Chapter 4

Convex optimization problems

4.1 Optimization problems

4.1.1 Basic terminology

We use the notation

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \le 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{array} \tag{4.1}$$

to describe the problem of finding an x that minimizes $f_0(x)$ among all x that satisfy the conditions $f_i(x) \le 0$, $i = 1, \ldots, m$, and $h_i(x) = 0$, $i = 1, \ldots, p$. We call $x \in \mathbf{R}^n$ the *optimization variable* and the function $f_0 : \mathbf{R}^n \to \mathbf{R}$ the *objective function* or *cost function*. The inequalities $f_i(x) \le 0$ are called *inequality constraints*, and the corresponding functions $f_i : \mathbf{R}^n \to \mathbf{R}$ are called the *inequality constraint functions*. The equations $h_i(x) = 0$ are called the *equality constraints*, and the functions $h_i : \mathbf{R}^n \to \mathbf{R}$ are the *equality constraint functions*. If there are no constraints (*i.e.*, $m = p = 0$) we say the problem (4.1) is *unconstrained*.

The set of points for which the objective and all constraint functions are defined,

$$\mathcal{D} = \bigcap_{i=0}^{m} \mathbf{dom}\, f_i \;\cap\; \bigcap_{i=1}^{p} \mathbf{dom}\, h_i,$$

is called the *domain* of the optimization problem (4.1). A point $x \in \mathcal{D}$ is *feasible* if it satisfies the constraints $f_i(x) \le 0$, $i = 1, \ldots, m$, and $h_i(x) = 0$, $i = 1, \ldots, p$. The problem (4.1) is said to be feasible if there exists at least one feasible point, and *infeasible* otherwise. The set of all feasible points is called the *feasible set* or the *constraint set*.

The *optimal value* $p^\star$ of the problem (4.1) is defined as

$$p^\star = \inf\left\{ f_0(x) \mid f_i(x) \le 0,\ i = 1, \ldots, m,\ h_i(x) = 0,\ i = 1, \ldots, p \right\}.$$

We allow $p^\star$ to take on the extended values $\pm\infty$. If the problem is infeasible, we have $p^\star = \infty$ (following the standard convention that the infimum of the empty set is ∞). If there are feasible points x_k with $f_0(x_k) \to -\infty$ as $k \to \infty$, then $p^\star = -\infty$, and we say the problem (4.1) is *unbounded below*.

Optimal and locally optimal points

We say $x^\star$ is an *optimal point*, or solves the problem (4.1), if $x^\star$ is feasible and $f_0(x^\star) = p^\star$. The set of all optimal points is the *optimal set*, denoted

$$X_{\text{opt}} = \{x \mid f_i(x) \le 0,\ i = 1, \ldots, m,\ h_i(x) = 0,\ i = 1, \ldots, p,\ f_0(x) = p^\star\}.$$

If there exists an optimal point for the problem (4.1), we say the optimal value is *attained* or *achieved*, and the problem is *solvable*. If X_{opt} is empty, we say the optimal value is not attained or not achieved. (This always occurs when the problem is unbounded below.) A feasible point x with $f_0(x) \le p^\star + \epsilon$ (where $\epsilon > 0$) is called *ϵ-suboptimal*, and the set of all ϵ-suboptimal points is called the *ϵ-suboptimal set* for the problem (4.1).

We say a feasible point x is *locally optimal* if there is an $R > 0$ such that

$$\begin{aligned} f_0(x) = \inf\{f_0(z) \mid\ & f_i(z) \le 0,\ i = 1, \ldots, m, \\ & h_i(z) = 0,\ i = 1, \ldots, p,\ \|z - x\|_2 \le R\}, \end{aligned}$$

or, in other words, x solves the optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(z) \\ \text{subject to} & f_i(z) \le 0, \quad i = 1, \ldots, m \\ & h_i(z) = 0, \quad i = 1, \ldots, p \\ & \|z - x\|_2 \le R \end{array}$$

with variable z. Roughly speaking, this means x minimizes f_0 over nearby points in the feasible set. The term 'globally optimal' is sometimes used for 'optimal' to distinguish between 'locally optimal' and 'optimal'. Throughout this book, however, optimal will mean globally optimal.

If x is feasible and $f_i(x) = 0$, we say the ith inequality constraint $f_i(x) \le 0$ is *active* at x. If $f_i(x) < 0$, we say the constraint $f_i(x) \le 0$ is *inactive*. (The equality constraints are active at all feasible points.) We say that a constraint is *redundant* if deleting it does not change the feasible set.

Example 4.1 We illustrate these definitions with a few simple unconstrained optimization problems with variable $x \in \mathbf{R}$, and $\mathbf{dom}\, f_0 = \mathbf{R}_{++}$.

- $f_0(x) = 1/x$: $p^\star = 0$, but the optimal value is not achieved.
- $f_0(x) = -\log x$: $p^\star = -\infty$, so this problem is unbounded below.
- $f_0(x) = x \log x$: $p^\star = -1/e$, achieved at the (unique) optimal point $x^\star = 1/e$.

Feasibility problems

If the objective function is identically zero, the optimal value is either zero (if the feasible set is nonempty) or ∞ (if the feasible set is empty). We call this the *feasibility problem*, and will sometimes write it as

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

The feasibility problem is thus to determine whether the constraints are consistent, and if so, find a point that satisfies them.

4.1.2 Expressing problems in standard form

We refer to (4.1) as an optimization problem in *standard form*. In the standard form problem we adopt the convention that the righthand side of the inequality and equality constraints are zero. This can always be arranged by subtracting any nonzero righthand side: we represent the equality constraint $g_i(x) = \tilde{g}_i(x)$, for example, as $h_i(x) = 0$, where $h_i(x) = g_i(x) - \tilde{g}_i(x)$. In a similar way we express inequalities of the form $f_i(x) \geq 0$ as $-f_i(x) \leq 0$.

Example 4.2 *Box constraints.* Consider the optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & l_i \leq x_i \leq u_i, \quad i = 1, \ldots, n, \end{array}$$

where $x \in \mathbf{R}^n$ is the variable. The constraints are called *variable bounds* (since they give lower and upper bounds for each x_i) or *box constraints* (since the feasible set is a box).

We can express this problem in standard form as

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & l_i - x_i \leq 0, \quad i = 1, \ldots, n \\ & x_i - u_i \leq 0, \quad i = 1, \ldots, n. \end{array}$$

There are $2n$ inequality constraint functions:

$$f_i(x) = l_i - x_i, \quad i = 1, \ldots, n,$$

and

$$f_i(x) = x_{i-n} - u_{i-n}, \quad i = n+1, \ldots, 2n.$$

Maximization problems

We concentrate on the minimization problem by convention. We can solve the *maximization* problem

$$\begin{array}{ll} \text{maximize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{array} \qquad (4.2)$$

by minimizing the function $-f_0$ subject to the constraints. By this correspondence we can define all the terms above for the maximization problem (4.2). For example the optimal value of (4.2) is defined as

$$p^\star = \sup\{f_0(x) \mid f_i(x) \le 0,\ i=1,\dots,m,\ h_i(x)=0,\ i=1,\dots,p\},$$

and a feasible point x is ϵ-suboptimal if $f_0(x) \ge p^\star - \epsilon$. When the maximization problem is considered, the objective is sometimes called the *utility* or *satisfaction level* instead of the cost.

4.1.3 Equivalent problems

In this book we will use the notion of equivalence of optimization problems in an informal way. We call two problems *equivalent* if from a solution of one, a solution of the other is readily found, and vice versa. (It is possible, but complicated, to give a formal definition of equivalence.)

As a simple example, consider the problem

$$\begin{array}{ll} \text{minimize} & \tilde{f}(x) = \alpha_0 f_0(x) \\ \text{subject to} & \tilde{f}_i(x) = \alpha_i f_i(x) \le 0, \quad i=1,\dots,m \\ & \tilde{h}_i(x) = \beta_i h_i(x) = 0, \quad i=1,\dots,p, \end{array} \tag{4.3}$$

where $\alpha_i > 0$, $i=0,\dots,m$, and $\beta_i \neq 0$, $i=1,\dots,p$. This problem is obtained from the standard form problem (4.1) by scaling the objective and inequality constraint functions by positive constants, and scaling the equality constraint functions by nonzero constants. As a result, the feasible sets of the problem (4.3) and the original problem (4.1) are identical. A point x is optimal for the original problem (4.1) if and only if it is optimal for the scaled problem (4.3), so we say the two problems are equivalent. The two problems (4.1) and (4.3) are not, however, the same (unless α_i and β_i are all equal to one), since the objective and constraint functions differ.

We now describe some general transformations that yield equivalent problems.

Change of variables

Suppose $\phi : \mathbf{R}^n \to \mathbf{R}^n$ is one-to-one, with image covering the problem domain $\mathcal{D}$, *i.e.*, $\phi(\mathbf{dom}\,\phi) \supseteq \mathcal{D}$. We define functions $\tilde{f}_i$ and $\tilde{h}_i$ as

$$\tilde{f}_i(z) = f_i(\phi(z)), \quad i=0,\dots,m, \qquad \tilde{h}_i(z) = h_i(\phi(z)), \quad i=1,\dots,p.$$

Now consider the problem

$$\begin{array}{ll} \text{minimize} & \tilde{f}_0(z) \\ \text{subject to} & \tilde{f}_i(z) \le 0, \quad i=1,\dots,m \\ & \tilde{h}_i(z) = 0, \quad i=1,\dots,p, \end{array} \tag{4.4}$$

with variable z. We say that the standard form problem (4.1) and the problem (4.4) are related by the *change of variable* or *substitution of variable* $x = \phi(z)$.

The two problems are clearly equivalent: if x solves the problem (4.1), then $z = \phi^{-1}(x)$ solves the problem (4.4); if z solves the problem (4.4), then $x = \phi(z)$ solves the problem (4.1).

Transformation of objective and constraint functions

Suppose that $\psi_0 : \mathbf{R} \to \mathbf{R}$ is monotone increasing, $\psi_1, \ldots, \psi_m : \mathbf{R} \to \mathbf{R}$ satisfy $\psi_i(u) \leq 0$ if and only if $u \leq 0$, and $\psi_{m+1}, \ldots, \psi_{m+p} : \mathbf{R} \to \mathbf{R}$ satisfy $\psi_i(u) = 0$ if and only if $u = 0$. We define functions $\tilde{f}_i$ and $\tilde{h}_i$ as the compositions

$$\tilde{f}_i(x) = \psi_i(f_i(x)), \quad i = 0, \ldots, m, \qquad \tilde{h}_i(x) = \psi_{m+i}(h_i(x)), \quad i = 1, \ldots, p.$$

Evidently the associated problem

$$\begin{array}{ll} \text{minimize} & \tilde{f}_0(x) \\ \text{subject to} & \tilde{f}_i(x) \leq 0, \quad i = 1, \ldots, m \\ & \tilde{h}_i(x) = 0, \quad i = 1, \ldots, p \end{array}$$

and the standard form problem (4.1) are equivalent; indeed, the feasible sets are identical, and the optimal points are identical. (The example (4.3) above, in which the objective and constraint functions are scaled by appropriate constants, is the special case when all ψ_i are linear.)

Example 4.3 *Least-norm and least-norm-squared problems.* As a simple example consider the unconstrained Euclidean norm minimization problem

$$\text{minimize} \quad \|Ax - b\|_2, \tag{4.5}$$

with variable $x \in \mathbf{R}^n$. Since the norm is always nonnegative, we can just as well solve the problem

$$\text{minimize} \quad \|Ax - b\|_2^2 = (Ax - b)^T(Ax - b), \tag{4.6}$$

in which we minimize the square of the Euclidean norm. The problems (4.5) and (4.6) are clearly equivalent; the optimal points are the same. The two problems are not the same, however. For example, the objective in (4.5) is not differentiable at any x with $Ax - b = 0$, whereas the objective in (4.6) is differentiable for all x (in fact, quadratic).

Slack variables

One simple transformation is based on the observation that $f_i(x) \leq 0$ if and only if there is an $s_i \geq 0$ that satisfies $f_i(x) + s_i = 0$. Using this transformation we obtain the problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & s_i \geq 0, \quad i = 1, \ldots, m \\ & f_i(x) + s_i = 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p, \end{array} \tag{4.7}$$

where the variables are $x \in \mathbf{R}^n$ and $s \in \mathbf{R}^m$. This problem has $n + m$ variables, m inequality constraints (the nonnegativity constraints on s_i), and $m + p$ equality constraints. The new variable s_i is called the *slack variable* associated with the original inequality constraint $f_i(x) \leq 0$. Introducing slack variables replaces each inequality constraint with an equality constraint, and a nonnegativity constraint.

The problem (4.7) is equivalent to the original standard form problem (4.1). Indeed, if (x, s) is feasible for the problem (4.7), then x is feasible for the original problem, since $s_i = -f_i(x) \geq 0$. Conversely, if x is feasible for the original problem, then (x, s) is feasible for the problem (4.7), where we take $s_i = -f_i(x)$. Similarly, x is optimal for the original problem (4.1) if and only if (x, s) is optimal for the problem (4.7), where $s_i = -f_i(x)$.

Eliminating equality constraints

If we can explicitly parametrize all solutions of the equality constraints

$$h_i(x) = 0, \quad i = 1, \ldots, p, \tag{4.8}$$

using some parameter $z \in \mathbf{R}^k$, then we can *eliminate* the equality constraints from the problem, as follows. Suppose the function $\phi : \mathbf{R}^k \to \mathbf{R}^n$ is such that x satisfies (4.8) if and only if there is some $z \in \mathbf{R}^k$ such that $x = \phi(z)$. The optimization problem

$$\begin{array}{ll} \text{minimize} & \tilde{f}_0(z) = f_0(\phi(z)) \\ \text{subject to} & \tilde{f}_i(z) = f_i(\phi(z)) \leq 0, \quad i = 1, \ldots, m \end{array}$$

is then equivalent to the original problem (4.1). This transformed problem has variable $z \in \mathbf{R}^k$, m inequality constraints, and no equality constraints. If z is optimal for the transformed problem, then $x = \phi(z)$ is optimal for the original problem. Conversely, if x is optimal for the original problem, then (since x is feasible) there is at least one z such that $x = \phi(z)$. Any such z is optimal for the transformed problem.

Eliminating linear equality constraints

The process of eliminating variables can be described more explicitly, and easily carried out numerically, when the equality constraints are all linear, *i.e.*, have the form $Ax = b$. If $Ax = b$ is inconsistent, *i.e.*, $b \notin \mathcal{R}(A)$, then the original problem is infeasible. Assuming this is not the case, let x_0 denote any solution of the equality constraints. Let $F \in \mathbf{R}^{n \times k}$ be any matrix with $\mathcal{R}(F) = \mathcal{N}(A)$, so the general solution of the linear equations $Ax = b$ is given by $Fz + x_0$, where $z \in \mathbf{R}^k$. (We can choose F to be full rank, in which case we have $k = n - \mathbf{rank}\, A$.)

Substituting $x = Fz + x_0$ into the original problem yields the problem

$$\begin{array}{ll} \text{minimize} & f_0(Fz + x_0) \\ \text{subject to} & f_i(Fz + x_0) \leq 0, \quad i = 1, \ldots, m, \end{array}$$

with variable z, which is equivalent to the original problem, has no equality constraints, and $\mathbf{rank}\, A$ fewer variables.

Introducing equality constraints

We can also *introduce* equality constraints and new variables into a problem. Instead of describing the general case, which is complicated and not very illuminating, we give a typical example that will be useful later. Consider the problem

$$\begin{array}{ll} \text{minimize} & f_0(A_0 x + b_0) \\ \text{subject to} & f_i(A_i x + b_i) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p, \end{array}$$

where $x \in \mathbf{R}^n$, $A_i \in \mathbf{R}^{k_i \times n}$, and $f_i : \mathbf{R}^{k_i} \to \mathbf{R}$. In this problem the objective and constraint functions are given as compositions of the functions f_i with affine transformations defined by $A_i x + b_i$.

We introduce new variables $y_i \in \mathbf{R}^{k_i}$, as well as new equality constraints $y_i = A_i x + b_i$, for $i = 0, \ldots, m$, and form the equivalent problem

$$\begin{array}{ll} \text{minimize} & f_0(y_0) \\ \text{subject to} & f_i(y_i) \leq 0, \quad i = 1, \ldots, m \\ & y_i = A_i x + b_i, \quad i = 0, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p. \end{array}$$

This problem has $k_0 + \cdots + k_m$ new variables,

$$y_0 \in \mathbf{R}^{k_0}, \quad \ldots, \quad y_m \in \mathbf{R}^{k_m},$$

and $k_0 + \cdots + k_m$ new equality constraints,

$$y_0 = A_0 x + b_0, \quad \ldots, \quad y_m = A_m x + b_m.$$

The objective and inequality constraints in this problem are *independent*, *i.e.*, involve different optimization variables.

Optimizing over some variables

We always have

$$\inf_{x,y} f(x,y) = \inf_x \tilde{f}(x)$$

where $\tilde{f}(x) = \inf_y f(x,y)$. In other words, we can always minimize a function by first minimizing over some of the variables, and then minimizing over the remaining ones. This simple and general principle can be used to transform problems into equivalent forms. The general case is cumbersome to describe and not illuminating, so we describe instead an example.

