary Slackness since we could it write as

$$\begin{cases} \lambda_i^* > 0 \Rightarrow f_i(x^*) = 0 \\ f_i(x^*) < 0 \Rightarrow \lambda_i^* = 0 \end{cases}$$

And we know $g(\lambda, \nu)$ is the minimum of the Lagrange dual function, we get it by taking the derivative of Lagrange function is 0. So we also have

$$\nabla f_0(x^\star) + \sum_{i=1}^m \lambda_i^\star \nabla f_i(x^\star) + \sum_{i=1}^p \nu_i^\star \nabla h_i(x^\star) = 0.$$

which is called the stationary condition.

Definition 2.102. *(KKT condition) We call the KKT condition is satisfied if we have the following 4 property: primal feasible, dual feasible, Complementary Slackness and stationary.*

To be more specific, the condition satisfies

$$\begin{aligned} f_i(x^\star) &\le 0, \quad i = 1, \dots, m \\ h_i(x^\star) &= 0, \quad i = 1, \dots, p \\ \lambda_i^\star &\ge 0, \quad i = 1, \dots, m \\ \lambda_i^\star f_i(x^\star) &= 0, \quad i = 1, \dots, m \\ \nabla f_0(x^\star) + \sum_{i=1}^{m} \lambda_i^\star \nabla f_i(x^\star) + \sum_{i=1}^{p} \nu_i^\star \nabla h_i(x^\star) &= 0, \end{aligned}$$

Proposition 2.103. *If the prime problem is convex, every function is differential, and the strong duality holds, then (x^*, λ^*, ν^*) is an optimal point if and only if it satisfies KKT condition.*

Proof. For sufficiency, it is obvious since it comes from how we get KKT-condition, even the problem is not convex.

For necessity, we first prove that the Lagrange dual function is convex since it is combination of non-negative convex functions. It satisfies stationary condition, so we get the global optimal point for x, so we conclude that

$$\begin{aligned} g(\lambda^*, \nu^*) &= L(x^*, \lambda^*, \nu^*) \\ &= f_0(x^*) + \sum_{i=1}^{m} \lambda_i^* f_i(x^*) + \sum_{i=1}^{p} \nu_i^* h_i(x^*) \\ &= f_0(x^*). \end{aligned}$$

This shows that x^* and (λ^*, ν^*) have zero duality gap, and therefore are primal and dual optimal. □

Remark 2.104. *We provide another equivalent statement: If a convex optimization problem with differentiable objective and constraint functions satisfies Slater's condition, then the KKT conditions are necessary and sufficient conditions for optimality.*

Example 2.4.7. *We now provide an interesting convex optimization problem related to water-filling.*

$$\begin{array}{ll} \text{minimize} & -\sum_{i=1}^{n} \log(\alpha_i + x_i) \\ \textit{subject to} & x \succeq 0, \quad \mathbf{1}^T x = 1, \end{array}$$

where $\alpha_i > 0$.

Introducing Lagrange multipliers $\lambda^\star \in \mathbf{R}^n$ for the inequality constraints $x^\star \succeq 0$, and a multiplier $\nu^* \in \mathbf{R}$ for the equality constraint $\mathbf{1}^T x = 1$, we obtain the KKT conditions

$$\begin{aligned} -(x^*) &\preceq 0 \\ 1^T x^\star &= 1 \\ \lambda^* &\succeq 0 \\ \lambda_i^* x_i^* &= 0, \quad i = 1, \dots, n \\ \frac{-1}{(\alpha_i + x_i^*)} - \lambda_i^* + \nu^* &= 0, \quad i = 1, \dots, n. \end{aligned}$$

So we have

$$\begin{cases} x_i^* \left(\nu^* - \frac{1}{(\alpha_i + x_i^*)} \right) = 0, & i = 1, \dots, n \\ \nu^* \ge \frac{1}{(\alpha_i + x_i^*)}, & i = 1, \dots, n. \end{cases}$$

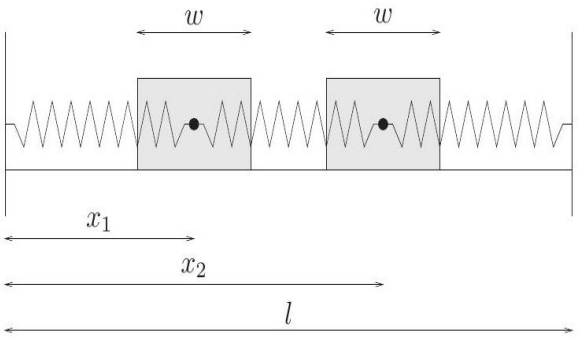
Thus we obtain

$$x_i^* = \begin{cases} 1/\nu^* - \alpha_i & \nu^* < 1/\alpha_i \\ 0 & \nu^* \geq 1/\alpha_i, \end{cases}$$

or, in a simpler form, $x_i^* = \max\{0, 1/\nu^* - \alpha_i\}$. *Substituting this expression for* x_i^* *into the condition* $1^T x^* = 1$ *we obtain*

$$\sum_{i=1}^{n} \max\{0, 1/\nu^* - \alpha_i\} = 1.$$

Example 2.4.8. *We now provide another example related to physical interpretation. We illustrate the idea with a simple example. There is a system which 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* $x \in \mathbf{R}^2$, *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 picture is shown below.*



To simplify the question, we first assume that the original length of each spring is 0, *so the potential energy in the springs, as a function of the block positions, is given by*

$$f_0(x_1, x_2) = \frac{1}{2}k_1 x_1^2 + \frac{1}{2}k_2 (x_2 - x_1)^2 + \frac{1}{2}k_3 (l - x_2)^2$$

where $k_i > 0$ *are the stiffness constants of the three springs. The equilibrium position* $x^\star$ *is the position that minimizes the potential energy subject to the inequalities*

$$w/2 - x_1 \leq 0, \quad w + x_1 - x_2 \leq 0, \quad w/2 - l + x_2 \leq 0.$$

The equilibrium position is given by the solution of the optimization problem

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\left(k_1 x_1^2 + k_2 (x_2 - x_1)^2 + k_3 (l - x_2)^2\right) \\ \textit{subject to} & w/2 - x_1 \leq 0 \\ & w + x_1 - x_2 \leq 0 \\ & w/2 - l + x_2 \leq 0. \end{array}$$

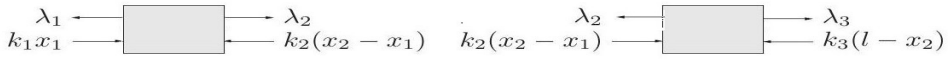
With $\lambda_1, \lambda_2, \lambda_3$ *as Lagrange multipliers, the KKT condition for this problem is now:*

$$\lambda_1 (w/2 - x_1) = 0, \quad \lambda_2 (w - x_2 + x_1) = 0, \quad \lambda_3 (w/2 - l + x_2) = 0$$

and

$$\begin{bmatrix} k_1 x_1 - k_2 (x_2 - x_1) \\ k_2 (x_2 - x_1) - k_3 (l - x_2) \end{bmatrix} + \lambda_1 \begin{bmatrix} -1 \\ 0 \end{bmatrix} + \lambda_2 \begin{bmatrix} 1 \\ -1 \end{bmatrix} + \lambda_3 \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 0.$$

We could also explain the KKT condition by physical interpretation. We firstly consider some extreme situations, if the length of spring is 0, then the wall will impose a force λ_1 to the left block, the compress force between two blocks is λ_2 and the right wall will impose a force λ_3 to the right block. So we will have three complement slackness conditions. Also based on the extreme situation provided before, we could make the force balance equation for two blocks. Details are shown below.



Definition 2.105. *(Perturbation for standard optimization problem) We now make some small changes to standard optimization problem and consider the following perturbed version:*

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \textit{subject to} & f_i(x) \le u_i, \quad i = 1, \dots, m \\ & h_i(x) = v_i, \quad i = 1, \dots, p \end{array}$$

with variable $x \in \mathbf{R}^n$.

When $u = 0, v = 0$, we then get the original optimization problem. When u_i is positive it means that we have relaxed the i th inequality constraint; when u_i is negative, it means that we have tightened the constraint.
We define $p^\star(u, v)$ as the optimal value of the perturbed problem :

$$\begin{aligned} p^\star(u, v) = \inf \{ f_0(x) \mid \exists x \in \mathcal{D}, \quad & f_i(x) \le u_i, i = 1, \dots, m \\ & h_i(x) = v_i, i = 1, \dots, p \} \end{aligned}$$

If $p^\star(u, v) = \infty$, it means that perturbations of the constraints result in infeasibility. The function $p^\star : \mathbf{R}^m \times \mathbf{R}^p \to \mathbf{R}$ gives the optimal value of the problem as a function of perturbations to the right hand sides of the constraints.

Proposition 2.106. *When the original problem is convex, the function $p^\star$ is a convex function of u and v*

Proof. Recall that a function is convex iff its epigraph is convex. So we only need to prove the epigraph of $p^*(u, v)$ is convex. Additionally, $h_i(x)$ is linear function since we could write $h_i(x) = v_i$ as $h_i(x) \le v_i$ and $h_i(x) \ge v_i$. The epigraph is defined by

$$\begin{aligned} \operatorname{epi} p^* = \{ (u, v, t) \mid \exists x \in \mathcal{D}, & f_i(x) \le u_i, i = 1, \dots, m \\ & h_i(x) = v_i, i = 1, \dots, p, \inf_x \{ f_0(x) \} \le t \Big\}, \end{aligned}$$

For any $(u_1, v_1, t_1), (u_2, v_2, t_2) \in \operatorname{epi} p^*$, there exist $x_1, x_2 \in D$ such that for any $\epsilon > 0$, $f_0(x_1) \le t_1 + \epsilon, f_i(x_1) \le u_1, h_i(x_1) = v_1$ and $f_0(x_2) \le t_2 + \epsilon, f_i(x_2) \le u_2, h_i(x_2) = v_2$. Since $f_i(x)$ is convex, we have

$$f_0(\theta x_1 + (1 - \theta)x_2) \le \theta f_0(x_1) + (1 - \theta) f_0(x_2) \le \theta t_1 + (1 - \theta) t_2 + \epsilon,$$

and

$$f_i(\theta x_1 + (1 - \theta)x_2) \le \theta f_i(x_1) + (1 - \theta) f_i(x_2) \le \theta u_1 + (1 - \theta) u_2.$$

We need to prove

$$(\theta u_1 + (1-\theta)u_2, \theta v_1 + (1-\theta)v_2, \theta t_1 + (1-\theta)t_2) \in \operatorname{epi} p^*.$$

Since there exists $x = \theta x_1 + (1-\theta)x_2$ and $f_0(x) \leq \theta t_1 + (1-\theta)t_2 + \epsilon$ for any $\epsilon > 0$. Therefore $f_0(x) \leq \theta t_1 + (1-\theta)t_2$, which concludes the proof. □

Remark 2.107. *We get the idea for this proof from previous proof of Minimization property of convex function. Actually we could also prove it just use this proposition.*

Now we assume that strong duality holds, and that the dual optimum is attained.

Proposition 2.108. *Let $(\lambda^\star, \nu^\star)$ be optimal for the dual of the unperturbed problem, then for all u and v we have*

$$p^\star(u,v) \geq p^\star(0,0) - \lambda^{\star T} u - \nu^{\star T} v.$$

Proof. Suppose that x is any feasible point for the perturbed problem, i.e., $f_i(x) \leq u_i$ for $i = 1, \ldots, m$, and $h_i(x) = v_i$ for $i = 1, \ldots, p$. Then we have, by strong duality,

$$\begin{aligned} p^\star(0,0) = g(\lambda^\star, \nu^\star) &\leq f_0(x) + \sum_{i=1}^{m} \lambda_i^\star f_i(x) + \sum_{i=1}^{p} \nu_i^\star h_i(x) \\ &\leq f_0(x) + \lambda^{\star T} u + \nu^{\star T} v. \end{aligned}$$

Since the feasible x for the original problem still holds feasible for perturbation problem. The first inequality in original problem will derive the second inequality for the new perturbation problem. So we conclude that for any x feasible for the perturbed problem, we have

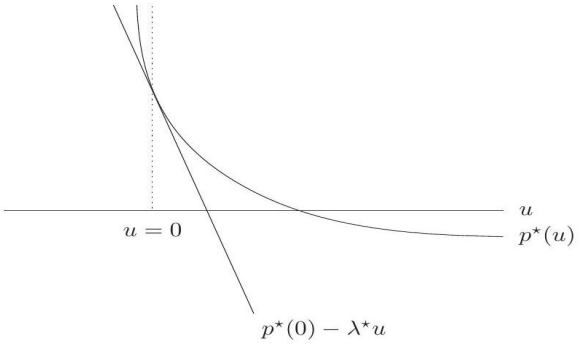
$$f_0(x) \geq p^\star(0,0) - \lambda^{\star T} u - \nu^{\star T} v.$$

□

When strong duality holds, from the above proposition, we have conclusions in four cases:

- If $\lambda_i^\star$ is large and we tighten the i th constraint (i.e., choose $u_i < 0$), then the growth rate of the optimal value $p^\star(u,v)$ is fast.
- If $\nu_i^\star$ is large and positive and we take $v_i < 0$, or if $\nu_i^\star$ is large and negative and we take $v_i > 0$, then the growth rate of the optimal value $p^\star(u,v)$ is also fast.
- If $\lambda_i^\star$ is small, and we loosen the i th constraint $(u_i > 0)$, then the decrease rate of the optimal value $p^\star(u,v)$ is slow.
- If $\nu_i^\star$ is small and positive, and $v_i > 0$, or if $\nu_i^\star$ is small and negative and $v_i < 0$, then the decrease rate of the optimal value $p^\star(u,v)$ is also slow.

We now assume $v = 0$, and only consider the relationship between u, p^*. From the graph, it is easy to see that $p^*(0) - \lambda^{\star T} u$ is the lower bound for $p^*(u)$.



Proposition 2.109. *(Local sensitivity analysis) If $p^\star(u,v)$ is differentiable at $u = 0, v = 0$. Then, provided strong duality holds, the optimal dual variables $\lambda^\star, \nu^\star$ are related to the gradient of $p^\star$ at $u = 0, v = 0$:*

$$\lambda_i^\star = -\frac{\partial p^\star(0,0)}{\partial u_i}, \quad \nu_i^\star = -\frac{\partial p^\star(0,0)}{\partial v_i}.$$

Proof. Suppose $p^\star(u,v)$ is differentiable and it holds strong duality. We now consider the directional derivative to analyze perturbation, i.e., $u = te_i, v = 0$, where e_i is the i th unit vector„ we will obtain

$$\lim_{t\to 0} \frac{p^\star(te_i, 0) - p^\star}{t} = \frac{\partial p^\star(0,0)}{\partial u_i}.$$

When $t > 0$, we will have

$$\frac{p^\star(te_i, 0) - p^\star}{t} \geq -\lambda_i^\star,$$

while for $t < 0$ we have

$$\frac{p^\star(te_i, 0) - p^\star}{t} \leq -\lambda_i^\star.$$

By letting the limit $t \to 0$, with $t > 0$, yields

$$\frac{\partial p^\star(0,0)}{\partial u_i} \geq -\lambda_i^\star,$$

while taking the limit with $t < 0$ yields the opposite inequality, so we conclude that

$$\frac{\partial p^\star(0,0)}{\partial u_i} = -\lambda_i^\star.$$

The same method can be used to establish

$$\frac{\partial p^\star(0,0)}{\partial v_i} = -\nu_i^\star.$$

□

Remark 2.110. *Shadow price interpretation is when we consider the optimal solution we already get and consider the perturbations engendered by the trend towards the optimal solution.*

Definition 2.111. *(Alternative inequality system) Given a system of inequalities and equalities with its domain non-empty*

$$f_i(x) \leq 0, \quad i = 1, \ldots, m, \quad h_i(x) = 0, \quad i = 1, \ldots, p.$$

we have its inequality system

$$\lambda \succeq 0, \quad g(\lambda, \nu) > 0,$$

where $g(\lambda, \nu)$ is the dual function.
Similarly, for strict inequality system

$$f_i(x) < 0, \quad i = 1, \ldots, m, \quad h_i(x) = 0, \quad i = 1, \ldots, p.$$

With g defined as for the nonstrict inequality system, we have the alternative inequality system

$$\lambda \succeq 0, \quad \lambda \neq 0, \quad g(\lambda, \nu) \geq 0.$$

We now study the optimal value for these systems of inequalities.
Consider the inequality

$$f_i(x) \leq 0, \quad i = 1, \ldots, m, \quad h_i(x) = 0, \quad i = 1, \ldots, p.$$

It is equivalent to make the system as the standard problem, with objective $f_0 = 0$, i.e.,

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p. \end{array}$$

So it is obvious that this problem has optimal value

$$p^\star = \begin{cases} 0 & \text{feasible} \\ \infty & \text{infeasible} . \end{cases}$$

Remember the dual function

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} \left(\sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right) \leq 0.$$

So for its alternative inequality system, we define its optimal point

$$d^\star = \begin{cases} \infty & \lambda \succeq 0, g(\lambda, \nu) > 0 \text{ is feasible} \\ 0 & \lambda \succeq 0, g(\lambda, \nu) > 0 \text{ is infeasible.} \end{cases}$$

Then, we shall find that if the original inequality system is feasible, then the inequality system must be infeasible, and if the inequality system

$$\lambda \succeq 0, \quad g(\lambda, \nu) > 0,$$

is feasible, then the original inequality system is infeasible for two system of inequalities.

Definition 2.112. *(Weak alternative) Two systems of inequalities (and equalities) are called weak alternatives if at most one of the two is feasible. The example beforehand is weak alternative.*

We now show another weak alternative for strict inequality system

$$f_i(x) < 0, \quad i = 1, \ldots, m, \quad h_i(x) = 0, \quad i = 1, \ldots, p,$$

and its the alternative inequality system

$$\lambda \succeq 0, \quad \lambda \neq 0, \quad g(\lambda, \nu) \geq 0.$$

Just by definition of alternative inequality, we suppose there exists an $\tilde{x}$ with $f_i(\tilde{x}) < 0, h_i(\tilde{x}) = 0$. Then for any $\lambda \succeq 0, \lambda \neq 0$, 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

$$\begin{aligned} g(\lambda, \nu) &= \inf_{x \in \mathcal{D}} \left(\sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right) \\ &\leq \sum_{i=1}^{m} \lambda_i f_i(\tilde{x}) + \sum_{i=1}^{p} \nu_i h_i(\tilde{x}) \\ &< 0. \end{aligned}$$

Thus it also follows weak alternative.

Definition 2.113. *(Strong alternative) Two systems of inequalities (and equalities) are called strong alternatives if exactly one of the two is feasible.*

We now give some special examples satisfying strong alternative with form has $Ax = b$, thus satisfying Slater's condition.

Example 2.4.9. *We first study the strict inequality system*

$$f_i(x) < 0, \quad i = 1, \ldots, m, \quad Ax = b,$$

and its alternative

$$\lambda \succeq 0, \quad \lambda \neq 0, \quad g(\lambda, \nu) \geq 0.$$

Considering Slater's condition, we assume that there exists an $x \in \operatorname{relint} \mathcal{D}$ with $Ax = b$. For original system, similar to the method we used before, we establish this result by considering the related optimization problem

$$\begin{array}{ll} \text{minimize} & s \\ \textit{subject to} & f_i(x) - s \leq 0, \quad i = 1, \ldots, m \\ & Ax = b, \end{array}$$

with variables x, s, and domain $\mathcal{D} \times \mathbf{R}$. The optimal value $p^\star$ of this problem is negative if and only if there exists a solution to the strict inequality system. So we then mainly study the optimal value of this related optimization problem to help us study original problem. The Lagrange dual function is now

$$\inf_{s, x \in \mathcal{D}} \left(s + \sum_{i=1}^{m} \lambda_i \left(f_i(x) - s \right) + \nu^T (Ax - b) \right) = \begin{cases} g(\lambda, \nu) & \mathbf{1}^T \lambda = 1 \\ -\infty & \textit{otherwise} \end{cases}$$

Therefore we can express the dual problem as

$$\begin{array}{ll} \textit{maximize} & g(\lambda, \nu) \\ \textit{subject to} & \lambda \succeq 0, \quad \mathbf{1}^T \lambda = 1. \end{array}$$

We now consider Slater's condition, and choosing any $\tilde{s} > \max_i f_i(\tilde{x})$ yields a point $(\tilde{x}, \tilde{s})$ which is strictly feasible. Therefore we have $d^\star = p^\star$, and there exist $(\lambda^\star, \nu^\star)$ such that

$$g(\lambda^\star, \nu^\star) = p^\star, \quad \lambda^\star \succeq 0, \quad \mathbf{1}^T \lambda^\star = 1.$$

Now suppose that the strict inequality system is infeasible, which means that $p^\star \geq 0$. Then $(\lambda^\star, \nu^\star)$ satisfy the alternate inequality system. Similarly, if the alternate inequality system is feasible, then $d^\star = p^\star \geq 0$, which shows that the original strict inequality system is infeasible. Thus, the two inequality systems are strong alternatives.

Example 2.4.10. *We now consider the nonstrict inequality system*

$$f_i(x) \leq 0, \quad i = 1, \ldots, m, \quad Ax = b,$$

and its alternative

$$\lambda \succeq 0, \quad g(\lambda, \nu) > 0.$$

We will show these are strong alternatives, provided the following conditions hold: there exists an $x \in \operatorname{relint} \mathcal{D}$ with $Ax = b$. We consider another related optimization problem.

$$\begin{array}{ll} \text{minimize} & s \\ \textit{subject to} & f_i(x) - s \leq 0, \quad i = 1, \ldots, m \\ & Ax = b. \end{array}$$

The optimal value $p^*$is non-positive if and only if original problem is feasible. So when original problem is feasible, then $p^* \leq 0$, and $d^* \leq p^* \leq 0$, so the alternative system is infeasible. When original problem is infeasible, we have $p^* > 0$ and now the alternative system is feasible. Thus, the inequality systems are strong alternatives.

Lemma 2.114. *(Farkas' lemma) The system of inequalities*

$$Ax \preceq 0, \quad c^T x < 0,$$

where $A \in \mathbf{R}^{m \times n}$ and $c \in \mathbf{R}^n$, and the system of equalities and inequalities

$$A^T y + c = 0, \quad y \succeq 0,$$

are strong alternatives.

Proof. We can prove Farkas' lemma directly using LP duality. Consider the LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq 0. \end{array}$$

Recall that we require $c^T x < 0$, so we have the property:

$$p^\star = \begin{cases} 0 & \text{infeasible} \\ -\infty & \text{feasible} \end{cases}$$

and its dual

$$\begin{array}{ll} \text{maximize} & 0 \\ \text{subject to} & A^T y + c = 0 \\ & y \succeq 0. \end{array}$$

with its optimal point

$$d^\star = \begin{cases} 0 & \text{feasible} \\ -\infty & \text{infeasible.} \end{cases}$$

So we say this two system of equalities and inequalities are strong alternatives. □

Remark 2.115. *We now use Farka's Lemma to give further proof of the relationship between linear programming and its dual problem.*

Proposition 2.116. *There are four situations for the relationship between linear programming and its dual problem.*

1. *Both primal and dual have no feasible solutions (are infeasible).*
2. *The primal is infeasible and the dual unbounded.*
3. *The dual is infeasible and the primal unbounded.*
4. *Both primal and dual have feasible solutions and their values are equal.*

Proof. We will now use the LP

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & Ax \preceq b, \end{aligned}$$

and its dual

$$\begin{aligned} \text{maximize} \quad & -b^T y \\ \text{subject to} \quad & A^T y + c = 0 \\ & y \succeq 0. \end{aligned}$$

to give the detail of the proof. Notice we change the Farka's Lemma to a new system of inequalities which directly corresponds to original constraints: (1) $Ax \preceq b$, and the system of equalities and inequalities (2) $A^T y = 0, y^T b < 0, y \succeq 0$ are strong alternatives.

1. The example with the special x and special A, b is shown in (2.13).
2. If primal is infeasible, then (2) must hold, so there exists $\hat{y}$ such that $A^T \hat{y} = 0, \hat{y}^T b < 0, \hat{y} \succeq 0$. If exists y' a feasible solution for the dual form, we now consider the family of solution $y = y' + \lambda \hat{y}, \lambda \succeq 0$. Then for each λ, y is feasible since

   $$A^T y + c = A^T (y' + \lambda \hat{y}) + c = A^T y' + c = 0$$

   and

   $$y = y' + \lambda \hat{y} \succeq 0.$$

   The objective value is now

   $$-b^T y = -b^T (y' + \lambda \hat{y}).$$

   Since $\hat{y}^T b < 0$,we have $b \preceq 0$, when $\lambda \to +\infty$, $-b^T y = -b^T (y' + \lambda \hat{y}) \to +\infty$, which is unbounded.
3. If dual is infeasible, we now use the original Farka's Lemma, so there exists $\hat{x}$ such that $A\hat{x} \preceq 0, \quad c^T \hat{x} < 0$, if exists x' a feasible solution for original LP, we now consider the family of solution $x = x' + \lambda \hat{x}, \lambda \succeq 0$. Then for each λ, x is feasible since

   $$Ax = A(x' + \lambda \hat{x}) \preceq b + 0 = b.$$

   The objective value is now

   $$c^T x = c^T (x' + \lambda \hat{x}).$$

   Since $c^T \hat{x} < 0$, when $\lambda \to +\infty$, $c^T (x' + \lambda \hat{x}) \to -\infty$, which is unbounded.

4. If exists x' and y' which are feasible solutions for original LP and its dual form. By weak duality, $c^T x' \geq -b^T y'$, so both the prime and dual are bounded now. Let d be the optimal value for the dual, so we have the optimal value of prime bigger than d,i.e, there does not exist x such that $Ax \preceq b, c^T x \leq d$, or we write it as $\begin{bmatrix} A \\ c^T \end{bmatrix} x \preceq \begin{bmatrix} b \\ d \end{bmatrix}$, so there must exist a vector $[y^T, k]^T$, where $k \in R$ and

$$\begin{bmatrix} A \\ c^T \end{bmatrix}^T \begin{bmatrix} y \\ k \end{bmatrix} = 0, \begin{bmatrix} b \\ d \end{bmatrix}^T \begin{bmatrix} y \\ k \end{bmatrix} < 0, \begin{bmatrix} y \\ k \end{bmatrix} \succeq 0.$$

If $k = 0$, then we get $A^T y = 0, y^T b < 0, y \succeq 0$. Then prime problem is infeasible, contradicts our assumption.
If $k > 0$, then we have $A^T y = -ck$, so $A^T \left(\frac{y}{k}\right) + c = 0$, then $\frac{y}{k}$ is feasible. But since $b^T y + kd < 0$, $d < -b^T \frac{y}{k}$, which gives us a new feasible optimal value and this contradicts d is the optimal value of the dual problem. Thus, if the primal and dual are both feasible, then their optimal values are equal.

□

Chapter 3

Optimization Algorithms

3.1 Unconstrained minimization Algorithms

Introduction to unconstrained minimization

In this chapter we will introduce some basic numerical methods for solving the unconstrained optimization problem

$$\text{minimize } f(x)$$

where $f : \mathbf{R}^n \to \mathbf{R}$ is convex and twice continuously differentiable. We now assume that there exists an optimal point $x^\star$ and denote the optimal value, $\inf_x f(x) = f(x^\star)$, as $p^\star$. Since f is differentiable and convex, from previous chapter, we know $x^\star$ is optimal if and only if $\nabla f(x^\star) = 0$. But for most of the time, the problem do not have analytical solution, and we have to solve it by an iterative algorithm which computes a sequence of points $x^{(0)}, x^{(1)}, \ldots \in \mathbf{dom}\, f$ with $f\left(x^{(k)}\right) \to p^\star$ as $k \to \infty$. Such a sequence of points is called a minimizing sequence. The algorithm is terminated when $f\left(x^{(k)}\right) - p^\star \leq \epsilon$, where $\epsilon > 0$ is some specified tolerance.
We now show two classical examples about unconstrained minimization.

Example 3.1.1. *The first example is about Quadratic minimization and least-squares with the form*

$$\textit{minimize } (1/2)x^T Px + q^T x + r,$$

where $P \in \mathbf{S}^n_+, q \in \mathbf{R}^n$, and $r \in \mathbf{R}$.
Let $f(x) = (1/2)x^T Px + q^T x + r$. So we could calculate the optimal point by $\nabla f = Px^\star + q = 0$. When $P \succ 0$, there is a unique solution, $x^\star = -P^{-1}q$. When P is non-positive semi-definite,then there must exist v s.t $v^T Pv < 0$, we define $z = \alpha v$, $f(z) = \frac{\alpha^2}{2} v^T Pv + \alpha q^T v + r$, when $\alpha \to \infty$, $f(z)$ will be unbounded below. If P is positive semidefinite, but $Px^* = -q$ does not have a solution. Since we know that $\nabla f(x) = 0$ gives the global minimization, so the problem must be unbounded below.
We now consider one special case of the quadratic minimization problem called least-squares problem

$$\textit{minimize } \|Ax - b\|_2^2 = x^T \left(A^T A\right) x - 2\left(A^T b\right)^T x + b^T b.$$

The optimality condition is

$$A^T Ax^\star = A^T b.$$

Example 3.1.2. *We now consider an unconstrained geometric program in convex form, which is disscussed in* (2.1.1).

$$\textit{minimize } f(x) = \log\left(\sum_{i=1}^{m} \exp\left(a_i^T x + b_i\right)\right).$$

The optimality condition is

$$\nabla f\left(x^\star\right) = \frac{1}{\sum_{j=1}^{m} \exp\left(a_j^T x^\star + b_j\right)} \sum_{i=1}^{m} \exp\left(a_i^T x^\star + b_i\right) a_i = 0,$$

with its domain $\operatorname{dom} f = R^n$. *We can not easily get analytical solution, so here we have to resort to an iterative algorithm. Details will be discussed in next sections.*

In the beginning of the section, we briefly discussed the idea of the algorithm. We now give a further discussion to show the validation of this idea.

Theorem 3.1. *(Existence of minimizing sequence) If the function is strongly convex and twice differentiable, then there exist a minimizing sequence such that* $f\left(x^{(k)}\right)$ *is convergent to its optimal value* p^*.

Remark 3.2. *This theorem is based on the theorem of sequential continuity, to be more specific, a function* $f : X \to Y$ *is continuous at* $x \in X$ *if and only if* (x_n) *converges to* x *in* $X \Longrightarrow f(x_n)$ *converges to* $f(x)$ *in* Y.

Before we prove this theorem, we first introduce one kind of sublevel based on initial point.

Definition 3.3. *If the starting point lies in* $\operatorname{dom} f$, *we define such kind of sublevel set*

$$S = \left\{x \in \operatorname{dom} f \mid f(x) \leq f\left(x^{(0)}\right)\right\}.$$

Remark 3.4. S *is closed since function* f *is convex and continuous, and the imagine of* f *is closed, the preimage now equilvalent to its sublevel set is closed.*

Proof. Assume that the objective function is strongly convex on S, which means that there exists an $m > 0$ such that

$$\nabla^2 f(x) \succeq mI,$$

for all $x \in S$. Since f is twice differentiable, we do second order Taylor expansion for $x, y \in S$ we have

$$f(y) = f(x) + \nabla f(x)^T (y - x) + \frac{1}{2}(y - x)^T \nabla^2 f(z)(y - x),$$

for some z on the line segment $[x, y]$. By the strong convexity assumption, the last term on the righthand side is at least $(m/2)\|y - x\|_2^2$, so we have the inequality

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2}\|y - x\|_2^2,$$

for all x and y in S.

We firstly derive the convergence for x, i.e., an upper bound for $\|x - x^\star\|_2$, the distance between x and any optimal point $x^\star$,in terms of $\|\nabla f(x)\|_2$: let $y = x^\star$, we will obtain

$$\begin{aligned} p^\star = f\left(x^\star\right) &\geq f(x) + \nabla f(x)^T \left(x^\star - x\right) + \frac{m}{2}\left\|x^\star - x\right\|_2^2 \\ &\geq f(x) - \|\nabla f(x)\|_2 \left\|x^\star - x\right\|_2 + \frac{m}{2}\left\|x^\star - x\right\|_2^2, \end{aligned}$$

where we use the Cauchy-Schwarz inequality in the second inequality. Since $p^\star \leq f(x)$, we must have

$$-\|\nabla f(x)\|_2 \|x^\star - x\|_2 + \frac{m}{2}\|x^\star - x\|_2^2 \leq 0,$$

So we finally get

$$\|x - x^\star\|_2 \leq \frac{2}{m}\|\nabla f(x)\|_2.$$

Secondly, we will show the convergence for $f(x)$. Again, we mainly consider the right hand side of the inequality, since it is a convex quadratic function, we fix x here and will find that $\tilde{y} = x - (1/m)\nabla f(x)$ minimizes the righthand side. Therefore we have

$$\begin{aligned} f(y) &\geq f(x) + \nabla f(x)^T(y - x) + \frac{m}{2}\|y - x\|_2^2 \\ &\geq f(x) + \nabla f(x)^T(\tilde{y} - x) + \frac{m}{2}\|\tilde{y} - x\|_2^2 \\ &= f(x) - \frac{1}{2m}\|\nabla f(x)\|_2^2. \end{aligned}$$

Since this holds for any $y \in S$, we have

$$p^\star \geq f(x) - \frac{1}{2m}\|\nabla f(x)\|_2^2.$$

So we will have

$$f(x) - p^* \leq \frac{1}{2m}\|\nabla f(x)\|_2^2.$$

This inequality shows that if the gradient is really small at a point, then the point is nearly optimal, to be more specific,

$$\|\nabla f(x)\|_2 \leq (2m\epsilon)^{1/2} \Longrightarrow f(x) - p^\star \leq \epsilon.$$

□

Remark 3.5. *Similarly, we could get the observation that if $\|\nabla f(x)\|_2$ is large enough, then $f(x) - p^*$ will be really large.*

Proof. Since $\lim_{\|x\|\to\infty} f(x) = +\infty$, for any $M > 0$, there exists R s.t. $||x|| > R, f(x) > M = f(x_0)$. So $S \subseteq \{x : ||x|| < R\}$, Since S is closed and bounded, the maximum eigenvalue of $\nabla^2 f(x)$ is bounded. We now give the upper bound for

$$\nabla^2 f(x) \preceq MI$$

for all $x \in S$. This upper bound on the Hessian implies for any $x, y \in S$,

$$f(y) \leq f(x) + \nabla f(x)^T(y - x) + \frac{M}{2}\|y - x\|_2^2.$$

Minimizing each side over y yields

$$p^\star \leq f(x) - \frac{1}{2M}\|\nabla f(x)\|_2^2,$$

□

Definition 3.6. *(Width of a convex set) We define the width of a convex set $C \subseteq \mathbf{R}^n$, in the direction q, where $\|q\|_2 = 1$, as*

$$W(C,q) = \sup_{z\in C} q^T z - \inf_{z\in C} q^T z.$$

So the minimum width and maximum width of C are determined by the direction of q we choose, given by

$$W_{\min} = \inf_{\|q\|_2=1} W(C,q), \quad W_{\max} = \sup_{\|q\|_2=1} W(C,q).$$

Definition 3.7. *(Condition number of the convex set) Similar to the definition of condition number for a matrix, we define the condition number of the convex set C is defined as*

$$\mathrm{cond}(C) = \frac{W_{\max}^2}{W_{\min}^2}.$$

Proposition 3.8. *Condition number of an ellipsoid is*

$$\mathrm{cond}(\mathcal{E}) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)} = \kappa(A),$$

where $\mathcal{E}$ is the ellipsoid defined by

$$\mathcal{E} = \left\{ x \mid (x-x_0)^T A^{-1} (x-x_0) \le 1 \right\},$$

$A \in \mathbf{S}^n_{++}$, and $\kappa(A)$ denotes the condition number of the matrix A.

Proof. Since for any $z \in \left\{ x \mid (x-x_0)^T A^{-1} (x-x_0) \le 1 \right\}$, we could write it as :

$$\left[(z-x_0) A^{-1/2}\right]^T \left[A^{-1/2}(z-x_0)\right] \le 1$$

we get that $||A^{-1/2}(z-x_0)||_2 \le 1$
So we could find the upper bound and lower bound for $q^T z$ by Cauchy-inequality, we now show the way to find the upper bound (Similar way to find lower bound).

$$\begin{aligned} q^T z &= q^T(z-x_0) + q^T x_0 \\ &= q^T A^{1/2} A^{-1/2}(z-x_0) + q^T x_0 \\ &\le ||q^T A^{1/2}||_2 ||A^{-1/2}(z-x_0)||_2 + q^T x_0 \\ &\le ||q^T A^{1/2}||_2 + q^T x_0. \end{aligned}$$

So the width of $\mathcal{E}$ in the direction q is

$$\begin{aligned} \sup_{z\in\mathcal{E}} q^T z - \inf_{z\in\mathcal{E}} q^T z &= \left(\left\|A^{1/2}q\right\|_2 + q^T x_0\right) - \left(-\left\|A^{1/2}q\right\|_2 + q^T x_0\right) \\ &= 2\left\|A^{1/2}q\right\|_2. \end{aligned}$$

It follows that its minimum and maximum width are

$$W_{\min} = 2\lambda_{\min}(A)^{1/2}, \quad W_{\max} = 2\lambda_{\max}(A)^{1/2},$$

and its condition number is

$$\mathrm{cond}(\mathcal{E}) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)} = \kappa(A),$$

where $\kappa(A)$ denotes the condition number of the matrix A, i.e., the ratio of its maximum singular value to its minimum singular value. □

Proposition 3.9. *(Condition number of sublevel set) The condition number of the sublevel sets of f with respect to it limit to optiaml value is*

$$\lim_{\alpha \to p^\star} \operatorname{cond}(C_\alpha) = \kappa\left(\nabla^2 f(x^\star)\right).$$

where α-sublevel is $C_\alpha = \{x \mid f(x) \leq \alpha\}$, and we have $p^\star < \alpha \leq f\left(x^{(0)}\right)$.

Proof. Previously we have proved

$$mI \preceq \nabla^2 f(x) \preceq MI,$$

for all $x \in S$. So the ratio $\kappa = M/m$ is thus an upper bound on the condition number of the matrix $\nabla^2 f(x)$, i.e., the ratio of its largest eigenvalue to its smallest eigenvalue.
Now suppose f satisfies $mI \preceq \nabla^2 f(x) \preceq MI$ for all $x \in S$. We will derive a bound on the condition number of the α-sublevel $C_\alpha = \{x \mid f(x) \leq \alpha\}$, where $p^\star < \alpha \leq f\left(x^{(0)}\right)$.
We first define two kinds of ball which satisfy $B_{\text{inner}} \subseteq C_\alpha \subseteq B_{\text{outer}}$

$$\begin{aligned} B_{\text{inner}} &= \left\{y \mid \|y - x^\star\|_2 \leq \left(2(\alpha - p^\star)/M\right)^{1/2}\right\}, \\ B_{\text{outer}} &= \left\{y \mid \|y - x^\star\|_2 \leq \left(2(\alpha - p^\star)/m\right)^{1/2}\right\}. \end{aligned}$$

From previous result, it is obvious that

$$p^\star + (M/2)\|y - x^\star\|_2^2 \geq f(y) \geq p^\star + (m/2)\|y - x^\star\|_2^2.$$

So for any $y \in B_{\text{inner}}$, we have $\frac{M}{2}\|y - x^\star\|_2^2 \leq \alpha - p^\star$, so $f(y) \leq \frac{M}{2}\|y - x^\star\|_2^2 + p^* \leq \alpha$, thus $y \in C_\alpha$, and $B_{\text{inner}} \subseteq C_\alpha$.
Similarly, for any $y \in C_\alpha$, we have

$$p^\star + (m/2)\|y - x^\star\|_2^2 \leq f(y) \leq \alpha,$$

this shows that $\|y - x^\star\|_2 \leq \left(2(\alpha - p^\star)/m\right)^{1/2}$, so $y \in B_{\text{outer}}$, thus giving $C_\alpha \subseteq B_{\text{outer}}$.
In other words, the α-sublevel set contains B_{inner}, and is contained in B_{outer}, which are balls with radius

$$\left(2(\alpha - p^\star)/M\right)^{1/2}, \quad \left(2(\alpha - p^\star)/m\right)^{1/2},$$

respectively. From previous definition, The ratio gives an upper bound on the condition number of C_α :

$$\operatorname{cond}(C_\alpha) \leq \frac{M}{m}.$$

□

Remark 3.10. *We can also give a geometric interpretation of the condition number $\kappa\left(\nabla^2 f(x^\star)\right)$ of the Hessian at the optimum. From the Taylor series expansion of f around $x^\star$,*

$$f(y) \approx p^\star + \frac{1}{2}(y - x^\star)^T \nabla^2 f(x^\star)(y - x^\star),$$

we see that, for α close to $p^\star$,

$$C_\alpha \approx \left\{y \mid (y - x^\star)^T \nabla^2 f(x^\star)(y - x^\star) \leq 2(\alpha - p^\star)\right\},$$

i.e., the sublevel set is well approximated by an ellipsoid with center $x^\star$. Therefore

$$\lim_{\alpha \to p^\star} \operatorname{cond}(C_\alpha) = \kappa\left(\nabla^2 f(x^\star)\right).$$

Descent method

In this section we will further discuss algorithms to deal with unconstraint minimization convex problems.

Definition 3.11. *(Minimizing sequence) We now define minimizing sequence* $x^{(k)}, k = 1, \ldots,$ *satisfies*

$$x^{(k+1)} = x^{(k)} + t^{(k)} \Delta x^{(k)},$$

and $t^{(k)} > 0$ *(except when* $x^{(k)}$ *is optimal), where* Δx *is the search direction and* $t^{(k)} > 0$ *is the step size at iteration* k*.*

Since we hope to find the minimum solution, we require $f\left(x^{(k+1)}\right) < f\left(x^{(k)}\right)$, except when $x^{(k)}$ is optimal. From first-order condition of convexity we know that

$$f(y) \geq f\left(x^{(k)}\right) + \nabla f\left(x^{(k)}\right)^T \left(y - x^{(k)}\right)$$

if we define $y - x^{(k)} = t^{(k)} \Delta x^{(k)}$, and still want to get previous requirement, the search direction in a descent method must satisfy

$$\nabla f\left(x^{(k)}\right)^T \Delta x^{(k)} < 0,$$

We call such a direction a descent direction.

Algorithm 2: Descent method for minimization algorithm

1: Given a start point $x_0 \in \mathbf{dom}\, f$, current value $f(x_0)$, tolerance $\epsilon > 0$, iteration times $k = 0$
2: **while** $f(x) - p^* \geq \epsilon$ **do**
3: Determine a descent direction Δx
4: Line search. Choose a step size $t > 0$
5: $x = x + t\Delta x$
6: Calculate $f(x)$, $k = k + 1$
7: **end while**

We will mainly introduce two kinds of line search to find a suitable iteration step t.

Definition 3.12. *(Exact line search) We choose* t *to minimize* f *along the ray* $\{x + t\Delta x \mid t \geq 0\}$ *:*

$$t = \mathbf{argmin}_{s \geq 0} f(x + s\Delta x).$$

Remark 3.13. *For most of the time, exact line search will not be considered since it is strenuous to get exact best* t*. So we will then introduce another line search method.*

Definition 3.14. *(Backtracking line search) We find the suitable iteration step size* t *by starting with unit step size and then reduces it by the factor* β *until the stopping condition* $f(x + t\Delta x) \leq f(x) + \alpha t \nabla f(x)^T \Delta x$ *holds, with two constants* α, β *and* $0 < \alpha < 0.5, 0 < \beta < 1$*.*

So for small enough t we have

$$f(x+t\Delta x) \approx f(x) + t\nabla f(x)^T \Delta x < f(x) + \alpha t \nabla f(x)^T \Delta x,$$

which shows that the backtracking line search eventually terminates. Additionally, backtracking exit inequality $f(x+t\Delta x) \leq f(x) + \alpha t \nabla f(x)^T \Delta x$ 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

$$t = 1, \quad \text{or} \quad t \in (\beta t_0, t_0].$$

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

$$t \geq \min\{1, \beta t_0\}.$$

Algorithm 3: Backtracking line search

1: Given a descent direction Δx and $\alpha \in (0, 0.5), \beta \in (0, 1)$. initial value of iteration step size $t = 1$.
2: **while** $f(x+t\Delta x) > f(x) + \alpha t \nabla f(x)^T \Delta x$ **do**
3: $t = \beta t$.
4: **end while**

Remark 3.15. *The parameter α is usually set between 0.01 and 0.3, indicating that we allow a 1% to 30% reduction in f based on the linear forecast. The parameter β is often selected between 0.1 (representing a very rough search) and 0.8 (representing a more refined search).*

Definition 3.16. *(Gradient descent method) We construct our minimizing sequence and do iteration with descent method satisfying*

$$\nabla f\left(x^{(k)}\right)^T \Delta x^{(k)} < 0,$$

where we choose $\Delta x = -\nabla f(x)$.

Algorithm 4: Gradient descent method for minimization algorithm

1: Given a start point $x_0 \in \mathbf{dom}\, f$, current value $f(x_0)$, tolerance $\epsilon > 0$, iteration times $k = 0$
2: **while** $f(x) - p^* \geq \epsilon$ **do**
3: Determine a descent direction $\Delta x = -\nabla f(x)$.
4: Line search: Choose a step size $t > 0$ via exact or backtracking line search.
5: $x = x + t\Delta x$.
6: Calculate $f(x)$, $k = k + 1$.
7: **end while**

Proposition 3.17. *(Convergence analysis for exact line search) For gradient descent method, we have two important properties:*

1. *The convergence rate is at least linear convergence.*

2. *If the Hessian of f, near $x^\star$, has a large condition number, the gradient method does in fact require a large number of iterations.*

Proof. 1. Remember previously we proved the upper bound for f, i.e., for any $x, y \in S$,

$$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{M}{2}\|y - x\|_2^2,$$

We now let $y = x - t\nabla f(x)$, and fix x, consider it as the inequality for t, we will get

$$f(x - t\nabla f(x)) \leq f(x) - t\|\nabla f(x)\|_2^2 + \frac{Mt^2}{2}\|\nabla f(x)\|_2^2.$$

The righthand side is a simple quadratic function, which is minimized by $t = 1/M$, and has minimum value $f(x) - (1/(2M))\|\nabla f(x)\|_2^2$. Since left hand is equal to the new iteration value, we can write the inequality in a new way:

$$f(x^{k+1}) \leq f(x^k) - \frac{1}{2M}\|\nabla(f(x^k))\|_2^2.$$

Subtracting $p^\star$ from both sides, we get

$$f(x^{k+1}) - p^* \leq f(x^k) - p^* - \frac{1}{2M}\|\nabla(f(x^k))\|_2^2.$$

We combine this with $\|\nabla f(x)\|_2^2 \geq 2m\,(f(x) - p^\star$ and we will get

$$f(x^{k+1}) - p^\star \leq f(x^k) - p^* - \frac{1}{2M}\|\nabla(f(x^k))\|_2^2 \leq (1 - m/M)\left(f(x^k) - p^\star\right).$$

We could calculate the convergence rate now, it is linearly convergent since

$$\lim_{k\to\infty} \frac{|f(x^{k+1}) - p^*|}{|f(x^k) - p^*|} \leq 1 - \frac{m}{M} < 1.$$

2. Applying this inequality recursively, we find that

$$f\left(x^{(k)}\right) - p^\star \leq c^k \left(f\left(x^{(0)}\right) - p^\star\right),$$

where $c = 1 - m/M < 1$, which shows that $f\left(x^{(k)}\right)$ converges to $p^\star$ as $k \to \infty$. In particular, we could get the suitable iteration times by calculating $c^k \left(f\left(x^{(0)}\right) - p^\star\right) = \epsilon$. So, after at most

$$k = \lceil \frac{\log\left(\left(f\left(x^{(0)}\right) - p^\star\right)/\epsilon\right)}{\log(1/c)} \rceil$$

iterations of the gradient method with exact line search, the algorithm will stop and give the approximate best solution.

For large condition number bound M/m, assume $\frac{m}{M} = x \to 0$, by L'Hopital's Rule, we could approximate $log(1/c)$ to $\frac{m}{M}$. Since

$$\lim_{x\to 0} \frac{-\log(1-x)}{x} = \lim_{x\to 0} \frac{\frac{1}{1-x}}{1} = 1.$$

Our bound on the number of iterations required increases approximately linearly with increasing M/m.

So if the Hessian of f, near $x^\star$, has a large condition number, the gradient method does in fact require a large number of iterations [Bertsekas (2014)].

□

Proposition 3.18. *(Convergence analysis for backtracking line search) For gradient descent method, we have two important properties:*

1. *The convergence rate is at least linear convergence.*
2. *If the Hessian of f, near $x^\star$, has a large condition number, the gradient method does in fact require a large number of iterations.*

Proof. 1. Similarly, let $y = x - t\nabla f(x)$,

$$f(x - t\nabla f(x)) \leq f(x) - \alpha t\|\nabla f(x)\|_2^2,$$

we will then show that this inequality is always satisfied whenever $0 \leq t \leq 1/M$. First note that

$$0 \leq t \leq 1/M \Longrightarrow -t + \frac{Mt^2}{2} \leq -t/2.$$

Using this result and the bound we derived for last proof, we have, for $0 \leq t \leq 1/M$,

$$\begin{aligned} f(x - t\nabla f(x)) &\leq f(x) - t\|\nabla f(x)\|_2^2 + \frac{Mt^2}{2}\|\nabla(f(x))\|_2^2 \\ &\leq f(x) - (t/2)\|\nabla f(x)\|_2^2 \\ &\leq f(x) - \alpha t\|\nabla f(x)\|_2^2, \end{aligned}$$

since $\alpha < 1/2$.

Therefore we can use the property for the backtracking line search. Since the beginning step size is $t = 1$, if $1/M \geq 1$, we have

$$f(x - t\nabla f(x)) \leq f(x) - \alpha\|\nabla f(x)\|_2^2,$$

otherwise, we choose $t = \beta/M$, and we also have

$$f(x - t\nabla f(x)) \leq f(x) - (\beta\alpha/M)\|\nabla f(x)\|_2^2.$$

Putting these together, we always have

$$f(x - t\nabla f(x)) \leq f(x) - \min\{\alpha, \beta\alpha/M\}\|\nabla f(x)\|_2^2.$$

Since left hand side is the new iteration step, we could also write it as

$$f(x^{k+1}) \leq f(x^k) - \min\{\alpha, \beta\alpha/M\}\|\nabla f(x^k)\|_2^2.$$

Now we can proceed exactly as in the case of exact line search. We subtract $p^\star$ from both sides to get

$$f(x^{k+1}) - p^* \leq f(x^k) - p^* - \min\{\alpha, \beta\alpha/M\}\|\nabla f(x^k)\|_2^2.$$

and combine this with $\|\nabla f(x)\|_2^2 \geq 2m\,(f(x) - p^\star)$ to obtain

$$f(x^{k+1}) - p^\star \leq (1 - \min\{2m\alpha, 2\beta\alpha m/M\})\,\left(f(x^k) - p^\star\right).$$

We could calculate the convergence rate now, it is also linearly convergent since

$$\lim_{k\to\infty} \frac{|f(x^{k+1}) - p^*|}{|f(x^k) - p^*|} \leq 1 - \min\{2m\alpha, 2\beta\alpha m/M\} < 1.$$

2. Similarly, we also conclude that

$$f\left(x^{(k)}\right) - p^\star \leq c^k \left(f\left(x^{(0)}\right) - p^\star\right)$$

where

$$c = 1 - \min\{2m\alpha, 2\beta\alpha m/M\} < 1.$$

In particular, we could get the suitable iteration times by calculating $c^k \left(f\left(x^{(0)}\right) - p^\star\right) = \epsilon$ so, after at most

$$k = \lceil \frac{\log\left(\left(f\left(x^{(0)}\right) - p^\star\right)/\epsilon\right)}{\log(1/c)} \rceil$$

iterations of the gradient method with backtracking line search, the algorithm will stop and give the approximate best solution.

So if the Hessian of f, near $x^\star$, has a large condition number, $c = 1 - 2\beta\alpha\frac{m}{M}$ and assume $\frac{m}{M} = x \to 0$, by L'Hopital's Rule, we could approximate $log(1/c)$ to $2\beta\alpha\frac{m}{M}$. Since

$$\lim_{x\to 0} \frac{-\log(1-2\beta\alpha x)}{2\beta\alpha x} = \lim_{x\to 0} \frac{\frac{2\beta\alpha}{1-2\beta\alpha x}}{2\beta\alpha} = 1.$$

So we also find that if the Hessian of f, near $x^\star$, has a large condition number, the gradient method does in fact require a large number of iterations [Bertsekas (2014)].