Suppose the variable $x \in \mathbf{R}^n$ is partitioned as $x = (x_1, x_2)$, with $x_1 \in \mathbf{R}^{n_1}$, $x_2 \in \mathbf{R}^{n_2}$, and $n_1 + n_2 = n$. We consider the problem

$$\begin{array}{ll} \text{minimize} & f_0(x_1, x_2) \\ \text{subject to} & f_i(x_1) \leq 0, \quad i = 1, \ldots, m_1 \\ & \tilde{f}_i(x_2) \leq 0, \quad i = 1, \ldots, m_2, \end{array} \tag{4.9}$$

in which the constraints are independent, in the sense that each constraint function depends on x_1 or x_2. We first minimize over x_2. Define the function $\tilde{f}_0$ of x_1 by

$$\tilde{f}_0(x_1) = \inf\{f_0(x_1, z) \mid \tilde{f}_i(z) \leq 0, \ i = 1, \ldots, m_2\}.$$

The problem (4.9) is then equivalent to

$$\begin{array}{ll} \text{minimize} & \tilde{f}_0(x_1) \\ \text{subject to} & f_i(x_1) \leq 0, \quad i = 1, \ldots, m_1. \end{array} \tag{4.10}$$

Example 4.4 *Minimizing a quadratic function with constraints on some variables.* Consider a problem with strictly convex quadratic objective, with some of the variables unconstrained:

$$\begin{array}{ll} \text{minimize} & x_1^T P_{11} x_1 + 2x_1^T P_{12} x_2 + x_2^T P_{22} x_2 \\ \text{subject to} & f_i(x_1) \leq 0, \quad i = 1, \ldots, m, \end{array}$$

where P_{11} and P_{22} are symmetric. Here we can analytically minimize over x_2:

$$\inf_{x_2} \left(x_1^T P_{11} x_1 + 2x_1^T P_{12} x_2 + x_2^T P_{22} x_2 \right) = x_1^T \left(P_{11} - P_{12} P_{22}^{-1} P_{12}^T \right) x_1$$

(see §A.5.5). Therefore the original problem is equivalent to

$$\begin{array}{ll} \text{minimize} & x_1^T \left(P_{11} - P_{12} P_{22}^{-1} P_{12}^T \right) x_1 \\ \text{subject to} & f_i(x_1) \leq 0, \quad i = 1, \ldots, m. \end{array}$$

Epigraph problem form

The *epigraph form* of the standard problem (4.1) is the problem

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & f_0(x) - t \leq 0 \\ & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p, \end{array} \tag{4.11}$$

with variables $x \in \mathbf{R}^n$ and $t \in \mathbf{R}$. We can easily see that it is equivalent to the original problem: (x, t) is optimal for (4.11) if and only if x is optimal for (4.1) and $t = f_0(x)$. Note that the objective function of the epigraph form problem is a *linear* function of the variables x, t.

The epigraph form problem (4.11) can be interpreted geometrically as an optimization problem in the 'graph space' (x, t): we minimize t over the epigraph of f_0, subject to the constraints on x. This is illustrated in figure 4.1.

Implicit and explicit constraints

By a simple trick already mentioned in §3.1.2, we can include any of the constraints *implicitly* in the objective function, by redefining its domain. As an extreme example, the standard form problem can be expressed as the *unconstrained* problem

$$\text{minimize} \quad F(x), \tag{4.12}$$

where we define the function F as f_0, but with domain restricted to the feasible set:

$$\mathbf{dom}\, F = \{x \in \mathbf{dom}\, f_0 \mid f_i(x) \leq 0,\ i = 1, \ldots, m,\ h_i(x) = 0,\ i = 1, \ldots, p\},$$

and $F(x) = f_0(x)$ for $x \in \mathbf{dom}\, F$. (Equivalently, we can define $F(x)$ to have value ∞ for x not feasible.) The problems (4.1) and (4.12) are clearly equivalent: they have the same feasible set, optimal points, and optimal value.

Of course this transformation is nothing more than a notational trick. Making the constraints implicit has not made the problem any easier to analyze or solve,

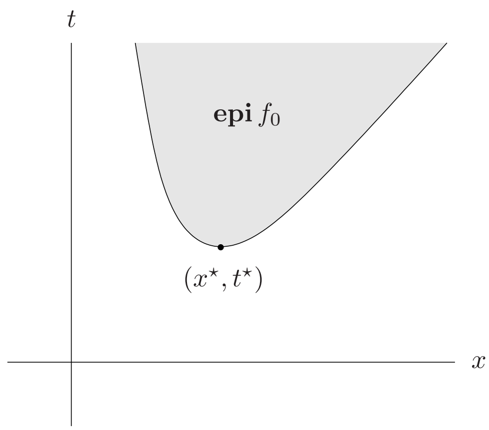


Figure 4.1 Geometric interpretation of epigraph form problem, for a problem with no constraints. The problem is to find the point in the epigraph (shown shaded) that minimizes t, *i.e.*, the 'lowest' point in the epigraph. The optimal point is $(x^\star, t^\star)$.

even though the problem (4.12) is, at least nominally, unconstrained. In some ways the transformation makes the problem more difficult. Suppose, for example, that the objective f_0 in the original problem is differentiable, so in particular its domain is open. The restricted objective function F is probably not differentiable, since its domain is likely not to be open.

Conversely, we will encounter problems with implicit constraints, which we can then make explicit. As a simple example, consider the unconstrained problem

$$\text{minimize} \quad f(x) \tag{4.13}$$

where the function f is given by

$$f(x) = \begin{cases} x^T x & Ax = b \\ \infty & \text{otherwise.} \end{cases}$$

Thus, the objective function is equal to the quadratic form $x^T x$ on the affine set defined by $Ax = b$, and ∞ off the affine set. Since we can clearly restrict our attention to points that satisfy $Ax = b$, we say that the problem (4.13) has an *implicit equality constraint* $Ax = b$ hidden in the objective. We can make the implicit equality constraint explicit, by forming the equivalent problem

$$\begin{array}{ll} \text{minimize} & x^T x \\ \text{subject to} & Ax = b. \end{array} \tag{4.14}$$

While the problems (4.13) and (4.14) are clearly equivalent, they are not the same. The problem (4.13) is unconstrained, but its objective function is not differentiable. The problem (4.14), however, has an equality constraint, but its objective and constraint functions are differentiable.

4.1.4 Parameter and oracle problem descriptions

For a problem in the standard form (4.1), there is still the question of how the objective and constraint functions are specified. In many cases these functions have some analytical or closed form, *i.e.*, are given by a formula or expression that involves the variable x as well as some parameters. Suppose, for example, the objective is quadratic, so it has the form $f_0(x) = (1/2)x^T Px + q^T x + r$. To specify the objective function we give the coefficients (also called *problem parameters* or *problem data*) $P \in \mathbf{S}^n$, $q \in \mathbf{R}^n$, and $r \in \mathbf{R}$. We call this a *parameter problem description*, since the specific problem to be solved (*i.e.*, the problem instance) is specified by giving the values of the parameters that appear in the expressions for the objective and constraint functions.

In other cases the objective and constraint functions are described by *oracle* models (which are also called *black box* or *subroutine* models). In an oracle model, we do not know f explicitly, but can evaluate $f(x)$ (and usually also some derivatives) at any $x \in \mathbf{dom}\, f$. This is referred to as *querying the oracle*, and is usually associated with some cost, such as time. We are also given some prior information about the function, such as convexity and a bound on its values. As a concrete example of an oracle model, consider an unconstrained problem, in which we are to minimize the function f. The function value $f(x)$ and its gradient $\nabla f(x)$ are evaluated in a subroutine. We can call the subroutine at any $x \in \mathbf{dom}\, f$, but do not have access to its source code. Calling the subroutine with argument x yields (when the subroutine returns) $f(x)$ and $\nabla f(x)$. Note that in the oracle model, we never really know the function; we only know the function value (and some derivatives) at the points where we have queried the oracle. (We also know some given prior information about the function, such as differentiability and convexity.)

In practice the distinction between a parameter and oracle problem description is not so sharp. If we are given a parameter problem description, we can construct an oracle for it, which simply evaluates the required functions and derivatives when queried. Most of the algorithms we study in part III work with an oracle model, but can be made more efficient when they are restricted to solve a specific parametrized family of problems.

4.2 Convex optimization

4.2.1 Convex optimization problems in standard form

A *convex optimization problem* is one of the form

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & a_i^T x = b_i, \quad i = 1, \ldots, p, \end{array} \tag{4.15}$$

where $f_0, \ldots, f_m$ are convex functions. Comparing (4.15) with the general standard form problem (4.1), the convex problem has three additional requirements:

- the objective function must be convex,
- the inequality constraint functions must be convex,
- the equality constraint functions $h_i(x) = a_i^T x - b_i$ must be affine.

We immediately note an important property: The feasible set of a convex optimization problem is convex, since it is the intersection of the domain of the problem

$$\mathcal{D} = \bigcap_{i=0}^{m} \mathbf{dom}\, f_i,$$

which is a convex set, with m (convex) sublevel sets $\{x \mid f_i(x) \le 0\}$ and p hyperplanes $\{x \mid a_i^T x = b_i\}$. (We can assume without loss of generality that $a_i \ne 0$: if $a_i = 0$ and $b_i = 0$ for some i, then the ith equality constraint can be deleted; if $a_i = 0$ and $b_i \ne 0$, the ith equality constraint is inconsistent, and the problem is infeasible.) Thus, in a convex optimization problem, we minimize a convex objective function over a convex set.

If f_0 is quasiconvex instead of convex, we say the problem (4.15) is a (standard form) *quasiconvex optimization problem*. Since the sublevel sets of a convex or quasiconvex function are convex, we conclude that for a convex or quasiconvex optimization problem the ϵ-suboptimal sets are convex. In particular, the optimal set is convex. If the objective is strictly convex, then the optimal set contains at most one point.

Concave maximization problems

With a slight abuse of notation, we will also refer to

$$\begin{array}{ll} \text{maximize} & f_0(x) \\ \text{subject to} & f_i(x) \le 0, \quad i = 1, \ldots, m \\ & a_i^T x = b_i, \quad i = 1, \ldots, p, \end{array} \tag{4.16}$$

as a convex optimization problem if the objective function f_0 is concave, and the inequality constraint functions $f_1, \ldots, f_m$ are convex. This *concave maximization problem* is readily solved by minimizing the convex objective function $-f_0$. All of the results, conclusions, and algorithms that we describe for the minimization problem are easily transposed to the maximization case. In a similar way the maximization problem (4.16) is called *quasiconvex* if f_0 is quasiconcave.

Abstract form convex optimization problem

It is important to note a subtlety in our definition of convex optimization problem. Consider the example with $x \in \mathbf{R}^2$,

$$\begin{array}{ll} \text{minimize} & f_0(x) = x_1^2 + x_2^2 \\ \text{subject to} & f_1(x) = x_1/(1 + x_2^2) \le 0 \\ & h_1(x) = (x_1 + x_2)^2 = 0, \end{array} \tag{4.17}$$

which is in the standard form (4.1). This problem is *not* a convex optimization problem in standard form since the equality constraint function h_1 is not affine, and the inequality constraint function f_1 is not convex. Nevertheless the feasible set, which is $\{x \mid x_1 \leq 0,\ x_1 + x_2 = 0\}$, is convex. So although in this problem we are minimizing a convex function f_0 over a convex set, it is not a convex optimization problem by our definition.

Of course, the problem is readily reformulated as

$$\begin{array}{ll} \text{minimize} & f_0(x) = x_1^2 + x_2^2 \\ \text{subject to} & \tilde{f}_1(x) = x_1 \leq 0 \\ & \tilde{h}_1(x) = x_1 + x_2 = 0, \end{array} \tag{4.18}$$

which is in standard convex optimization form, since f_0 and $\tilde{f}_1$ are convex, and $\tilde{h}_1$ is affine.

Some authors use the term *abstract convex optimization problem* to describe the (abstract) problem of minimizing a convex function over a convex set. Using this terminology, the problem (4.17) is an abstract convex optimization problem. *We will not use this terminology in this book.* For us, a convex optimization problem is not just one of minimizing a convex function over a convex set; it is also required that the feasible set be described specifically by a set of inequalities involving convex functions, and a set of linear equality constraints. The problem (4.17) is *not* a convex optimization problem, but the problem (4.18) *is* a convex optimization problem. (The two problems are, however, equivalent.)

Our adoption of the stricter definition of convex optimization problem does not matter much in practice. To solve the abstract problem of minimizing a convex function over a convex set, we need to find a description of the set in terms of convex inequalities and linear equality constraints. As the example above suggests, this is usually straightforward.

4.2.2 Local and global optima

A fundamental property of convex optimization problems is that any locally optimal point is also (globally) optimal. To see this, suppose that x is locally optimal for a convex optimization problem, *i.e.*, x is feasible and

$$f_0(x) = \inf\{f_0(z) \mid z \text{ feasible},\ \|z - x\|_2 \leq R\}, \tag{4.19}$$

for some $R > 0$. Now suppose that x is *not* globally optimal, *i.e.*, there is a feasible y such that $f_0(y) < f_0(x)$. Evidently $\|y - x\|_2 > R$, since otherwise $f_0(x) \leq f_0(y)$. Consider the point z given by

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

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 (4.19). Hence there exists no feasible y with $f_0(y) < f_0(x)$, *i.e.*, x is globally optimal.

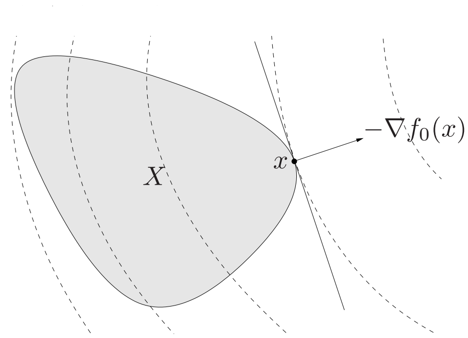


Figure 4.2 Geometric interpretation of the optimality condition (4.21). The feasible set X is shown shaded. Some level curves of f_0 are shown as dashed lines. The point x is optimal: $-\nabla f_0(x)$ defines a supporting hyperplane (shown as a solid line) to X at x.

It is not true that locally optimal points of quasiconvex optimization problems are globally optimal; see §4.2.5.

4.2.3 An optimality criterion for differentiable f_0

Suppose that the objective f_0 in a convex optimization problem is differentiable, so that for all $x, y \in \mathbf{dom}\, f_0$,

$$f_0(y) \geq f_0(x) + \nabla f_0(x)^T (y - x) \tag{4.20}$$

(see §3.1.3). Let X denote the feasible set, *i.e.*,

$$X = \{x \mid f_i(x) \leq 0,\ i = 1, \ldots, m,\ h_i(x) = 0,\ i = 1, \ldots, p\}.$$

Then x is optimal if and only if $x \in X$ and

$$\nabla f_0(x)^T (y - x) \geq 0 \text{ for all } y \in X. \tag{4.21}$$

This optimality criterion can be understood geometrically: If $\nabla f_0(x) \neq 0$, it means that $-\nabla f_0(x)$ defines a supporting hyperplane to the feasible set at x (see figure 4.2).

Proof of optimality condition

First suppose $x \in X$ and satisfies (4.21). Then if $y \in X$ we have, by (4.20), $f_0(y) \geq f_0(x)$. This shows x is an optimal point for (4.1).

Conversely, suppose x is optimal, but the condition (4.21) does not hold, *i.e.*, for some $y \in X$ we have

$$\nabla f_0(x)^T (y - x) < 0.$$

Consider the point $z(t) = ty + (1-t)x$, where $t \in [0,1]$ is a parameter. Since $z(t)$ is on the line segment between x and y, and the feasible set is convex, $z(t)$ is feasible. We claim that for small positive t we have $f_0(z(t)) < f_0(x)$, which will prove that x is not optimal. To show this, note that

$$\left.\frac{d}{dt} f_0(z(t))\right|_{t=0} = \nabla f_0(x)^T (y - x) < 0,$$

so for small positive t, we have $f_0(z(t)) < f_0(x)$.

We will pursue the topic of optimality conditions in much more depth in chapter 5, but here we examine a few simple examples.

Unconstrained problems

For an unconstrained problem (*i.e.*, $m = p = 0$), the condition (4.21) reduces to the well known necessary and sufficient condition

$$\nabla f_0(x) = 0 \tag{4.22}$$

for x to be optimal. While we have already seen this optimality condition, it is useful to see how it follows from (4.21). Suppose x is optimal, which means here that $x \in \mathbf{dom}\, f_0$, and for all feasible y we have $\nabla f_0(x)^T (y - x) \geq 0$. Since f_0 is differentiable, its domain is (by definition) open, so all y sufficiently close to x are feasible. Let us take $y = x - t\nabla f_0(x)$, where $t \in \mathbf{R}$ is a parameter. For t small and positive, y is feasible, and so

$$\nabla f_0(x)^T (y - x) = -t \|\nabla f_0(x)\|_2^2 \geq 0,$$

from which we conclude $\nabla f_0(x) = 0$.

There are several possible situations, depending on the number of solutions of (4.22). If there are no solutions of (4.22), then there are no optimal points; the optimal value of the problem is not attained. Here we can distinguish between two cases: the problem is unbounded below, or the optimal value is finite, but not attained. On the other hand we can have multiple solutions of the equation (4.22), in which case each such solution is a minimizer of f_0.

Example 4.5 *Unconstrained quadratic optimization.* Consider the problem of minimizing the quadratic function

$$f_0(x) = (1/2)x^T P x + q^T x + r,$$

where $P \in \mathbf{S}_+^n$ (which makes f_0 convex). The necessary and sufficient condition for x to be a minimizer of f_0 is

$$\nabla f_0(x) = Px + q = 0.$$

Several cases can occur, depending on whether this (linear) equation has no solutions, one solution, or many solutions.

- If $q \notin \mathcal{R}(P)$, then there is no solution. In this case f_0 is unbounded below.
- If $P \succ 0$ (which is the condition for f_0 to be strictly convex), then there is a unique minimizer, $x^\star = -P^{-1}q$.

- If P is singular, but $q \in \mathcal{R}(P)$, then the set of optimal points is the (affine) set $X_{\rm opt} = -P^\dagger q + \mathcal{N}(P)$, where $P^\dagger$ denotes the pseudo-inverse of P (see §A.5.4).