□

Example 3.1.3. *We first provide a simple example. Use gradient descent method to find the minimum of quadratic function $f(x) = \frac{1}{2}(x_1^2 + \gamma x_2^2)$, where $\gamma > 0$. We now show the steps for exact line-search with beginning point at $x^0 = (\gamma, 1)$.*

$$x^{k+1} = x^k - \nabla f(x_k)t,$$

where $t = \mathrm{argmin}_s f(x - \nabla f(x_k)s)$.
We calculate $\nabla f(x) = [x_1, \gamma x_2]^T$ and the derivative of $f(x - \nabla f(x_k)s) = 0$, we then derive

$$(2s-2)x_1^2 + 2\gamma^2(\gamma s - 1)x_2^2 = 0.$$

So the minimum t should be $t = \frac{\gamma^2 x_2^2 + x_1^2}{x_1^2 + \gamma^3 x_2^2}$. Thus

$$x_1^{k+1} = (1-t)x_1^k = \frac{x_1\gamma^2 x_2^2(\gamma-1)}{x_1^2 + \gamma^3 x_2^2}, \text{ and } x_2^{k+1} = (1-\gamma t)x_2^k = \frac{(1-\gamma)x_1^2 x_2}{x_1^2 + \gamma^3 x_2^2}.$$

We then use induction to prove

$$x_1^k = \gamma\left(\frac{\gamma-1}{\gamma+1}\right)^k, \text{ and } x_2^k = \left(-\frac{\gamma-1}{\gamma+1}\right)^k.$$

Assume $n = k$ satisfies, then $n = k+1$ we have

$$x_1^{k+1} = \frac{\gamma\left(\frac{\gamma-1}{\gamma+1}\right)^k \gamma^2 \left(-\frac{\gamma-1}{\gamma+1}\right)^{2k}(\gamma-1)}{\gamma^2\left(\frac{\gamma-1}{\gamma+1}\right)^2 k + \gamma^3\left(\frac{\gamma-1}{\gamma+1}\right)^{2k}} = \gamma\left(\frac{\gamma-1}{\gamma+1}\right)^{k+1},$$

and

$$x_2^{k+1} = \frac{(1-\gamma)\left(\frac{\gamma-1}{\gamma+1}\right)^{2k}\gamma^2\left(-\frac{\gamma-1}{\gamma+1}\right)^k}{\gamma^2\left(\frac{\gamma-1}{\gamma+1}\right)^2 k + \gamma^3\left(\frac{\gamma-1}{\gamma+1}\right)^{2k}} = \left(-\frac{\gamma-1}{\gamma+1}\right)^{k+1}.$$

So we obtain

$$f(x^k) = \left(\frac{\gamma-1}{\gamma+1}\right)^{2k} f(x^0) = \frac{\gamma(\gamma-1)^{2k}}{2(\gamma+1)^{2k-1}}.$$

We then do a further discussion about its iteration steps. We let

$$f(x^k) = \left(\frac{\gamma-1}{\gamma+1}\right)^{2k} f(x^0) < \epsilon,$$

and derive

$$k > \frac{\log(\epsilon/f(0))}{\log(\frac{\gamma-1}{\gamma+1})^2}. \tag{3.1}$$

Notice that the Hessian matrix of f has eigenvalues $1, \gamma$, we obtain that $m = \min\{1,\gamma\}$, $M = \max\{1,\gamma\}$. So we write (3.1) as

$$k > \frac{\log(\epsilon/f(0))}{\log(\frac{1-m/M}{1+m/M})^2},$$

Assume $\frac{m}{M} = x \to 0$, by L'Hopital's Rule, we could approximate $\log(\frac{1-m/M}{1+m/M})^2$ to $-4\frac{m}{M}$. Since

$$\lim_{x\to 0} \frac{\log(\frac{1-x}{1+x})^2}{-4x} = \lim_{x\to 0} \frac{1}{1-x^2} = 1.$$

Therefore this example satisfies that the number of iteration steps grows approximately like $M/4m$.

Example 3.1.4. *We now consider another example,*

$$f(x_1, x_2) = e^{x_1+3x_2-0.1} + e^{x_1-3x_2-0.1} + e^{-x_1-0.1}.$$

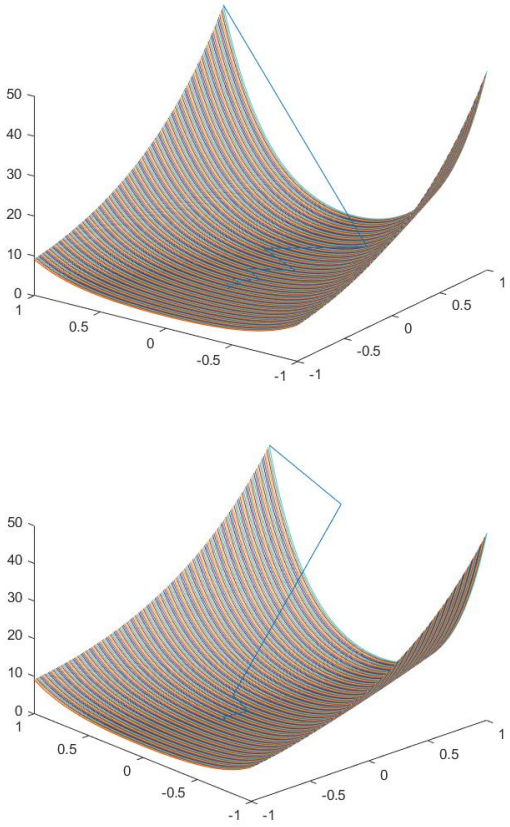
We now apply the gradient method with a backtracking line search and exact-line search to solve this problem, with $\alpha = 0.1$, $\beta = 0.7$. The Matlab code is shown in appendix and the graph of line search looks like (left graph is backtracking line search and right graph is exact-line search method).And the convergence rate for two methods are linearly convergent. Their error versus iteration is also shown below.

Remark 3.19. *1. The gradient method often exhibits approximately linear convergence.*

2. *The choice of backtracking parameters α, β has a noticeable but not dramatic effect on the convergence. An exact line search may improve the convergence of the gradient method to some extent, but the effect is not that salient while costing much time to compute derivative, so most of time we don't consider this method.*

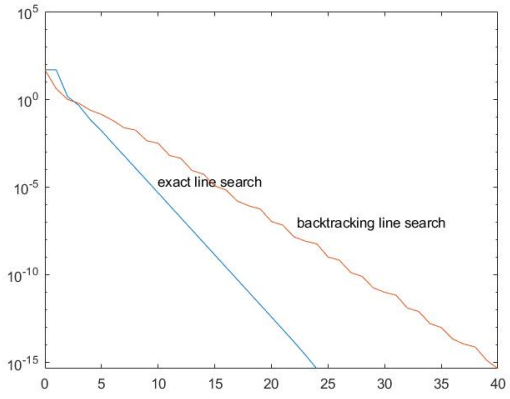
3. *The convergence rate really depends on the condition number of its Hessian matrix. When the condition number is large enough then the gradient method is quite slow and is not practical.*

Definition 3.20. *(Steepest descent method) Considering first order Taylor approximation of $f(x+v)$ around x as $f(x+v) \approx f(x) + \nabla f(x)^T v$, $\nabla f(x)^T v$ is the directional derivative of f at x in the direction v. If we choose v to make the directional derivative as negative as possible, we call this choosing idea as Steepest descent method.*



The first graph is backtracking line search, the second graph is exact line search, this briefly shows that both two method converge fast to get the optimal result.

Figure 3.1: Gradient descent example(3.1.4)



From this graph, it's easy to see that both two method have the same convergence rate. The error Exact line search gives seems better than backtracking line search, but not so salient.

Figure 3.2: graph of Error versus iteration

Definition 3.21. *(Normalized steepest descent direction) We define a normalized steepest descent direction (with respect to the norm $\|\cdot\|$) as*

$$\Delta x_{\text{nsd}} = \operatorname{argmin}\left\{\nabla f(x)^T v \mid \|v\| = 1\right\}.$$

A normalized steepest descent direction Δx_{nsd} is a step of unit norm that gives the largest decrease in the linear approximation of f.

Remark 3.22. *It is also convenient to consider a steepest descent step Δx_{sd} that is unnormalized, since*

$$\|\nabla f(x)\|_* = \max \frac{\langle \nabla f(x), v\rangle}{||v||} = \max_{||v||=1} \langle \nabla f(x), v\rangle = -\nabla f(x)^T \Delta x_{\text{nsd}},$$

we derive a new steepest descent direction:

$$\Delta x_{\text{sd}} = \|\nabla f(x)\|_* \Delta x_{\text{nsd}},$$

where $\|\cdot\|_*$ denotes the dual norm. Note that for the steepest descent step, we have

$$\nabla f(x)^T \Delta x_{\text{sd}} = \|\nabla f(x)\|_* \nabla f(x)^T \Delta x_{\text{nsd}} = -\|\nabla f(x)\|_*^2.$$

Algorithm 5: Steepest descent method

1: Given a start point $x_0 \in \operatorname{dom} f$, current value $f(x_0)$, tolerance $\epsilon > 0$, iteration times $k = 0$
2: **while** $f(x) - p^* \geq \epsilon$ **do**
3: Determine a descent direction Δx
4: Line search. Choose a step size $t > 0$
5: $x = x + t\Delta x_{sd}$
6: Calculate $f(x)$, $k = k + 1$
7: **end while**

Remark 3.23. *When taking ℓ_2-norm $\|\cdot\|_2$, we find that the steepest descent direction is simply the negative gradient, i.e., $\Delta x_{\text{sd}} = -\nabla f(x)$. Then the steepest descent method for the Euclidean norm is now the same as the gradient descent method.*

Remark 3.24. *When taking ℓ_∞-norm $\|\cdot\|_\infty$, we find that the steepest descent direction is $\Delta x_{\text{sd}} = -\|\nabla f(x)\|_1 \Delta x_{\text{nsd}}$. The normalized steepest descent direction x_{nsd} is now $\pm \mathbf{1}$, which satisfies its product with $\|\nabla f(x)\|_1$ is the minimized result.*

Remark 3.25. *When taking ℓ_1-norm $\|\cdot\|_1$, we find that the steepest descent direction is $\Delta x_{\text{sd}} = -\|\nabla f(x)\|_\infty \Delta x_{\text{nsd}}$. 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*

$$\Delta x_{\text{nsd}} = -\operatorname{sign}\left(\frac{\partial f(x)}{\partial x_i}\right) e_i,$$

where e_i is the i th standard basis vector. An unnormalized steepest descent step is then

$$\Delta x_{\text{sd}} = \Delta x_{\text{nsd}} \|\nabla f(x)\|_\infty = -\frac{\partial f(x)}{\partial x_i} e_i.$$

Remark 3.26. *When taking quadratic-norm for vector z, $\|z\|_P = (z^T P z)^{1/2} = \left\|P^{1/2} z\right\|_2$, where $P \in \mathbf{S}^n_{++}$, we find that the steepest descent direction is*

$$\Delta x_{\text{nsd}} = \text{argmin}\left\{\nabla f(x)^T v \mid ||v||_P = 1\right\} = \text{argmin}\left\{\nabla f(x)^T v \mid ||P^{1/2} v||_2 = 1\right\}.$$

Let $w = P^{1/2} v$, we obtain

$$\begin{aligned}
\Delta x_{\text{nsd}} &= \text{argmin}\left\{\nabla f(x)^T v \mid ||P^{1/2} v||_2 = 1\right\} \\
&= P^{-1/2} \min_{||w||_2 = 1} \langle \nabla f(x), P^{-1/2} w \rangle \\
&= P^{-1/2} \min_{||w||_2 = 1} \langle P^{-1/2} \nabla f(x), w \rangle \\
&= P^{-1/2} \frac{-P^{-1/2} \nabla f(x)}{||P^{-1/2} \nabla f(x)||_2} \\
&= \frac{-P^{-1} \nabla f(x)}{\sqrt{\nabla f(x)^T P^{-1} \nabla f(x)}}.
\end{aligned}$$

Since the dual norm for quadratic norm is $||z||_* = \left\|P^{-1/2} z\right\|_2$. Then the unnormalized steepest descent step is then

$$\Delta x_{\text{sd}} = \|\nabla f(x)\|_* \Delta x_{\text{nsd}} = \left\|P^{-1/2} \nabla f(x)\right\|_2 \Delta x_{\text{nsd}} = -P^{-1} \nabla f(x).$$

Proposition 3.27. *(Convergence rate for steepest descent method) For steepest descent method, the convergence rate is at least linear convergence.*

Proof. We will use the fact that any norm can be bounded in terms of the Euclidean norm, so there exists constants $\gamma, \tilde{\gamma} \in (0, 1]$ such that

$$\|x\| \geq \gamma \|x\|_2, \quad \|x\|_* \geq \tilde{\gamma} \|x\|_2$$

Assuming f is strongly convex on the initial sublevel set S. Remember previously we proved the upper bound for f, i.e., for any $x, y \in S$,

$$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{M}{2} \|y - x\|_2^2,$$

We now let $y = x + t\Delta x_{\text{sd}}$, and fix x, consider it as the inequality for t, we will get

$$f(x + t\Delta x_{\text{sd}}) \leq f(x) + t \nabla f(x)^T \Delta x_{\text{sd}} + \frac{M \|\Delta x_{\text{sd}}\|_2^2}{2} t^2.$$

We then write the new inequality with arbitrary norm.

$$\begin{aligned}
f\left(x + t\Delta x_{\text{sd}}\right) &\leq f(x) + t \nabla f(x)^T \Delta x_{\text{sd}} + \frac{M \|\Delta x_{\text{sd}}\|_2^2}{2} t^2 \\
&\leq f(x) + t \nabla f(x)^T \Delta x_{\text{sd}} + \frac{M \|\Delta x_{\text{sd}}\|^2}{2\gamma^2} t^2 \\
&= f(x) - t \|\nabla f(x)\|_*^2 + \frac{M}{2\gamma^2} t^2 \|\nabla f(x)\|_*^2.
\end{aligned}$$

The right hand side is a simple quadratic function, which is minimized by $\hat{t} = \gamma^2 / M$, and has minimum value $f(x) - \frac{\gamma^2}{2M} \|\nabla f(x)\|_*^2$. Since left hand is equal to the new iteration value, we can write the inequality in a new way:

$$f(x^{k+1}) \leq f(x^k) - \frac{\gamma^2}{2M} \|\nabla f(x^k)\|_*^2.$$

Since $\alpha < 1/2$, we obtain

$$f(x^{k+1}) \leq f(x^k) - \frac{\gamma^2}{2M}\|\nabla f(x^k)\|_*^2 \leq f(x^k) - \frac{\alpha\gamma^2}{M}\|\nabla f(x^k)\|_*^2$$

With backtracking line search, we require step size $t \geq \min\{1, \beta\gamma^2/M\}$, and we have

$$\begin{aligned} f\left(x^{k+1}\right) &\leq f(x^k) - \alpha \min\left\{1, \beta\gamma^2/M\right\} \|\nabla f(x^k)\|_*^2 \\ &\leq f(x^k) - \alpha\tilde{\gamma}^2 \min\left\{1, \beta\gamma^2/M\right\} \|\nabla f(x^k)\|_2^2, \end{aligned}$$

where for second inequality, we also use the property that any norm can be bounded in terms of the Euclidean norm.

Subtracting $p^\star$ from both sides and combine this with $\|\nabla f(x)\|_2^2 \geq 2m\left(f(x) - p^\star\right.$ and we will get

$$f\left(x^{k+1}\right) - p^\star \leq c\left(f(x^k) - p^\star\right),$$

where

$$c = 1 - 2m\alpha\tilde{\gamma}^2 \min\left\{1, \beta\gamma^2/M\right\} < 1.$$

We could calculate the convergence rate now, it is linearly convergent since

$$\lim_{k\to\infty} \frac{|f(x^{k+1}) - p^*|}{|f(x^k) - p^*|} \leq 1 - 2m\alpha\tilde{\gamma}^2 \min\left\{1, \beta\gamma^2/M\right\} < 1.$$

□

Newton's method

In this subsection, we will introduce a new descent method called Newton's method, quite different to previous methods, this method converges at quadratic rate.

If we do second-order Taylor expansion for function f, i.e.,

$$\widehat{f}(x+v) = f(x) + \nabla f(x)^T v + \frac{1}{2} v^T \nabla^2 f(x) v,$$

where $\widehat{f}$ is the second-order Taylor expansion and is convex quadratic function of v, so by previous proof, it is minimized if we choose $v = -\nabla^2 f(x)^{-1}\nabla f(x)$, based on on this property, we introduce a new definition called Newton step.

Definition 3.28. *(Newton step) For $x \in \operatorname{dom} f$, the vector*

$$\Delta x_{\mathrm{nt}} = -\nabla^2 f(x)^{-1}\nabla f(x)$$

is called the Newton step. Positive definiteness of $\nabla^2 f(x)$ implies that

$$\nabla f(x)^T \Delta x_{\mathrm{nt}} = -\nabla f(x)^T \nabla^2 f(x)^{-1}\nabla f(x) < 0,$$

unless $\nabla f(x) = 0$, so the Newton step is a descent direction (unless x is optimal).

Proposition 3.29. *Newton step is independent of linear changes of coordinates.*

Proof. Suppose $A \in \mathbf{R}^{n\times n}$ is nonsingular, and define $\bar{f}(y) = f(Ay)$. Then we have

$$\begin{aligned} \bar{f}(y + \nabla y) &= f(Ay + A\nabla y) = f(Ay) + \langle \nabla f(Ay), A\nabla y\rangle + \frac{1}{2}(A\nabla y)^T \nabla^2 f(Ay)(A\nabla y) + \mathrm{O(y^2)} \\ &= f(Ay) + \langle A^T \nabla f(x), \nabla y\rangle + \frac{1}{2}(\nabla y)^T A^T \nabla^2 f(Ay) A(\nabla y) + \mathrm{O(y^2)} \\ &= \bar{f}(y) + \langle \nabla \bar{f}(y), \nabla y\rangle + \frac{1}{2}(\nabla y)^T \nabla^2 \bar{f}(y)\nabla y + \mathrm{O(y^2)}\,. \end{aligned}$$

So we obtain that

$$\nabla \bar{f}(y) = A^T \nabla f(x), \quad \nabla^2 \bar{f}(y) = A^T \nabla^2 f(x) A,$$

where $x = Ay$. The Newton step for $\bar{f}$ at y is therefore

$$\begin{aligned} \Delta y_{\text{nt}} &= -\left(A^T \nabla^2 f(x) A\right)^{-1} \left(A^T \nabla f(x)\right) \\ &= -A^{-1} \nabla^2 f(x)^{-1} \nabla f(x) \\ &= A^{-1} \Delta x_{\text{nt}}, \end{aligned}$$

where Δx_{nt} is the Newton step for f at x. Hence the Newton steps of f and $\bar{f}$ are related by the same linear transformation, and

$$x + \Delta x_{\text{nt}} = A\left(y + \Delta y_{\text{nt}}\right).$$

□

Definition 3.30. *(The Newton decrement) If the quantity satisfies the form*

$$\lambda(x) = \left(\nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x)\right)^{1/2}.$$

We call it as Newton decrement at x.

Proposition 3.31. *Some propositions with Newton decrement.*

1. $\lambda(x) = \left(\Delta x_{\text{nt}}^T \nabla^2 f(x) \Delta x_{\text{nt}}\right)^{1/2}$.
2. $\nabla f(x)^T \Delta x_{\text{nt}} = -\lambda(x)^2$.
3. *Newton decrement is independent of linear changes of coordinates.*
4. *Newton decrement can be used for stopping criterion, i.e., $\lambda^2/2 \leq \epsilon$.*

Proof. Property 1-2 can be proved directly by calculation, for property 3, define $\bar{f}(y) = f(Ay)$, so the Newton decrement for $\bar{f}$ is

$$\begin{aligned} \lambda(y) &= \left(\nabla \bar{f}(y)^T \nabla^2 \bar{f}(y)^{-1} \nabla \bar{f}(y)\right)^{1/2} \\ &= \left(A^T \nabla f(x)^T (A^T \nabla^2 f(x) A)^{-1} A^T \nabla f(x)\right)^{1/2} \\ &= \left(\nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x)\right)^{1/2} \\ &= \lambda(x). \end{aligned}$$

For property 4, since

$$\widehat{f}(x+v) - f(x) = \nabla f(x)^T v + \frac{1}{2} v^T \nabla^2 f(x) v, \tag{3.2}$$

where $\widehat{f}$ is the second-order Taylor expansion for f, and Newton step provides the minimum value for (3.2), we derive

$$\lambda^2/2 = \widehat{f}(x + \Delta x_{nt}) - f(x).$$

So we can use this as stopping criterion. □

Algorithm 6: Newton's method

1: Given a start point $x_0 \in \mathbf{dom}\, f$, current value $f(x_0)$, tolerance $\epsilon > 0$, iteration times $k = 0$
2: **while** true **do**
3: Compute the Newton step and decrement.

$$\Delta x_{\text{nt}} := -\nabla^2 f(x)^{-1} \nabla f(x); \quad \lambda^2 := \nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x).$$

4: **if** $\lambda^2/2 \leq \epsilon$ **then** break
5: **end if**
6: Line search. Choose a step size $t > 0$
7: $x = x + t\Delta x_{nt}$
8: Calculate $f(x)$, $k = k + 1$
9: **end while**

Remark 3.32. *Newton's method can be divided into two stage, the first stage is often called as damped Newton stage and the second stage is called as quadratic convergent stage. During first stage, we need backtracking line search to find suitable step size, while for second stage, a fixed step size $t = 1$ is always valid. We will further discuss this in the following theorem and its proof.*

Theorem 3.33. *(Convergence rate for Newton's method) Assume f is twice continuously differentiable, and strongly convex. In addition, if Hessian of f is Lipschitz continuous on S with constant L, i.e.,*

$$\left\|\nabla^2 f(x) - \nabla^2 f(y)\right\|_2 \leq L\|x - y\|_2, \tag{3.3}$$

for all $x, y \in S$, then there are two stages of Newton's method based on absolute value of $\left\|\nabla f\left(x^{(k)}\right)\right\|_2$, where $x^{(k)}$ is the k step of iteration in the Newton's method and there are numbers η and γ with $0 < \eta \leq m^2/L$ and $\gamma > 0$ such that

- *If $\left\|\nabla f\left(x^{(k)}\right)\right\|_2 \geq \eta$, then the method is at damped Newton stage and we have*

$$f\left(x^{(k+1)}\right) - f\left(x^{(k)}\right) \leq -\gamma.$$

- *If $\left\|\nabla f\left(x^{(k)}\right)\right\|_2 < \eta$, then the method is at quadratic convergent stage and we select step size $t = 1$ with the backtracking line search and*

$$\frac{L}{2m^2}\left\|\nabla f\left(x^{(k+1)}\right)\right\|_2 \leq \left(\frac{L}{2m^2}\left\|\nabla f\left(x^{(k)}\right)\right\|_2\right)^2.$$

Proof. We will divide this proof into three parts. The first part contains some lemmas to be used later. The second part is the proof for damped Newton stage, and the third part is for quadratic convergent stage.

1. Since f is twice continuously differentiable, and strongly convex then we have two constant m, M such that $mI \preceq \nabla^2 f(x) \preceq MI$ for $x \in S$. Based on this, we directly derive two properties:

$$\lambda(x)^2 = \Delta x_{\text{nt}}^T \nabla^2 f(x) \Delta x_{\text{nt}}, \; m\left\|\Delta x_{\text{nt}}\right\|_2^2 \leq \lambda(x)^2 \leq M\left\|\Delta x_{\text{nt}}\right\|_2^2. \tag{3.4}$$

$$\lambda(x)^2 = \nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x), \; (1/M)\|\nabla f(x)\|_2^2 \leq \lambda(x)^2 \leq (1/m)\|\nabla f(x)\|_2^2. \tag{3.5}$$

2. Assume $\|\nabla f(x)\|_2 \geq \eta$. We check the next iteration step with backtracking line search.

$$\begin{aligned} f\left(x + t\Delta x_{\text{nt}}\right) &\leq f(x) + t\nabla f(x)^T \Delta x_{\text{nt}} + \frac{M\left\|\Delta x_{\text{nt}}\right\|_2^2}{2} t^2 \\ &= f(x) - t\lambda(x)^2 + \frac{M}{2} t^2 \lambda(x)^2 \\ &\leq f(x) - \alpha t \lambda(x)^2, \end{aligned}$$

where we use proposition (3.31) and (3.4). Therefore we obtain $t \leq \frac{2}{M}(1-\alpha)$, so the range of t should be $t \in \left[\frac{2\beta}{M}(1-\alpha), \frac{2}{M}(1-\alpha)\right]$. Then we derive

$$\begin{aligned} f\left(x^{(k+1)}\right) - f\left(x^{(k)}\right) &\leq -\alpha t \lambda(x)^2 \\ &\leq -2\alpha\beta \frac{1-\alpha}{M} \lambda(x)^2 \\ &\leq -2\alpha\beta \frac{1-\alpha}{M^2} \|\nabla f(x)\|_2^2 \\ &\leq -2\alpha\beta\eta^2 \frac{1-\alpha}{M^2}, \end{aligned}$$

Thus we obtain $\gamma = 2\alpha\beta\eta^2 \frac{1-\alpha}{M^2}$.

3. Assume $\|\nabla f(x)\|_2 < \eta$. We first show that the backtracking line search selects unit steps, provided

$$\eta \leq 3(1-2\alpha)\frac{m^2}{L}.$$

By the Lipschitz condition (3.3), we have, for $t \geq 0$,

$$\left\|\nabla^2 f\left(x + t\Delta x_{\text{nt}}\right) - \nabla^2 f(x)\right\|_2 \leq tL\left\|\Delta x_{\text{nt}}\right\|_2,$$

and therefore

$$\left|\Delta x_{\text{nt}}^T \left(\nabla^2 f\left(x + t\Delta x_{\text{nt}}\right) - \nabla^2 f(x)\right) \Delta x_{\text{nt}}\right| \leq tL\left\|\Delta x_{\text{nt}}\right\|_2^3.$$

With $\tilde{f}(t) = f\left(x + t\Delta x_{\text{nt}}\right)$, we have $\tilde{f}'(t) = \nabla f\left(x + t\Delta x_{\text{nt}}\right)\Delta x_{\text{nt}}$, and $\tilde{f}''(t) = \Delta x_{\text{nt}}^T \nabla^2 f\left(x + t\Delta x_{\text{nt}}\right)\Delta x_{\text{nt}}$, so the inequality above is

$$\left|\tilde{f}''(t) - \tilde{f}''(0)\right| \leq tL\left\|\Delta x_{\text{nt}}\right\|_2^3.$$

We will use this inequality to determine an upper bound on $\tilde{f}(t)$. We start with

$$\tilde{f}''(t) \leq \tilde{f}''(0) + tL\left\|\Delta x_{\text{nt}}\right\|_2^3 \leq \lambda(x)^2 + t\frac{L}{m^{3/2}}\lambda(x)^3$$

where we use $\tilde{f}''(0) = \lambda(x)^2$ and (3.4). We integrate the inequality to get

$$\begin{aligned} \tilde{f}'(t) &\leq \tilde{f}'(0) + t\lambda(x)^2 + t^2\frac{L}{2m^{3/2}}\lambda(x)^3 \\ &= -\lambda(x)^2 + t\lambda(x)^2 + t^2\frac{L}{2m^{3/2}}\lambda(x)^3, \end{aligned}$$

using $\tilde{f}'(0) = -\lambda(x)^2$. We integrate once more to get

$$\tilde{f}(t) \leq \tilde{f}(0) - t\lambda(x)^2 + t^2\frac{1}{2}\lambda(x)^2 + t^3\frac{L}{6m^{3/2}}\lambda(x)^3.$$

Finally, notice that $\tilde{f}(0) = f(x)$, and we take $t = 1$ to obtain

$$f\left(x + \Delta x_{\text{nt}}\right) \leq f(x) - \frac{1}{2}\lambda(x)^2 + \frac{L}{6m^{3/2}}\lambda(x)^3. \tag{3.6}$$

Since $\|\nabla f(x)\|_2 \leq \eta \leq 3(1-2\alpha)m^2/L$. By (3.5), we can obtain

$$\lambda(x) \leq 3(1-2\alpha)m^{3/2}/L$$

and by (3.6) we have

$$\begin{aligned} f\left(x+\Delta x_{\mathrm{nt}}\right) &\leq f(x)-\lambda(x)^2\left(\frac{1}{2}-\frac{L\lambda(x)}{6m^{3/2}}\right) \\ &\leq f(x)-\alpha\lambda(x)^2 \\ &= f(x)+\alpha\nabla f(x)^T\Delta x_{\mathrm{nt}} \end{aligned}$$

which shows that the unit step $t=1$ is accepted by the backtracking line search. We then use this unit step to prove the convergence rate is quadratic.
Since $x_{t+1}=x_t--\nabla^2 f\left(x_t\right)^{-1}\nabla f\left(x_t\right)$, subtracting both sides with optimal point x^*, and we notice that $\nabla f\left(x^*\right)=0$. So we have

$$\begin{aligned} x_{t+1}-x^* &= x_t-x^*-\nabla^2 f\left(x_t\right)^{-1}\nabla f\left(x_t\right) \\ &= x_t-x^*+\nabla^2 f\left(x_t\right)^{-1}\left(\nabla f\left(x^*\right)-\nabla f\left(x_t\right)\right). \end{aligned}$$

Define $g(t)=\nabla f\left(x_t+t\left(x^*-x_t\right)\right)$, so we obtain $g'(t)=\nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)\left(x^*-x_t\right)$. Thus we have

$$\begin{aligned} \nabla f\left(x^*\right)-\nabla f\left(x_t\right) &= g(1)-g(0) \\ &= \int_0^1 \nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)\left(x^*-x_t\right)dt. \end{aligned}$$

So we can obtain

$$\begin{aligned} x_{t+1}-x^* &= x_t-x^*+\nabla^2 f\left(x_t\right)^{-1}\int_0^1 \nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)\left(x^*-x_t\right)dt \\ &= \nabla^2 f\left(x_t\right)^{-1}\left(\nabla^2 f\left(x_t\right)\left(x_t-x^*\right)+\int_0^1 \nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)\left(x^*-x_t\right)dt\right) \\ &= \nabla^2 f\left(x_t\right)^{-1}\int_0^1 \left(\nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)-\nabla^2 f\left(x_t\right)\right)\left(x^*-x_t\right)dt. \end{aligned}$$

Taking the 2-norm of both sides of the equation, and applying the Cauchy-Schwarz inequality and $\left\|\nabla^2 f\left(x_t\right)^{-1}\right\|_2 \leq \frac{1}{m}$, we obtain

$$\begin{aligned} \left\|x_{t+1}-x^*\right\|_2 &= \left\|\nabla^2 f\left(x_t\right)^{-1}\int_0^1 \left(\nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)-\nabla^2 f\left(x_t\right)\right)\left(x^*-x_t\right)dt\right\|_2 \\ &\leq \left\|\nabla^2 f\left(x_t\right)^{-1}\right\|_2\left\|\int_0^1 \left(\nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)-\nabla^2 f\left(x_t\right)\right)\left(x^*-x_t\right)dt\right\|_2 \\ &\leq \frac{1}{m}\int_0^1 \left\|\left(\nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)-\nabla^2 f\left(x_t\right)\right)\right\|_2\left\|\left(x^*-x_t\right)\right\|_2 dt \\ &= \frac{1}{m}\left\|x^*-x_t\right\|\int_0^1 \left\|\left(\nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)-\nabla^2 f\left(x_t\right)\right)\right\|_2 dt \end{aligned}$$

By Lipschitz condition (3.3), we get

$$\begin{aligned} \int_0^1 \left\|\nabla^2 f\left(x_t+t\left(x^*-x_t\right)\right)-\nabla^2 f\left(x_t\right)\right\|_2 dt &\leq \int_0^1 L\left\|t\left(x^*-x_t\right)\right\|_2 dt \\ &= \frac{L}{2}\left\|x^*-x_t\right\|_2 \end{aligned}$$

Finally, we will derive

$$\|x_{t+1} - x^*\|_2 \leq \frac{1}{m} \|x^* - x_t\|_2 \frac{L}{2} \|x^* - x_t\|_2 = \frac{L}{2m} \|x^* - x_t\|_2^2$$

Thus the convergence rate is quadratic.