Example 4.6 *Analytic centering.* Consider the (unconstrained) problem of minimizing the (convex) function $f_0 : \mathbf{R}^n \to \mathbf{R}$, defined as

$$f_0(x) = -\sum_{i=1}^m \log(b_i - a_i^T x), \qquad \mathbf{dom}\, f_0 = \{x \mid Ax \prec b\},$$

where $a_1^T, \ldots, a_m^T$ are the rows of A. The function f_0 is differentiable, so the necessary and sufficient conditions for x to be optimal are

$$Ax \prec b, \qquad \nabla f_0(x) = \sum_{i=1}^m \frac{1}{b_i - a_i^T x} a_i = 0. \tag{4.23}$$

(The condition $Ax \prec b$ is just $x \in \mathbf{dom}\, f_0$.) If $Ax \prec b$ is infeasible, then the domain of f_0 is empty. Assuming $Ax \prec b$ is feasible, there are still several possible cases (see exercise 4.2):

- There are no solutions of (4.23), and hence no optimal points for the problem. This occurs if and only if f_0 is unbounded below.
- There are many solutions of (4.23). In this case it can be shown that the solutions form an affine set.
- There is a unique solution of (4.23), *i.e.*, a unique minimizer of f_0. This occurs if and only if the open polyhedron $\{x \mid Ax \prec b\}$ is nonempty and bounded.

Problems with equality constraints only

Consider the case where there are equality constraints but no inequality constraints, *i.e.*,

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

Here the feasible set is affine. We assume that it is nonempty; otherwise the problem is infeasible. The optimality condition for a feasible x is that

$$\nabla f_0(x)^T (y - x) \geq 0$$

must hold for all y satisfying $Ay = b$. Since x is feasible, every feasible y has the form $y = x + v$ for some $v \in \mathcal{N}(A)$. The optimality condition can therefore be expressed as:

$$\nabla f_0(x)^T v \geq 0 \text{ for all } v \in \mathcal{N}(A).$$

If a linear function is nonnegative on a subspace, then it must be zero on the subspace, so it follows that $\nabla f_0(x)^T v = 0$ for all $v \in \mathcal{N}(A)$. In other words,

$$\nabla f_0(x) \perp \mathcal{N}(A).$$

Using the fact that $\mathcal{N}(A)^\perp = \mathcal{R}(A^T)$, this optimality condition can be expressed as $\nabla f_0(x) \in \mathcal{R}(A^T)$, *i.e.*, there exists a $\nu \in \mathbf{R}^p$ such that

$$\nabla f_0(x) + A^T \nu = 0.$$

Together with the requirement $Ax = b$ (*i.e.*, that x is feasible), this is the classical Lagrange multiplier optimality condition, which we will study in greater detail in chapter 5.

Minimization over the nonnegative orthant

As another example we consider the problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & x \succeq 0, \end{array}$$

where the only inequality constraints are nonnegativity constraints on the variables.

The optimality condition (4.21) is then

$$x \succeq 0, \qquad \nabla f_0(x)^T (y - x) \geq 0 \text{ for all } y \succeq 0.$$

The term $\nabla f_0(x)^T y$, which is a linear function of y, is unbounded below on $y \succeq 0$, unless we have $\nabla f_0(x) \succeq 0$. The condition then reduces to $-\nabla f_0(x)^T x \geq 0$. But $x \succeq 0$ and $\nabla f_0(x) \succeq 0$, so we must have $\nabla f_0(x)^T x = 0$, *i.e.*,

$$\sum_{i=1}^n (\nabla f_0(x))_i x_i = 0.$$

Now each of the terms in this sum is the product of two nonnegative numbers, so we conclude that each term must be zero, *i.e.*, $(\nabla f_0(x))_i \, x_i = 0$ for $i = 1, \ldots, n$.

The optimality condition can therefore be expressed as

$$x \succeq 0, \qquad \nabla f_0(x) \succeq 0, \qquad x_i \left(\nabla f_0(x)\right)_i = 0, \quad i = 1, \ldots, n.$$

The last condition is called *complementarity*, since it means that the sparsity patterns (*i.e.*, the set of indices corresponding to nonzero components) of the vectors x and $\nabla f_0(x)$ are complementary (*i.e.*, have empty intersection). We will encounter complementarity conditions again in chapter 5.

4.2.4 Equivalent convex problems

It is useful to see which of the transformations described in §4.1.3 preserve convexity.

Eliminating equality constraints

For a convex problem the equality constraints must be linear, *i.e.*, of the form $Ax = b$. In this case they can be eliminated by finding a particular solution x_0 of

$Ax = b$, and a matrix F whose range is the nullspace of A, which results in the problem

$$\begin{array}{ll} \text{minimize} & f_0(Fz + x_0) \\ \text{subject to} & f_i(Fz + x_0) \leq 0, \quad i = 1, \ldots, m, \end{array}$$

with variable z. Since the composition of a convex function with an affine function is convex, eliminating equality constraints preserves convexity of a problem. Moreover, the process of eliminating equality constraints (and reconstructing the solution of the original problem from the solution of the transformed problem) involves standard linear algebra operations.

At least in principle, this means we can restrict our attention to convex optimization problems which have no equality constraints. In many cases, however, it is better to retain the equality constraints, since eliminating them can make the problem harder to understand and analyze, or ruin the efficiency of an algorithm that solves it. This is true, for example, when the variable x has very large dimension, and eliminating the equality constraints would destroy sparsity or some other useful structure of the problem.

Introducing equality constraints

We can introduce new variables and equality constraints into a convex optimization problem, provided the equality constraints are linear, and the resulting problem will also be convex. For example, if an objective or constraint function has the form $f_i(A_i x + b_i)$, where $A_i \in \mathbf{R}^{k_i \times n}$, we can introduce a new variable $y_i \in \mathbf{R}^{k_i}$, replace $f_i(A_i x + b_i)$ with $f_i(y_i)$, and add the linear equality constraint $y_i = A_i x + b_i$.

Slack variables

By introducing slack variables we have the new constraints $f_i(x) + s_i = 0$. Since equality constraint functions must be affine in a convex problem, we must have f_i affine. In other words: introducing slack variables for *linear inequalities* preserves convexity of a problem.

Epigraph problem form

The epigraph form of the convex optimization problem (4.15) is

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

The objective is linear (hence convex) and the new constraint function $f_0(x) - t$ is also convex in (x, t), so the epigraph form problem is convex as well.

It is sometimes said that a linear objective is *universal* for convex optimization, since any convex optimization problem is readily transformed to one with linear objective. The epigraph form of a convex problem has several practical uses. By assuming the objective of a convex optimization problem is linear, we can simplify theoretical analysis. It can also simplify algorithm development, since an algorithm that solves convex optimization problems with linear objective can, using the transformation above, solve any convex optimization problem (provided it can handle the constraint $f_0(x) - t \le 0$).

Minimizing over some variables

Minimizing a convex function over some variables preserves convexity. Therefore, if f_0 in (4.9) is jointly convex in x_1 and x_2, and f_i, $i = 1, \ldots, m_1$, and $\tilde{f}_i$, $i = 1, \ldots, m_2$, are convex, then the equivalent problem (4.10) is convex.

4.2.5 Quasiconvex optimization

Recall that a quasiconvex optimization problem has the standard form

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \le 0, \quad i = 1, \ldots, m \\ & Ax = b, \end{array} \tag{4.24}$$

where the inequality constraint functions $f_1, \ldots, f_m$ are convex, and the objective f_0 is quasiconvex (instead of convex, as in a convex optimization problem). (Quasiconvex constraint functions can be replaced with equivalent convex constraint functions, *i.e.*, constraint functions that are convex and have the same 0-sublevel set, as in §3.4.5.)

In this section we point out some basic differences between convex and quasiconvex optimization problems, and also show how solving a quasiconvex optimization problem can be reduced to solving a sequence of convex optimization problems.

Locally optimal solutions and optimality conditions

The most important difference between convex and quasiconvex optimization is that a quasiconvex optimization problem can have locally optimal solutions that are not (globally) optimal. This phenomenon can be seen even in the simple case of unconstrained minimization of a quasiconvex function on $\mathbf{R}$, such as the one shown in figure 4.3.

Nevertheless, a variation of the optimality condition (4.21) given in §4.2.3 does hold for quasiconvex optimization problems with differentiable objective function. Let X denote the feasible set for the quasiconvex optimization problem (4.24). It follows from the first-order condition for quasiconvexity (3.20) that x is optimal if

$$x \in X, \qquad \nabla f_0(x)^T (y - x) > 0 \text{ for all } y \in X \setminus \{x\}. \tag{4.25}$$

There are two important differences between this criterion and the analogous one (4.21) for convex optimization:

- The condition (4.25) is only *sufficient* for optimality; simple examples show that it need not hold for an optimal point. In contrast, the condition (4.21) is necessary and sufficient for x to solve the convex problem.
- The condition (4.25) requires the gradient of f_0 to be nonzero, whereas the condition (4.21) does not. Indeed, when $\nabla f_0(x) = 0$ in the convex case, the condition (4.21) is satisfied, and x is optimal.

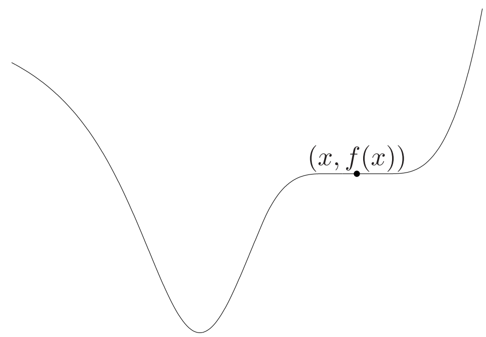


Figure 4.3 A quasiconvex function f on $\mathbf{R}$, with a locally optimal point x that is not globally optimal. This example shows that the simple optimality condition $f'(x) = 0$, valid for convex functions, does not hold for quasiconvex functions.

Quasiconvex optimization via convex feasibility problems

One general approach to quasiconvex optimization relies on the representation of the sublevel sets of a quasiconvex function via a family of convex inequalities, as described in §3.4.5. Let $\phi_t : \mathbf{R}^n \to \mathbf{R}$, $t \in \mathbf{R}$, be a family of convex functions that satisfy

$$f_0(x) \le t \iff \phi_t(x) \le 0,$$

and also, for each x, $\phi_t(x)$ is a nonincreasing function of t, *i.e.*, $\phi_s(x) \le \phi_t(x)$ whenever $s \ge t$.

Let $p^\star$ denote the optimal value of the quasiconvex optimization problem (4.24). If the feasibility problem

$$\begin{array}{ll} \text{find} & x \\ \text{subject to} & \phi_t(x) \le 0 \\ & f_i(x) \le 0, \quad i = 1, \dots, m \\ & Ax = b, \end{array} \tag{4.26}$$

is feasible, then we have $p^\star \le t$. Conversely, if the problem (4.26) is infeasible, then we can conclude $p^\star \ge t$. The problem (4.26) is a convex feasibility problem, since the inequality constraint functions are all convex, and the equality constraints are linear. Thus, we can check whether the optimal value $p^\star$ of a quasiconvex optimization problem is less than or more than a given value t by solving the convex feasibility problem (4.26). If the convex feasibility problem is feasible then we have $p^\star \le t$, and any feasible point x is feasible for the quasiconvex problem and satisfies $f_0(x) \le t$. If the convex feasibility problem is infeasible, then we know that $p^\star \ge t$.

This observation can be used as the basis of a simple algorithm for solving the quasiconvex optimization problem (4.24) using *bisection*, solving a convex feasibility problem at each step. We assume that the problem is feasible, and start with an interval $[l, u]$ known to contain the optimal value $p^\star$. We then solve the convex feasibility problem at its midpoint $t = (l + u)/2$, to determine whether the optimal value is in the lower or upper half of the interval, and update the interval accordingly. This produces a new interval, which also contains the optimal value, but has half the width of the initial interval. This is repeated until the width of the interval is small enough:

Algorithm 4.1 *Bisection method for quasiconvex optimization.*

given $l \leq p^\star$, $u \geq p^\star$, tolerance $\epsilon > 0$.

repeat
1. $t := (l + u)/2$.
2. Solve the convex feasibility problem (4.26).
3. **if** (4.26) is feasible, $u := t$; **else** $l := t$.

until $u - l \leq \epsilon$.

The interval $[l, u]$ is guaranteed to contain $p^\star$, *i.e.*, we have $l \leq p^\star \leq u$ at each step. In each iteration the interval is divided in two, *i.e.*, bisected, so the length of the interval after k iterations is $2^{-k}(u - l)$, where $u - l$ is the length of the initial interval. It follows that exactly $\lceil \log_2((u - l)/\epsilon) \rceil$ iterations are required before the algorithm terminates. Each step involves solving the convex feasibility problem (4.26).

4.3 Linear optimization problems

When the objective and constraint functions are all affine, the problem is called a *linear program* (LP). A general linear program has the form

$$\begin{array}{ll} \text{minimize} & c^T x + d \\ \text{subject to} & Gx \preceq h \\ & Ax = b, \end{array} \tag{4.27}$$

where $G \in \mathbf{R}^{m \times n}$ and $A \in \mathbf{R}^{p \times n}$. Linear programs are, of course, convex optimization problems.

It is common to omit the constant d in the objective function, since it does not affect the optimal (or feasible) set. Since we can maximize an affine objective $c^T x + d$, by minimizing $-c^T x - d$ (which is still convex), we also refer to a maximization problem with affine objective and constraint functions as an LP.

The geometric interpretation of an LP is illustrated in figure 4.4. The feasible set of the LP (4.27) is a polyhedron $\mathcal{P}$; the problem is to minimize the affine function $c^T x + d$ (or, equivalently, the linear function $c^T x$) over $\mathcal{P}$.

Standard and inequality form linear programs

Two special cases of the LP (4.27) are so widely encountered that they have been given separate names. In a *standard form LP* the only inequalities are componen-

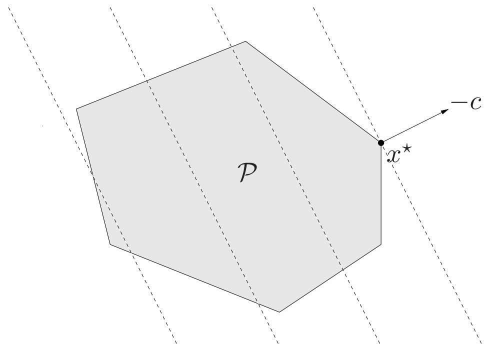


Figure 4.4 Geometric interpretation of an LP. The feasible set $\mathcal{P}$, which is a polyhedron, is shaded. The objective $c^T x$ is linear, so its level curves are hyperplanes orthogonal to c (shown as dashed lines). The point $x^\star$ is optimal; it is the point in $\mathcal{P}$ as far as possible in the direction $-c$.

twise nonnegativity constraints $x \succeq 0$:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \succeq 0. \end{array} \tag{4.28}$$

If the LP has no equality constraints, it is called an *inequality form LP*, usually written as

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b. \end{array} \tag{4.29}$$

Converting LPs to standard form

It is sometimes useful to transform a general LP (4.27) to one in standard form (4.28) (for example in order to use an algorithm for standard form LPs). The first step is to introduce slack variables s_i for the inequalities, which results in

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

The second step is to express the variable x as the difference of two nonnegative variables x^+ and x^-, *i.e.*, $x = x^+ - x^-$, $x^+, x^- \succeq 0$. This yields the problem

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

which is an LP in standard form, with variables x^+, x^-, and s. (For equivalence of this problem and the original one (4.27), see exercise 4.10.)

These techniques for manipulating problems (along with many others we will see in the examples and exercises) can be used to formulate many problems as linear programs. With some abuse of terminology, it is common to refer to a problem that can be formulated as an LP as an LP, even if it does not have the form (4.27).

4.3.1 Examples

LPs arise in a vast number of fields and applications; here we give a few typical examples.

Diet problem

A healthy diet contains m different nutrients in quantities at least equal to $b_1, \ldots, b_m$. We can compose such a diet by choosing nonnegative quantities $x_1, \ldots, x_n$ of n different foods. One unit quantity of food j contains an amount a_{ij} of nutrient i, and has a cost of c_j. We want to determine the cheapest diet that satisfies the nutritional requirements. This problem can be formulated as the LP

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

Several variations on this problem can also be formulated as LPs. For example, we can insist on an exact amount of a nutrient in the diet (which gives a linear equality constraint), or we can impose an upper bound on the amount of a nutrient, in addition to the lower bound as above.

Chebyshev center of a polyhedron

We consider the problem of finding the largest Euclidean ball that lies in a polyhedron described by linear inequalities,

$$\mathcal{P} = \{x \in \mathbf{R}^n \mid a_i^T x \le b_i,\ i = 1, \ldots, m\}.$$

(The center of the optimal ball is called the *Chebyshev center* of the polyhedron; it is the point deepest inside the polyhedron, *i.e.*, farthest from the boundary; see §8.5.1.) We represent the ball as

$$\mathcal{B} = \{x_c + u \mid \|u\|_2 \le r\}.$$

The variables in the problem are the center $x_c \in \mathbf{R}^n$ and the radius r; we wish to maximize r subject to the constraint $\mathcal{B} \subseteq \mathcal{P}$.

We start by considering the simpler constraint that $\mathcal{B}$ lies in one halfspace $a_i^T x \le b_i$, *i.e.*,

$$\|u\|_2 \le r \implies a_i^T (x_c + u) \le b_i. \tag{4.30}$$

Since

$$\sup\{a_i^T u \mid \|u\|_2 \le r\} = r\|a_i\|_2$$

we can write (4.30) as

$$a_i^T x_c + r\|a_i\|_2 \le b_i, \tag{4.31}$$

which is a linear inequality in x_c and r. In other words, the constraint that the ball lies in the halfspace determined by the inequality $a_i^T x \le b_i$ can be written as a linear inequality.

Therefore $\mathcal{B} \subseteq \mathcal{P}$ if and only if (4.31) holds for all $i = 1, \ldots, m$. Hence the Chebyshev center can be determined by solving the LP

$$\begin{array}{ll} \text{maximize} & r \\ \text{subject to} & a_i^T x_c + r\|a_i\|_2 \le b_i, \quad i = 1, \ldots, m, \end{array}$$

with variables r and x_c. (For more on the Chebyshev center, see §8.5.1.)