We then examine $\eta \leq m^2/L$. Similarly, applying the Lipschitz condition, we have

$$\begin{aligned}
\left\|\nabla f\left(x^{k+1}\right)\right\|_2 &= \left\|\nabla f\left(x^k + \Delta x_{\mathrm{nt}}\right) - \nabla f(x^k) - \nabla^2 f(x^k)\Delta x_{\mathrm{nt}}\right\|_2 \\
&= \left\|\int_0^1 \left(\nabla^2 f\left(x^k + t\Delta x_{\mathrm{nt}}\right) - \nabla^2 f(x^k)\right)\Delta x_{\mathrm{nt}} dt\right\|_2 \\
&\leq \frac{L}{2}\left\|\Delta x_{\mathrm{nt}}\right\|_2^2 \\
&= \frac{L}{2}\left\|\nabla^2 f(x^k)^{-1}\nabla f(x^k)\right\|_2^2 \\
&\leq \frac{L}{2m^2}\|\nabla f(x^k)\|_2^2
\end{aligned}.$$

This gives us an upper bound for η.

In conclusion, the algorithm selects unit steps if $\left\|\nabla f\left(x^{(k)}\right)\right\|_2 < \eta$, where

$$\eta = \min\{1, 3(1-2\alpha)\}\frac{m^2}{L}$$

□

Remark 3.34. *Generally speaking, the Newton's method consists of two stage, the first stage we use backtracking line search and its convergence rate is at least linear (proved before), and the second stage the convergence rate is quadratic. We then estimate the total number of iteration steps. We derive an upper bound on the number of iterations in the damped Newton stage. Since f decreases by at least γ at each iteration, the number k of damped Newton steps is at most*

$$k = \frac{f\left(x^{(0)}\right) - p^\star}{\gamma}.$$

Then Newton's method will go into second stage and we have the formula

$$\frac{L}{2m^2}\left\|\nabla f\left(x^{(n+1)}\right)\right\|_2 \leq \left(\frac{L}{2m^2}\left\|\nabla f\left(x^{(n)}\right)\right\|_2\right)^2, n = k+1, k+2, \ldots$$

Applying this inequality recursively, we find that for $l \geq k$,

$$\frac{L}{2m^2}\left\|\nabla f\left(x^{(l)}\right)\right\|_2 \leq \left(\frac{L}{2m^2}\left\|\nabla f\left(x^{(k)}\right)\right\|_2\right)^{2^{l-k}} \leq \left(\frac{1}{2}\right)^{2^{l-k}},$$

and combine this with $\|\nabla f(x)\|_2^2 \geq 2m\left(f(x) - p^\star\right)$, and $\|\nabla f(x)\|_2 \leq \eta \leq m^2/L$, we finally obtain

$$f\left(x^{(l)}\right) - p^\star \leq \frac{1}{2m}\left\|\nabla f\left(x^{(l)}\right)\right\|_2^2 \leq \frac{m^3}{L^2}\left(\frac{1}{2}\right)^{2^{l-k}}.$$

Define $\epsilon_0 = m^3/L^2$. To calculate the iteration steps for quadratic term we should satisfy $f(x) - p^\star \leq \epsilon$, i.e., $\epsilon_0\left(\frac{1}{2}\right)^{2^{l-k}} \leq \epsilon$.So the iteration steps for this part is

$$l - k = \log_2\log_2\left(\epsilon_0/\epsilon\right).$$

Overall, the total number of iterations until $f(x) - p^\star \leq \epsilon$ is bounded above by

$$\frac{f\left(x^{(0)}\right) - p^\star}{\gamma} + \log_2 \log_2 \left(\epsilon_0/\epsilon\right).$$

Substitute $\gamma = 2\alpha\beta\eta^2 \frac{1-\alpha}{M^2}$, and $\eta = \min\{1, 3(1-2\alpha)\} \frac{m^2}{L}$, we finally get the total number of iteration is

$$\frac{M^2L^2/m^4}{2\alpha\beta(1-\alpha)\min\{1, 9(1-2\alpha)^2\}} \left(f\left(x^{(0)}\right) - p^\star\right) + \log_2 \log_2 \left(\frac{m^3}{L^2\epsilon}\right).$$

Remark 3.35. *Summary of Newton's method.*

1. *Convergence rate of Newton's method is quite fast, and quadratic near $x^\star$.*
2. *Newton's method is not affected by the choice of coordinates or the condition number of the sublevel sets.*
3. *Newton' s method need a high cost of forming and storing the Hessian, and calculating the Newton step. Sometimes, we may even could not find the Hessian matrix for the object function.*

Further study with Newton's method

In this section, we will introduce some special functions called self-concordant function, these functions can preserve some good properties and work as an alternative for conditions like $mI \preceq \nabla^2 f(x) \preceq MI$, and $\|\nabla^2 f(x) - \nabla^2 f(y)\|_2 \leq L\|x - y\|_2$.

Definition 3.36. *(Self-concordant function on $\mathbf{R}$) A convex function $f : \mathbf{R} \to \mathbf{R}$ is self-concordant if it satisfies*

$$|f'''(x)| \leq 2f''(x)^{3/2}$$

for all $x \in \operatorname{dom} f$.

Definition 3.37. *(Self-concordant function on $\mathbf{R}^n$) We say a function $f : \mathbf{R}^n \to \mathbf{R}$ is self-concordant if it is self-concordant along each line in its domain, i.e., if the function $\tilde{f}(t) = f(x + ay)$ is a self-concordant function of a for all $x \in \operatorname{dom} f$ and for all y.*

Proposition 3.38. *(Operations to preserve self-concordant) Self-concordance is preserved by following operations.*

1. *Self-concordance is preserved with scaling by a factor bigger than 1.*
2. *Self-concordance is preserved under addition, i.e., If f_1, f_2 are self-concordant, then $f_1 + f_2$ is self-concordant.*
3. *Self-concordance is preserved under affine function, i.e., if $f : \mathbf{R}^n \to \mathbf{R}$ is self-concordant, and $A \in \mathbf{R}^{n\times m}, b \in \mathbf{R}^n$, then $f(Ax + b)$ is self-concordant.*

Proposition 3.39. *Lower and upper bounds on $f''(t)$ for strictly convex self-concordant function f is:*

$$\frac{f''(0)}{\left(1 + tf''(0)^{1/2}\right)^2} \leq f''(t) \leq \frac{f''(0)}{\left(1 - tf''(0)^{1/2}\right)^2}$$

Lemma 3.40. $\lambda\left(x^{k+1}\right) \leq \frac{\lambda(x^k)^2}{(1-\lambda(x^k))^2}$, *where* x^k *stands for* k *iteration, and* λ *is the newton decrement.*

Theorem 3.41. *(Improved theorem for Newton's method) Newton's method is a global convergent algorithm, and if the function is strictly convex and self-concordant, then two stages of Newton's method can be based on value of* $\lambda(x)$*,i.e., there are numbers* η *and* γ *with* $0 < \eta \leq 1 - \frac{\sqrt{2}}{2}$ *and* $\gamma > 0$ *such that*

- *If* $\lambda(x) \geq \eta$*, then the method is at damped Newton stage and we have*
$$f\left(x^{(k+1)}\right) - f\left(x^{(k)}\right) \leq -\gamma.$$
- *If* $\lambda(x) < \eta$*, then the method is at quadratic convergent stage and we select step size* $t = 1$ *with the backtracking line search and*
$$2\lambda\left(x^{(k+1)}\right) \leq \left(2\lambda\left(x^{(k)}\right)\right)^2.$$

Proof. Based on lemma (3.40) and the proposition(3.39), we then provide a improved proof for Newton's method without condition number for matrix and Lipschitz continuous condition. The whole proof is divided into two parts similar to original proof.

1. Let $\tilde{f}(t) = f\left(x + t\Delta x_{\text{nt}}\right)$, so we have
$$\tilde{f}'(0) = -\lambda(x)^2, \quad \tilde{f}''(0) = \lambda(x)^2.$$
If we integrate the upper bound in proposition (3.39) twice, we obtain an upper bound for $\tilde{f}(t)$:
$$\begin{aligned} \tilde{f}(t) &\leq \tilde{f}(0) + t\tilde{f}'(0) - t\tilde{f}''(0)^{1/2} - \log\left(1 - t\tilde{f}''(0)^{1/2}\right) \\ &= \tilde{f}(0) - t\lambda(x)^2 - t\lambda(x) - \log(1 - t\lambda(x)), \end{aligned} \tag{3.7}$$
valid for $0 \leq t < 1/\lambda(x)$. We then check it by backtracking line search:
$$\tilde{f}(\hat{t}) \leq \tilde{f}(0) - \hat{t}\lambda(x)^2 - \hat{t}\lambda(x) - \log(1 - \hat{t}\lambda(x)) \leq \tilde{f}(0) - \alpha\lambda(x)^2\hat{t}.$$
Let $t\lambda(x) = 1 - m$, we simplify the inequality as
$$(\alpha\lambda(x) - \lambda - 1)m + \log(m) \geq \alpha\lambda(x) - \lambda(x) - 1.$$
Additionally, let $\alpha\lambda(x) - \lambda - 1 = y < 0$, we only need to find suitable m satisfying $m + \log m/y \leq 1$. Define $g(m) = m + \log m/y$, notice that g is decreasing first, then increasing and $g(1) = 0$. We choose $1/(1 + \lambda(x)) < 1/(1 + \lambda(x) - \alpha\lambda(x))$. Notice that
$$(1 + \lambda(x) - \alpha\lambda(x))\frac{1}{1 + \lambda(x)} - \log(\frac{1}{1 + \lambda(x)}) \leq 1 + \lambda(x) - \alpha\lambda(x)$$
always satisfies when $\alpha \leq 1/2$, so we derive when $m \in [0, \frac{1}{1+\lambda(x)}]$, the inequality always satisfies, therefore we obtain
$$t \in [0, \frac{1}{1 + \lambda(x)}].$$
Thus by backtracking line search, we choose $t \in [\frac{\beta}{1+\lambda(x)}, \frac{1}{1+\lambda(x)}]$, and we get
$$\gamma = \alpha\beta\frac{\eta^2}{1 + \eta}.$$

2. We will then show the improved proof for quadratic convergent part, we choose a suitable

$$\eta = (1-2\alpha)/(2+\sqrt{2})$$

which satisfies $0 < \eta < 1 - \frac{\sqrt{2}}{2}$, since $0 < \alpha < 1/2$. We will first prove that the upper bound (3.7) implies that a unit step $t = 1$ always satisfies backtracking line search,

$$\begin{aligned}\tilde{f}(1) &\leq \tilde{f}(0) - \lambda(x)^2 - \lambda(x) - \log(1-\lambda(x)) \\ &\leq \tilde{f}(0) - \alpha\lambda(x)^2.\end{aligned}$$

This inequality always satisfies for $0 \leq x \leq 1/2$. Additionally, by lemma (3.40), since $\lambda(x) < 1 - \frac{\sqrt{2}}{2} < 1$, and $x^{k+1} = x^k - \nabla^2 f(x^k)^{-1}\nabla f(x^k)$, we have

$$\lambda\left(x^{k+1}\right) \leq \frac{\lambda(x^k)^2}{(1-\lambda(x^k))^2}.$$

Notice that $\lambda(x^k) \leq 1 - \frac{\sqrt{2}}{2}$, we simplify the inequality and obtain

$$\lambda\left(x^{k+1}\right) \leq 2\lambda(x^k)^2,$$

□

Remark 3.42. *Similar to Remark*(3.34)*, the total number of iterations until* $f(x) - p^\star \leq \epsilon$ *now is bounded above by*

$$\frac{f\left(x^{(0)}\right) - p^\star}{\gamma} + \log_2\log_2\left(\epsilon_0/\epsilon\right).$$

Substitute $\gamma = \alpha\beta\frac{\eta^2}{1+\eta}$*, and* $\eta = 1 - \frac{\sqrt{2}}{2}$*, we finally get the total number of iteration is*

$$\frac{4-\sqrt{2}}{\alpha\beta(3-2\sqrt{2})}\left(f\left(x^{(0)}\right) - p^\star\right) + \log_2\log_2\left(\frac{\epsilon_0}{\epsilon}\right).$$

Example 3.1.5. *We now give an example to compare the effect between gradient descent method and Newton's method. Consider the unconstrained problem*

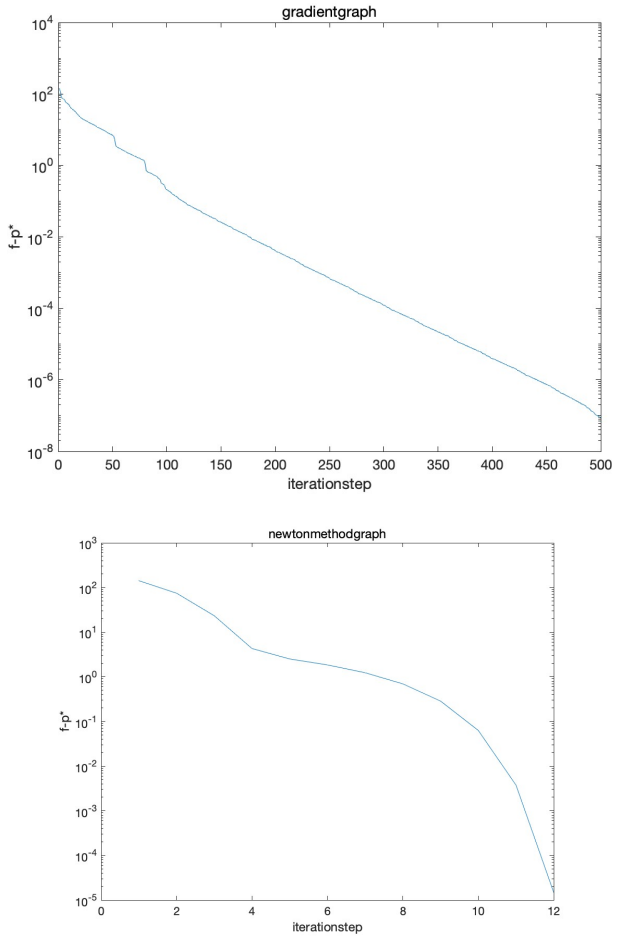
$$\textit{minimize } f(x) = -\sum_{i=1}^{m}\log\left(1 - a_i^T x\right) - \sum_{i=1}^{n}\log\left(1 - x_i^2\right),$$

with variable $x \in \mathbf{R}^n$*, and* $\operatorname{dom} f = \{x \,|a_i^T x < 1, i = 1,\ldots,m, |\, x_i \,| < 1, i = 1,\ldots,n\}$*. For this example, we derive* a_i *by random number from* $[0,1]$ *satisfying Gauss distribution and choose the zero point as our initial point.The Matlab code in shown in appendix part and the graph is shown here.*

Conjugate Gradient Method

In this section we will mainly discussed a new new method called Conjugate Gradient Method. This method is quite effective when solving problems like quadratic optimization problems based on special properties with conjugation. We will first apply this method to solve traditional quadratic optimization problems and then consider some more complex and nonlinear optimization problems with this method.

We first introduce the definition of conjugation and its basic properties.



From this graph, it's easy to see that Newton's method converges much faster than gradient descent method, and there is a salient decreasing trend in Newton's method when error bound between $f(x^k)$ and p^* is small, this tells us the Newton's method begins the quadratic convergent stage.

Figure 3.3: graph of Error versus iteration

Definition 3.43. *(A-Conjugate) Let A be a symmetric positive-definite $n \times n$ matrix. For two n-vectors v and w, define the A-inner product*

$$(v, w)_A = v^T A w.$$

The vectors v and w are A-conjugate if $(v, w)_A = 0$.

Definition 3.44. *(Residual of the linear system) When using iterative method for solving a linear system of equations. We define the residual of the function at k iteration is*

$$r_k = Ax_k - b.$$

In conjugate gradient method, we still consider the minimized sequences(3.11). Specially, we write this minimized sequences as the form

$$x_{k+1} = x_k + \alpha_k \Delta x_k, \tag{3.8}$$

where the searching direction Δx is now the conjugate direction and linear independent to each other, we can find.

Theorem 3.45. *For any $x_0 \in \mathbb{R}^n$ the sequence $\{x_k\}$ generated by the conjugate direction algorithm, the sequence converges to the solution x^* of the linear system $Ax = b$ in at most n steps.*

Proof. Since the searching directions $\{\Delta x_i\}$ are linearly independent, they must span the whole space $\mathbb{R}^n$. Hence, we can write the difference between x_0 and the solution x^* in the following way:

$$x^* - x_0 = \sigma_0 \Delta x_0 + \sigma_1 \Delta x_1 + \cdots + \sigma_{n-1} \Delta x_{n-1},$$

for some choice of scalars σ_k. By premultiplying this expression by $\Delta x_k^T A$ and using the conjugacy property, we obtain

$$\Delta x_k^T A (x^* - x_0) = \sigma_k \Delta x_k^T A \Delta x_k.$$

We now establish the result by showing that these coefficients σ_k coincide with the step lengths α_k generated by the formula(3.8). If x_k is generated by algorithm, then we have

$$x_k = x_0 + \alpha_0 \Delta x_0 + \alpha_1 \Delta x_1 + \cdots + \alpha_{k-1} \Delta x_{i-1}.$$

By premultiplying this expression by $\Delta x_k^T A$ and using the conjugacy property, we have that

$$\Delta x_k^T A (x_k - x_0) = 0,$$

and therefore

$$\Delta x_k^T A (x^* - x_0) = \Delta x_k^T A (x^* - x_k + x_k - x_0) = \Delta x_k^T A (x^* - x_k) = \Delta x_k^T (b - Ax_k) = -\Delta x_k^T r_k.$$

Thus we find that $\sigma_k = \alpha_k$ and derive the formula

$$\alpha_k = -\frac{r_k^T \Delta x_k}{\Delta x_k^T A \Delta x_k}. \tag{3.9}$$

□

Based on the formula of α_k, we then derive the iteration formula for residual variable r_k.

$$r_{k+1} = Ax_{k+1} - b = A(x_k + \alpha_k \Delta x_k) - b = r_k + \alpha_k A \Delta x_k. \tag{3.10}$$

Theorem 3.46. *Let $x_0 \in \mathbb{R}^n$ be any starting point and consider the minimized sequence $\{x_k\}$ is generated by the conjugate direction algorithm. Then*

$$r_k^T \Delta x_i = 0, \quad \text{for } i = 0, 1, \ldots, k-1, \tag{3.11}$$

and x_k is the minimizer of $\phi(x) = \frac{1}{2}x^T Ax - b^T x$ over the set

$$\{x \mid x = x_0 + \text{span}\{\Delta x_0, \Delta x_1, \ldots, \Delta x_{i-1}\}\}. \tag{3.12}$$

Proof. We first show that ϕ obtain its minimization if and only if $r(\bar{x})^T \Delta x_i = 0$, for each $i = 0, 1, \ldots, k-1$. Let us define $g(\sigma) = \phi(x_0 + \sigma_0 p_0 + \cdots + \sigma_{k-1}\Delta x_{i-1})$, where $\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_{k-1})^T$. Since $g(\sigma)$ is a strictly convex quadratic, the unique optimal solution σ^* satisfies

$$\frac{\partial g(\sigma^*)}{\partial \sigma_i} = 0, \quad i = 0, 1, \ldots, k-1.$$

By the chain rule, we obtain the equation as:

$$\nabla \phi (x_0 + \sigma_0^* \Delta x_0 + \cdots + \sigma_{k-1}^* \Delta x_{i-1})^T \Delta x_i = 0, \quad i = 0, 1, \ldots, k-1.$$

Thus we derive the optimal solution $\bar{x} = x_0 + \sigma_0^* \Delta x_0 + \sigma_1^* \Delta x_2 + \cdots + \sigma_{k-1}^* \Delta x_{i-1}$ on the set (3.12) which satisfies $r(\bar{x})^T \Delta x_i = 0$.

We now use induction to show that x_k satisfies (3.11). For the case $k = 1$, we have $x_1 = x_0 + \alpha_0 p_0$ minimizes ϕ along p_0 that $r_1^T p_0 = 0$. Assuming that $r_{k-1}^T \Delta x_i = 0$ for $i = 0, 1, \ldots, k-2$. By (3.10), we have

$$r_k = r_{k-1} + \alpha_{k-1} A \Delta x_{i-1},$$

By the formula (3.9) of α_{k-1}, we obtain that

$$\Delta x_{i-1}^T r_k = \Delta x_{i-1}^T r_{k-1} + \alpha_{k-1} \Delta x_{i-1}^T A \Delta x_{i-1} = 0.$$

Additionally, for the other vectors $\Delta x_i, i = 0, 1, \ldots, k-2$, we have

$$\Delta x_i^T r_k = \Delta x_i^T r_{k-1} + \alpha_{k-1} \Delta x_i^T A \Delta x_{i-1} = 0,$$

where $\Delta x_i^T r_{k-1} = 0$ because of the induction hypothesis and $\Delta x_i^T A \Delta x_{i-1} = 0$ because of conjugacy of the vectors Δx_i. We have shown that $r_k^T \Delta x_i = 0$, for $i = 0, 1, \ldots, k-1$, so the proof is complete. □

In the conjugate gradient method, we require that each direction Δx_k is chosen to be a linear combination of the negative residual $-r_k$ and the previous direction Δx_{k-1}. Thus the iteration equation is now

$$\Delta x_k = -r_k + \beta_k \Delta x_{k-1}. \tag{3.13}$$

Since each searching direction is conjugate to each other, we derive the formula for β_k if we multiply $\Delta x_{k-1}^T A$ on both sides of the equation.

$$\beta_k = \frac{r_k^T A \Delta x_{k-1}}{\Delta x_{k-1}^T A \Delta x_{k-1}}$$

Now we have derived all basic formulas for conjugate gradient method. Additionally, we choose the first search direction Δx_0 to be the steepest descent direction at the initial point x_0. Then the total algorithm is performed as follows:

Algorithm 7: Preliminary Conjugate gradient method

1: Given a start point $x_0 \in \mathbf{dom}\, f$, iteration times $k = 0$.
2: Calculate $r_0 \leftarrow Ax_0 - b, p_0 \leftarrow -r_0, k \leftarrow 0$;
3: **while** $r_k \neq 0$ **do**
4: $\quad \alpha_k \leftarrow -\frac{r_k^T \Delta x_k}{\Delta x_k^T A \Delta x_k}$;
5: $\quad x_{k+1} \leftarrow x_k + \alpha_k \Delta x_k$;
6: $\quad r_{k+1} \leftarrow Ax_{k+1} - b$;
7: $\quad \beta_{k+1} \leftarrow \frac{r_{k+1}^T A \Delta x_k}{\Delta x_k^T A \Delta x_k}$
8: $\quad \Delta x_{k+1} \leftarrow -r_{k+1} + \beta_{k+1} \Delta x_k$;
9: $\quad k \leftarrow k + 1$.
10: **end while**

Theorem 3.47. *Suppose that the kth iterate generated by the conjugate gradient method is not the solution point x^*. We then have the following four properties hold:*

1.
$$r_k^T r_i = 0, \quad \text{for } i = 0, 1, \ldots, k-1, \tag{3.14}$$

2.
$$\operatorname{span}\{r_0, r_1, \ldots, r_k\} = \operatorname{span}\left\{r_0, Ar_0, \ldots, A^k r_0\right\} \tag{3.15}$$

3.
$$\operatorname{span}\{\Delta x_0, \Delta x_1, \ldots, \Delta x_k\} = \operatorname{span}\left\{r_0, Ar_0, \ldots, A^k r_0\right\} \tag{3.16}$$

4.
$$\Delta x_k^T A \Delta x_i = 0, \quad \text{for } i = 0, 1, \ldots, k-1. \tag{3.17}$$

Therefore, the sequence $\{x_k\}$ converges to x^* in at most n steps.

Proof. We will show that the whole proof is generally by induction. It is obvious that the equations (3.15) and (3.16) hold trivially for $k = 0$, while (3.17) holds when we consider $k = 1$. We now Assume that these three equations are true for some k, our next step is to show that they still hold for $k + 1$.
To prove (3.15), we first show that

$$\operatorname{span}\{\Delta x_0, \Delta x_1, \ldots, \Delta x_k\} \subseteq \operatorname{span}\left\{r_0, Ar_0, \ldots, A^k r_0\right\}.$$

Since (3.15) and (3.16) hold for $n = k$, we derive that

$$r_k \in \operatorname{span}\left\{r_0, Ar_0, \ldots, A^k r_0\right\}, \quad \Delta x_k \in \operatorname{span}\left\{r_0, Ar_0, \ldots, A^k r_0\right\},$$

while by multiplying the second of these expressions by A, we obtain

$$A\Delta x_k \in \operatorname{span}\left\{Ar_0, \ldots, A^{k+1} r_0\right\}.$$

By applying $r_k = r_{k-1} + \alpha_{k-1} A\Delta x_{i-1}$, we find that

$$r_{k+1} \in \text{span}\left\{r_0, Ar_0, \ldots, A^{k+1}r_0\right\}.$$

We then combine this with the induction hypothesis for (3.15), we obtain that

$$\text{span}\left\{r_0, r_1, \ldots, r_k, r_{k+1}\right\} \subset \text{span}\left\{r_0, Ar_0, \ldots, A^{k+1}r_0\right\}.$$

To prove

$$\text{span}\left\{\Delta x_0, \Delta x_1, \ldots, \Delta x_k\right\} \supseteq \text{span}\left\{r_0, Ar_0, \ldots, A^{k}r_0\right\},$$

we use the induction hypothesis on (3.16) to deduce that

$$A^{k+1}r_0 = A\left(A^k r_0\right) \in \text{span}\left\{A\Delta x_0, A\Delta x_1, \ldots, A\Delta x_k\right\}$$

Since by $r_k = r_{k-1} + \alpha_{k-1} A\Delta x_{i-1}$, we have $A\Delta x_i = \left(r_{i+1} - r_i\right)/\alpha_i$ for $i = 0, 1, \ldots, k$, it follows that

$$A^{k+1}r_0 \in \text{span}\left\{r_0, r_1, \ldots, r_{k+1}\right\}.$$

By combining this expression with the induction hypothesis for (3.15), we find that

$$\text{span}\left\{r_0, Ar_0, \ldots, A^{k+1}r_0\right\} \subset \text{span}\left\{r_0, r_1, \ldots, r_k, r_{k+1}\right\}.$$

Therefore, the relation (3.15) continues to hold at $k+1$.
We then show that (3.16) continues to hold when k is replaced by $k+1$ by the following argument:

$$\begin{aligned}
&\text{span}\left\{\Delta x_0, \Delta x_1, \ldots, \Delta x_k, \Delta x_{k+1}\right\} \\
&\quad = \text{span}\left\{\Delta x_0, \Delta x_1, \ldots, \Delta x_k, r_{k+1}\right\} && \text{by (3.13)} \\
&\quad = \text{span}\left\{r_0, Ar_0, \ldots, A^k r_0, r_{k+1}\right\} && \text{by induction hypothesis for (3.16)} \\
&\quad = \text{span}\left\{r_0, Ar_0, \ldots, A^{k+1} r_0\right\} && \text{by (3.15) for } k+1.
\end{aligned}$$

Next, we prove the conjugacy condition (3.17) with k replaced by $k+1$. By multiplying (3.13) by $A\Delta x_i, i = 0, 1, \ldots, k$, we obtain

$$\Delta x_{k+1}^T A\Delta x_i = -r_{k+1}^T A\Delta x_i + \beta_{k+1}\Delta x_k^T A\Delta x_i. \tag{3.18}$$

By the formula of β_k, the right-hand-side of (3.18) vanishes when $i = k$. For $i \leq k-1$ we need to collect a number of observations. Note first that our induction hypothesis for (3.17) implies that the directions $p_0, p_1, \ldots, \Delta x_k$ are conjugate, so we can apply Theorem (3.46) to deduce that

$$r_{k+1}^T \Delta x_i = 0, \quad \text{for } i = 0, 1, \ldots, k. \tag{3.19}$$

Second, by repeatedly applying (3.16), we find that for $i = 0, 1, \ldots, k-1$, the following inclusion holds:

$$\begin{aligned}
A\Delta x_i \in A\,\text{span}\left\{r_0, Ar_0, \ldots, A^i r_0\right\} &= \text{span}\left\{Ar_0, A^2 r_0, \ldots, A^{i+1} r_0\right\} \\
&\subset \text{span}\left\{\Delta x_0, \Delta x_1, \ldots, \Delta x_{i+1}\right\}
\end{aligned} \tag{3.20}$$

We then combine (3.19) and (3.20) and then derive that

$$r_{k+1}^T A\Delta x_i = 0, \quad \text{for } i = 0, 1, \ldots, k-1,$$

so we have

$$\Delta x_{k+1}^T A \Delta x_i = \beta_{k+1} \Delta x_k^T A \Delta x_i, \text{ for } i = 0, 1, \ldots, k-1.$$

Because of the induction hypothesis for (3.17), we finally obtain that $\Delta x_{k+1}^T A \Delta x_i = 0, i = 0, 1, \ldots, k$. Hence, the induction argument holds for (3.17) also.

It follows that the direction set generated by the conjugate gradient method is indeed a conjugate direction set, so Theorem (3.45) tells us that the algorithm terminates in at most n iterations.

Finally, we prove (3.14) by a noninductive argument. Because the direction set is conjugate, we have from (3.11) that $r_k^T \Delta x_i = 0$ for all $i = 0, 1, \ldots, k-1$ and any $k = 1, 2, \ldots, n-1$. By rearranging (3.13), we find that

$$\Delta x_i = -r_i + \beta_i \Delta x_{i-1}$$

so that $r_i \in \operatorname{span}\{\Delta x_i, \Delta x_{i-1}\}$ for all $i = 1, \ldots, k-1$. We conclude that $r_k^T r_i = 0$ for all $i = 1, \ldots, k-1$. To complete the proof, we note that $r_k^T r_0 = -r_k^T \Delta x_0 = 0$, by definition of Δx_0 in this Algorithm and by (3.11). □

Remark 3.48. *The theorem's demonstration hinges on the assertion that the initial direction Δx_0 corresponds to the steepest descent direction $-r_0$. However, this conclusion is not applicable to alternative selections of Δx_0 since the gradients r_k are mutually orthogonal.*

Based on (3.14) and theorem (3.46), we can now rewrite the equation for α_k and β_k in a more economic way.

$$\alpha_k = -\frac{r_k^T \Delta x_k}{\Delta x_k^T A \Delta x_k} = \frac{r_k^T (r_k - \beta_k \Delta x_{k-1})}{\Delta x_k^T A \Delta x_k} = \frac{r_k^T r_k}{\Delta x_k^T A \Delta x_k}.$$

$$\beta_k = \frac{r_k^T A \Delta x_{k-1}}{\Delta x_{k-1}^T A \Delta x_{k-1}} = \frac{r_k^T (r_k - r_{k-1})/\alpha_{k-1}}{\Delta x_{k-1}^T (r_k - r_{k-1})/\alpha_{k-1}} = \frac{r_k^T r_k}{\Delta x_{k-1}^T r_k} = \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}}.$$

Thus we derive a more economic way for conjugate gradient method algorithm.

Algorithm 8: Conjugate gradient method

1: Given a start point $x_0 \in \operatorname{dom} f$, iteration times $k = 0$.
2: Calculate $r_0 \leftarrow Ax_0 - b, p_0 \leftarrow -r_0, k \leftarrow 0$;
3: **while** $r_k \neq 0$ **do**
4: $\quad \alpha_k \leftarrow -\frac{r_k^T r_k}{\Delta x_k^T A \Delta x_k}$;
5: $\quad x_{k+1} \leftarrow x_k + \alpha_k \Delta x_k$;
6: $\quad r_{k+1} \leftarrow Ax_{k+1} - b$;
7: $\quad \beta_{k+1} \leftarrow \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$
8: $\quad \Delta x_{k+1} \leftarrow -r_{k+1} + \beta_{k+1} \Delta x_k$;
9: $\quad k \leftarrow k + 1$.
10: **end while**

We then consider the convergence rate of this algorithm.

Convergence rate of Conjugate gradient method

From the definition of minimized sequence and (3.15), we have that

$$\begin{aligned} x_{k+1} &= x_0 + \alpha_0 \Delta x_0 + \cdots + \alpha_k \Delta x_k \\ &= x_0 + \gamma_0 r_0 + \gamma_1 A r_0 + \cdots + \gamma_k A^k r_0 \end{aligned}$$

for some constants γ_i. We now define $Poly^*(\cdot)$ to be a polynomial of degree k with coefficients $\gamma_0, \gamma_1, \ldots, \gamma_k$. Like any polynomial, $Poly^*$ can take either a scalar or a square matrix as its argument. For the matrix argument A, we have

$$Poly^*(A) = \gamma_0 I + \gamma_1 A + \cdots + \gamma_k A^k,$$

which allows us to express the equation as follows:

$$x_{k+1} = x_0 + Poly^*(A) r_0.$$

We now show that the Algorithm does the best job of minimizing the distance to the solution after k steps, when this distance is measured by the weighted norm measure $\|\cdot\|_A$ defined by

$$\|z\|_A^2 = z^T A z.$$

Consider $\phi(x) = \frac{1}{2} x^T A x - b^T x$, and the fact that x^* minimizes ϕ, we obtain that

$$\frac{1}{2} \|x - x^*\|_A^2 = \frac{1}{2} (x - x^*)^T A (x - x^*) = \phi(x) - \phi(x^*).$$

Since

$$r_0 = A x_0 - b = A x_0 - A x^* = A (x_0 - x^*),$$

we have that

$$x_{k+1} - x^* = x_0 + Poly^*(A) r_0 - x^* = [I + Poly^*(A) A] (x_0 - x^*). \tag{3.21}$$

Let $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ be the eigenvalues of A, and let $v_1, v_2, \ldots, v_n$ be the corresponding orthonormal eigenvectors, so that

$$A = \sum_{i=1}^{n} \lambda_i \mathrm{v}_i \mathrm{v}_i^T.$$

Since the eigenvectors span the whole space $\mathbb{R}^n$, we can write

$$x_0 - x^* = \sum_{i=1}^{n} \xi_i \mathrm{v}_i, \tag{3.22}$$

for some coefficients ξ_i. It is easy to show that any eigenvector of A is also an eigenvector of $Poly(A)$ for any polynomial $Poly$. For our particular matrix A and its eigenvalues λ_i and eigenvectors v_i, we have

$$Poly(A) \mathrm{v}_i = Poly(\lambda_i) \mathrm{v}_i, \quad i = 1, 2, \ldots, n.$$

By substituting (3.22) into (3.21) we have

$$x_{k+1} - x^* = \sum_{i=1}^{n} [1 + \lambda_i Poly^*(\lambda_i)] \xi_i \mathrm{v}_i.$$

By using the fact that $\|z\|_A^2 = z^T A z = \sum_{i=1}^n \lambda_i (\mathrm{v}_i^T z)^2$, we have

$$\|x_{k+1} - x^*\|_A^2 = \sum_{i=1}^{n} \lambda_i [1 + \lambda_i Poly^*(\lambda_i)]^2 \xi_i^2.$$

Since the polynomial $Poly^*$ generated by the CG method is optimal with respect to this norm, we have

$$\|x_{k+1} - x^*\|_A^2 = \min_{Poly} \sum_{i=1}^{n} \lambda_i [1 + \lambda_i Poly(\lambda_i)]^2 \xi_i^2.$$

By extracting the largest of the terms $[1+\lambda_i Poly(\lambda_i)]^2$ from this expression, we obtain that

$$\begin{aligned}\|x_{k+1}-x^*\|_A^2 &\le \min_{Poly}\max_{1\le i\le n}[1+\lambda_i Poly(\lambda_i)]^2\left(\sum_{j=1}^n \lambda_j\xi_j^2\right)\\ &= \min_{Poly}\max_{1\le i\le n}[1+\lambda_i Poly(\lambda_i)]^2\|x_0-x^*\|_A^2,\end{aligned}$$

where we have used the fact that $\|x_0-x^*\|_A^2=\sum_{j=1}^n\lambda_j\xi_j^2$.
We then consider the precondition of conjugate gradient method.

Precondition of conjugate gradient method

The process of preconditioning involves transforming the linear system to enhance the eigenvalue distribution of A, thereby accelerating the conjugate gradient method.
We now use a nonsingular matrix B to change variables from x to $\hat{x}$, expressed as $\hat{x}=Bx$. Correspondingly, the quadratic function ϕ, undergoes a transformation to

$$\hat{\phi}(\hat{x})=\frac{1}{2}\hat{x}^T\left(B^{-T}AB^{-1}\right)\hat{x}-\left(B^{-T}b\right)^T\hat{x}.$$

Thus minimizing $\hat{\phi}$ is equivalent to solving the linear system $\left(B^{-T}AB^{-1}\right)\hat{x}=B^{-T}b$.
We denote the transformation of A and b as

$$\hat{A}=B^{-T}AB^{-1}\qquad \hat{b}=B^{-T}b.$$

Thus the residual of the equation $\hat{r}_k$ for each iteration is now

$$\begin{aligned}\hat{r}_k &= \hat{A}\hat{x}_k-\hat{b}\\ &= B^{-T}AB^{-1}Bx_k-B^{-T}b\\ &= B^{-T}(Ax_k-b)\\ &= B^{-T}r_k.\end{aligned}$$

Similarly, we derive the transformation for other part of the algorithm. For $\hat{\alpha}_k$ we have

$$\hat{\alpha}_k=\frac{\hat{r}_k^T\hat{r}_k}{\Delta\hat{x}_k^T\hat{A}\Delta\hat{x}_k}=\frac{r_k^TB^{-1}B^{-T}r_k}{\Delta x_k^TA\Delta x_k}.$$

For $\hat{\beta}_{k+1}$, we obtain that

$$\hat{\beta}_{k+1}=\frac{\hat{r}_{k+1}^T\hat{r}_{k+1}}{\hat{r}_k^T\hat{r}_k}=\frac{r_{k+1}^TB^{-1}B^{-T}r_k}{r_k^TB^{-1}B^{-T}r_k}.$$

Considering the equation for $\Delta\hat{x}_k$, we have

$$\Delta\hat{x}_{k+1}=-\hat{r}_{k+1}+\beta_{k+1}\Delta\hat{x}_k=-B^{-T}r_{k+1}+\beta_{k+1}\Delta\hat{x}_k.$$

Thus for Δx_k, we obtain

$$\Delta x_{k+1}=-B^{-1}B^{-T}r_{k+1}+\beta_{k+1}\Delta x_k.$$

Additionally, consider the equation for $\hat{r}_{k+1}$:

$$\hat{r}_{k+1}=\hat{r}_k+\alpha_k\hat{A}\Delta\hat{x}_k.$$

We obtain the equation that

$$r_{k+1}=r_k+B^T\alpha_k\hat{A}\Delta\hat{x}_k=r_k+\alpha_kB^T\hat{A}B\Delta x_k=r_k+\alpha_kA\Delta x_k.$$

We define the matrix $C = B^T B$. Therefore, we can now rewrite the algorithm as precondition way:

Algorithm 9: Conjugate gradient method(Precondition)

Given a start point $x_0 \in \mathbf{dom}\, f$, precondition matrix C, iteration times $k = 0$.
Calculate $r_0 \leftarrow Ax_0 - b$;
Solve $Cy_0 = r_0$ Let $\Delta x_0 \leftarrow -y_0, k \leftarrow 0$;
while $r_k \neq 0$ **do**
    $\alpha_k \leftarrow -\frac{r_k^T y_k}{\Delta x_k^T A \Delta x_k}$;
    $x_{k+1} \leftarrow x_k + \alpha_k \Delta x_k$;
    $r_{k+1} \leftarrow Ax_{k+1} - b$;
    Solve $Cy_{k+1} = r_{k+1}$
    $\beta_{k+1} \leftarrow \frac{r_{k+1}^T y_{k+1}}{r_k^T y_k}$
    $\Delta x_{k+1} \leftarrow -y_{k+1} + \beta_{k+1} \Delta x_k$;
    $k \leftarrow k + 1$.
end while

3.2 Equality constrained minimization problems

Introduction to Equality constrained minimization problem

In this section, we will mainly discuss about methods to solve optimization problems with equality constraints. In general, there are two main ideas to solve this kind of problems, one is eliminating constraints to unconstrained problem and use the method in previous sections, another idea is solve dual problems since if the problem is convex, then it satisfies Slater's condition, and we will have the optimal value for the dual problem is the same as prime problem.

Definition 3.49. *(Equality constrained minimization problems) If the problem has the form*

$$\begin{array}{ll} \textit{minimize} & f(x) \\ \textit{subject to} & Ax = b, \end{array} \tag{3.23}$$

where $f : \mathbf{R}^n \to \mathbf{R}$ is convex and $A \in \mathbf{R}^{p \times n}$ with $\mathbf{rank}\, A = p < n$, then we call this equality constrained minimization problem.

Recall in chapter 2, we have definition(2.102), this gives that that a point $x^\star \in \mathbf{dom}\, f$ is optimal for (3.23) if and only if there is a $\nu^\star \in \mathbf{R}^p$ such that

$$Ax^\star = b, \quad \nabla f(x^\star) + A^T \nu^\star = 0. \tag{3.24}$$

We now introduce a classical convex quadratic minimization.

Example 3.2.1. *Consider the equality constrained convex quadratic minimization problem*

$$\begin{array}{ll} \textit{minimize} & f(x) = (1/2)x^T P x + q^T x + r \\ \textit{subject to} & Ax = b, \end{array}$$

where $P \in \mathbf{S}_+^n$ and $A \in \mathbf{R}^{p\times n}$. Its KKT condition (3.24) are

$$Ax^\star = b, \quad Px^\star + q + A^T\nu^\star = 0.$$

We can write these two conditions as the form

$$\begin{bmatrix} P & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x^\star \\ \nu^\star \end{bmatrix} = \begin{bmatrix} -q \\ b \end{bmatrix}.$$

Specially, we call the coefficient matrix as KKT matrix. If the KKT matrix is nonsingular, then we derive a unique optimal primal-dual pair $(x^\star, \nu^\star)$. If the KKT matrix is singular, but the KKT system is solvable, any solution which can be written as the combination of trivial solutions and a specific solution, thus yielding multiple optimal pairs $(x^\star, \nu^\star)$. If the KKT system is not solvable, the quadratic optimization problem is unbounded below or infeasible.
We now give a detailed proof that there exist $v \in \mathbf{R}^n$ and $w \in \mathbf{R}^p$ such that

$$Pv + A^Tw = 0, \quad Av = 0, \quad -q^Tv + b^Tw > 0.$$

The first two conditions mean that v, w belong to nullspace of KKT matrix and the third condition is a special case such that KKT system is not solvable (If KKT system is not solvable, then we require $-q^Tv + b^Tw \neq 0$).
Let $\hat{x}$ be any feasible point. We have $A\hat{x} = b$, so the point $x = \hat{x} + tv$ is feasible for all t and we derive

$$\begin{aligned} f(\hat{x} + tv) &= 1/2(\hat{x} + tv)^T P(\hat{x} + tv) + q^T(\hat{x} + tv) + r \\ &= 1/2\left[\hat{x}^T P\hat{x} + t\hat{x}^T Pv + tv^T P\hat{x} + t^2 v^T Pv\right] + q^T\hat{x} + tq^Tv + r \\ &= f(\hat{x}) + t\left(v^T P\hat{x} + q^Tv\right) + (1/2)t^2 v^T Pv \\ &= f(\hat{x}) + t\left(-\hat{x}^T A^T w + q^Tv\right) - (1/2)t^2 w^T Av \\ &= f(\hat{x}) + t\left(-b^Tw + q^Tv\right). \end{aligned}$$

This function will decrease without bound with $t \to \infty$.

Remark 3.50. *Generally speaking, we can use Jacobi, Gauss-Seidel and other classical methods to solve this equality.*

We now give a proposition which shows some equivalent conditions satisfying nonsingularity of the KKT matrix.

Proposition 3.51. *Consider the KKT matrix $\begin{bmatrix} P & A^T \\ A & 0 \end{bmatrix}$, where $P \in \mathbf{S}_+^n, A \in \mathbf{R}^{p\times n}$, and $\operatorname{rank} A = p < n$, is nonsingular, then the following conditions are equivalent:*

1. *$\mathcal{N}(P) \cap \mathcal{N}(A) = \{0\}$, i.e., P and A have no nontrivial common nullspace.*
2. *$Ax = 0, x \neq 0 \Longrightarrow x^TPx > 0$, i.e., P is positive definite on the nullspace of A.*
3. *$F^TPF \succ 0$, where $F \in \mathbf{R}^{n\times(n-p)}$ is a matrix for which $\mathcal{R}(F) = \mathcal{N}(A)$.*

Additionally, the KKT matrix has exactly n positive and p negative eigenvalues.

Proof. We first prove nonsingular is equivalent to first condition, and give proof for three equivalent conditions.

1. We prove it by contradiction. For sufficiency, if KKT matrix is singular, we have x, z, not both zero, such that
$$\begin{bmatrix} P & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} = 0.$$
This means that $Px + A^T z = 0$ and $Ax = 0$, if $x = 0$, then $z \neq 0$, but $A^T z = 0$ contradicts $\mathbf{rank}\, A = p$, so if $z = 0$, $x \neq 0$, multiple previous two equations with x^T, we have $x^T Px + x^T A^T z = 0$. Using $Ax = 0$, this reduces to $x^T Px = 0$. Since $P \in \mathbf{S}_+^n$, we have $Px = 0$, this contradicts condition 1.
For necessity, if $\mathcal{N}(P) \cap \mathcal{N}(A) \neq \{0\}$, there exists x, z not all zero such that
$$\begin{bmatrix} P & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} = 0.$$
Then the KKT matrix is singular.

2. We now prove condition 1 is equivalent to condition 2, it is also proved by contradiction, if condition 1 fails, then similar to the proof before, we have $x^T Px = 0$, contradicts condition 2. The proof for necessity is also similar.

3. We now prove condition 2 is equivalent to condition 3, let $x = Fy$, we then get $y^T F^T PFy \succ 0$, thus finishing the proof.

4. Finally, we prove the KKT matrix is nonsingular, then it has exactly n positive and p negative eigenvalues. Since $P \in \mathbf{S}_+^n$, we have $P + A^T A \succ 0$, therefore there exists a nonsingular matrix $R \in \mathbf{R}^{n\times n}$ such that
$$R^T \left(P + A^T A\right) R = I.$$
Let $AR = U\Sigma V_1^T$ be the singular value decomposition of AR, with $U \in \mathbf{R}^{p\times p}$, $\Sigma = \mathbf{diag}\left(\sigma_1, \ldots, \sigma_p\right) \in \mathbf{R}^{p\times p}$ and $V_1 \in \mathbf{R}^{n\times p}$. Let $V_2 \in \mathbf{R}^{n\times(n-p)}$ be such that
$$V = \begin{bmatrix} V_1 & V_2 \end{bmatrix}$$
is orthogonal, and define
$$S = \begin{bmatrix} \Sigma & 0 \end{bmatrix} \in \mathbf{R}^{p\times n}.$$
We have $AR = USV^T$, so
$$V^T R^T \left(P + A^T A\right) RV = V^T R^T PRV + S^T S = I.$$
Therefore $V^T R^T PRV = I - S^T S$ is diagonal, we have :
$$V^T R^T PRV = \mathbf{diag}\left(1 - \sigma_1^2, \ldots, 1 - \sigma_p^2, 1, \ldots, 1\right).$$
Applying a congruence transformation to the KKT system gives
$$\begin{bmatrix} V^T R^T & 0 \\ 0 & U^T \end{bmatrix} \begin{bmatrix} P & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} RV & 0 \\ 0 & U \end{bmatrix} = \begin{bmatrix} V^T R^T PRV & S^T \\ S & 0. \end{bmatrix}$$
Notice that for the right hand matrix with its size as $(n+p)\times(n+p)$, we can write $p \times p$ subpart as 2-block small matrix and it has additional $n - p$ diagnal elements equals to 1. So for each term, we have both positive and negative eigenvalue. Thus there are total n positive eigenvalues and p negative eigenvalues.

□

Eliminate equality constraints

A common strategy for addressing the equality-constrained problem described in (3.23) involves first eliminating the equality constraints and subsequently tackling the resultant unconstrained problem by employing some methods in previous sections for unconstrained minimization. We first find a matrix $F \in \mathbf{R}^{n\times(n-p)}$ and vector $\hat{x} \in \mathbf{R}^n$ that work to replace the condition $Ax = b$:

$$\{x \mid Ax = b\} = \{Fz + \hat{x} \mid z \in \mathbf{R}^{n-p}\}.$$

Here we can choose any particular solution of $Ax = b$ as our $\hat{x}$. Additionally, we employ a matrix $F \in \mathbf{R}^{n\times(n-p)}$ that spans the nullspace of A. Subsequently, we construct a modified optimization problem, denoted as:

$$\text{minimize} \quad \tilde{f}(z) = f(Fz + \hat{x}),$$

which is an unconstrained problem with variable $z \in \mathbf{R}^{n-p}$. From its solution $z^\star$, we can find the solution of the equality constrained problem as $x^\star = Fz^\star + \hat{x}$. Since we have

$$\nabla f(x^\star) + A^T\left(-\left(AA^T\right)^{-1} A\nabla f(x^\star)\right) = 0,$$

we can choose optimal dual variable $\nu^\star$ as

$$\nu^\star = -\left(AA^T\right)^{-1} A\nabla f(x^\star).$$

Use Lagrange dual function to solve prime problem

An alternative method for addressing (3.23) involves first solving the dual problem and subsequently retrieving the optimal primal variable $x^\star$. The dual function for (3.23) is given by:

$$\begin{aligned} g(\nu) &= -b^T\nu + \inf_x\left(f(x) + \nu^T Ax\right) \\ &= -b^T\nu - \sup_x\left(\left(-A^T\nu\right)^T x - f(x)\right) \\ &= -b^T\nu - f^*\left(-A^T\nu\right), \end{aligned}$$

where f^* is the conjugate of f, so the dual problem is

$$\text{maximize} \ -b^T\nu - f^*\left(-A^T\nu\right).$$

Newton's method with equality constraints

In this part, we will introduce an algorithm related to Newton's method to help solve optimization problems with constraints. Recall that in Newton's method, we used second-order Taylor expansion to derive Newton step. We now combine this idea and its constraint to derive a new Newton step. Consider second-order Taylor expansion for function f, i.e.,

$$\widehat{f}(x+v) = f(x) + \nabla f(x)^T v + \frac{1}{2}v^T\nabla^2 f(x)v,$$

where $\widehat{f}$ is the second-order Taylor expansion and is convex quadratic function of v. If we consider this function as a quadratic function for v and use the idea in Example (3.2.1), we derive a new KKT system:

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix}\begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ 0 \end{bmatrix},$$

where w is the dual optimal point for this quadratic optimization problem. Based on this, we derive a new Newton step $\Delta x_{\text{nt}} = v$.

Definition 3.52. *(Newton step for equality constrained problems) For $x \in \mathbf{dom}\, f$, and x is feasible, i.e., $Ax = b$, the vector*

$$\Delta x_{\text{nt}} = -\nabla^2 f(x)^{-1}(\nabla f(x) + A^T w)$$

is called the Newton step.

Similarly, we can also define Newton's decrement for constraint problems.