Dynamic activity planning

We consider the problem of choosing, or planning, the activity levels of n activities, or sectors of an economy, over N time periods. We let $x_j(t) \ge 0$, $t = 1, \ldots, N$, denote the activity level of sector j, in period t. The activities both consume and produce products or goods in proportion to their activity levels. The amount of good i produced per unit of activity j is given by a_{ij}. Similarly, the amount of good i consumed per unit of activity j is b_{ij}. The total amount of goods produced in period t is given by $Ax(t) \in \mathbf{R}^m$, and the amount of goods consumed is $Bx(t) \in \mathbf{R}^m$. (Although we refer to these products as 'goods', they can also include unwanted products such as pollutants.)

The goods consumed in a period cannot exceed those produced in the previous period: we must have $Bx(t+1) \preceq Ax(t)$ for $t = 1, \ldots, N$. A vector $g_0 \in \mathbf{R}^m$ of initial goods is given, which constrains the first period activity levels: $Bx(1) \preceq g_0$. The (vectors of) excess goods not consumed by the activities are given by

$$\begin{array}{rcl} s(0) & = & g_0 - Bx(1) \\ s(t) & = & Ax(t) - Bx(t+1), \quad t = 1, \ldots, N-1 \\ s(N) & = & Ax(N). \end{array}$$

The objective is to maximize a discounted total value of excess goods:

$$c^T s(0) + \gamma c^T s(1) + \cdots + \gamma^N c^T s(N),$$

where $c \in \mathbf{R}^m$ gives the values of the goods, and $\gamma > 0$ is a discount factor. (The value c_i is negative if the ith product is unwanted, *e.g.*, a pollutant; $|c_i|$ is then the cost of disposal per unit.)

Putting it all together we arrive at the LP

$$\begin{array}{ll} \text{maximize} & c^T s(0) + \gamma c^T s(1) + \cdots + \gamma^N c^T s(N) \\ \text{subject to} & x(t) \succeq 0, \quad t = 1, \ldots, N \\ & s(t) \succeq 0, \quad t = 0, \ldots, N \\ & s(0) = g_0 - Bx(1) \\ & s(t) = Ax(t) - Bx(t+1), \quad t = 1, \ldots, N-1 \\ & s(N) = Ax(N), \end{array}$$

with variables $x(1), \ldots, x(N)$, $s(0), \ldots, s(N)$. This problem is a standard form LP; the variables $s(t)$ are the slack variables associated with the constraints $Bx(t+1) \preceq Ax(t)$.

Chebyshev inequalities

We consider a probability distribution for a discrete random variable x on a set $\{u_1, \ldots, u_n\} \subseteq \mathbf{R}$ with n elements. We describe the distribution of x by a vector $p \in \mathbf{R}^n$, where

$$p_i = \mathbf{prob}(x = u_i),$$

so p satisfies $p \succeq 0$ and $\mathbf{1}^T p = 1$. Conversely, if p satisfies $p \succeq 0$ and $\mathbf{1}^T p = 1$, then it defines a probability distribution for x. We assume that u_i are known and fixed, but the distribution p is not known.

If f is any function of x, then

$$\mathbf{E}\, f = \sum_{i=1}^{n} p_i f(u_i)$$

is a linear function of p. If S is any subset of $\mathbf{R}$, then

$$\mathbf{prob}(x \in S) = \sum_{u_i \in S} p_i$$

is a linear function of p.

Although we do not know p, we are given prior knowledge of the following form: We know upper and lower bounds on expected values of some functions of x, and probabilities of some subsets of $\mathbf{R}$. This prior knowledge can be expressed as linear inequality constraints on p,

$$\alpha_i \leq a_i^T p \leq \beta_i, \quad i = 1, \ldots, m.$$

The problem is to give lower and upper bounds on $\mathbf{E}\, f_0(x) = a_0^T p$, where f_0 is some function of x.

To find a lower bound we solve the LP

$$\begin{array}{ll} \text{minimize} & a_0^T p \\ \text{subject to} & p \succeq 0, \quad \mathbf{1}^T p = 1 \\ & \alpha_i \leq a_i^T p \leq \beta_i, \quad i = 1, \ldots, m, \end{array}$$

with variable p. The optimal value of this LP gives the lowest possible value of $\mathbf{E}\, f_0(X)$ for any distribution that is consistent with the prior information. Moreover, the bound is sharp: the optimal solution gives a distribution that is consistent with the prior information and achieves the lower bound. In a similar way, we can find the best upper bound by maximizing $a_0^T p$ subject to the same constraints. (We will consider Chebyshev inequalities in more detail in §7.4.1.)

Piecewise-linear minimization

Consider the (unconstrained) problem of minimizing the piecewise-linear, convex function

$$f(x) = \max_{i=1,\ldots,m} (a_i^T x + b_i).$$

This problem can be transformed to an equivalent LP by first forming the epigraph problem,

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \max_{i=1,\ldots,m} (a_i^T x + b_i) \leq t, \end{array}$$

and then expressing the inequality as a set of m separate inequalities:

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & a_i^T x + b_i \leq t, \quad i = 1, \ldots, m. \end{array}$$

This is an LP (in inequality form), with variables x and t.

4.3.2 Linear-fractional programming

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

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array} \tag{4.32}$$

where the objective function is given by

$$f_0(x) = \frac{c^T x + d}{e^T x + f}, \qquad \mathbf{dom}\, f_0 = \{x \mid e^T x + f > 0\}.$$

The objective function is quasiconvex (in fact, quasilinear) so linear-fractional programs are quasiconvex optimization problems.

Transforming to a linear program

If the feasible set

$$\{x \mid Gx \preceq h,\ Ax = b,\ e^T x + f > 0\}$$

is nonempty, the linear-fractional program (4.32) can be transformed to an equivalent linear program

$$\begin{array}{ll} \text{minimize} & c^T y + dz \\ \text{subject to} & Gy - hz \preceq 0 \\ & Ay - bz = 0 \\ & e^T y + fz = 1 \\ & z \geq 0 \end{array} \tag{4.33}$$

with variables y, z.

To show the equivalence, we first note that if x is feasible in (4.32) then the pair

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

is feasible in (4.33), with the same objective value $c^T y + dz = f_0(x)$. It follows that the optimal value of (4.32) is greater than or equal to the optimal value of (4.33).

Conversely, if (y, z) is feasible in (4.33), with $z \neq 0$, then $x = y/z$ is feasible in (4.32), with the same objective value $f_0(x) = c^T y + dz$. If (y, z) is feasible in (4.33) with $z = 0$, and x_0 is feasible for (4.32), then $x = x_0 + ty$ is feasible in (4.32) for all $t \geq 0$. Moreover, $\lim_{t\to\infty} f_0(x_0 + ty) = c^T y + dz$, so we can find feasible points in (4.32) with objective values arbitrarily close to the objective value of (y, z). We conclude that the optimal value of (4.32) is less than or equal to the optimal value of (4.33).

Generalized linear-fractional programming

A generalization of the linear-fractional program (4.32) is the *generalized linear-fractional program* in which

$$f_0(x) = \max_{i=1,\dots,r} \frac{c_i^T x + d_i}{e_i^T x + f_i}, \qquad \mathbf{dom}\, f_0 = \{x \mid e_i^T x + f_i > 0,\ i = 1,\dots,r\}.$$

The objective function is the pointwise maximum of r quasiconvex functions, and therefore quasiconvex, so this problem is quasiconvex. When $r = 1$ it reduces to the standard linear-fractional program.

Example 4.7 *Von Neumann growth problem.* We consider an economy with n sectors, and activity levels $x_i > 0$ in the current period, and activity levels $x_i^+ > 0$ in the next period. (In this problem we only consider one period.) There are m goods which are consumed, and also produced, by the activity: An activity level x consumes goods $Bx \in \mathbf{R}^m$, and produces goods Ax. The goods consumed in the next period cannot exceed the goods produced in the current period, *i.e.*, $Bx^+ \preceq Ax$. The *growth rate* in sector i, over the period, is given by x_i^+/x_i.

Von Neumann's growth problem is to find an activity level vector x that maximizes the minimum growth rate across all sectors of the economy. This problem can be expressed as a generalized linear-fractional problem

$$\begin{array}{ll} \text{maximize} & \min_{i=1,\dots,n} x_i^+/x_i \\ \text{subject to} & x^+ \succeq 0 \\ & Bx^+ \preceq Ax \end{array}$$

with domain $\{(x, x^+) \mid x \succ 0\}$. Note that this problem is homogeneous in x and x^+, so we can replace the implicit constraint $x \succ 0$ by the explicit constraint $x \succeq \mathbf{1}$.

4.4 Quadratic optimization problems

The convex optimization problem (4.15) 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

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & Gx \preceq h \\ & Ax = b, \end{array} \tag{4.34}$$

where $P \in \mathbf{S}_+^n$, $G \in \mathbf{R}^{m\times n}$, and $A \in \mathbf{R}^{p\times n}$. In a quadratic program, we minimize a convex quadratic function over a polyhedron, as illustrated in figure 4.5.

If the objective in (4.15) as well as the inequality constraint functions are (convex) quadratic, as in

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} & (1/2)x^T P_i x + q_i^T x + r_i \le 0, \quad i = 1,\dots,m \\ & Ax = b, \end{array} \tag{4.35}$$

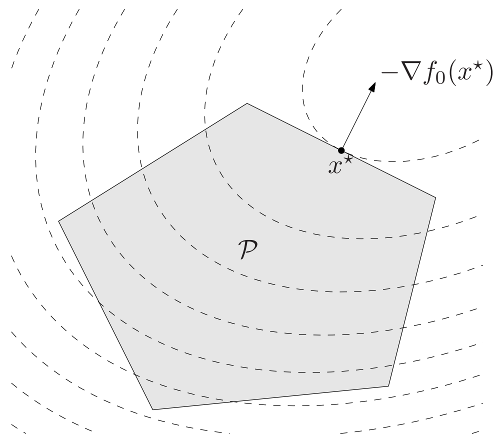


Figure 4.5 Geometric illustration of QP. The feasible set $\mathcal{P}$, which is a polyhedron, is shown shaded. The contour lines of the objective function, which is convex quadratic, are shown as dashed curves. The point $x^\star$ is optimal.

where $P_i \in \mathbf{S}_+^n$, $i = 0, 1 \ldots, m$, the problem is called a *quadratically constrained quadratic program* (QCQP). In a QCQP, we minimize a convex quadratic function over a feasible region that is the intersection of ellipsoids (when $P_i \succ 0$).

Quadratic programs include linear programs as a special case, by taking $P = 0$ in (4.34). Quadratically constrained quadratic programs include quadratic programs (and therefore also linear programs) as a special case, by taking $P_i = 0$ in (4.35), for $i = 1, \ldots, m$.

4.4.1 Examples

Least-squares and regression

The problem of minimizing the convex quadratic function

$$\|Ax - b\|_2^2 = x^T A^T A x - 2b^T A x + b^T b$$

is an (unconstrained) QP. It arises in many fields and has many names, *e.g.*, *regression analysis* or *least-squares approximation*. This problem is simple enough to have the well known analytical solution $x = A^\dagger b$, where $A^\dagger$ is the pseudo-inverse of A (see §A.5.4).

When linear inequality constraints are added, the problem is called *constrained regression* or *constrained least-squares*, and there is no longer a simple analytical solution. As an example we can consider regression with lower and upper bounds on the variables, *i.e.*,

$$\begin{array}{ll}\text{minimize} & \|Ax - b\|_2^2 \\ \text{subject to} & l_i \le x_i \le u_i, \quad i = 1, \ldots, n,\end{array}$$

which is a QP. (We will study least-squares and regression problems in far more depth in chapters 6 and 7.)

Distance between polyhedra

The (Euclidean) distance between the polyhedra $\mathcal{P}_1 = \{x \mid A_1 x \preceq b_1\}$ and $\mathcal{P}_2 = \{x \mid A_2 x \preceq b_2\}$ in $\mathbf{R}^n$ is defined as

$$\mathbf{dist}(\mathcal{P}_1, \mathcal{P}_2) = \inf\{\|x_1 - x_2\|_2 \mid x_1 \in \mathcal{P}_1,\ x_2 \in \mathcal{P}_2\}.$$

If the polyhedra intersect, the distance is zero.

To find the distance between $\mathcal{P}_1$ and $\mathcal{P}_2$, we can solve the QP

$$\begin{array}{ll} \text{minimize} & \|x_1 - x_2\|_2^2 \\ \text{subject to} & A_1 x_1 \preceq b_1, \quad A_2 x_2 \preceq b_2, \end{array}$$

with variables $x_1,\ x_2 \in \mathbf{R}^n$. This problem is infeasible if and only if one of the polyhedra is empty. The optimal value is zero if and only if the polyhedra intersect, in which case the optimal x_1 and x_2 are equal (and is a point in the intersection $\mathcal{P}_1 \cap \mathcal{P}_2$). Otherwise the optimal x_1 and x_2 are the points in $\mathcal{P}_1$ and $\mathcal{P}_2$, respectively, that are closest to each other. (We will study geometric problems involving distance in more detail in chapter 8.)

Bounding variance

We consider again the Chebyshev inequalities example (page 150), where the variable is an unknown probability distribution given by $p \in \mathbf{R}^n$, about which we have some prior information. The variance of a random variable $f(x)$ is given by

$$\mathbf{E}\, f^2 - (\mathbf{E}\, f)^2 = \sum_{i=1}^n f_i^2 p_i - \left(\sum_{i=1}^n f_i p_i \right)^2,$$

(where $f_i = f(u_i)$), which is a concave quadratic function of p.

It follows that we can maximize the variance of $f(x)$, subject to the given prior information, by solving the QP

$$\begin{array}{ll} \text{maximize} & \sum_{i=1}^n f_i^2 p_i - \left(\sum_{i=1}^n f_i p_i\right)^2 \\ \text{subject to} & p \succeq 0, \quad \mathbf{1}^T p = 1 \\ & \alpha_i \leq a_i^T p \leq \beta_i, \quad i = 1, \ldots, m. \end{array}$$

The optimal value gives the maximum possible variance of $f(x)$, over all distributions that are consistent with the prior information; the optimal p gives a distribution that achieves this maximum variance.

Linear program with random cost

We consider an LP,

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

with variable $x \in \mathbf{R}^n$. We suppose that the cost function (vector) $c \in \mathbf{R}^n$ is *random*, with mean value $\bar{c}$ and covariance $\mathbf{E}(c - \bar{c})(c - \bar{c})^T = \Sigma$. (We assume for simplicity that the other problem parameters are deterministic.) For a given $x \in \mathbf{R}^n$, the cost $c^T x$ is a (scalar) random variable with mean $\mathbf{E}\, c^T x = \bar{c}^T x$ and variance

$$\mathbf{var}(c^T x) = \mathbf{E}(c^T x - \mathbf{E}\, c^T x)^2 = x^T \Sigma x.$$

In general there is a trade-off between small expected cost and small cost variance. One way to take variance into account is to minimize a linear combination of the expected value and the variance of the cost, *i.e.*,

$$\mathbf{E}\, c^T x + \gamma\, \mathbf{var}(c^T x),$$

which is called the *risk-sensitive cost*. The parameter $\gamma \geq 0$ is called the *risk-aversion parameter*, since it sets the relative values of cost variance and expected value. (For $\gamma > 0$, we are willing to trade off an increase in expected cost for a sufficiently large decrease in cost variance).

To minimize the risk-sensitive cost we solve the QP

$$\begin{array}{ll} \text{minimize} & \bar{c}^T x + \gamma x^T \Sigma x \\ \text{subject to} & Gx \preceq h \\ & Ax = b. \end{array}$$

Markowitz portfolio optimization

We consider a classical portfolio problem with n assets or stocks held over a period of time. We let x_i denote the amount of asset i held throughout the period, with x_i in dollars, at the price at the beginning of the period. A normal long position in asset i corresponds to $x_i > 0$; a short position in asset i (*i.e.*, the obligation to buy the asset at the end of the period) corresponds to $x_i < 0$. We let p_i denote the relative price change of asset i over the period, *i.e.*, its change in price over the period divided by its price at the beginning of the period. The overall return on the portfolio is $r = p^T x$ (given in dollars). The optimization variable is the portfolio vector $x \in \mathbf{R}^n$.

A wide variety of constraints on the portfolio can be considered. The simplest set of constraints is that $x_i \geq 0$ (*i.e.*, no short positions) and $\mathbf{1}^T x = B$ (*i.e.*, the total budget to be invested is B, which is often taken to be one).

We take a stochastic model for price changes: $p \in \mathbf{R}^n$ is a random vector, with known mean $\bar{p}$ and covariance Σ. Therefore with portfolio $x \in \mathbf{R}^n$, the return r is a (scalar) random variable with mean $\bar{p}^T x$ and variance $x^T \Sigma x$. The choice of portfolio x involves a trade-off between the mean of the return, and its variance.

The classical portfolio optimization problem, introduced by Markowitz, is the QP

$$\begin{array}{ll} \text{minimize} & x^T \Sigma x \\ \text{subject to} & \bar{p}^T x \geq r_{\min} \\ & \mathbf{1}^T x = 1, \quad x \succeq 0, \end{array}$$

where x, the portfolio, is the variable. Here we find the portfolio that minimizes the return variance (which is associated with the *risk* of the portfolio) subject to achieving a minimum acceptable mean return $r_{\min}$, and satisfying the portfolio budget and no-shorting constraints.

Many extensions are possible. One standard extension, for example, is to allow short positions, *i.e.*, $x_i < 0$. To do this we introduce variables x_{long} and x_{short}, with

$$x_{\text{long}} \succeq 0, \qquad x_{\text{short}} \succeq 0, \qquad x = x_{\text{long}} - x_{\text{short}}, \qquad \mathbf{1}^T x_{\text{short}} \leq \eta \mathbf{1}^T x_{\text{long}}.$$

The last constraint limits the total short position at the beginning of the period to some fraction η of the total long position at the beginning of the period.

As another extension we can include linear transaction costs in the portfolio optimization problem. Starting from a given initial portfolio x_{init} we buy and sell assets to achieve the portfolio x, which we then hold over the period as described above. We are charged a transaction fee for buying and selling assets, which is proportional to the amount bought or sold. To handle this, we introduce variables u_{buy} and u_{sell}, which determine the amount of each asset we buy and sell before the holding period. We have the constraints

$$x = x_{\text{init}} + u_{\text{buy}} - u_{\text{sell}}, \qquad u_{\text{buy}} \succeq 0, \qquad u_{\text{sell}} \succeq 0.$$

We replace the simple budget constraint $\mathbf{1}^T x = 1$ with the condition that the initial buying and selling, including transaction fees, involves zero net cash:

$$(1 - f_{\text{sell}}) \mathbf{1}^T u_{\text{sell}} = (1 + f_{\text{buy}}) \mathbf{1}^T u_{\text{buy}}$$

Here the lefthand side is the total proceeds from selling assets, less the selling transaction fee, and the righthand side is the total cost, including transaction fee, of buying assets. The constants $f_{\text{buy}} \geq 0$ and $f_{\text{sell}} \geq 0$ are the transaction fee rates for buying and selling (assumed the same across assets, for simplicity).

The problem of minimizing return variance, subject to a minimum mean return, and the budget and trading constraints, is a QP with variables x, u_{buy}, u_{sell}.

4.4.2 Second-order cone programming

A problem that is closely related to quadratic programming is the *second-order cone program* (SOCP):

$$\begin{array}{ll} \text{minimize} & f^T x \\ \text{subject to} & \|A_i x + b_i\|_2 \leq c_i^T x + d_i, \quad i = 1, \ldots, m \\ & Fx = g, \end{array} \tag{4.36}$$

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 $A \in \mathbf{R}^{k \times n}$, 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 $\mathbf{R}^{k+1}$.

When $c_i = 0$, $i = 1, \ldots, m$, the SOCP (4.36) is equivalent to a QCQP (which is obtained by squaring each of the constraints). Similarly, if $A_i = 0$, $i = 1, \ldots, m$, then the SOCP (4.36) reduces to a (general) LP. Second-order cone programs are, however, more general than QCQPs (and of course, LPs).

Robust linear programming

We consider a linear program in inequality form,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m, \end{array}$$

in which there is some uncertainty or variation in the parameters c, a_i, b_i. To simplify the exposition we assume that c and b_i are fixed, and that a_i are known to lie in given ellipsoids:

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

where $P_i \in \mathbf{R}^{n \times n}$. (If P_i is singular we obtain 'flat' ellipsoids, of dimension $\mathbf{rank}\, P_i$; $P_i = 0$ 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*

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i \text{ for all } a_i \in \mathcal{E}_i, \quad i = 1, \ldots, m. \end{array} \tag{4.37}$$

The robust linear constraint, $a_i^T x \leq b_i$ for all $a_i \in \mathcal{E}_i$, can be expressed as

$$\sup\{a_i^T x \mid a_i \in \mathcal{E}_i\} \leq b_i,$$

the lefthand side of which can be expressed as

$$\begin{array}{rcl} \sup\{a_i^T x \mid a_i \in \mathcal{E}_i\} & = & \bar{a}_i^T x + \sup\{u^T P_i^T x \mid \|u\|_2 \leq 1\} \\ & = & \bar{a}_i^T x + \|P_i^T x\|_2. \end{array}$$

Thus, the robust linear constraint can be expressed as

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

which is evidently a second-order cone constraint. Hence the robust LP (4.37) can be expressed as the SOCP

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

Note that the additional norm terms act as *regularization terms*; they prevent x from being large in directions with considerable uncertainty in the parameters a_i.

Linear programming with random constraints

The robust LP described above can also be considered in a statistical framework. Here we suppose that the parameters a_i are independent Gaussian random vectors, with mean $\bar{a}_i$ and covariance Σ_i. We require that each constraint $a_i^T x \leq b_i$ should hold with a probability (or confidence) exceeding η, where $\eta \geq 0.5$, *i.e.*,

$$\mathbf{prob}(a_i^T x \leq b_i) \geq \eta. \tag{4.38}$$

We will show that this probability constraint can be expressed as a second-order cone constraint.

Letting $u = a_i^T x$, with σ^2 denoting its variance, this constraint can be written as

$$\mathbf{prob}\left(\frac{u - \overline{u}}{\sigma} \leq \frac{b_i - \overline{u}}{\sigma}\right) \geq \eta.$$

Since $(u - \overline{u})/\sigma$ is a zero mean unit variance Gaussian variable, the probability above is simply $\Phi((b_i - \overline{u})/\sigma)$, where

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-t^2/2}\, dt$$

is the cumulative distribution function of a zero mean unit variance Gaussian random variable. Thus the probability constraint (4.38) can be expressed as

$$\frac{b_i - \overline{u}}{\sigma} \geq \Phi^{-1}(\eta),$$

or, equivalently,

$$\overline{u} + \Phi^{-1}(\eta)\sigma \leq b_i.$$

From $\overline{u} = \overline{a}_i^T x$ and $\sigma = (x^T \Sigma_i x)^{1/2}$ we obtain

$$\overline{a}_i^T x + \Phi^{-1}(\eta) \|\Sigma_i^{1/2} x\|_2 \leq b_i.$$

By our assumption that $\eta \geq 1/2$, we have $\Phi^{-1}(\eta) \geq 0$, so this constraint is a second-order cone constraint.

In summary, the problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{prob}(a_i^T x \leq b_i) \geq \eta, \quad i = 1, \ldots, m \end{array}$$

can be expressed as the SOCP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \overline{a}_i^T x + \Phi^{-1}(\eta) \|\Sigma_i^{1/2} x\|_2 \leq b_i, \quad i = 1, \ldots, m. \end{array}$$

(We will consider robust convex optimization problems in more depth in chapter 6. See also exercises 4.13, 4.28, and 4.59.)

Example 4.8 *Portfolio optimization with loss risk constraints.* We consider again the classical Markowitz portfolio problem described above (page 155). We assume here that the price change vector $p \in \mathbf{R}^n$ is a Gaussian random variable, with mean $\overline{p}$ and covariance Σ. Therefore the return r is a Gaussian random variable with mean $\overline{r} = \overline{p}^T x$ and variance $\sigma_r^2 = x^T \Sigma x$.

Consider a *loss risk constraint* of the form

$$\mathbf{prob}(r \leq \alpha) \leq \beta, \tag{4.39}$$

where α is a given unwanted return level (*e.g.*, a large loss) and β is a given maximum probability.

As in the stochastic interpretation of the robust LP given above, we can express this constraint using the cumulative distribution function Φ of a unit Gaussian random variable. The inequality (4.39) is equivalent to

$$\overline{p}^T x + \Phi^{-1}(\beta)\,\|\Sigma^{1/2}x\|_2 \geq \alpha.$$

Provided $\beta \leq 1/2$ (*i.e.*, $\Phi^{-1}(\beta) \leq 0$), this loss risk constraint is a second-order cone constraint. (If $\beta > 1/2$, the loss risk constraint becomes nonconvex in x.)

The problem of maximizing the expected return subject to a bound on the loss risk (with $\beta \leq 1/2$), can therefore be cast as an SOCP with one second-order cone constraint:

$$\begin{array}{ll} \text{maximize} & \overline{p}^T x \\ \text{subject to} & \overline{p}^T x + \Phi^{-1}(\beta)\,\|\Sigma^{1/2}x\|_2 \geq \alpha \\ & x \succeq 0, \quad \mathbf{1}^T x = 1. \end{array}$$

There are many extensions of this problem. For example, we can impose several loss risk constraints, *i.e.*,

$$\mathbf{prob}(r \leq \alpha_i) \leq \beta_i, \quad i = 1, \ldots, k,$$

(where $\beta_i \leq 1/2$), which expresses the risks (β_i) we are willing to accept for various levels of loss (α_i).

Minimal surface

Consider a differentiable function $f : \mathbf{R}^2 \to \mathbf{R}$ with $\mathbf{dom}\, f = C$. The surface area of its graph is given by

$$A = \int_C \sqrt{1 + \|\nabla f(x)\|_2^2}\, dx = \int_C \|(\nabla f(x), 1)\|_2\, dx,$$

which is a convex functional of f. The *minimal surface problem* is to find the function f that minimizes A subject to some constraints, for example, some given values of f on the boundary of C.

We will approximate this problem by discretizing the function f. Let $C = [0,1] \times [0,1]$, and let f_{ij} denote the value of f at the point $(i/K, j/K)$, for $i,\ j = 0, \ldots, K$. An approximate expression for the gradient of f at the point $x = (i/K, j/K)$ can be found using forward differences:

$$\nabla f(x) \approx K \left[\begin{array}{c} f_{i+1,j} - f_{i,j} \\ f_{i,j+1} - f_{i,j} \end{array} \right].$$

Substituting this into the expression for the area of the graph, and approximating the integral as a sum, we obtain an approximation for the area of the graph:

$$A \approx A_{\text{disc}} = \frac{1}{K^2} \sum_{i,j=0}^{K-1} \left\| \left[\begin{array}{c} K(f_{i+1,j} - f_{i,j}) \\ K(f_{i,j+1} - f_{i,j}) \\ 1 \end{array} \right] \right\|_2$$

The discretized area approximation A_{disc} is a convex function of f_{ij}.

We can consider a wide variety of constraints on f_{ij}, such as equality or inequality constraints on any of its entries (for example, on the boundary values), or on its moments. As an example, we consider the problem of finding the minimal area surface with fixed boundary values on the left and right edges of the square:

$$\begin{array}{ll} \text{minimize} & A_{\text{disc}} \\ \text{subject to} & f_{0j} = l_j, \quad j = 0, \ldots, K \\ & f_{Kj} = r_j, \quad j = 0, \ldots, K \end{array} \tag{4.40}$$

where f_{ij}, $i, j = 0, \ldots, K$, are the variables, and l_j, r_j are the given boundary values on the left and right sides of the square.

We can transform the problem (4.40) into an SOCP by introducing new variables t_{ij}, i, $j = 0, \ldots, K-1$:

$$\begin{array}{ll} \text{minimize} & (1/K^2)\sum_{i,j=0}^{K-1} t_{ij} \\ \text{subject to} & \left\| \begin{bmatrix} K(f_{i+1,j} - f_{i,j}) \\ K(f_{i,j+1} - f_{i,j}) \\ 1 \end{bmatrix} \right\|_2 \leq t_{ij}, \quad i, \ j = 0, \ldots, K-1 \\ & f_{0j} = l_j, \quad j = 0, \ldots, K \\ & f_{Kj} = r_j, \quad j = 0, \ldots, K. \end{array}$$

4.5 Geometric programming

In this section we describe a family of optimization problems that are *not* convex in their natural form. These problems can, however, be transformed to convex optimization problems, by a change of variables and a transformation of the objective and constraint functions.

4.5.1 Monomials and posynomials

A function $f : \mathbf{R}^n \to \mathbf{R}$ with $\mathbf{dom}\, f = \mathbf{R}^n_{++}$, defined as

$$f(x) = c x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}, \tag{4.41}$$

where $c > 0$ and $a_i \in \mathbf{R}$, is called a *monomial function*, or simply, a *monomial*. The exponents a_i of a monomial can be any real numbers, including fractional or negative, but the coefficient c can only be positive. (The term 'monomial' conflicts with the standard definition from algebra, in which the exponents must be nonnegative integers, but this should not cause any confusion.) A sum of monomials, *i.e.*, a function of the form

$$f(x) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}}, \tag{4.42}$$

where $c_k > 0$, is called a *posynomial function* (with K terms), or simply, a *posynomial*.

Posynomials are closed under addition, multiplication, and nonnegative scaling. Monomials are closed under multiplication and division. If a posynomial is multiplied by a monomial, the result is a posynomial; similarly, a posynomial can be divided by a monomial, with the result a posynomial.

4.5.2 Geometric programming

An optimization problem of the form

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \le 1, \quad i = 1, \ldots, m \\ & h_i(x) = 1, \quad i = 1, \ldots, p \end{array} \tag{4.43}$$

where $f_0, \ldots, f_m$ are posynomials and $h_1, \ldots, h_p$ are monomials, is called a *geometric program* (GP). The domain of this problem is $\mathcal{D} = \mathbf{R}^n_{++}$; the constraint $x \succ 0$ is implicit.

Extensions of geometric programming

Several extensions are readily handled. If f is a posynomial and h is a monomial, then the constraint $f(x) \le h(x)$ can be handled by expressing it as $f(x)/h(x) \le 1$ (since f/h is posynomial). This includes as a special case a constraint of the form $f(x) \le a$, where f is posynomial and $a > 0$. In a similar way if h_1 and h_2 are both nonzero monomial functions, then we can handle the equality constraint $h_1(x) = h_2(x)$ by expressing it as $h_1(x)/h_2(x) = 1$ (since h_1/h_2 is monomial). We can maximize a nonzero monomial objective function, by minimizing its inverse (which is also a monomial).

For example, consider the problem

$$\begin{array}{ll} \text{maximize} & x/y \\ \text{subject to} & 2 \le x \le 3 \\ & x^2 + 3y/z \le \sqrt{y} \\ & x/y = z^2, \end{array}$$

with variables $x, y, z \in \mathbf{R}$ (and the implicit constraint $x, y, z > 0$). Using the simple transformations described above, we obtain the equivalent standard form GP

$$\begin{array}{ll} \text{minimize} & x^{-1}y \\ \text{subject to} & 2x^{-1} \le 1, \quad (1/3)x \le 1 \\ & x^2 y^{-1/2} + 3y^{1/2}z^{-1} \le 1 \\ & xy^{-1}z^{-2} = 1. \end{array}$$

We will refer to a problem like this one, that is easily transformed to an equivalent GP in the standard form (4.43), also as a GP. (In the same way that we refer to a problem easily transformed to an LP as an LP.)

4.5.3 Geometric program in convex form

Geometric programs are not (in general) convex optimization problems, but they can be transformed to convex problems by a change of variables and a transformation of the objective and constraint functions.

We will use the variables defined as $y_i = \log x_i$, so $x_i = e^{y_i}$. If f is the monomial function of x given in (4.41), *i.e.*,

$$f(x) = c x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n},$$

then

$$\begin{aligned} f(x) &= f(e^{y_1}, \ldots, e^{y_n}) \\ &= c(e^{y_1})^{a_1} \cdots (e^{y_n})^{a_n} \\ &= e^{a^T y + b}, \end{aligned}$$

where $b = \log c$. The change of variables $y_i = \log x_i$ turns a monomial function into the exponential of an affine function.

Similarly, if f is the posynomial given by (4.42), *i.e.*,

$$f(x) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}},$$

then

$$f(x) = \sum_{k=1}^{K} e^{a_k^T y + b_k},$$

where $a_k = (a_{1k}, \ldots, a_{nk})$ and $b_k = \log c_k$. After the change of variables, a posynomial becomes a sum of exponentials of affine functions.

The geometric program (4.43) can be expressed in terms of the new variable y as

$$\begin{array}{ll} \text{minimize} & \sum_{k=1}^{K_0} e^{a_{0k}^T y + b_{0k}} \\ \text{subject to} & \sum_{k=1}^{K_i} e^{a_{ik}^T y + b_{ik}} \leq 1, \quad i = 1, \ldots, m \\ & e^{g_i^T y + h_i} = 1, \quad i = 1, \ldots, p, \end{array}$$

where $a_{ik} \in \mathbf{R}^n$, $i = 0, \ldots, m$, contain the exponents of the posynomial inequality constraints, and $g_i \in \mathbf{R}^n$, $i = 1, \ldots, p$, contain the exponents of the monomial equality constraints of the original geometric program.

Now we transform the objective and constraint functions, by taking the logarithm. This results in the problem

$$\begin{array}{ll} \text{minimize} & \tilde{f}_0(y) = \log\left(\sum_{k=1}^{K_0} e^{a_{0k}^T y + b_{0k}}\right) \\ \text{subject to} & \tilde{f}_i(y) = \log\left(\sum_{k=1}^{K_i} e^{a_{ik}^T y + b_{ik}}\right) \leq 0, \quad i = 1, \ldots, m \\ & \tilde{h}_i(y) = g_i^T y + h_i = 0, \quad i = 1, \ldots, p. \end{array} \tag{4.44}$$

Since the functions $\tilde{f}_i$ are convex, and $\tilde{h}_i$ are affine, this problem is a convex optimization problem. We refer to it as a *geometric program in convex form.* To distinguish it from the original geometric program, we refer to (4.43) as a *geometric program in posynomial form*.

Note that the transformation between the posynomial form geometric program (4.43) and the convex form geometric program (4.44) does not involve any computation; the problem data for the two problems are the same. It simply changes the form of the objective and constraint functions.

If the posynomial objective and constraint functions all have only one term, *i.e.*, are monomials, then the convex form geometric program (4.44) reduces to a (general) linear program. We can therefore consider geometric programming to be a generalization, or extension, of linear programming.

4.5.4 Examples

Frobenius norm diagonal scaling

Consider a matrix $M \in \mathbf{R}^{n\times n}$, and the associated linear function that maps u into $y = Mu$. Suppose we scale the coordinates, *i.e.*, change variables to $\tilde{u} = Du$, $\tilde{y} = Dy$, where D is diagonal, with $D_{ii} > 0$. In the new coordinates the linear function is given by $\tilde{y} = DMD^{-1}\tilde{u}$.

Now suppose we want to choose the scaling in such a way that the resulting matrix, DMD^{-1}, is small. We will use the Frobenius norm (squared) to measure the size of the matrix:

$$\begin{aligned} \|DMD^{-1}\|_F^2 &= \mathbf{tr}\left(\left(DMD^{-1}\right)^T \left(DMD^{-1}\right)\right) \\ &= \sum_{i,j=1}^{n} \left(DMD^{-1}\right)_{ij}^2 \\ &= \sum_{i,j=1}^{n} M_{ij}^2 d_i^2/d_j^2, \end{aligned}$$

where $D = \mathbf{diag}(d)$. Since this is a posynomial in d, the problem of choosing the scaling d to minimize the Frobenius norm is an unconstrained geometric program,

$$\begin{array}{ll} \text{minimize} & \sum_{i,j=1}^{n} M_{ij}^2 d_i^2/d_j^2, \end{array}$$

with variable d. The only exponents in this geometric program are 0, 2, and -2.