Definition 3.53. *(Newton decrement for equality constrained problems) The Newton decrement for the equality constrained problem has the form*

$$\lambda(x) = \left(\Delta x_{\text{nt}}^T \nabla^2 f(x) \Delta x_{\text{nt}}\right)^{1/2}.$$

Proposition 3.54. *Some propositions with Newton step and Newton decrement.*

1. *Newton step is always feasible.*
2. *$\nabla f(x)^T \Delta x_{\text{nt}} = -\lambda(x)^2$.*
3. *Newton step and Newton decrement are independent of linear changes of coordinates.*
4. *Newton decrement can be used for stopping criterion, i.e., $\lambda^2/2 \leq \epsilon$.*

Proof.

1. Since $Ax = b$ and $A\Delta x_{\text{nt}} = 0$, so we have $A(x + \Delta x_{\text{nt}}) = b$.
2. Notice that $\nabla^2 f(x)\Delta x_{\text{nt}} + A^T w = -\nabla f(x)$, multiply Δx_{nt}^T on both sides we derive

$$\Delta x_{\text{nt}}^T \nabla^2 f(x) \Delta x_{\text{nt}} + \Delta x_{\text{nt}}^T A^T w = \Delta x_{\text{nt}}^T \nabla^2 f(x) \Delta x_{\text{nt}} = -\nabla f(x)^T \Delta x_{\text{nt}}.$$

3. Suppose $T \in \mathbf{R}^{n \times n}$ is nonsingular, and define $\bar{f}(y) = f(Ty)$. Then we obtain that

$$\nabla \bar{f}(y) = T^T \nabla f(x), \quad \nabla^2 \bar{f}(y) = T^T \nabla^2 f(x) T,$$

where $x = Ty$ (This is proved in proposition(3.29)), for equality constraint, we have $ATy = b$. Thus the KKT system for $\bar{f}(y)$ is

$$\begin{bmatrix} T^T \nabla^2 f(Ty) T & T^T A^T \\ AT & 0 \end{bmatrix} \begin{bmatrix} \Delta y_{\text{nt}} \\ \bar{w} \end{bmatrix} = \begin{bmatrix} -T^T \nabla f(Ty) \\ 0 \end{bmatrix},$$

where Δy_{nt} is the Newton step for $\bar{f}(y)$, and $\bar{w}$ is the dual optimal point. So we derive $T\Delta y_{\text{nt}} = \Delta x_{\text{nt}}$, $\bar{w} = w$. And we can obtain the Newton decrement for $\bar{f}$ by

$$\begin{aligned} \lambda(x) &= \left(\Delta x_{\text{nt}}^T \nabla^2 f(x)^{-1} \Delta x_{\text{nt}}\right)^{1/2} \\ &= \left(\Delta y_{\text{nt}}^T T^T (\nabla^2 f(x))^{-1} T \Delta y_{\text{nt}}\right)^{1/2} \\ &= \left(\Delta y_{\text{nt}}^T \nabla^2 \bar{f}(y)^{-1} \Delta y_{\text{nt}}\right)^{1/2} \\ &= \lambda(y). \end{aligned}$$

Thus Newton step and Newton decrement are independent of linear changes of coordinates.

4. Since we have

$$\inf\{\widehat{f}(x+v)|A(x+v)=b\} = \widehat{f}(x+\Delta x_{\text{nt}}) = f(x) - (1/2)\lambda(x)^2,$$

where we use $\nabla f(x)^T \Delta x_{\text{nt}} = -\lambda(x)^2$. So the stopping criterion satisfies

$$f(x) - p^* = f(x) - \inf\{\widehat{f}(x+v)|A(x+v)=b\} = \lambda(x)^2/2.$$

□

Base on these theories, we now provide the algorithm for Newton method with equality constraints.

Algorithm 10: Newton's method for equality constrained optimization

1: Given a start point x_0 satisfying $Ax_0 = b$, current value $f(x_0)$, tolerance $\epsilon > 0$, iteration times $k = 0$
2: **while** true **do**
3: Compute the Newton step and decrement by solving the KKT system.
4: **if** $\lambda^2/2 \leq \epsilon$ **then** break
5: **end if**
6: Line search. Choose a step size $t > 0$
7: $x = x + t\Delta x_{nt}$
8: Calculate $f(x)$, $k = k+1$
9: **end while**

Improvement for Newton's method with equality constraints

Recall that we require the starting point must be feasible when using Newton method to solve equality constrained optimization, in this part we will introduce an improvement for previous algorithm so that the initial can be chosen randomly .
Since initial x may not be feasible, we intend to find Δx such that $x + \Delta x$ approximate the optimal point x^*, and we will then have

$$A(x+\Delta x) = b, \quad \nabla f(x+\Delta x) + A^T w = 0,$$

where w is the dual optimal point. We use first order expansion for $\nabla f(x+\Delta x)$, we will derive

$$\nabla f(x+\Delta x) \approx \nabla f(x) + \nabla^2 f(x)\Delta x.$$

Thus we then derive new KKT system:

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ w \end{bmatrix} = - \begin{bmatrix} \nabla f(x) \\ Ax - b \end{bmatrix}.$$

Remark 3.55. *If the initial point is infeasible, then the Newton direction may not be a descent direction for f.*

$$\begin{aligned} \frac{d}{dt} f(x+t\Delta x)\Big|_{t=0} &= \nabla f(x)^T \Delta x = \Delta x^T \nabla f(x) \\ &= -\Delta x^T \left(\nabla^2 f(x)\Delta x + A^T w\right) \\ &= -\Delta x^T \nabla^2 f(x)\Delta x + (Ax-b)^T w \end{aligned}$$

This may not be negative.

We then introduce another way to help make approximation for x^*. Since when reaching optimal point, we have $Ax = b,\ \Delta x = 0$, this is equivalent to $Ax - b = 0,\ \nabla f(x) + A^T w = 0$. We then consider the residual part of these two equations.

Definition 3.56. *(Residual function) A residual function $r : \mathbf{R}^n \times \mathbf{R}^p \to \mathbf{R}^n \times \mathbf{R}^p$ is defined as*

$$r(x,\nu) = \left(r_{dual}\left(x,\nu\right), r_{pri}\left(x,\nu\right)\right).$$

where each part of r is defined as

$$r_{dual}\left(x,\nu\right) = \nabla f(x) + A^T\nu, \quad r_{pri}\left(x,\nu\right) = Ax - b.$$

Based on this definition, we then approximate x^* by vanishing the value of r. The first order Taylor expansion for r at current point $\bar{v}$ is $r(\bar{v} + \Delta\bar{v}) \approx r(\bar{v}) + \nabla r(\bar{v})^T\Delta\bar{v}$. So if we choose $\nabla r(\bar{v})^T\Delta\bar{v} = -r(\bar{v})$, then r will vanish.

Explanation for residual function
For residual function, we have $\Delta\bar{v} = (\Delta x, \Delta\nu)$. Since the KKT system is

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ w \end{bmatrix} = - \begin{bmatrix} \nabla f(x) \\ Ax - b \end{bmatrix}.$$

and if we write w as the new iterated dual solution,i.e., $w = \nu + \Delta\nu$, we then derive a new KKT system:

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta\nu \end{bmatrix} = - \begin{bmatrix} \nabla f(x) + A^T\nu \\ Ax - b \end{bmatrix} = - \begin{bmatrix} r_{\text{dual}} \\ r_{\text{pri}} \end{bmatrix}.$$

So residual function works well to explain the original KKT system.

We then consider norm of the residual decreases in the Newton direction for Residual function, i.e.,

$$\frac{d}{dt}\left\|r\left(\bar{v} + t\Delta\bar{v}\right)\right\|_2^2\Big|_{t=0} = 2r(y)^T\nabla r(\bar{v})^T\Delta\bar{v} = -2r(\bar{v})^T r(\bar{v}).$$

Taking the derivative of the square, we obtain

$$\frac{d}{dt}\left\|r\left(\bar{v} + t\Delta\bar{v}\right)\right\|_2\Big|_{t=0} = -\|r(\bar{v})\|_2$$

Thus we find a new criterion to measure the progress of the infeasible start for Newton method.

Algorithm 11: Newton's method with infeasible initial point for equality constrained optimization

Given a start point $x_0 \in \mathbf{dom}\, f$ and arbitrary initial v, current value $f(x_0)$, tolerance $\epsilon > 0$, iteration times $k = 0$, $\alpha \in (0, 1/2), \beta \in (0, 1)$.
while $Ax \neq b$ and $||r||_2 > \epsilon$ **do**
    Compute the Newton step Δx and dual point w by solving KKT system.
    Compute dual Newton step $\Delta v = w - v$.
    Use backtracking line search: Choose initial step size $t = 1$
    while $\|r(x + t\Delta x, v + t\Delta v)\|_2 > (1 - \alpha t)\|r(x, v)\|_2$. **do**
        $t := \beta t$
    end while
    $x = x + t\Delta x$, $v = v + t\Delta v$.
    Calculate $f(x)$, $k = k + 1$
end while

We then provide two classical examples and using Lagrange dual function method, Newton method and improved Newton method separately to solve these two problems.

Example 3.2.2. *The first example is the equality constrained analytic centering problem*

$$\begin{array}{ll} \text{minimize} & f(x) = -\sum_{i=1}^{n} \log x_i \\ \textit{subject to} & Ax = b. \end{array} \tag{3.25}$$

For this example, we first solve it directly by feasible Newton method with equality constraints. We derive the Newton step by solving its KKT system:

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x_{\text{nt}} \\ w \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ 0 \end{bmatrix},$$

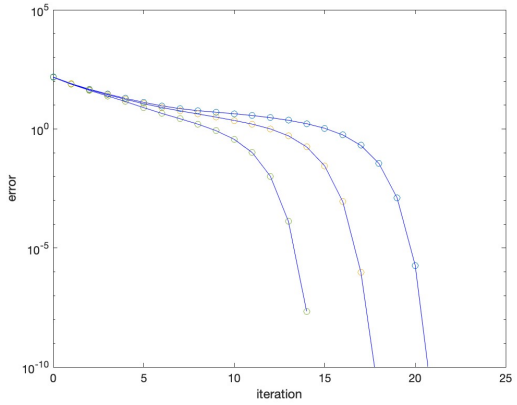
where w is the dual optimal point for this quadratic optimization problem. So we derive

$$\Delta x_{\text{nt}} = -\nabla^2 f(x)^{-1}(A^T w + \nabla f(x)), \quad A\nabla^2 f(x)^{-1} A^T w = -A\nabla^2 f(x)^{-1}\nabla f(x).$$

Specially, since $\nabla^2 f(x)^{-1}\nabla f(x) = x$, thus we have

$$\Delta x_{\text{nt}} = -\nabla^2 f(x)^{-1} A^T w + x, \quad A\nabla^2 f(x)^{-1} A^T w = Ax.$$

The details are shown in Matlab code. For this example, we choose $\alpha = 0.1, \beta = 0.5$, the size of A is 100×500 and we generate it by random function, and our initial point is also generated randomly. Here is the graph showing the outcome of this method.



In the test, we randomly choose three initial feasible points and it is easy to check Newton's method converges quadratic.

Figure 3.4: Iteration step versus error for Newton's method by KKT system

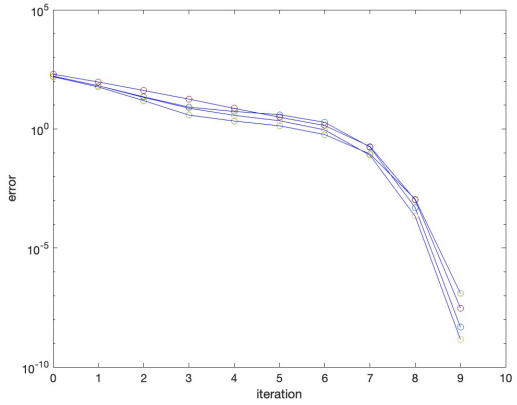
We then consider the Lagrange dual function to solve this problem.
We first calculate the conjugate function f^* for f:

$$\begin{aligned} f^*(y) &= \sup_x(y^T x - f(x)) \\ &= \sup_x(\sum_{i=1}^n (x_i y_i + \log x_i)) \\ &= \sum_{i=1}^n (-1 + \log(-1/y_i)) \\ &= -n - \sum_{i=1}^n \log(-y_i). \end{aligned}$$

The Lagrange dual function for (3.25) *is then*

$$\begin{aligned} g(\nu) &= \inf_x \nu^T(Ax - b) + f(x) \\ &= -b^T\nu + \inf_x(\nu^T Ax + f(x)) \\ &= -b^T\nu - \sup_x(-v^T Ax - f(x)) \\ &= -b^T\nu - f^*(-A^T\nu). \\ &= -b^T\nu + \sum_{i=1}^n \log(A^T\nu)_i + n. \end{aligned}$$

Thus we only need to solve this unconstrained optimization problem. We now use Newton's method to find the optimal solution and the Matlab code is shown in appendix. Here we provide the graph showing the outcome of this method.



In the test, we randomly choose four initial points. The graph provides error versus iteration steps.

Figure 3.5: Iteration step versus error for Newton's method by Lagrange dual method

Finally, we consider solving this problem by improved Newton's method with infeasible initial point. We randomly choose our initial point x_0 and initial dual point v. The Newton step is obtained by solving

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ w \end{bmatrix} = - \begin{bmatrix} \nabla f(x) \\ Ax - b \end{bmatrix}.$$

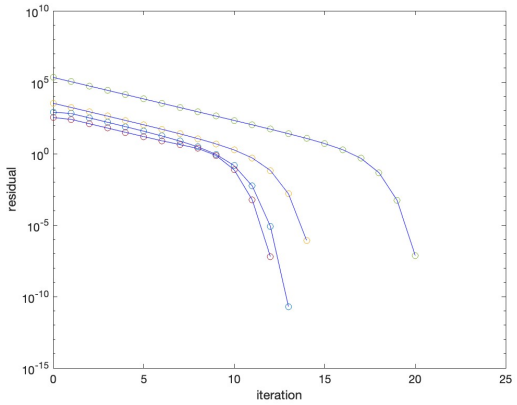
We obtain that

$$\Delta x_{\text{nt}} = -\nabla^2 f(x)^{-1}(A^T w + \nabla f(x)),\ A\nabla^2 f(x)^{-1} A^T w = Ax - b - A\nabla^2 f(x)^{-1}\nabla f(x).$$

Notice that $\nabla^2 f(x)^{-1}\nabla f(x) = x$, thus we have

$$\Delta x_{\text{nt}} = -\nabla^2 f(x)^{-1} A^T w + x, \quad A\nabla^2 f(x)^{-1} A^T w = 2Ax - b.$$

We then use Matlab to solve this problem. Here we provide the graph showing the outcome of this method.



In the test, we randomly choose our initial point and initial dual point, the graph provides residual versus iteration steps.

Figure 3.6: Iteration step versus error for modified Newton's method by residual function

Example 3.2.3. *Another example is the equality constrained entropy maximization problem*

$$\begin{array}{ll} \text{minimize} & f(x) = \sum_{i=1}^{n} x_i \log x_i \\ \textit{subject to} & Ax = b, \end{array}$$

with $\operatorname{dom} f = \mathbf{R}^n_{++}$ *and* $A \in \mathbf{R}^{p\times n}$, *with* $p < n$.

3.3 Inequality constrained minimization

Introduction to Inequality constrained minimization

In this section we will mainly focus on inequality constrained minimization with the form

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \le 0, \quad i = 1, \dots, m \\ & Ax = b, \end{array} \tag{3.26}$$

where $f_0, \dots, f_m : \mathbf{R}^n \to \mathbf{R}$ are convex and twice continuously differentiable, and $A \in \mathbf{R}^{p\times n}$ with $\operatorname{rank} A = p < n$. We assume that the problem is solvable, i.e., an optimal $x^\star$ exists. We denote the optimal value $f_0(x^\star)$ as $p^\star$.
Additionally, for this chapter, we also assume that the problem is strictly feasible, i.e., there exists $x \in \mathcal{D}$ that satisfies $Ax = b$ and $f_i(x) < 0$ for $i = 1, \dots, m$. This satisfies Slater's condition and we derive strong duality, so there exist dual optimal $\lambda^\star \in \mathbf{R}^m, \nu^\star \in \mathbf{R}^p$, which together with $x^\star$ satisfy the KKT conditions

$$\begin{aligned} Ax^\star = b, \quad f_i(x^\star) \le 0, \quad i = 1, \dots, m \\ \lambda^\star \succeq 0 \\ \nabla f_0(x^\star) + \sum_{i=1}^{m} \lambda_i^\star \nabla f_i(x^\star) + A^T \nu^\star = 0 \\ \lambda_i^\star f_i(x^\star) = 0, \quad i = 1, \dots, m. \end{aligned}$$

There are many methods to solve this kind of problems, one way is considering Lagrange dual function and find the optimal solutions for dual unconstrained optimization problem. Another way is to find its relationship with equality constrained optimization problem, and then use the method in section (3.2) to solve the problem.
For this section, we will then mainly discuss the second method and call it interior-point method.

Relationship with Inequality constrained minimization and Equality constrained minimization

In this part we will discuss how we transform a Inequality constrained minimization into Equality constrained minimization and give a detailed proof of the validation for this transformation. We first rewrite the problem (3.26) as the form

$$\begin{array}{ll} \text{minimize} & f_0(x) + \sum_{i=1}^{m} I_-(f_i(x)) \\ \text{subject to} & Ax = b, \end{array} \tag{3.27}$$

where $I_- : \mathbf{R} \to \mathbf{R}$ is the indicator function for the nonpositive reals,

$$I_-(u) = \begin{cases} 0 & u \le 0 \\ \infty & u > 0. \end{cases}$$

Thus we successfully transform the Inequality constrained minimization into Equality constrained minimization. Notice that (3.27) is not differentiable, so till now we still can not use the method in in section (3.2) to solve it. We then consider a new function to approximate the indicator function.

Definition 3.57. *(Logarithmic barrier function) We define the logarithmic function $\widehat{I}_-$ to approximate indicator function with the form*

$$\widehat{I}_-(u) = -(1/t)\log(-u), \quad \operatorname{dom}\widehat{I}_- = -\mathbf{R}_{++},$$

where $t > 0$ is a parameter that sets the accuracy of the approximation.

From this definition, it is obvious that as t increases, the approximation becomes more accurate. Additionally, the logarithmic barrier function is convex, thus we derive that the approximation

$$\begin{array}{ll} \text{minimize} & f_0(x) + \sum_{i=1}^m -(1/t)\log\left(-f_i(x)\right) \\ \text{subject to} & Ax = b \end{array} \tag{3.28}$$

is still a convex optimization problem. We will then give a detailed proof that this approximation still have a good approximated optimal solution for original inequality constrained minimization. For simplicity, we now provide a new definition called log barrier.

Definition 3.58. *(Log barrier) We call a function $\phi(x)$ is a log barrier for problem* (3.26) *if it has the form*

$$\phi(x) = -\sum_{i=1}^m \log\left(-f_i(x)\right),$$

with $\operatorname{dom}\phi = \{x \in \mathbf{R}^n \mid f_i(x) < 0, i = 1, \ldots, m\}$.

The gradient and Hessian of the Log barrier function ϕ are given by

$$\begin{aligned} \nabla\phi(x) &= \sum_{i=1}^m \frac{1}{-f_i(x)}\nabla f_i(x) \\ \nabla^2\phi(x) &= \sum_{i=1}^m \frac{1}{f_i(x)^2}\nabla f_i(x)\nabla f_i(x)^T + \sum_{i=1}^m \frac{1}{-f_i(x)}\nabla^2 f_i(x). \end{aligned}$$

So we obtain the KKT condition for this approximated problem(3.28) as

$$\begin{aligned} Ax^\star = b, \quad f_i\left(x^\star\right) < 0, \quad i = 1, \ldots, m \\ \nabla f_0\left(x^\star\right) + \sum_{i=1}^m \frac{1}{-tf_i(x)}\nabla f_i(x) + A^T\nu^\star = 0 \end{aligned}$$

where x^* is the optimal solution for this approximated problem and ν^* is the optimal solution for its dual problem. We also define $\lambda_i^\star(t) = -\frac{1}{tf_i(x^\star(t))}, \quad i = 1, \ldots, m.$
Since x^* is still feasible for problem(3.26), we use the idea in (2.101), i.e., we consider the Lagrange dual function for original function:

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \nu^T(Ax - b),$$

Taking x^*, λ_i^*, ν^* inside, we derive the dual function function $g(\lambda^\star(t), \nu^\star(t))$

$$\begin{aligned} g(\lambda^\star(t), \nu^\star(t)) &= f_0(x^\star(t)) + \sum_{i=1}^m \lambda_i^\star(t) f_i(x^\star(t)) + \nu^\star(t)^T (Ax^\star(t) - b) \\ &= f_0(x^\star(t)) - m/t. \end{aligned}$$

So we have

$$f_0(x^\star(t)) - p^\star \leq f_0(x^\star(t)) - g(\lambda^\star(t), \nu^\star(t)) \leq m/t.$$

Thus we derive the optimal point x^* for approximated problem converges to optimal point in original problem (3.26) as $t \to \infty$.

KKT condition analysis for approximated problem

Since we have defined $\lambda_i^\star(t) = -\frac{1}{tf_i(x^\star(t))}, \quad i = 1, \ldots, m$. We derive a point x is equal to $x^\star(t)$ if and only if there exists λ, ν such that

$$\begin{aligned} Ax = b, \quad f_i(x) \leq 0, \quad i = 1, \ldots, m \\ \lambda \succeq 0 \\ \nabla f_0(x) + \sum_{i=1}^m \lambda_i \nabla f_i(x) + A^T \nu = 0 \\ -\lambda_i f_i(x) = 1/t, \quad i = 1, \ldots, m. \end{aligned} \tag{3.29}$$

When $t \to \infty$, we also derive the approximated problem almost have the same optimal solution as the original problem.

We then mainly discuss the implement of Newton's method on this approximated problem. Using the same idea in (3.2), the KKT matrix and the equation for this problem is now:

$$\begin{bmatrix} \nabla^2 f_0(x) + 1/t \nabla^2 \phi(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x_{\text{nt}} \\ \nu_{\text{nt}} \end{bmatrix} = - \begin{bmatrix} \nabla f_0(x) + 1/t \nabla \phi(x) \\ 0 \end{bmatrix}.$$

This can be interpreted by Taylor expansion of (3.29), for small Δx_{nt}, we have:

$$\begin{aligned} &\nabla f_0(x + \Delta x_{\text{nt}}) + \sum_{i=1}^m \frac{1}{-tf_i(x + \Delta x_{\text{nt}})} \nabla f_i(x + \Delta x_{\text{nt}}) \\ &\approx \nabla f_0(x) + \sum_{i=1}^m \frac{1}{-tf_i(x)} \nabla f_i(x) + \nabla^2 f_0(x) \Delta x_{\text{nt}} \\ &\quad + \sum_{i=1}^m \frac{1}{-tf_i(x)} \nabla^2 f_i(x) \Delta x_{\text{nt}} + \sum_{i=1}^m \frac{1}{tf_i(x)^2} \nabla f_i(x) \nabla f_i(x)^T \Delta x_{\text{nt}}. \end{aligned}$$

Thus we also derive the KKT matrix and the equation:

$$H \Delta x_{\text{nt}} + A^T \nu_{\text{nt}} = -g, \quad A \Delta x_{\text{nt}} = 0,$$

where

$$\begin{aligned} H &= \nabla^2 f_0(x) + \sum_{i=1}^m \frac{1}{-tf_i(x)} \nabla^2 f_i(x) + \sum_{i=1}^m \frac{1}{tf_i(x)^2} \nabla f_i(x) \nabla f_i(x)^T \\ g &= \nabla f_0(x) + \sum_{i=1}^m \frac{1}{-tf_i(x)} \nabla f_i(x). \end{aligned}$$

The barrier method

Based on previous analysis, we now provide the general idea to solve problem(3.26). For simplicity, we consider the equivalent problem:

$$\begin{array}{ll} \text{minimize} & tf_0(x) + \phi(x) \\ \text{subject to} & Ax = b, \end{array}$$

which has the same minimization and the procedure to implement it is almost the same as (3.28). The approach to solving this problem involves tackling a series of unconstrained (or linearly constrained) minimization problems, with each problem using the last obtained solution as the initial point for the subsequent unconstrained minimization problem. Put differently, we calculate $x^\star(t)$ for an ascending sequence of t values, continuing until t reaches or exceeds m/ϵ. This ensures that we attain an ϵ-suboptimal solution for the original problem.

Algorithm 12: Barrier method

1: Given strictly feasible $x, t := t^{(0)} > 0, \mu > 1$, tolerance $\epsilon > 0$.
2: **while** true **do**
3: Centering step: Compute $x^\star(t)$ by minimizing $tf_0 + \phi$, subject to $Ax = b$, starting at x.
4: Update. $x := x^\star(t)$.
5: **if** $m/t < \epsilon$ **then**
6: break
7: **end if**
8: Increase t. $t := \mu t$.
9: **end while**

We now give a classical example from (2.4.3) related to LP problem.

Example 3.3.1. *The classical LP example has the form*

$$\begin{array}{ll} \text{minimize} & c^T x \\ \textit{subject to} & Ax \preceq b. \end{array}$$

Its dual problem is now

$$\begin{array}{ll} \textit{maximize} & -b^T\lambda \\ \textit{subject to} & A^T\lambda + c = 0 \\ & \lambda \succeq 0, \end{array} \tag{3.30}$$

So we use the barrier method by considering the function

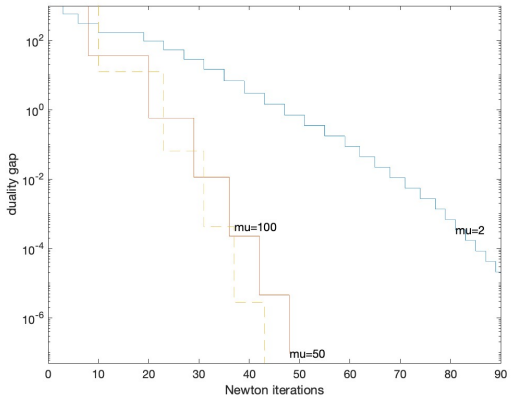
$$tc^T x - \sum_{i=1}^{n} (\log(b - Ax)_i).$$

Thus we choose $\lambda = 1/t(Ax^* - b)$, where x^* is the approximated optimal point at one iteration. So the dual gap is now

$$c^T x^* - (-b^T\lambda) = -(A^T\lambda)^T x^* + b^T\lambda = -(Ax^*)^T\lambda + b^T\lambda = (b - Ax^*)^T\lambda.$$

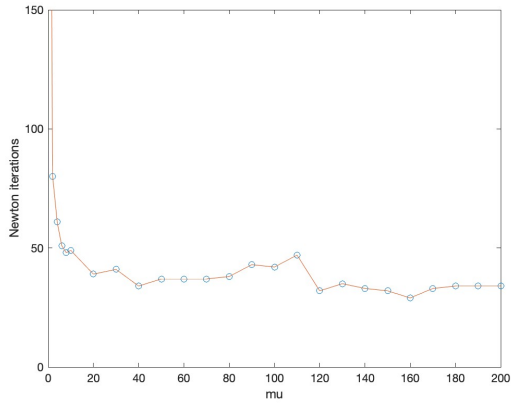
For this example, we use Matlab to make an implement. We consider $A \in \mathbf{R}^{100\times 50}$ and use Newton's method with backtracking line search. We choose the parameter $\alpha = 0.01, \beta = 0.5$. The

Matlab code is shown in appendix and we provide the graph here to make an explanation.



In the test, we randomly choose the initial feasible points and compare the procedure with different μ.