Design of a cantilever beam

We consider the design of a cantilever beam, which consists of N segments, numbered from right to left as $1, \dots, N$, as shown in figure 4.6. Each segment has unit length and a uniform rectangular cross-section with width w_i and height h_i. A vertical load (force) F is applied at the right end of the beam. This load causes the beam to deflect (downward), and induces stress in each segment of the beam. We assume that the deflections are small, and that the material is linearly elastic, with Young's modulus E.

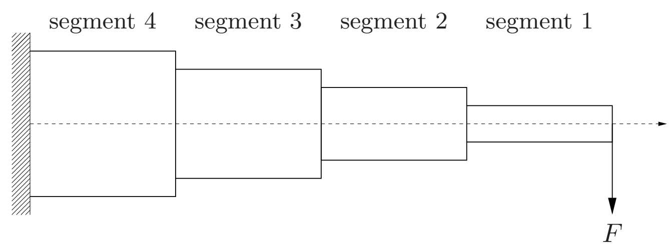


Figure 4.6 Segmented cantilever beam with 4 segments. Each segment has unit length and a rectangular profile. A vertical force F is applied at the right end of the beam.

The design variables in the problem are the widths w_i and heights h_i of the N segments. We seek to minimize the total volume of the beam (which is proportional to its weight),

$$w_1 h_1 + \cdots + w_N h_N,$$

subject to some design constraints. We impose upper and lower bounds on width and height of the segments,

$$w_{\min} \leq w_i \leq w_{\max}, \quad h_{\min} \leq h_i \leq h_{\max}, \quad i = 1, \ldots, N,$$

as well as the aspect ratios,

$$S_{\min} \leq h_i / w_i \leq S_{\max}.$$

In addition, we have a limit on the maximum allowable stress in the material, and on the vertical deflection at the end of the beam.

We first consider the maximum stress constraint. The maximum stress in segment i, which we denote σ_i, is given by $\sigma_i = 6iF/(w_i h_i^2)$. We impose the constraints

$$\frac{6iF}{w_i h_i^2} \leq \sigma_{\max}, \quad i = 1, \ldots, N,$$

to ensure that the stress does not exceed the maximum allowable value $\sigma_{\max}$ anywhere in the beam.

The last constraint is a limit on the vertical deflection at the end of the beam, which we will denote y_1:

$$y_1 \leq y_{\max}.$$

The deflection y_1 can be found by a recursion that involves the deflection and slope of the beam segments:

$$v_i = 12(i - 1/2) \frac{F}{E w_i h_i^3} + v_{i+1}, \qquad y_i = 6(i - 1/3) \frac{F}{E w_i h_i^3} + v_{i+1} + y_{i+1}, \quad (4.45)$$

for $i = N, N-1, \ldots, 1$, with starting values $v_{N+1} = y_{N+1} = 0$. In this recursion, y_i is the deflection at the right end of segment i, and v_i is the slope at that point. We can use the recursion (4.45) to show that these deflection and slope quantities are in fact posynomial functions of the variables w and h. We first note that v_{N+1} and y_{N+1} are zero, and therefore posynomials. Now assume that v_{i+1} and y_{i+1} are posynomial functions of w and h. The lefthand equation in (4.45) shows that v_i is the sum of a monomial and a posynomial (*i.e.*, v_{i+1}), and therefore is a posynomial. From the righthand equation in (4.45), we see that the deflection y_i is the sum of a monomial and two posynomials (v_{i+1} and y_{i+1}), and so is a posynomial. In particular, the deflection at the end of the beam, y_1, is a posynomial.

The problem is then

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^N w_i h_i \\ \text{subject to} & w_{\min} \leq w_i \leq w_{\max}, \quad i = 1, \dots, N \\ & h_{\min} \leq h_i \leq h_{\max}, \quad i = 1, \dots, N \\ & S_{\min} \leq h_i/w_i \leq S_{\max}, \quad i = 1, \dots, N \\ & 6iF/(w_i h_i^2) \leq \sigma_{\max}, \quad i = 1, \dots, N \\ & y_1 \leq y_{\max}, \end{array} \tag{4.46}$$

with variables w and h. This is a GP, since the objective is a posynomial, and the constraints can all be expressed as posynomial inequalities. (In fact, the constraints can be all be expressed as monomial inequalities, with the exception of the deflection limit, which is a complicated posynomial inequality.)

When the number of segments N is large, the number of monomial terms appearing in the posynomial y_1 grows approximately as N^2. Another formulation of this problem, explored in exercise 4.31, is obtained by introducing $v_1, \dots, v_N$ and $y_1, \dots, y_N$ as variables, and including a modified version of the recursion as a set of constraints. This formulation avoids this growth in the number of monomial terms.

Minimizing spectral radius via Perron-Frobenius theory

Suppose the matrix $A \in \mathbf{R}^{n \times n}$ is elementwise nonnegative, *i.e.*, $A_{ij} \geq 0$ for $i, j = 1, \dots, n$, and irreducible, which means that the matrix $(I + A)^{n-1}$ is elementwise positive. The Perron-Frobenius theorem states that A has a positive real eigenvalue λ_{pf} equal to its spectral radius, *i.e.*, the largest magnitude of its eigenvalues. The Perron-Frobenius eigenvalue λ_{pf} determines the asymptotic rate of growth or decay of A^k, as $k \to \infty$; in fact, the matrix $((1/\lambda_{\mathrm{pf}})A)^k$ converges. Roughly speaking, this means that as $k \to \infty$, A^k grows like λ_{pf}^k, if $\lambda_{\mathrm{pf}} > 1$, or decays like λ_{pf}^k, if $\lambda_{\mathrm{pf}} < 1$.

A basic result in the theory of nonnegative matrices states that the Perron-Frobenius eigenvalue is given by

$$\lambda_{\mathrm{pf}} = \inf\{\lambda \mid Av \preceq \lambda v \text{ for some } v \succ 0\}$$

(and moreover, that the infimum is achieved). The inequality $Av \preceq \lambda v$ can be expressed as

$$\sum_{j=1}^n A_{ij} v_j / (\lambda v_i) \leq 1, \quad i = 1, \dots, n, \tag{4.47}$$

which is a set of posynomial inequalities in the variables A_{ij}, v_i, and λ. Thus, the condition that $\lambda_{\mathrm{pf}} \leq \lambda$ can be expressed as a set of posynomial inequalities in A, v, and λ. This allows us to solve some optimization problems involving the Perron-Frobenius eigenvalue using geometric programming.

Suppose that the entries of the matrix A are posynomial functions of some underlying variable $x \in \mathbf{R}^k$. In this case the inequalities (4.47) are posynomial inequalities in the variables $x \in \mathbf{R}^k$, $v \in \mathbf{R}^n$, and $\lambda \in \mathbf{R}$. We consider the problem of choosing x to minimize the Perron-Frobenius eigenvalue (or spectral radius) of A, possibly subject to posynomial inequalities on x,

$$\begin{array}{ll} \text{minimize} & \lambda_{\text{pf}}(A(x)) \\ \text{subject to} & f_i(x) \leq 1, \quad i = 1, \ldots, p, \end{array}$$

where f_i are posynomials. Using the characterization above, we can express this problem as the GP

$$\begin{array}{ll} \text{minimize} & \lambda \\ \text{subject to} & \sum_{j=1}^n A_{ij} v_j / (\lambda v_i) \leq 1, \quad i = 1, \ldots, n \\ & f_i(x) \leq 1, \quad i = 1, \ldots, p, \end{array}$$

where the variables are x, v, and λ.

As a specific example, we consider a simple model for the population dynamics for a bacterium, with time or period denoted by $t = 0, 1, 2, \ldots$, in hours. The vector $p(t) \in \mathbf{R}_+^4$ characterizes the population age distribution at period t: $p_1(t)$ is the total population between 0 and 1 hours old; $p_2(t)$ is the total population between 1 and 2 hours old; and so on. We (arbitrarily) assume that no bacteria live more than 4 hours. The population propagates in time as $p(t+1) = Ap(t)$, where

$$A = \begin{bmatrix} b_1 & b_2 & b_3 & b_4 \\ s_1 & 0 & 0 & 0 \\ 0 & s_2 & 0 & 0 \\ 0 & 0 & s_3 & 0 \end{bmatrix}.$$

Here b_i is the birth rate among bacteria in age group i, and s_i is the survival rate from age group i into age group $i+1$. We assume that $b_i > 0$ and $0 < s_i < 1$, which implies that the matrix A is irreducible.

The Perron-Frobenius eigenvalue of A determines the asymptotic growth or decay rate of the population. If $\lambda_{\text{pf}} < 1$, the population converges to zero like λ_{pf}^t, and so has a half-life of $-1/\log_2 \lambda_{\text{pf}}$ hours. If $\lambda_{\text{pf}} > 1$ the population grows geometrically like λ_{pf}^t, with a doubling time of $1/\log_2 \lambda_{\text{pf}}$ hours. Minimizing the spectral radius of A corresponds to finding the fastest decay rate, or slowest growth rate, for the population.

As our underlying variables, on which the matrix A depends, we take c_1 and c_2, the concentrations of two chemicals in the environment that affect the birth and survival rates of the bacteria. We model the birth and survival rates as monomial functions of the two concentrations:

$$\begin{array}{rcll} b_i &=& b_i^{\text{nom}} (c_1/c_1^{\text{nom}})^{\alpha_i} (c_2/c_2^{\text{nom}})^{\beta_i}, & i = 1, \ldots, 4 \\ s_i &=& s_i^{\text{nom}} (c_1/c_1^{\text{nom}})^{\gamma_i} (c_2/c_2^{\text{nom}})^{\delta_i}, & i = 1, \ldots, 3. \end{array}$$

Here, b_i^{nom} is nominal birth rate, s_i^{nom} is nominal survival rate, and c_i^{nom} is nominal concentration of chemical i. The constants α_i, β_i, γ_i, and δ_i give the effect on the birth and survival rates due to changes in the concentrations of the chemicals away from the nominal values. For example $\alpha_2 = -0.3$ and $\gamma_1 = 0.5$ means that an increase in concentration of chemical 1, over the nominal concentration, causes a decrease in the birth rate of bacteria that are between 1 and 2 hours old, and an increase in the survival rate of bacteria from 0 to 1 hours old.

We assume that the concentrations c_1 and c_2 can be independently increased or decreased (say, within a factor of 2), by administering drugs, and pose the problem of finding the drug mix that maximizes the population decay rate (*i.e.*, minimizes $\lambda_{\rm pf}(A)$). Using the approach described above, this problem can be posed as the GP

$$\begin{array}{ll} \text{minimize} & \lambda \\ \text{subject to} & b_1v_1 + b_2v_2 + b_3v_3 + b_4v_4 \leq \lambda v_1 \\ & s_1v_1 \leq \lambda v_2 \\ & s_2v_2 \leq \lambda v_3 \\ & s_3v_3 \leq \lambda v_4 \\ & 1/2 \leq c_i/c_i^{\rm nom} \leq 2, \quad i=1,2 \\ & b_i = b_i^{\rm nom}(c_1/c_1^{\rm nom})^{\alpha_i}(c_2/c_2^{\rm nom})^{\beta_i}, \quad i=1,\ldots,4 \\ & s_i = s_i^{\rm nom}(c_1/c_1^{\rm nom})^{\gamma_i}(c_2/c_2^{\rm nom})^{\delta_i}, \quad i=1,\ldots,3, \end{array}$$

with variables b_i, s_i, c_i, v_i, and λ.

4.6 Generalized inequality constraints

One very useful generalization of the standard form convex optimization problem (4.15) is obtained by allowing the inequality constraint functions to be vector valued, and using generalized inequalities in the constraints:

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \preceq_{K_i} 0, \quad i=1,\ldots,m \\ & Ax = b, \end{array} \tag{4.48}$$

where $f_0 : \mathbf{R}^n \to \mathbf{R}$, $K_i \subseteq \mathbf{R}^{k_i}$ are proper cones, and $f_i : \mathbf{R}^n \to \mathbf{R}^{k_i}$ are K_i-convex. We refer to this problem as a (standard form) *convex optimization problem with generalized inequality constraints*. Problem (4.15) is a special case with $K_i = \mathbf{R}_+$, $i = 1,\ldots,m$.

Many of the results for ordinary convex optimization problems hold for problems with generalized inequalities. Some examples are:

- The feasible set, any sublevel set, and the optimal set are convex.
- Any point that is locally optimal for the problem (4.48) is globally optimal.
- The optimality condition for differentiable f_0, given in §4.2.3, holds without any change.

We will also see (in chapter 11) that convex optimization problems with generalized inequality constraints can often be solved as easily as ordinary convex optimization problems.

4.6.1 Conic form problems

Among the simplest convex optimization problems with generalized inequalities are the *conic form problems* (or *cone programs*), which have a linear objective and one inequality constraint function, which is affine (and therefore K-convex):

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Fx + g \preceq_K 0 \\ & Ax = b. \end{array} \tag{4.49}$$

When K is the nonnegative orthant, the conic form problem reduces to a linear program. We can view conic form problems as a generalization of linear programs in which componentwise inequality is replaced with a generalized linear inequality.

Continuing the analogy to linear programming, we refer to the conic form problem

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

as a *conic form problem in standard form.* Similarly, the problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Fx + g \preceq_K 0 \end{array}$$

is called a *conic form problem in inequality form.*

4.6.2 Semidefinite programming

When K is $\mathbf{S}^k_+$, the cone of positive semidefinite $k \times k$ matrices, the associated conic form problem is called a *semidefinite program* (SDP), and has the form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x_1 F_1 + \cdots + x_n F_n + G \preceq 0 \\ & Ax = b, \end{array} \tag{4.50}$$

where $G,\ F_1, \ldots, F_n \in \mathbf{S}^k$, and $A \in \mathbf{R}^{p \times n}$. The inequality here is a linear matrix inequality (see example 2.10).

If the matrices $G,\ F_1, \ldots, F_n$ are all diagonal, then the LMI in (4.50) is equivalent to a set of n linear inequalities, and the SDP (4.50) reduces to a linear program.

Standard and inequality form semidefinite programs

Following the analogy to LP, a *standard form SDP* has linear equality constraints, and a (matrix) nonnegativity constraint on the variable $X \in \mathbf{S}^n$:

$$\begin{array}{ll} \text{minimize} & \mathbf{tr}(CX) \\ \text{subject to} & \mathbf{tr}(A_i X) = b_i, \quad i = 1, \ldots, p \\ & X \succeq 0, \end{array} \tag{4.51}$$

where $C,\ A_1, \dots, A_p \in \mathbf{S}^n$. (Recall that $\mathbf{tr}(CX) = \sum_{i,j=1}^n C_{ij}X_{ij}$ is the form of a general real-valued linear function on $\mathbf{S}^n$.) This form should be compared to the standard form linear program (4.28). In LP and SDP standard forms, we minimize a linear function of the variable, subject to p linear equality constraints on the variable, and a nonnegativity constraint on the variable.

An *inequality form SDP*, analogous to an inequality form LP (4.29), has no equality constraints, and one LMI:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x_1 A_1 + \cdots + x_n A_n \preceq B, \end{array}$$

with variable $x \in \mathbf{R}^n$, and parameters $B,\ A_1, \dots, A_n \in \mathbf{S}^k$, $c \in \mathbf{R}^n$.

Multiple LMIs and linear inequalities

It is common to refer to a problem with linear objective, linear equality and inequality constraints, and several LMI constraints, *i.e.*,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & F^{(i)}(x) = x_1 F_1^{(i)} + \cdots + x_n F_n^{(i)} + G^{(i)} \preceq 0, \quad i = 1, \dots, K \\ & Gx \preceq h, \qquad Ax = b, \end{array}$$

as an SDP as well. Such problems are readily transformed to an SDP, by forming a large block diagonal LMI from the individual LMIs and linear inequalities:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{diag}(Gx - h, F^{(1)}(x), \dots, F^{(K)}(x)) \preceq 0 \\ & Ax = b. \end{array}$$

4.6.3 Examples

Second-order cone programming

The SOCP (4.36) can be expressed as a conic form problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & -(A_i x + b_i, c_i^T x + d_i) \preceq_{K_i} 0, \quad i = 1, \dots, m \\ & Fx = g, \end{array}$$

in which

$$K_i = \{(y,t) \in \mathbf{R}^{n_i+1} \mid \|y\|_2 \le t\},$$

i.e., the second-order cone in $\mathbf{R}^{n_i+1}$. This explains the name *second-order cone program* for the optimization problem (4.36).

Matrix norm minimization

Let $A(x) = A_0 + x_1 A_1 + \cdots + x_n A_n$, where $A_i \in \mathbf{R}^{p \times q}$. We consider the unconstrained problem

$$\text{minimize} \quad \|A(x)\|_2,$$

where $\|\cdot\|_2$ denotes the spectral norm (maximum singular value), and $x \in \mathbf{R}^n$ is the variable. This is a convex problem since $\|A(x)\|_2$ is a convex function of x.

Using the fact that $\|A\|_2 \leq s$ if and only if $A^T A \preceq s^2 I$ (and $s \geq 0$), we can express the problem in the form

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & A(x)^T A(x) \preceq sI, \end{array}$$

with variables x and s. Since the function $A(x)^T A(x) - sI$ is matrix convex in (x, s), this is a convex optimization problem with a single $q \times q$ matrix inequality constraint.

We can also formulate the problem using a single linear matrix inequality of size $(p+q) \times (p+q)$, using the fact that

$$A^T A \preceq t^2 I \text{ (and } t \geq 0) \iff \left[\begin{array}{cc} tI & A \\ A^T & tI \end{array}\right] \succeq 0.$$

(see §A.5.5). This results in the SDP

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \left[\begin{array}{cc} tI & A(x) \\ A(x)^T & tI \end{array}\right] \succeq 0 \end{array}$$

in the variables x and t.

Moment problems

Let t be a random variable in $\mathbf{R}$. The expected values $\mathbf{E}\, t^k$ (assuming they exist) are called the (power) *moments* of the distribution of t. The following classical results give a characterization of a moment sequence.