Figure 3.7: Iteration steps versus dual gap for classical LP problem with interior point method



This graph gives a brief idea of the relationship between μ we choose and the iteration step.

Figure 3.8: Variable μ versus iteration step for classical LP problem with interior point method

The first graph demonstrates key aspects of the barrier method. Firstly, the method exhibits effective performance, displaying nearly linear convergence in the duality gap. This is due to the consistent number of Newton steps needed for re-centering, regardless of the value of μ*. For* $\mu = 50$ *and* $\mu = 150$*, the barrier method successfully solves the problem using roughly 35 to 40 Newton steps.*

Furthermore, it illustrates the trade-off associated with selecting μ*. For* $\mu = 2$*, the iterations are shorter, typically requiring only 2 or 3 Newton steps. However, the improvements are limited since the duality gap only reduces by a factor of 2 in each outer iteration. Conversely, when* $\mu = 150$*, the iterations are longer, usually involving around 8 Newton steps, but the gap decreases significantly as it is reduced by a factor of 150 in each outer iteration.*

This trade-off is further explored in the second graph. The barrier method is used to solve the LP problem, stopping when the duality gap falls below 10^{-3}*, for 25 values of* μ *ranging from 1.2*

to 200. The plot displays the total number of Newton steps required to solve the problem as a function of μ.

The second graph demonstrates the barrier method's strong performance across a broad range of μ values, from approximately 3 to 200. As expected, the total number of Newton steps increases when μ is too small, resulting in more outer iterations. Notably, the total number of Newton steps remains relatively constant for μ values greater than approximately 3. Consequently, as μ increases within this range, the decrease in the number of outer iterations is balanced by an increase in the number of Newton steps per outer iteration. However, for extremely high μ values, the barrier method's performance becomes less predictable, depending more on the specific problem instance. Given that larger μ values do not yield improved performance, an optimal choice typically falls within the range of 10 to 100.

Use two phase method to solve minimization problem

Previously we discussed the barrier method which necessitates a strictly feasible initial point, denoted as $x^{(0)}$. When such a starting point is not readily available, the barrier method is preceded by an initial phase, known as "Phase I," during which the primary objective is to compute a strictly feasible point or determine the feasibility of the constraints. The strictly feasible point obtained during Phase I serves as the starting point for the barrier method's main phase, referred to as "Phase II."

In the subsequent sections, we will provide a more detailed description of the Phase I methods used to establish a feasible starting point for the barrier method.

Phase I method

In Phase I method, we need to do iterations to make our initial infeasible point become strictly feasible. Our requirements for constraints are

$$f_i(x) \leq 0, \quad i = 1, \ldots, m, \quad Ax = b, \tag{3.31}$$

where $f_i : \mathbf{R}^n \to \mathbf{R}$ are convex. We then transform the constraints as problem satisfying:

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & f_i(x) \leq s, \quad i = 1, \ldots, m \\ & Ax = b \end{array} \tag{3.32}$$

in the variables $x \in \mathbf{R}^n, s \in \mathbf{R}$. Considering this transformed problem and original requirements. Assume the optimal solution for this problem is $\bar{p}^\star$. We can categorize the scenarios based on the optimal value $\bar{p}^\star$ as follows:

1. When $\bar{p}^\star < 0$, there exists a strictly feasible solution for equation(3.31). Furthermore, if (x, s) is a feasible solution for (3.32) with $s < 0$, it implies that x satisfies the condition $f_i(x) < 0$. In this case, it is unnecessary to solve the optimization problem (3.32) with high precision, and we can terminate the process when $s < 0$ is met.

2. If $\bar{p}^\star > 0$, it indicates that equation (3.31) is infeasible. Similar to case 1, there is no need to solve the Phase I optimization problem (3.32) with high precision. Termination

can occur when a dual feasible point is discovered with a positive dual objective value, which serves as proof that $\bar{p}^\star > 0$. In such situations, we can construct an alternative that demonstrates the feasibility of equation (3.31) based on the dual feasible point.

3. When $\bar{p}^\star = 0$ and the minimum is achieved at x^* with $s^\star = 0$, it implies that the set of inequalities is feasible but not strictly feasible. However, if $\bar{p}^\star = 0$ and the minimum is not attained, it indicates that the inequalities are infeasible.

Prime dual method

In this section, we will consider prime dual method to solve interior point method which do not require our initial point is feasible. The processing method is similar to section (3.2). Previously, we discussed the KKT condition for approximated problems with (3.29). We now modify this KKT condition into residual function expressed as $r_t(x, \lambda, \nu) = 0$, where we define

$$r_t(x, \lambda, \nu) = \begin{bmatrix} \nabla f_0(x) + \sum_{i=1}^m \lambda_i \nabla f_i(x) + A^T\nu \\ -\sum_{i=1}^m \lambda_i f_i(x) - (1/t)1 \\ Ax - b \end{bmatrix},$$

and $t > 0$. If x, λ, ν satisfy $r_t(x, \lambda, \nu) = 0$ (and $f_i(x) < 0$), then we will derive $x = x^\star(t), \lambda = \lambda^\star(t)$, and $\nu = \nu^\star(t)$. Formally, we call x is primal feasible, and λ, ν are dual feasible, with duality gap $1/t$.
We define $f : \mathbf{R}^n \to \mathbf{R}^m$ and its derivative matrix Df as the form

$$f(x) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{bmatrix}, \quad Df(x) = \begin{bmatrix} \nabla f_1(x)^T \\ \vdots \\ \nabla f_m(x)^T \end{bmatrix}.$$

So we can rewrite the residual function as

$$r_t(x, \lambda, \nu) = \begin{bmatrix} \nabla f_0(x) + Df(x)^T\lambda + A^T\nu \\ -\operatorname{diag}(\lambda)f(x) - (1/t)\mathbf{1} \\ Ax - b \end{bmatrix}.$$

Additionally, we define the first block component of r_t as the dual residual with its form

$$r_{\text{dual}} = \nabla f_0(x) + Df(x)^T\lambda + A^T\nu,$$

the last block component as the primal residual with its form $r_{\text{pri}} = Ax - b$, and define the middle block as the centrality residual with its form

$$r_{\text{cent}} = -\operatorname{diag}(\lambda)f(x) - (1/t)\mathbf{1}.$$

We then approximate x^* by vanishing the value of r. The first order Taylor expansion for r at current point $\bar{v}$ is $r(\bar{v} + \Delta\bar{v}) \approx r(\bar{v}) + \nabla r(\bar{v})^T\Delta\bar{v}$. So if we choose $\nabla r(\bar{v})^T\Delta\bar{v} = -r(\bar{v})$, then r will vanish. We denote the current point $\bar{v}$ and vanishing direction $\Delta\bar{v}$ as

$$\bar{v} = (x, \lambda, \nu), \quad \Delta\bar{v} = (\Delta x, \Delta\lambda, \Delta\nu),$$

respectively. Therefore we derive the Jacobian matrix $\nabla r(\bar{v})^T$ in terms of x, λ, ν, and the equation

$$\begin{bmatrix} \nabla^2 f_0(x) + \sum_{i=1}^m \lambda_i \nabla^2 f_i(x) & Df(x)^T & A^T \\ -\operatorname{diag}(\lambda) Df(x) & -\operatorname{diag}(f(x)) & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \\ \Delta \nu \end{bmatrix} = - \begin{bmatrix} r_{\text{dual}} \\ r_{\text{cent}} \\ r_{\text{pri}} \end{bmatrix}. \tag{3.33}$$

The primal-dual search direction $\Delta \bar{v}_{\text{pd}} = \left(\Delta x_{\text{pd}}, \Delta \lambda_{\text{pd}}, \Delta \nu_{\text{pd}}\right)$ is defined as the solution of (3.33).

Remark 3.59. *For prime-dual method, the new iteration point obtained may not always be feasible, so we can not directly use the original dual gap for this method. By* (2.11)*, we consider a new definition which approximately reflects the dual gap called surrogate gap.*

Definition 3.60. *(Surrogate gap) For any x that satisfies $f(x) \prec 0$ and $\lambda \succeq 0$, we define surrogate gap as*

$$\hat{g}(x, \lambda) = -f(x)^T \lambda.$$

Based on this definition, we now provide the prime-dual method algorithm for interior point method.

Algorithm 13: Prime-dual method

1: Given a random starting point x_0, λ, ν, tolerance $\epsilon > 0$, iteration times $k = 0$, $\mu > 1$.
2: Calculate row number m of matrix A, current $\hat{g}$, r_{pri} , r_{cent} , and $r_{\text{dual.}}$
3: **while** $\|r_{\text{pri}}\|_2 \geq \epsilon_{\text{feas}}$, $\|r_{\text{dual}}\|_2 \geq \epsilon_{\text{feas}}$, and $\hat{g} \geq \epsilon$ **do**
4: Determine the suitable $t = \mu m / \hat{g}$.
5: Determine a prime-dual direction $[\Delta x, \Delta \lambda, \Delta \nu]^T$ by Jacobian matrix.
6: Initialize step size $s = 1$.
7: Derive suitable step size s by Line search.
8: $x = x + s\Delta x, \lambda = \lambda + s\Delta \lambda, \nu = \nu + s\Delta \nu$.
9: Calculate $f(x)$, $k = k + 1$, new gap $\hat{g}$.
10: Calculate new r_{pri} , r_{cent} , and $r_{\text{dual.}}$
11: **end while**

We now give an example to implement this prime-dual method.

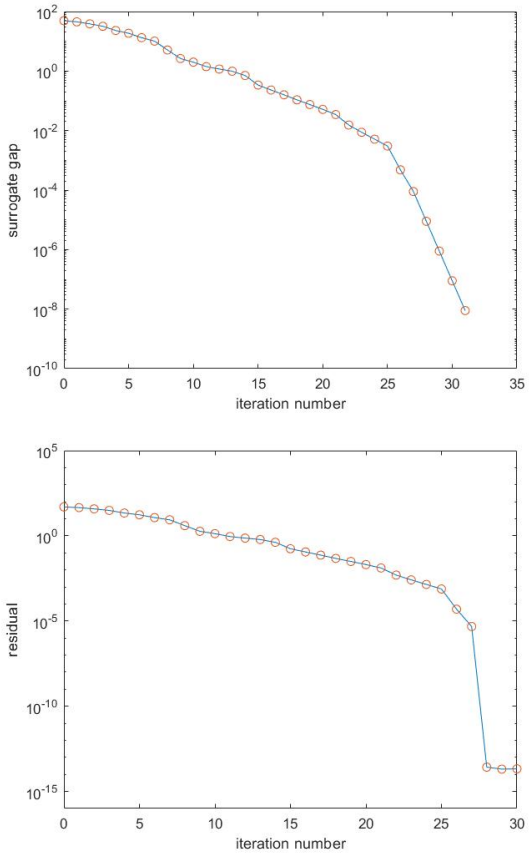
Example 3.3.2. *Reconsider example* (3.3.1) *with prime-dual method. We start the primal-dual interior point method at a randomly generated $x^{(0)}$, that satisfies $f(x) \prec 0$. The parameter values we use for the primal-dual interior-point method are*

$$\mu = 10, \quad \beta = 0.5, \quad \epsilon = 10^{-8}, \quad \alpha = 0.01.$$

We initialize the problem with $m = 200$ inequalities and $n = 100$ variables. The two pictures show the progress of the primal-dual interior-point method. The performance of the method is demonstrated via two graphs: one showing the surrogate gap $\hat{g}$, and the norm of the primal and dual residuals,

$$r_{feas} = \left(\|r_{pri}\|_2^2 + \|r_{dual}\|_2^2\right)^{1/2},$$

After approximately 30 iterations, the residual reaches zero. The primal-dual interior-point method is found to be faster than the barrier method, particularly when a high degree of accuracy is needed.



These two graphs show the progress of the primal-dual interior-point method. If compared with the previous example, we find that the iteration number for prime-dual method is much smaller than barrier method.

Figure 3.9: Result of using prime-dual method in interior point method for classical LP problem

Comparison between Prime-dual method and barrier method

The search directions in the primal-dual method are similar but not identical to those used in the barrier method. We start with the linear equations (3.33) that define the primal-dual search directions. We eliminate the second variable $\Delta\lambda_{\text{pd}}$, using

$$\Delta\lambda_{\text{pd}} = -\operatorname{diag}(f(x))^{-1}\operatorname{diag}(\lambda)Df(x)\Delta x_{\text{pd}} + \operatorname{diag}(f(x))^{-1}r_{\text{cent}}\ ,$$

which comes from the second block of equations. Substituting this into the first block of equations gives

$$\begin{aligned}
&\begin{bmatrix} H_{\text{pd}} & A^T \\ A & 0 \end{bmatrix}\begin{bmatrix} \Delta x_{\text{pd}} \\ \Delta\nu_{\text{pd}} \end{bmatrix} \\
&= -\begin{bmatrix} r_{\text{dual}} + Df(x)^T\operatorname{diag}(f(x))^{-1}r_{\text{cent}} \\ r_{\text{pri}} \end{bmatrix} \\
&= -\begin{bmatrix} \nabla f_0(x) + (1/t)\sum_{i=1}^m \frac{1}{-f_i(x)}\nabla f_i(x) + A^T\nu \\ r_{\text{pri}} \end{bmatrix},
\end{aligned} \tag{3.34}$$

where

$$H_{\text{pd}} = \nabla^2 f_0(x) + \sum_{i=1}^{m} \lambda_i \nabla^2 f_i(x) + \sum_{i=1}^{m} \frac{\lambda_i}{-f_i(x)} \nabla f_i(x) \nabla f_i(x)^T.$$

Recall that in barrier method, we have the KKT system for approximated problem with the form

$$\begin{bmatrix} \nabla^2 f_0(x) + 1/t \nabla^2 \phi(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x_{\text{nt}} \\ \nu_{\text{nt}} \end{bmatrix} = - \begin{bmatrix} \nabla f_0(x) + 1/t \nabla \phi(x) \\ 0 \end{bmatrix}.$$

The gradient and Hessian of the Log barrier function ϕ are given by

$$\begin{aligned} \nabla \phi(x) &= \sum_{i=1}^{m} \frac{1}{-f_i(x)} \nabla f_i(x) \\ \nabla^2 \phi(x) &= \sum_{i=1}^{m} \frac{1}{f_i(x)^2} \nabla f_i(x) \nabla f_i(x)^T + \sum_{i=1}^{m} \frac{1}{-f_i(x)} \nabla^2 f_i(x). \end{aligned}$$

Thus, if we consider $\Delta\nu_{\text{nt}} = \nu_{\text{nt}} - \nu$, we will derive

$$\begin{bmatrix} \nabla^2 f_0(x) + 1/t \nabla^2 \phi(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x_{\text{nt}} \\ \Delta\nu_{\text{nt}} \end{bmatrix} = - \begin{bmatrix} \nabla f_0(x) + 1/t \nabla \phi(x) + A^T \nu \\ 0 \end{bmatrix}.$$

Therefore, when λ satisfies $\lambda_i = 1/-tf_i(x)$ and we consider $Ax \neq b$, i.e., there exists $r_{\text{pri}} = Ax - b$. We can rewrite this KKT system the same as (3.34).

3.4 Interior point method solver with standard LP problem

Previously we have discussed using interior method to solve an easy LP model with inequality constraints in (3.3). In this section, we will provide a further discussion related to LP problems and provide my solver related to standard LP problem.

The background of the standard LP problem

We now consider standard LP (2.2.1) with its form:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b, \quad x \succeq 0, \end{array}$$

with $A \in \mathbf{R}^{m \times n}$. Its dual problem is

$$\begin{array}{ll} \text{minimize} & -b^T \nu \\ \text{subject to} & A^T \nu + c = \lambda, \quad \lambda \succeq 0. \end{array}$$

Rewrite the dual form and we derive

$$\begin{array}{ll} \text{minimize} & b^T \nu \\ \text{subject to} & A^T \nu \preceq c. \end{array}$$

To solve this problem, we first do not assume the initial point is feasible, i.e. , we will randomly choose our start point and do iterations to find the optimal.

Ideas to solve the problem

We will use the interior method to solve this problem. Using barrier function we first derive its approximated problem:

$$\begin{array}{ll} \text{minimize} & tc^T x - \sum_{i=1}^{n} (\log x_i) \\ \text{subject to} & Ax = b, \end{array}$$

This approximated problem is the equality constrained minimization problem, we now consider Newton's method with infeasible start. We randomly choose our initial point x_0 and initial dual point v. The Newton step is obtained by solving

$$\begin{bmatrix} \nabla^2 f(x) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ w \end{bmatrix} = - \begin{bmatrix} \nabla f(x) \\ Ax - b \end{bmatrix}.$$

We obtain that

$$\Delta x_{\text{nt}} = -\nabla^2 f(x)^{-1}(A^T w + \nabla f(x)),\ A\nabla^2 f(x)^{-1} A^T w = Ax - b - A\nabla^2 f(x)^{-1}\nabla f(x).$$

Our next step is to use the residual function to measure the progress of the infeasible start for Newton method.

$$r(x, \nu) = \left(r_{\text{dual}}\,(x, \nu), r_{\text{pri}}\,(x, \nu)\right),$$

where each part of r is defined as

$$r_{\text{dual}}\,(x, \nu) = \nabla f(x) + A^T \nu, \quad r_{\text{pri}}\,(x, \nu) = Ax - b.$$

The iterations stops if $Ax = b$ and the $r_{\text{dual}}\,(x, \nu) \to 0$.
Based on previous steps, we satisfies the requirements that the solution points are feasible. Our next step is to use the barrier method to do iterations for suitable t such that the approximated function do have small error to real optimal solution.
With the increasing t, we derive our new dual points to approximate original LP problem and calculate both the gap between original LP problem and the new residual functions. Then repeat previous steps until the error is really small.
We now provide the whole procedures of the algorithm.

Algorithm 14: Interior point method for standard LP problem

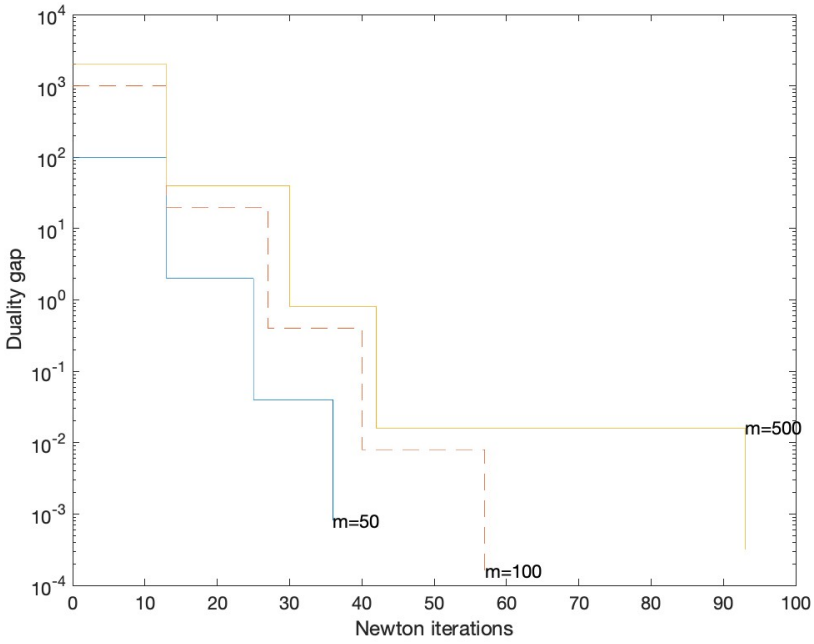
Given a random start point x_0, v, current value $f(x_0)$, tolerance $\epsilon > 0$, iteration times $k = 0$, $\alpha \in (0, 1/2), \beta \in (0, 1)$, initial variable $t > 0$, increase rate μ.
while true **do**
    Derive $v = -v/t$ for approximated problem and calculate the dual gap
    if gap $< \epsilon$ **then**
        break
    end if
    Increase t. $t := \mu t$.
    while $Ax \neq b$ and $||r||_2 > \epsilon$ **do**
        Compute the Newton step Δx and dual point w by solving the KKT system.
        Compute dual Newton step $\Delta v = w - v$.
        Use backtracking line search: Choose initial step size $s = 1$
        while $\|r(x + s\Delta x, v + s\Delta v)\|_2 > (1 - \alpha s)\|r(x, v)\|_2$. **do**
            $s := \beta s$
        end while
        $x = x + s\Delta x$, $v = v + s\Delta v$.
        $k = k + 1$
    end while
end while

Implement

We now implement the interior point method.
We utilized algorithm parameters with $\mu = 50$, employing the same settings for the centering steps as described in the previous examples: backtracking parameters $\alpha = 0.01$ and $\beta = 0.5$, along with a stopping criterion of $norm(residual) \leq 10^{-5}$. The initial point was selected randomly. The algorithm was concluded when the initial duality gap was reduced by a factor of 10^4, which corresponds to completing two outer iterations.

The graph below displays the relationship between the duality gap and the number of iterations for three different problem instances, each with varying dimensions: $m = 50$, $m = 100$, and $m = 500$. The plotted data closely resembles previous cases, demonstrating an almost linear convergence of the duality gap. It is evident from the plots that there is a slight increase in the number of Newton steps required as the problem size transitions from 50 constraints to 500 constraints.

The plot illustrates that the growth in the number of required Newton steps is minimal, rising from approximately 21 to roughly 50, even when the problem dimensions increase by a factor of 500. This behavior is consistent with the general characteristics of the barrier method: the number of Newton steps needed increases very gradually with problem dimensions, typically remaining within the range of a few tens. Naturally, the computational effort associated with executing one Newton step increases with the problem's higher dimensions.



In the test, we randomly choose the initial feasible points and compare the procedure with different size of problem.

Figure 3.10: Iteration steps versus dual gap for standard LP problem with interior point method

Chapter 4

Conclusion

This final year project mainly focus on optimization theory and application, and it consists of two main sections. The first section extensively investigates optimization theory, covering crucial topics such as convex sets, convex functions, convex optimization problems, Lagrange dual function, and the Karush-Kuhn-Tucker (KKT) condition, which serve as foundational principles in numerical optimization methods. The second section focuses on numerical optimization and its practical applications, employing methods like the interior point method, Newton's method, and conjugate gradient method to solve traditional optimization problems such as linear programming and quadratic programming.

In the theoretical section, The definitions and properties of convex sets and convex functions serve as the fundamental framework throughout this report. Meanwhile, the Lagrange dual function, Slater condition, and KKT conditions represent the central conceptual content, not only embodying crucial properties and conclusions in the theoretical analysis of optimization but also constituting the core ideology for subsequent numerical optimization.

In the numerical optimization section, the report progresses from simpler to more complex optimization problems based on underlying convex optimization issues. It begins with the most basic unconstrained optimization problems, then advances to optimization problems with slightly more complex transformed forms of equality constraints, and finally addresses the complex yet prevalent optimization problems with inequality constraints. Throughout the transformations of problem forms, our approach involves converting complex problems into familiar simpler ones. For instance, we approximate equality-constrained problems as unconstrained problems by approximating them using the KKT system, and transform inequality-constrained problems into equality-constrained ones using barrier methods before approximating them as unconstrained optimization problems. Central to these methodologies remains the pivotal Newton's method.

This report serves merely as an introductory exploration into numerical optimization. Both the theoretical analysis and numerical optimization sections exhibit areas for further improvement. For instance, the analytical section could delve deeper into the investigation of specific Quadratically Constrained Quadratic Programming (QCQP) problems, particularly those where

the optimized function can achieve strong duality despite non-convex conditions. Additionally, the introduction of the conjugate gradient method in the numerical optimization section appears somewhat cursory. Future studies could reference literature to explore nonlinear methods and convergence rates in greater detail. Furthermore, a deeper investigation into convergence analysis and common enhancements of interior point methods could be pursued in subsequent research endeavors.

However, the field of numerical optimization transcends the conventional methodologies of the past century. In contemporary optimization realms, traditional approaches intersect with modern machine algorithms, incorporating tools such as neural networks, which are extensively applied in both academia and industry. It is anticipated that optimization will witness epochal scientific breakthroughs in the near future. Moreover, there is a hope that humanity will acquire the capacity to elucidate and interpret current popular AI algorithms, including neural networks, akin to how humans once mastered the Lagrange dual function and KKT conditions, integrating interpretable knowledge and theories into practical applications in daily life and production processes.

Appendix A

Some interesting topic with vector optimization

Proposition A.1. *In vector optimization problems, if the object function is convex, then the scalarization is the same as original problem.*

Remark A.2. *This proof this quite hard, I have not yet found something related to this proof.*

Lagrange duality extends to a problem with generalized inequality constraints
Given the optimization problem with the form:

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \preceq_{K_i} 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{array}$$

where $K_i \subseteq \mathbf{R}^{k_i}$ are proper cones, with the domain of this problem $\mathcal{D} = \bigcap_{i=0}^{m} \operatorname{dom} f_i \cap \bigcap_{i=1}^{p} \operatorname{dom} h_i$, is nonempty.
With each generalized inequality $f_i(x) \preceq_{K_i} 0$, Now the Lagrange function the same as we stated before

$$L(x, \lambda, \nu) = f_0(x) + \lambda_1^T f_1(x) + \cdots + \lambda_m^T f_m(x) + \nu_1 h_1(x) + \cdots + \nu_p h_p(x),$$

where $\lambda = (\lambda_1, \ldots, \lambda_m)$ and $\nu = \left(\nu_1, \ldots, \nu_p\right)$. The dual function is also same as before:

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i^T f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right).$$

Remember for generally inequality, we define the condition for λ_i as:

$$\lambda_i \succeq K_i^* 0, \quad i = 1, \ldots, m,$$

where K_i^* denotes the dual cone of K_i. We also have weak duality for generalized inequality, to be more specific, the Lagrange dual optimization problem is

$$\begin{array}{ll} \text{maximize} & g(\lambda, \nu) \\ \text{subject to} & \lambda_i \succeq_{K_i^*} 0, \quad i = 1, \ldots, m. \end{array}$$

weak duality, i.e., $d^\star \leq p^\star$, where $d^\star$ denotes the optimal value of the dual problem.

Definition A.3. *(Slater's condition) Similar to previous chapter, Strong duality ($d^\star = p^\star$) holds when the primal problem is convex and satisfies some special requirements.*

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \textit{subject to} & f_i(x) \preceq K_t 0, \quad i = 1, \ldots, m \\ & Ax = b, \end{array}$$

where f_0 is convex and f_i is K_i-convex, Slater's condition means that there exists an $x \in \operatorname{relint} \mathcal{D}$ with the affine function condition $Ax = b$ and $f_i(x) \prec_{K_i} 0, i = 1, \ldots, m$.

Definition A.4. *(KKT condition for generalized inequality) Assume that the functions f_i, h_i are differentiable. Since $x^\star$ minimizes $L(x, \lambda^\star, \nu^\star)$, its gradient with respect to x vanishes at $x^\star$:*

$$\nabla f_0(x^\star) + \sum_{i=1}^{m} Df_i(x^\star)^T \lambda_i^\star + \sum_{i=1}^{p} \nu_i^\star \nabla h_i(x^\star) = 0,$$

where $Df_i(x^\star) \in \mathbf{R}^{k_t \times n}$ is the derivative of f_i evaluated at $x^\star$. Thus, if strong duality holds, any primal optimal $x^\star$ and any dual optimal $(\lambda^\star, \nu^\star)$ must satisfy the optimal conditions

$$\begin{array}{rcll} f_i(x^\star) & \preceq_{K_i} & 0, & i = 1, \ldots, m \\ h_i(x^\star) & = & 0, & i = 1, \ldots, p \\ \lambda_i^\star & \succeq_{K_i^*} & 0, & i = 1, \ldots, m \\ \lambda_i^{\star T} f_i(x^\star) & = & 0, & i = 1, \ldots, m \\ \nabla f_0(x^\star) + \sum_{i=1}^{m} Df_i(x^\star)^T \lambda_i^\star + \sum_{i=1}^{p} \nu_i^\star \nabla h_i(x^\star) & = & 0. & \end{array}$$

Proposition A.5. *If the prime problem is convex, every function is differential, and the strong duality holds, then (x^*, λ^*, ν^*) is the optimal points if and only if it satisfies KKT condition.*

Appendix B

Matlab code

```
%The matlab code for backtracking line search.
clear;clc;
a=0.1;b=0.7;x_0=[1;1];iteration_time=100;
syms x y
F = exp(x+3*y-0.1)+exp(x-3*y-0.1)+exp(-x-0.1);
Fx1 = diff(F,x);
Fx2 = diff(F,y);
f = matlabFunction(F);
dx1 = matlabFunction(Fx1);
dx2 = matlabFunction(Fx2);
x1=zeros(1,iteration_time+1);
x2=zeros(1,iteration_time+1);
x1(1,1)=x_0(1,1);
x2(1,1)=x_0(2,1);
result=zeros(1,iteration_time+1);
result(1,1)=f(x1(1,1),x2(1,1));
k = 1:iteration_time;
format long
for i=k
t=1;
while f(x1(1,i)-t*dx1(x1(1,i),x2(1,i)),x2(1,i)-t*dx2(x1(1,i),
    x2(1,i)))...
>f(x1(1,i),x2(1,i))-a*t*(dx1(x1(1,i),x2(1,i))^2....
+dx2(x1(1,i),x2(1,i))^2)
t=b*t;
end
x1(1,i+1)=x1(1,i)-t*dx1(x1(1,i),x2(1,i));
x2(1,i+1)=x2(1,i)-t*dx2(x1(1,i),x2(1,i));
result(1,i+1)=vpa(f(x1(1,i+1),x2(1,i+1)),8);
end
```

```
plot3(x1,x2,result)
hold on
x=-1:0.01:1;
y=-1:0.01:1;
[x,y]=meshgrid(x);
F = exp(x+3*y-0.1)+exp(x-3*y-0.1)+exp(-x-0.1);
plot3(x,y,F)


%The matlab code for exact line search.
clear;clc;
x_0=[1;1];iteration_time=100;
f=@(x,y) exp(x+3*y-0.1)+exp(x-3*y-0.1)+exp(-x-0.1);
fx = @(x,y) exp(x - 3*y - 1/10) + exp(x + 3*y - 1/10) - exp(-
    x - 1/10);
fy = @(x,y) 3*exp(x + 3*y - 1/10) - 3*exp(x - 3*y - 1/10);
x1=zeros(1,iteration_time+1);
x2=zeros(1,iteration_time+1);
x1(1,1)=x_0(1,1); x2(1,1)=x_0(2,1);
result=zeros(1,iteration_time);
result(1,1)=f(x1(1,1),x2(1,1));
k = 1:iteration_time;
for i=k
dx=(f(x1(1,i)+1/100,x2(1,i))-f(x1(1,i)-1/100,x2(1,i)))
    /(2/100);
dy=(f(x1(1,i),x2(1,i)+1/100)-f(x1(1,i),x2(1,i)-1/100))
    /(2/100);
g = @(t) -dx*fx(x1(1,i)-t*dx,x2(1,i)-t*dy)+-dy*fy(x1(1,i)-t*
    dx,x2(1,i)-t*dy);
dt = fzero(g,0);
x1(1,i+1)=x1(1,i)-dt*dx;
x2(1,i+1)=x2(1,i)-dt*dy;
result(1,i+1)=vpa(f(x1(1,i),x2(1,i)),8);
end
disp(result(1,101));
plot3(x1,x2,result)
hold on
x=-1:0.01:1;
y=-1:0.01:1;
[x,y]=meshgrid(x);
F = exp(x+3*y-0.1)+exp(x-3*y-0.1)+exp(-x-0.1);
plot3(x,y,F)
```

```
%This is the matlab code with Newton's method to solve
    example.
function testfornewton(A)
%test function f(x)=-sum(log(1-a_i*x))-sum(log(1-x_i^2))
%with newton's method
n=100;
%m=200;
% a=randn(m);
% x=zeros(n,1);
alpha = 0.2;
beta = 0.5;
max_iteration = 1000;
episilo = 1e-8;
%A=randn(m,n);
value=zeros(1,1);
x = zeros(n,1);
k=1;
for iter = 1:max_iteration
value(iter) = -sum(log(1-A*x)) - sum(log(1+x)) - sum(log(1-x)
    );
d = 1./(1-A*x);
grad = A'*d - 1./(1+x) + 1./(1-x);
hess = A'*diag(d.^2)*A + diag(1./(1+x).^2 + 1./(1-x).^2);
v = -hess\grad;
f1 = grad'*v;%newton_step
if abs(f1)/2 < episilo
    break;
end
t = 1;
while ((max(A*(x+t*v)) >= 1) | (max(abs(x+t*v)) >= 1))
t = beta*t;
end
while ( -sum(log(1-A*(x+t*v))) - sum(log(1-(x+t*v).^2)) > ...
value(iter) + alpha*t*f1 )
t = beta*t;
end
x = x+t*v;
k=k+1;
end
value=value-value(1,end);
semilogy([1:k],value(1,[1:end]));
xlabel('iterationstep')
```

```
ylabel('f-p*')
title('newtonmethodgraph');

%%This is the matlab code with gradient descent method to solve
    example.
function testwithgradient(A)
%test function f(x)=-sum(log(1-a_i*x))-sum(log(1-x_i^2))
%with gradient descent method
n=100;
%m=200;
alpha = 0.2;
beta = 0.5;
max_iteration = 1000;
episilo = 1e-3;
x = zeros(n,1);
value=zeros(1,1);
for iter = 1:max_iteration
value(iter) = -sum(log(1-A*x)) - sum(log(1+x)) - sum(log(1-x)
    );
grad = A'*(1./(1-A*x)) - 1./(1+x) + 1./(1-x);
if norm(grad) < episilo
    break;
end
v = -grad;
f1 = grad'*v;%descent direction
t = 1;
while ((max(A*(x+t*v)) >= 1) | (max(abs(x+t*v)) >= 1))
t = beta*t;
end
while ( -sum(log(1-A*(x+t*v))) - sum(log(1-(x+t*v).^2)) > ...
value(iter) + alpha*t*f1 )
t = beta*t;
end
x = x+t*v;
end
value=value-value(1,end);
semilogy([1:500],value(1,[1:500]));
xlabel('iterationstep');
ylabel('f-p*');
title('gradientgraph');

%Feasible Newton method applied to primal problem,
    % with three different starting points.
```

```
clc;clear
randn('state',26);
rand('state',26);
m = 100;  n = 500;
A = randn(m,n);
% make z0 dual feasible (A'*z0 = 1)
z0 = randn(m,1);
A = A + z0*(1 - A'*z0)'/(z0'*z0);
% make x0 primal feasible
x0 = rand(n,1);
b = A*x0;
max_iteration = 50;
alpha = 0.1;
beta  = 0.5;
TOL = 1e-12;
figure(1);
randn('state',2);
nostarts = 3;
for starts = 1:nostarts
% Starting point: generate random vector v in nullspace of A
    and
% take x = x0 + 0.99*s*v where s is maximum step to the
    boundary.
v = randn(n,1);    v = v - A'*(A'\v);
x = x0 + (0.99/max(-v./x0))*v;
fvals = [];
for iters=1:max_iteration
f = -sum(log(x));  fvals = [fvals, f];
g = -1./x;  invH = x.^2;%invH is H^(-1)
dw = -( A*diag(invH)*A') \  (A*(invH.*g));
dx = - invH.* (g+A'*dw);
lambda = -g'*dx;
if (lambda/2 < TOL)
break;
end
t = 1;
while (-sum(log(1+x+t*dx)) > f - alpha*lambda)
t = beta*t;
end
x = x+t*dx;
end
figure(1);
```

```
    semilogy([0:iters-2], fvals(1:(end-1))-fvals(end), 'o', ...
    [0:iters-2], fvals(1:(iters-1))-fvals(iters), 'b-');
    %This is approximated points and curves
    hold on
    end
    axis([0 25 1e-10 1e5]);
    xlabel('iteration');
    ylabel('error');
    hold off

\begin{lstlisting}
    % Newton's method applied to the dual.
    % mininimize b'*z - sum(log(A'*z)
    randn('state',3);
    rand('state',3);
    m = 100;  n = 500;
    A = randn(m,n);
    % make z0 dual feasible (A'*z0 = 1)
    z0 = randn(m,1);
    A = A + z0*(1 - A'*z0)'/(z0'*z0);
    % make x0 primal feasible, use the same b
    x0 = rand(n,1);
    b = A*x0;
    MAXITERS = 50;
    alpha = 0.1;
    beta  = 0.5;
    TOL = 1e-12;
    figure(2);
    randn('state',1);
    nostarts = 4;
    for starts = 1:nostarts
    % Starting point: generate random vector v and take
    % z = z0 + 0.99*s*v where s is maximum step to the boundary.
    v = randn(m,1);
    z = z0 + (0.99/max(-(A'*v)./(A'*z0)))*v;
    fvals = [];
    for iters=1:MAXITERS
    y = A'*z;
    f = b'*z - sum(log(y));  fvals = [fvals, f];
    gradient = b + -A*(1./y);
    d = 1./(y.^2);
    Hessan = (A .* (d(:,ones(1,m))'))*A';
    newton_step = -Hessan\gradient;
```

```
lambda = -gradient'*newton_step;
if (lambda/2 < TOL)
break;
end
dy = A'*newton_step;
t = 1;
while (b'*(z+t*newton_step)-sum(log(A'*(z+t*newton_step))) >
    f-alpha*lambda)
t = beta*t;
end
z = z+t*newton_step;
end
figure(2);
semilogy([0:iters-2], fvals(1:(iters-1)) - fvals(iters), 'o',
    ...
[0:iters-2], fvals(1:(iters-1)) - fvals(iters), 'b-');
hold on;
end
xlabel('iteration'); ylabel('error');
axis([0 10 1e-10 1e5]);
hold off

% Infeasible Newton method.
clear; clc
randn('state',3);
rand('state',3);
m = 100;  n = 500;
A = randn(m,n);
% make z0 dual feasible (A'*z0 = 1)
z0 = randn(m,1);
A = A + z0*(1 - A'*z0)'/(z0'*z0);
% make x0 primal feasible
x0 = rand(n,1);
b = A*x0;
max_iteration = 50;
alpha = 0.1;
beta  = 0.5;
TOL = 1e-12;
figure(3);
randn('state',2);
nostarts = 4;
for starts = 1:nostarts
x = rand(n,1);
```

```
        v = randn(m,1);
        residules = [];
        for iters=1:max_iteration
        g = -1./x;
        r = [g + A'*v;  A*x-b];
        normr = norm(r);  residules = [residules; normr];
        if (normr) < sqrt(TOL)
        break;
        end
        H = 1./(x.^2);  invH = x.^2;
        w = (A*diag(invH)*A') \  (2*A*x-b);
        dual_newton_step = w-v;
        newton_step = x - x.^2 .* (A'*w);
        t = 1;
        while norm([-1./(x+t*newton_step) + A'*(w+t*dual_newton_step)
            ; ...
        A*(x+t*newton_step)-b ]) > ...
        (1-alpha*t)*normr
        t = beta*t;
        end
        x = x+t*newton_step;  v = v+t*dual_newton_step;
        end
        semilogy([0:iters-1], residules(1:iters), 'o', [0:iters-1],
            ...
        residules(1:iters), 'b-');
        hold on
        end
        axis([0 25 1e-15 1e10]);
        xlabel('iteration');
        ylabel('residual');
        axis([0 25 1e-15 1e10]);
        hold off

    % Generate a strictly primal and dual feasible LP.
        %
        % (primal) minimize    c'*x
        %          subject to  A*x <= b
        %
        % (dual)   maximize    -b'*z
%           subject to  A'*z + c = 0, z >= 0
alpha = 0.01;
beta = 0.5;
randn('state',0);
```

```
rand('state',0);
m = 100;
n = 50;
A = randn(m,n);
b = rand(m,1);
%c = A'*rand(m,1);
c = rand(n,1);

% Solve using linprog.
x = linprog(c, A, b);
opt_val = c'*x;

% Make a change of variables  x := x + s*c, so that optimal value is
    1.
t = (1-opt_val)/(c'*c);
b = b + t*A*c;
x0 = t*c;



% xc is the point on the central path with barrier parameter t=1.
t = 1;
x = x0;
for k=1:100
    d = b-A*x;
    val = t*c'*x - sum(log(d));
    gradient = t*c + A'*(1./d);
    Hessian = A'*diag(1./(d.^2))*A;
    newton_step = -Hessian\gradient;
    lambda = gradient'*newton_step;
    t = 1;
    while (t*c'*(x+t*newton_step) - sum(log(b-A*(x+t*newton_step))) >
        val + alpha*t*lambda)
        t = t*beta;
    end
    x = x+t*newton_step;
    if ((-lambda/2) < 1e-6)
        break;
    end
end
xc = x;
```

```
% Figure  Barrier method starting at xc, for three values of mu.

mu_vals = [2 50 150];
[x,iters,gaps] = lp(A,b,c,xc,mu_vals(1),1e-6);
l = length(gaps);  iters1 = [];  gaps1 = [];
for i=1:l-1
    iters1 = [iters1 iters(i)-1 iters(i+1)-1];
    gaps1 = [gaps1 gaps(i) gaps(i)];
end
iters1 = [iters1 iters(l)-1]; gaps1 = [gaps1 gaps(l)];

[x,iters,gaps] = lp(A,b,c,xc,mu_vals(2),1e-6);
l = length(gaps);  iters2 = [];  gaps2 = [];
for i=1:l-1
    iters2 = [iters2 iters(i)-1 iters(i+1)-1];
    gaps2 = [gaps2 gaps(i) gaps(i)];
end
iters2 = [iters2 iters(l)-1]; gaps2 = [gaps2 gaps(l)];

[x,iters,gaps] = lp(A,b,c,xc,mu_vals(3),1e-6);
l = length(gaps);  iters3 = [];  gaps3 = [];
for i=1:l-1
    iters3 = [iters3 iters(i)-1 iters(i+1)-1];
    gaps3 = [gaps3 gaps(i) gaps(i)];
end
iters3 = [iters3 iters(l)-1]; gaps3 = [gaps3 gaps(l)];

figure(1)
semilogy(iters1,gaps1, iters2,gaps2, '-', iters3,gaps3, '--');
axis([0 90 1e-7 1e3]);
text(iters1(length(iters1)-20), gaps1(length(iters1)-20),'mu=2');
text(iters2(length(iters2)-2), gaps2(length(iters2)-2),'mu=50');
text(iters3(length(iters3)-5), gaps3(length(iters3)-5),'mu=100');
axis([0 90 0.5e-7 1e3]);
xlabel('Newton iterations');  ylabel('duality gap');

% Figure   Number of Newton iterations vs. mu.

muvals = [1.5 2 4 6 8 10:10:200];
noiters=zeros(1,length(muvals));
for i=1:length(muvals)
    [x,iters,gaps] = lp(A,b,c,xc,muvals(i),1e-3);
```

```
    noiters(i) = iters(length(iters));
end
figure(2)
plot(muvals,noiters,'o', muvals,noiters,'-');
axis([ 0 200 0 150]);
xlabel('mu'); ylabel('Newton iterations');

function [x, inniters, gaps] = lp(A, b, c, x0, mu, tol)

% [x, inniters, gaps] = lp(A, b, c, x0, mu, tol)
%
%      minimize    c'*x
%      subject to  A*x <= b
%
%      maximize    -b'*z
%      subject to  A'*z + c = 0
%                  z >= 0
%
% Barrier method for solving LP within absolute accuracy tol,
   starting
% with initial t = 1, at a strictly feasible x0.  We assume the
% problem is strictly dual feasible.
%
% inniters:  array with number of Newton iters per outer iteration
% gaps:   array with duality gaps at the end of each outer iteration

MAXITERS = 500;
ALPHA = 0.01;
BETA = 0.5;
NTTOL = 1e-4;      % stop inner iteration if lambda^2/2 < NTTOL

[m,n] = size(A);
t = 1;
x = x0;
gaps = [];
inniters = [];
for k=1:MAXITERS
    d = b-A*x;
    val = t*c'*x - sum(log(d));
    g = t*c + A'*(1./d);
    H = A'*diag(1./(d.^2))*A;
    newton_step = -H\g;
```

```
    lambda = g'*newton_step;
    s = 1;
    while (min(b-A*(x+s*newton_step)) < 0)
        s = BETA*s;
    end
    while (t*c'*(x+s*newton_step) - sum(log(b-A*(x+s*newton_step))) >
        val + ALPHA*s*lambda)
        s = BETA*s;
    end
    x = x+s*newton_step;
    if ((-lambda/2) < NTTOL)
        inniters = [inniters, k];
        z = 1./t*d; %-1/t(Ax-b)=1./t*d
        %z = (1./d) .* (1 + (A*newton_step)./d) / t;here we use
            improved method in ex11.9
        %gap=c'x+b'z, c=-A'z, b'z-z'Ax=b'z-(Ax)'z=(b-Ax)'z
        gap = (b-A*x)'*z;
        gaps = [gaps,gap];
        if (gap < tol)
            break;
        end
        t = t*mu;
    end
end
inniters = [inniters, k];
gaps = [gaps, gap];
disp(['Maxiters (', int2str(MAXITERS), ') exceeded.']);

    % Barrier method for standard form LPs
%
%     minimize   c'*x
%     subject to A*x = b
%                x >= 0


% Figure  Gap versus Newton iterations for three problems.
% for this method, we directly consider the initial point is strictly
% feasible, otherwise, we need phase I method.
mu=50;
disp('m=50, n=100.')
m = 50;
n = 100;
A = randn(m,n);
```

```
x0 = rand(n,1);
b = A*x0;
z = randn(m,1);
c = A'*z + rand(n,1);
[x, iters, gaps] = stdlp(A,b,c, mu,1e-3,1e-5);
l = length(gaps);  iters1 = [];  gaps1 = [];
%generate one gap from once time of iterations
for i=1:l-1
    iters1 = [iters1, iters(i)-1, iters(i+1)-1];
    gaps1 = [gaps1, gaps(i), gaps(i)];
end
iters1 = [iters1, iters(l)-1] - iters1(1);
gaps1 = [gaps1, gaps(l)];

disp('m=500, n=1000.')
m = 500;
n = 1000;
A = randn(m,n);
x0 = rand(n,1);
b = A*x0;
z = rand(m,1);
c = A'*z + rand(n,1);
[x, iters, gaps] = stdlp(A,b,c,mu,1e-3,1e-5);
l = length(gaps);  iters2 = [];  gaps2 = [];
for i=1:l-1
   iters2 = [iters2, iters(i)-1, iters(i+1)-1];
   gaps2 = [gaps2, gaps(i), gaps(i)];
end
iters2 = [iters2, iters(l)-1] - iters2(1);
gaps2 = [gaps2, gaps(l)];

disp('m=1000, n=2000.')
m = 1000;
n = 2000;
A = randn(m,n);
x0 = rand(n,1);
b = A*x0;
z = rand(m,1);
c = A'*z + rand(n,1);
[x, iters, gaps] = stdlp(A,b,c,mu,1e-3,1e-5);
l = length(gaps);  iters3 = [];  gaps3 = [];
for i=1:l-1
```

```
    iters3 = [iters3, iters(i)-1, iters(i+1)-1];
    gaps3 = [gaps3, gaps(i), gaps(i)];
end
iters3 = [iters3, iters(l)-1] - iters3(1);
gaps3 = [gaps3, gaps(l)];


semilogy(iters1,gaps1, iters2,gaps2, '--', iters3,gaps3, '-');
text(iters1(length(iters1)), gaps1(length(iters1)),'m=50');
text(iters2(length(iters2)), gaps2(length(iters2)),'m=100');
text(iters3(length(iters3)), gaps3(length(iters3)-1),'m=500');
xlabel('Newton iterations'); ylabel('Duality gap');


function [x, inniters, gaps] = stdlp(A, b, c,mu, tol, reltol)

% [x, inniters, gaps] = stdlp(A, b, c, x0, mu, tol, reltol)
%
% (primal) minimize  c'*x        (dual)  maximize   b'*z
%          subjec to A*x = b             subject to A'*z <= c
%                    x >= 0
%
% x0:  strictly feasible starting point, not necesarily on central
    path
% inniters:  array with number of Newton iterations per outer
    iteration
% gaps:  array with duality gaps at the end of each outer iteration

MAXITERS = 500;
alpha = 0.01;
beta = 0.5;
NTTOL = 1e-5;     % stop inner iteration if lambda^2/2 < NTTOL
[m,n] = size(A);
%random initia x and dual initial
randn('state',245);
x = rand(n,1);
w = randn(m,1);
t = 1;
resdls = [];
gaps = [];
inniters = [];
for k=1:MAXITERS
```

```
    %solve the equivalent approximated problem and find its optimal
        points x^* and z,
    %notice that we times t for function, we need to divide it.
    %approximated function is t*c'*x - sum(log(x));
    g = t*c - 1./x;
    %Hessan matrix is diag(1/x.^2)
    r = [g + A'*w;  A*x-b];
    normr = norm(r);  resdls = [resdls; normr];
    if (normr) < sqrt(NTTOL)
        w = -w/t;
        %dual = b'*z=(Ax)'*z=x'*A'*z;
        gap = x'*(c-A'*w);
        inniters = [inniters, k];
        gaps = [gaps,gap];
        if (gap < tol)
            return;
        end
        t = t*mu;
    else
    z =  (A*spdiags(x.^2,0,n,n)*A') \ (A*x-b-A*(g.*(x.^2)) );
    dw = z-w;
    dx= - spdiags(x.^2,0,n,n)*(A'*z+g);
    s = 1;
    while (min(x+s*dx) < 0)
        s = beta*s;
    end
    while norm([t*c-1./(x+s*dx) + A'*(w+s*dw);  A*(x+s*dx)-b ]) > ...
            (1-alpha*s)*normr
            s = beta*s;
    end
        x = x+s*dx;  w = w+s*dw;
    end
end
disp(normr);
end

        % Primal-dual method for a small LP.
        randn('state',0);
        rand('state',0);
        m = 200;
        n = 100;
        A = randn(m,n);
        b = rand(m,1);
```

```
z0 = rand(m,1);
c = -A'*z0;
x = linprog(c, A, b);
s = (1-c'*x)/norm(c)^2;
b = b + s*A*c;
x0 = s*c;

[x, z, iters, gaps, rs] = lp_pd(A, b, c, x0);

figure(1)
semilogy([0:length(gaps)-1], gaps,'-', [0:length(gaps)-1],
    gaps,'o');
xlabel('iteration number'); ylabel('surrogate gap');

figure(2)
semilogy([0:length(gaps)-1], rs, '-', [0:length(gaps)-1], rs,
     'o');
axis([0 30 1e-16 1e5]);
xlabel('iteration number'); ylabel('residual');


function [x, lambda, iters, gaps, resdls] = lp_pd(A, b, c, x0
    )
% Solve
%
%     minimize    c'*x                maximize    -b'*z
%     subject to  A*x <= b            subject to  A'*z + c =
     0
%                                                 z >= 0
%
MAXITERS = 500;
TOL = 1e-8;
RESTOL = 1e-8;
MU = 10;
alpha = 0.01;
beta = 0.5;

[m,n] = size(A);
gaps = [];  resdls = [];
x = x0;
f_i = b-A*x;  %here we derive-f_i for the real question
%first lambda is arbitary
```

```
lambda = rand(m,1);
for iters = 1:MAXITERS
gap = f_i'*lambda;  gaps = [gaps, gap];
res = A'*lambda  + c; %residual_dual
resdls = [resdls, norm(res)];
if ((gap < TOL) && (norm(res) < RESTOL))
return;
end
t_inverse = gap/(m*MU);
sol = -[zeros(n,n), A'; -lambda.*A, -diag(-f_i)] \ [A'*lambda
   +c; lambda.*f_i-t_inverse];
%A' is the derivative of f_i
dx = sol(1:n);
dlambda = sol(n+[1:m]);
df = -A*dx;

% backtracking line search
r = [c+A'*lambda;  lambda.*f_i-t_inverse];
step = min(1.0, 0.99/max(-dlambda./lambda));
while (min(f_i+step*df) <= 0)
step = beta*step;
end
newlambda = lambda+step*dlambda;
newx = x+step*dx;
newf_i = f_i+step*df;
newr = [c+A'*newlambda;newlambda.*newf_i-t_inverse];
while (norm(newr) > (1-alpha*step)*norm(r))
step = beta*step;
newlambda = lambda+step*dlambda;  newx = x+step*dx;  newf_i =
    f_i+step*df;
newr = [c+A'*newlambda;  newlambda.*newf_i-t_inverse];
end
x = x+step*dx;  lambda = lambda +step*dlambda;  f_i = b-A*x;
end
```

Bibliography

Andrei, N. (2017). *Sequential Quadratic Programming (SQP)*, pages 269–288. Springer International Publishing, Cham.

Andrei, N. et al. (2020). *Nonlinear conjugate gradient methods for unconstrained optimization*. Springer.

Bertsekas, D. P. (2014). *Constrained optimization and Lagrange multiplier methods*. Academic press.

Boyd, S. P. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

Conn, A., Gould, N., and Toint, P. L. (1994). A note on using alternative second-order models for the subproblems arising in barrier function methods for minimization. *Numerische Mathematik*, 68:17–33.

Dantzig, G. B. (1955). Linear programming under uncertainty. *Management science*, 1(3-4):197–206.

Dantzig, G. B. (1991). Linear programming. *History of mathematical programming*, pages 19–31.

Dennis Jr, J. E. and Schnabel, R. B. (1996). *Numerical methods for unconstrained optimization and nonlinear equations*. SIAM.

Du, D.-Z., Pardalos, P. M., and Wu, W. (2009). *History of optimizationHistory of Optimization*, pages 1538–1542. Springer US, Boston, MA.

El-Bakry, A., Tapia, R., Tsuchiya, T., and Zhang, Y. (1996). On the formulation of the newton interior-point method for non-linear programming. *Journal of Optimization Theory and Applications*, 89:507–541.

Golub, G. H. and van Loan, C. F. (2013). *Matrix Computations*. JHU Press, fourth edition.

Hager, W. and Zhang, H. (2006). A survey of nonlinear conjugate gradient method. 2.

Hillier, F. and Lieberman, G. (2015). *Introduction to operations research*.

Polyak, B. (2007). Newton' s method and its use in optimization. *European Journal of Operational Research*, 181:1086–1096.

Šolcová, A. (2004). *The Founders of the Conjugate Gradient Method*, pages 3–10. Springer Berlin Heidelberg, Berlin, Heidelberg.

Steihaug, T. (1983). The conjugate gradient method and trust regions in large scale optimization. *SIAM Journal on Numerical Analysis*, 20(3):626–637.

Wright, M. (2005). The interior-point revolution in optimization: history, recent developments, and lasting consequences. *Bulletin of the American mathematical society*, 42(1):39–56.