If there is a probability distribution on $\mathbf{R}$ such that $x_k = \mathbf{E}\, t^k$, $k = 0, \ldots, 2n$, then $x_0 = 1$ and

$$H(x_0, \ldots, x_{2n}) = \left[\begin{array}{cccccc} x_0 & x_1 & x_2 & \cdots & x_{n-1} & x_n \\ x_1 & x_2 & x_3 & \cdots & x_n & x_{n+1} \\ x_2 & x_3 & x_4 & \cdots & x_{n+1} & x_{n+2} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ x_{n-1} & x_n & x_{n+1} & \cdots & x_{2n-2} & x_{2n-1} \\ x_n & x_{n+1} & x_{n+2} & \cdots & x_{2n-1} & x_{2n} \end{array}\right] \succeq 0. \qquad (4.52)$$

(The matrix H is called the *Hankel matrix* associated with $x_0, \ldots, x_{2n}$.) This is easy to see: Let $x_i = \mathbf{E}\, t^i$, $i = 0, \ldots, 2n$ be the moments of some distribution, and let $y = (y_0, y_1, \ldots y_n) \in \mathbf{R}^{n+1}$. Then we have

$$y^T H(x_0, \ldots, x_{2n}) y = \sum_{i,j=0}^{n} y_i y_j \mathbf{E}\, t^{i+j} = \mathbf{E}(y_0 + y_1 t^1 + \cdots + y_n t^n)^2 \geq 0.$$

The following partial converse is less obvious: If $x_0 = 1$ and $H(x) \succ 0$, then there exists a probability distribution on $\mathbf{R}$ such that $x_i = \mathbf{E}\, t^i$, $i = 0, \ldots, 2n$. (For a proof, see exercise 2.37.) Now suppose that $x_0 = 1$, and $H(x) \succeq 0$ (but possibly $H(x) \not\succ 0$), *i.e.*, the linear matrix inequality (4.52) holds, but possibly not strictly. In this case, there is a sequence of distributions on $\mathbf{R}$, whose moments converge to x. In summary: the condition that $x_0, \ldots, x_{2n}$ be the moments of some distribution on $\mathbf{R}$ (or the limit of the moments of a sequence of distributions) can be expressed as the linear matrix inequality (4.52) in the variable x, together with the linear equality $x_0 = 1$. Using this fact, we can cast some interesting problems involving moments as SDPs.

Suppose t is a random variable on $\mathbf{R}$. We do not know its distribution, but we do know some bounds on the moments, *i.e.*,

$$\underline{\mu}_k \le \mathbf{E}\, t^k \le \overline{\mu}_k, \quad k = 1, \ldots, 2n$$

(which includes, as a special case, knowing exact values of some of the moments). Let $p(t) = c_0 + c_1 t + \cdots + c_{2n} t^{2n}$ be a given polynomial in t. The expected value of $p(t)$ is linear in the moments $\mathbf{E}\, t^i$:

$$\mathbf{E}\, p(t) = \sum_{i=0}^{2n} c_i \,\mathbf{E}\, t^i = \sum_{i=0}^{2n} c_i x_i.$$

We can compute upper and lower bounds for $\mathbf{E}\, p(t)$,

$$\begin{array}{ll} \text{minimize (maximize)} & \mathbf{E}\, p(t) \\ \text{subject to} & \underline{\mu}_k \le \mathbf{E}\, t^k \le \overline{\mu}_k, \quad k = 1, \ldots, 2n, \end{array}$$

over all probability distributions that satisfy the given moment bounds, by solving the SDP

$$\begin{array}{ll} \text{minimize (maximize)} & c_1 x_1 + \cdots + c_{2n} x_{2n} \\ \text{subject to} & \underline{\mu}_k \le x_k \le \overline{\mu}_k, \quad k = 1, \ldots, 2n \\ & H(1, x_1, \ldots, x_{2n}) \succeq 0 \end{array}$$

with variables $x_1, \ldots, x_{2n}$. This gives bounds on $\mathbf{E}\, p(t)$, over all probability distributions that satisfy the known moment constraints. The bounds are sharp in the sense that there exists a sequence of distributions, whose moments satisfy the given moment bounds, for which $\mathbf{E}\, p(t)$ converges to the upper and lower bounds found by these SDPs.

Bounding portfolio risk with incomplete covariance information

We consider once again the setup for the classical Markowitz portfolio problem (see page 155). We have a portfolio of n assets or stocks, with x_i denoting the amount of asset i that is held over some investment period, and p_i denoting the relative price change of asset i over the period. The change in total value of the portfolio is $p^T x$. The price change vector p is modeled as a random vector, with mean and covariance

$$\overline{p} = \mathbf{E}\, p, \qquad \Sigma = \mathbf{E}(p - \overline{p})(p - \overline{p})^T.$$

The change in value of the portfolio is therefore a random variable with mean $\overline{p}^T x$ and standard deviation $\sigma = (x^T \Sigma x)^{1/2}$. The risk of a large loss, *i.e.*, a change in portfolio value that is substantially below its expected value, is directly related to the standard deviation σ, and increases with it. For this reason the standard deviation σ (or the variance σ^2) is used as a measure of the risk associated with the portfolio.

In the classical portfolio optimization problem, the portfolio x is the optimization variable, and we minimize the risk subject to a minimum mean return and other constraints. The price change statistics $\overline{p}$ and Σ are known problem parameters. In the risk bounding problem considered here, we turn the problem around: we assume the portfolio x is known, but only partial information is available about the covariance matrix Σ. We might have, for example, an upper and lower bound on each entry:

$$L_{ij} \leq \Sigma_{ij} \leq U_{ij}, \quad i,\ j = 1, \ldots, n,$$

where L and U are given. We now pose the question: what is the maximum risk for our portfolio, over all covariance matrices consistent with the given bounds? We define the *worst-case variance* of the portfolio as

$$\sigma_{\mathrm{wc}}^2 = \sup\{x^T \Sigma x \mid L_{ij} \leq \Sigma_{ij} \leq U_{ij},\ i,j = 1, \ldots, n,\ \Sigma \succeq 0\}.$$

We have added the condition $\Sigma \succeq 0$, which the covariance matrix must, of course, satisfy.

We can find σ_{wc} by solving the SDP

$$\begin{array}{ll} \text{maximize} & x^T \Sigma x \\ \text{subject to} & L_{ij} \leq \Sigma_{ij} \leq U_{ij}, \quad i,\ j = 1, \ldots, n \\ & \Sigma \succeq 0 \end{array}$$

with variable $\Sigma \in \mathbf{S}^n$ (and problem parameters x, L, and U). The optimal Σ is the worst covariance matrix consistent with our given bounds on the entries, where 'worst' means largest risk with the (given) portfolio x. We can easily construct a distribution for p that is consistent with the given bounds, and achieves the worst-case variance, from an optimal Σ for the SDP. For example, we can take $p = \overline{p} + \Sigma^{1/2} v$, where v is any random vector with $\mathbf{E}\, v = 0$ and $\mathbf{E}\, vv^T = I$.

Evidently we can use the same method to determine σ_{wc} for any prior information about Σ that is convex. We list here some examples.

- *Known variance of certain portfolios.* We might have equality constraints such as
$$u_k^T \Sigma u_k = \sigma_k^2,$$
where u_k and σ_k are given. This corresponds to prior knowledge that certain known portfolios (given by u_k) have known (or very accurately estimated) variance.

- *Including effects of estimation error.* If the covariance Σ is estimated from empirical data, the estimation method will give an estimate $\hat{\Sigma}$, and some information about the reliability of the estimate, such as a confidence ellipsoid. This can be expressed as
$$C(\Sigma - \hat{\Sigma}) \leq \alpha,$$
where C is a positive definite quadratic form on $\mathbf{S}^n$, and the constant α determines the confidence level.

- *Factor models.* The covariance might have the form

$$\Sigma = F\Sigma_{\text{factor}}F^T + D,$$

where $F \in \mathbf{R}^{n\times k}$, $\Sigma_{\text{factor}} \in \mathbf{S}^k$, and D is diagonal. This corresponds to a model of the price changes of the form

$$p = Fz + d,$$

where z is a random variable (the underlying *factors* that affect the price changes) and d_i are independent (additional volatility of each asset price). We assume that the factors are known. Since Σ is linearly related to Σ_{factor} and D, we can impose any convex constraint on them (representing prior information) and still compute σ_{wc} using convex optimization.

- *Information about correlation coefficients.* In the simplest case, the diagonal entries of Σ (*i.e.*, the volatilities of each asset price) are known, and bounds on correlation coefficients between price changes are known:

$$l_{ij} \le \rho_{ij} = \frac{\Sigma_{ij}}{\Sigma_{ii}^{1/2}\Sigma_{jj}^{1/2}} \le u_{ij}, \quad i,\ j = 1,\ldots,n.$$

Since Σ_{ii} are known, but Σ_{ij} for $i \ne j$ are not, these are linear inequalities.

Fastest mixing Markov chain on a graph

We consider an undirected graph, with nodes $1,\ldots,n$, and a set of edges

$$\mathcal{E} \subseteq \{1,\ldots,n\} \times \{1,\ldots,n\}.$$

Here $(i,j) \in \mathcal{E}$ means that nodes i and j are connected by an edge. Since the graph is undirected, $\mathcal{E}$ is symmetric: $(i,j) \in \mathcal{E}$ if and only if $(j,i) \in \mathcal{E}$. We allow the possibility of self-loops, *i.e.*, we can have $(i,i) \in \mathcal{E}$.

We define a Markov chain, with state $X(t) \in \{1,\ldots,n\}$, for $t \in \mathbf{Z}_+$ (the set of nonnegative integers), as follows. With each edge $(i,j) \in \mathcal{E}$ we associate a probability P_{ij}, which is the probability that X makes a transition between nodes i and j. State transitions can only occur across edges; we have $P_{ij} = 0$ for $(i,j) \notin \mathcal{E}$. The probabilities associated with the edges must be nonnegative, and for each node, the sum of the probabilities of links connected to the node (including a self-loop, if there is one) must equal one.

The Markov chain has transition probability matrix

$$P_{ij} = \mathbf{prob}(X(t+1) = i \mid X(t) = j), \quad i,j = 1,\ldots,n.$$

This matrix must satisfy

$$P_{ij} \ge 0, \quad i,\ j = 1,\ldots,n, \qquad \mathbf{1}^T P = \mathbf{1}^T, \qquad P = P^T, \tag{4.53}$$

and also

$$P_{ij} = 0 \quad \text{for } (i,j) \notin \mathcal{E}. \tag{4.54}$$

Since P is symmetric and $\mathbf{1}^T P = \mathbf{1}^T$, we conclude $P\mathbf{1} = \mathbf{1}$, so the uniform distribution $(1/n)\mathbf{1}$ is an equilibrium distribution for the Markov chain. Convergence of the distribution of $X(t)$ to $(1/n)\mathbf{1}$ is determined by the second largest (in magnitude) eigenvalue of P, *i.e.*, by $r = \max\{\lambda_2, -\lambda_n\}$, where

$$1 = \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$$

are the eigenvalues of P. We refer to r as the *mixing rate* of the Markov chain. If $r = 1$, then the distribution of $X(t)$ need not converge to $(1/n)\mathbf{1}$ (which means the Markov chain does not mix). When $r < 1$, the distribution of $X(t)$ approaches $(1/n)\mathbf{1}$ asymptotically as r^t, as $t \to \infty$. Thus, the smaller r is, the faster the Markov chain mixes.

The *fastest mixing Markov chain problem* is to find P, subject to the constraints (4.53) and (4.54), that minimizes r. (The problem data is the graph, *i.e.*, $\mathcal{E}$.) We will show that this problem can be formulated as an SDP.

Since the eigenvalue $\lambda_1 = 1$ is associated with the eigenvector $\mathbf{1}$, we can express the mixing rate as the norm of the matrix P, restricted to the subspace $\mathbf{1}^\perp$: $r = \|QPQ\|_2$, where $Q = I - (1/n)\mathbf{1}\mathbf{1}^T$ is the matrix representing orthogonal projection on $\mathbf{1}^\perp$. Using the property $P\mathbf{1} = \mathbf{1}$, we have

$$\begin{array}{rcl} r & = & \|QPQ\|_2 \\ & = & \|(I - (1/n)\mathbf{1}\mathbf{1}^T)P(I - (1/n)\mathbf{1}\mathbf{1}^T)\|_2 \\ & = & \|P - (1/n)\mathbf{1}\mathbf{1}^T\|_2. \end{array}$$

This shows that the mixing rate r is a convex function of P, so the fastest mixing Markov chain problem can be cast as the convex optimization problem

$$\begin{array}{ll} \text{minimize} & \|P - (1/n)\mathbf{1}\mathbf{1}^T\|_2 \\ \text{subject to} & P\mathbf{1} = \mathbf{1} \\ & P_{ij} \geq 0, \quad i, j = 1, \ldots, n \\ & P_{ij} = 0 \text{ for } (i, j) \notin \mathcal{E}, \end{array}$$

with variable $P \in \mathbf{S}^n$. We can express the problem as an SDP by introducing a scalar variable t to bound the norm of $P - (1/n)\mathbf{1}\mathbf{1}^T$:

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & -tI \preceq P - (1/n)\mathbf{1}\mathbf{1}^T \preceq tI \\ & P\mathbf{1} = \mathbf{1} \\ & P_{ij} \geq 0, \quad i, j = 1, \ldots, n \\ & P_{ij} = 0 \text{ for } (i, j) \notin \mathcal{E}. \end{array} \tag{4.55}$$

4.7 Vector optimization

4.7.1 General and convex vector optimization problems

In §4.6 we extended the standard form problem (4.1) to include vector-valued constraint functions. In this section we investigate the meaning of a vector-valued *objective function*. We denote a general *vector optimization problem* as

$$\begin{array}{ll} \text{minimize (with respect to } K) & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p. \end{array} \tag{4.56}$$

Here $x \in \mathbf{R}^n$ 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. The only difference between this problem and the standard optimization problem (4.1) is that here, the objective function takes values in $\mathbf{R}^q$, and the problem specification includes a proper cone K, which is used to compare objective values. In the context of vector optimization, the standard optimization problem (4.1) is sometimes called a *scalar optimization problem*.

We say the vector optimization problem (4.56) is a *convex vector optimization problem* if the objective function f_0 is K-convex, the inequality constraint functions $f_1, \ldots, f_m$ are convex, and the equality constraint functions $h_1, \ldots, h_p$ are affine. (As in the scalar case, we usually express the equality constraints as $Ax = b$, where $A \in \mathbf{R}^{p \times n}$.)

What meaning can we give to the vector optimization problem (4.56)? Suppose x and y are two feasible points (*i.e.*, they satisfy the constraints). Their associated objective values, $f_0(x)$ and $f_0(y)$, are to be compared using the generalized inequality $\preceq_K$. We interpret $f_0(x) \preceq_K f_0(y)$ as meaning that x is 'better than or equal' in value to y (as judged by the objective f_0, with respect to K). The confusing aspect of vector optimization is that the two objective values $f_0(x)$ and $f_0(y)$ need not be comparable; we can have neither $f_0(x) \preceq_K f_0(y)$ nor $f_0(y) \preceq_K f_0(x)$, *i.e.*, neither is better than the other. This cannot happen in a scalar objective optimization problem.

4.7.2 Optimal points and values

We first consider a special case, in which the meaning of the vector optimization problem is clear. 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*. If this set has a minimum element (see §2.4.2), *i.e.*, there is a feasible x such that $f_0(x) \preceq_K f_0(y)$ for all feasible y, then we say x is *optimal* for the problem (4.56), and refer to $f_0(x)$ as the *optimal value* of the problem. (When a vector optimization problem has an optimal value, it is unique.) If $x^\star$ is an optimal point, then $f_0(x^\star)$, the objective at $x^\star$, can be compared to the objective at every other feasible point, and is better than or equal to it. Roughly speaking, $x^\star$ is unambiguously a best choice for x, among feasible points.

A point $x^\star$ is optimal if and only if it is feasible and

$$\mathcal{O} \subseteq f_0(x^\star) + K \tag{4.57}$$

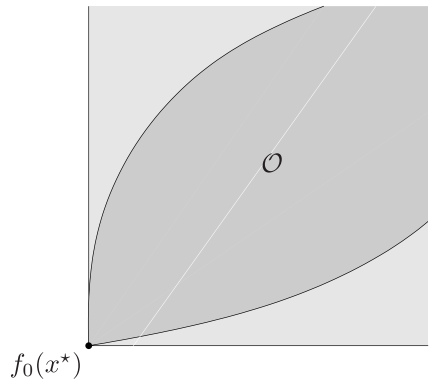


Figure 4.7 The set $\mathcal{O}$ of achievable values for a vector optimization with objective values in $\mathbf{R}^2$, with cone $K = \mathbf{R}^2_+$, is shown shaded. In this case, the point labeled $f_0(x^\star)$ is the optimal value of the problem, and $x^\star$ is an optimal point. The objective value $f_0(x^\star)$ can be compared to every other achievable value $f_0(y)$, and is better than or equal to $f_0(y)$. (Here, 'better than or equal to' means 'is below and to the left of'.) The lightly shaded region is $f_0(x^\star)+K$, which is the set of all $z \in \mathbf{R}^2$ corresponding to objective values worse than (or equal to) $f_0(x^\star)$.

(see §2.4.2). The set $f_0(x^\star) + K$ can be interpreted as the set of values that are worse than, or equal to, $f_0(x^\star)$, so the condition (4.57) states that every achievable value falls in this set. This is illustrated in figure 4.7. Most vector optimization problems do not have an optimal point and an optimal value, but this does occur in some special cases.

Example 4.9 *Best linear unbiased estimator.* Suppose $y = Ax + v$, where $v \in \mathbf{R}^m$ is a measurement noise, $y \in \mathbf{R}^m$ is a vector of measurements, and $x \in \mathbf{R}^n$ is a vector to be estimated, given the measurement y. We assume that A has rank n, and that the measurement noise satisfies $\mathbf{E}\, v = 0$, $\mathbf{E}\, vv^T = I$, *i.e.*, its components are zero mean and uncorrelated.

A *linear estimator* of x has the form $\widehat{x} = Fy$. The estimator is called *unbiased* if for all x we have $\mathbf{E}\, \widehat{x} = x$, *i.e.*, if $FA = I$. The error covariance of an unbiased estimator is

$$\mathbf{E}(\widehat{x} - x)(\widehat{x} - x)^T = \mathbf{E}\, Fvv^T F^T = FF^T.$$

Our goal is to find an unbiased estimator that has a 'small' error covariance matrix. We can compare error covariances using matrix inequality, *i.e.*, with respect to $\mathbf{S}^n_+$. This has the following interpretation: Suppose $\widehat{x}_1 = F_1 y$, $\widehat{x}_2 = F_2 y$ are two unbiased estimators. Then the first estimator is at least as good as the second, *i.e.*, $F_1 F_1^T \preceq F_2 F_2^T$, if and only if for all c,

$$\mathbf{E}(c^T \widehat{x}_1 - c^T x)^2 \le \mathbf{E}(c^T \widehat{x}_2 - c^T x)^2.$$

In other words, for any linear function of x, the estimator F_1 yields at least as good an estimate as does F_2.

We can express the problem of finding an unbiased estimator for x as the vector optimization problem

$$\begin{array}{ll} \text{minimize (w.r.t. } \mathbf{S}_+^n) & FF^T \\ \text{subject to} & FA = I, \end{array} \tag{4.58}$$

with variable $F \in \mathbf{R}^{n \times m}$. The objective FF^T is convex with respect to $\mathbf{S}_+^n$, so the problem (4.58) is a convex vector optimization problem. An easy way to see this is to observe that $v^T FF^T v = \|F^T v\|_2^2$ is a convex function of F for any fixed v.

It is a famous result that the problem (4.58) has an optimal solution, the least-squares estimator, or pseudo-inverse,

$$F^\star = A^\dagger = (A^T A)^{-1} A^T.$$

For any F with $FA = I$, we have $FF^T \succeq F^\star F^{\star T}$. The matrix

$$F^\star F^{\star T} = A^\dagger A^{\dagger T} = (A^T A)^{-1}$$

is the optimal value of the problem (4.58).

4.7.3 Pareto optimal points and values

We now consider the case (which occurs in most vector optimization problems of interest) in which the set of achievable objective values does not have a minimum element, so the problem does not have an optimal point or optimal value. In these cases *minimal* elements of the set of achievable values play an important role. We say that a feasible point x is *Pareto optimal* (or *efficient*) if $f_0(x)$ is a minimal element of the set of achievable values $\mathcal{O}$. In this case we say that $f_0(x)$ is a *Pareto optimal value* for the vector optimization problem (4.56). Thus, a point x is Pareto optimal if it is feasible and, for any feasible y, $f_0(y) \preceq_K f_0(x)$ implies $f_0(y) = f_0(x)$. In other words: any feasible point y that is better than or equal to x (*i.e.*, $f_0(y) \preceq_K f_0(x)$) has exactly the same objective value as x.

A point x is Pareto optimal if and only if it is feasible and

$$(f_0(x) - K) \cap \mathcal{O} = \{f_0(x)\} \tag{4.59}$$

(see §2.4.2). The set $f_0(x) - K$ can be interpreted as the set of values that are better than or equal to $f_0(x)$, so the condition (4.59) states that the only achievable value better than or equal to $f_0(x)$ is $f_0(x)$ itself. This is illustrated in figure 4.8.

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

$$\mathcal{P} \subseteq \mathcal{O} \cap \mathbf{bd}\, \mathcal{O},$$

i.e., every Pareto optimal value is an achievable objective value that lies in the boundary of the set of achievable objective values (see exercise 4.52).

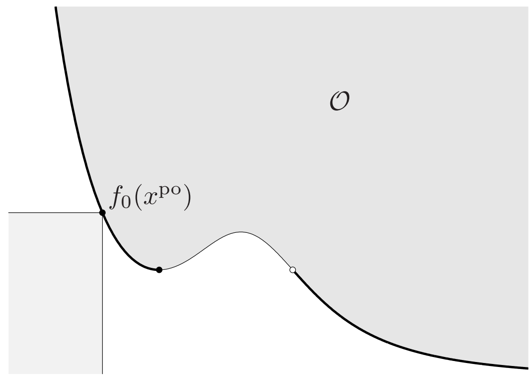


Figure 4.8 The set $\mathcal{O}$ of achievable values for a vector optimization problem with objective values in $\mathbf{R}^2$, with cone $K = \mathbf{R}^2_+$, is shown shaded. This problem does not have an optimal point or value, but it does have a set of Pareto optimal points, whose corresponding values are shown as the darkened curve on the lower left boundary of $\mathcal{O}$. The point labeled $f_0(x^{\rm po})$ is a Pareto optimal value, and $x^{\rm po}$ is a Pareto optimal point. The lightly shaded region is $f_0(x^{\rm po}) - K$, which is the set of all $z \in \mathbf{R}^2$ corresponding to objective values better than (or equal to) $f_0(x^{\rm po})$.

4.7.4 Scalarization

Scalarization is a standard technique for finding Pareto optimal (or optimal) points for a vector optimization problem, based on the characterization of minimum and minimal points via dual generalized inequalities given in §2.6.3. Choose any $\lambda \succ_{K^*} 0$, *i.e.*, any vector that is positive in the dual generalized inequality. Now consider the *scalar* optimization problem

$$\begin{array}{ll} \text{minimize} & \lambda^T f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p, \end{array} \tag{4.60}$$

and let x be an optimal point. Then x is Pareto optimal for the vector optimization problem (4.56). This follows from the dual inequality characterization of minimal points given in §2.6.3, and is also easily shown directly. If x were not Pareto optimal, then there is a y that is feasible, satisfies $f_0(y) \preceq_K f_0(x)$, 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 (4.60).

Using scalarization, we can find Pareto optimal points for *any* vector optimization problem by solving the ordinary scalar optimization problem (4.60). The vector λ, which is sometimes called the *weight vector*, must satisfy $\lambda \succ_{K^*} 0$. The weight vector is a free parameter; by varying it we obtain (possibly) different Pareto optimal solutions of the vector optimization problem (4.56). This is illustrated in figure 4.9. The figure also shows an example of a Pareto optimal point that cannot

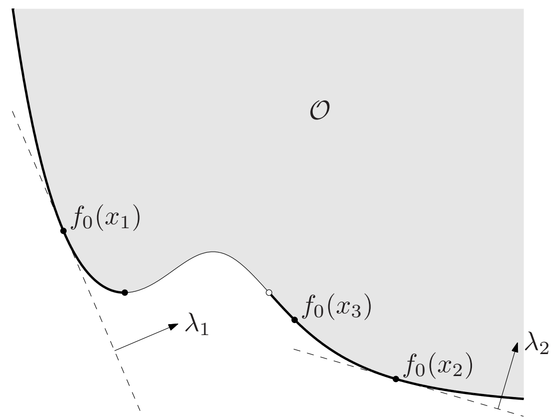


Figure 4.9 *Scalarization.* The set $\mathcal{O}$ of achievable values for a vector optimization problem with cone $K = \mathbf{R}^2_+$. Three Pareto optimal values $f_0(x_1)$, $f_0(x_2)$, $f_0(x_3)$ are shown. The first two values can be obtained by scalarization: $f_0(x_1)$ minimizes $\lambda_1^T u$ over all $u \in \mathcal{O}$ and $f_0(x_2)$ minimizes $\lambda_2^T u$, where $\lambda_1, \lambda_2 \succ 0$. The value $f_0(x_3)$ is Pareto optimal, but cannot be found by scalarization.

be obtained via scalarization, for any value of the weight vector $\lambda \succ_{K^*} 0$.

The method of scalarization can be interpreted geometrically. A point x is optimal for the scalarized problem, *i.e.*, minimizes $\lambda^T f_0$ over the feasible set, if and only if $\lambda^T(f_0(y) - f_0(x)) \geq 0$ for all feasible y. But this is the same as saying that $\{u \mid -\lambda^T(u - f_0(x)) = 0\}$ is a supporting hyperplane to the set of achievable objective values $\mathcal{O}$ at the point $f_0(x)$; in particular

$$\{u \mid \lambda^T(u - f_0(x)) < 0\} \cap \mathcal{O} = \emptyset. \tag{4.61}$$

(See figure 4.9.) Thus, when we find an optimal point for the scalarized problem, we not only find a Pareto optimal point for the original vector optimization problem; we also find an entire halfspace in $\mathbf{R}^q$, given by (4.61), of objective values that cannot be achieved.

Scalarization of convex vector optimization problems

Now suppose the vector optimization problem (4.56) is convex. Then the scalarized problem (4.60) is also convex, since $\lambda^T f_0$ is a (scalar-valued) convex function (by the results in §3.6). This means that we can find Pareto optimal points of a convex vector optimization problem by solving a convex scalar optimization problem. For each choice of the weight vector $\lambda \succ_{K^*} 0$ we get a (usually different) Pareto optimal point.

For convex vector optimization problems we have a partial converse: For every Pareto optimal point x^{po}, there is some nonzero $\lambda \succeq_{K^*} 0$ such that x^{po} is a solution of the scalarized problem (4.60). So, roughly speaking, for convex problems the method of scalarization yields all Pareto optimal points, as the weight vector λ

varies over the K^*-nonnegative, nonzero values. We have to be careful here, because it is *not* true that every solution of the scalarized problem, with $\lambda \succeq_{K^*} 0$ and $\lambda \neq 0$, is a Pareto optimal point for the vector problem. (In contrast, *every* solution of the scalarized problem with $\lambda \succ_{K^*} 0$ is Pareto optimal.)

In some cases we can use this partial converse to find *all* Pareto optimal points of a convex vector optimization problem. Scalarization with $\lambda \succ_{K^*} 0$ gives a set of Pareto optimal points (as it would in a nonconvex vector optimization problem as well). To find the remaining Pareto optimal solutions, we have to consider nonzero weight vectors λ that satisfy $\lambda \succeq_{K^*} 0$. For each such weight vector, we first identify all solutions of the scalarized problem. Then among these solutions we must check which are, in fact, Pareto optimal for the vector optimization problem. These 'extreme' Pareto optimal points can also be found as the limits of the Pareto optimal points obtained from positive weight vectors.

To establish this partial converse, we consider the set

$$\mathcal{A} = \mathcal{O} + K = \{t \in \mathbf{R}^q \mid f_0(x) \preceq_K t \text{ for some feasible } x\}, \tag{4.62}$$

which consists of all values that are worse than or equal to (with respect to $\preceq_K$) some achievable objective value. While the set $\mathcal{O}$ of achievable objective values need not be convex, the set $\mathcal{A}$ is convex, when the problem is convex. Moreover, the minimal elements of $\mathcal{A}$ are exactly the same as the minimal elements of the set $\mathcal{O}$ of achievable values, *i.e.*, they are the same as the Pareto optimal values. (See exercise 4.53.) Now we use the results of §2.6.3 to conclude that any minimal element of $\mathcal{A}$ minimizes $\lambda^T z$ over $\mathcal{A}$ for some nonzero $\lambda \succeq_{K^*} 0$. This means that every Pareto optimal point for the vector optimization problem is optimal for the scalarized problem, for some nonzero weight $\lambda \succeq_{K^*} 0$.

Example 4.10 *Minimal upper bound on a set of matrices.* We consider the (convex) vector optimization problem, with respect to the positive semidefinite cone,

$$\begin{array}{ll} \text{minimize (w.r.t. } \mathbf{S}_+^n) & X \\ \text{subject to} & X \succeq A_i, \quad i = 1, \ldots, m, \end{array} \tag{4.63}$$

where $A_i \in \mathbf{S}^n$, $i = 1, \ldots, m$, are given. The constraints mean that X is an upper bound on the given matrices $A_1, \ldots, A_m$; a Pareto optimal solution of (4.63) is a *minimal upper bound* on the matrices.

To find a Pareto optimal point, we apply scalarization: we choose any $W \in \mathbf{S}_{++}^n$ and form the problem

$$\begin{array}{ll} \text{minimize} & \mathbf{tr}(WX) \\ \text{subject to} & X \succeq A_i, \quad i = 1, \ldots, m, \end{array} \tag{4.64}$$

which is an SDP. Different choices for W will, in general, give different minimal solutions.

The partial converse tells us that if X is Pareto optimal for the vector problem (4.63) then it is optimal for the SDP (4.64), for some nonzero weight matrix $W \succeq 0$. (In this case, however, not every solution of (4.64) is Pareto optimal for the vector optimization problem.)

We can give a simple geometric interpretation for this problem. We associate with each $A \in \mathbf{S}_{++}^n$ an ellipsoid centered at the origin, given by

$$\mathcal{E}_A = \{u \mid u^T A^{-1} u \leq 1\},$$

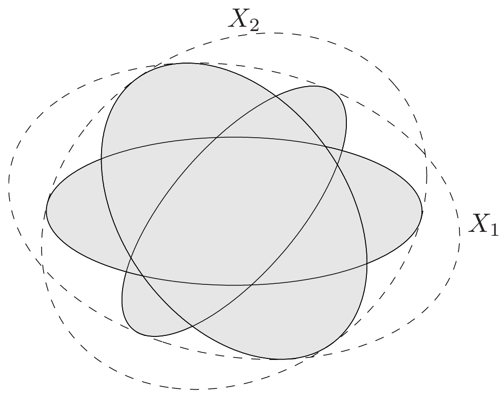


Figure 4.10 Geometric interpretation of the problem (4.63). The three shaded ellipsoids correspond to the data A_1, A_2, $A_3 \in \mathbf{S}^2_{++}$; the Pareto optimal points correspond to minimal ellipsoids that contain them. The two ellipsoids, with boundaries labeled X_1 and X_2, show two minimal ellipsoids obtained by solving the SDP (4.64) for two different weight matrices W_1 and W_2.

so that $A \preceq B$ if and only if $\mathcal{E}_A \subseteq \mathcal{E}_B$. A Pareto optimal point X for the problem (4.63) corresponds to a minimal ellipsoid that contains the ellipsoids associated with $A_1, \ldots, A_m$. An example is shown in figure 4.10.

4.7.5 Multicriterion optimization

When a vector optimization problem involves the cone $K = \mathbf{R}^q_+$, it is called a *multicriterion* or *multi-objective* optimization problem. The components of f_0, say, $F_1, \ldots, F_q$, can be interpreted as q different scalar objectives, each of which we would like to minimize. We refer to F_i as the *ith objective* of the problem. A multicriterion optimization problem is convex if $f_1, \ldots, f_m$ are convex, $h_1, \ldots, h_p$ are affine, and the objectives $F_1, \ldots, F_q$ are convex.

Since multicriterion problems are vector optimization problems, all of the material of §4.7.1–§4.7.4 applies. For multicriterion problems, though, we can be a bit more specific in the interpretations. If x is feasible, we can think of $F_i(x)$ as its score or value, according to the ith objective. If x and y are both feasible, $F_i(x) \leq F_i(y)$ means that x is at least as good as y, according to the ith objective; $F_i(x) < F_i(y)$ means that x is better than y, or x beats y, according to the ith objective. If x and y are both feasible, we say that x is *better than* y, or x *dominates* y, if $F_i(x) \leq F_i(y)$ for $i = 1, \ldots, q$, and for at least one j, $F_j(x) < F_j(y)$. Roughly speaking, x is better than y if x meets or beats y on all objectives, and beats it in at least one objective.

In a multicriterion problem, an optimal point $x^\star$ satisfies

$$F_i(x^\star) \leq F_i(y), \quad i = 1, \ldots, q,$$

for every feasible y. In other words, $x^\star$ is simultaneously optimal for each of the scalar problems

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

for $j = 1, \ldots, q$. When there is an optimal point, we say that the objectives are *noncompeting*, since no compromises have to be made among the objectives; each objective is as small as it could be made, even if the others were ignored.

A Pareto optimal point x^{po} satisfies the following: if y is feasible and $F_i(y) \leq F_i(x^{\text{po}})$ for $i = 1, \ldots, q$, then $F_i(x^{\text{po}}) = F_i(y)$, $i = 1, \ldots, q$. This can be restated as: a point is Pareto optimal if and only if it is feasible and there is no better feasible point. In particular, if a feasible point is not Pareto optimal, there is at least one other feasible point that is better. In searching for good points, then, we can clearly limit our search to Pareto optimal points.

Trade-off analysis

Now suppose that x and y are Pareto optimal points with, say,

$$\begin{array}{ll} F_i(x) < F_i(y), & i \in A \\ F_i(x) = F_i(y), & i \in B \\ F_i(x) > F_i(y), & i \in C, \end{array}$$

where $A \cup B \cup C = \{1, \ldots, q\}$. In other words, A is the set of (indices of) objectives for which x beats y, B is the set of objectives for which the points x and y are tied, and C is the set of objectives for which y beats x. If A and C are empty, then the two points x and y have exactly the same objective values. If this is not the case, then both A and C must be nonempty. In other words, when comparing two Pareto optimal points, they either obtain the same performance (*i.e.*, all objectives equal), or, each beats the other in at least one objective.

In comparing the point x to y, we say that we have *traded* or *traded off* better objective values for $i \in A$ for worse objective values for $i \in C$. *Optimal trade-off analysis* (or just trade-off analysis) is the study of how much worse we must do in one or more objectives in order to do better in some other objectives, or more generally, the study of what sets of objective values are achievable.

As an example, consider a bi-criterion (*i.e.*, two criterion) problem. Suppose x is a Pareto optimal point, with objectives $F_1(x)$ and $F_2(x)$. We might ask how much larger $F_2(z)$ would have to be, in order to obtain a feasible point z with $F_1(z) \leq F_1(x) - a$, where $a > 0$ is some constant. Roughly speaking, we are asking how much we must pay in the second objective to obtain an improvement of a in the first objective. If a large increase in F_2 must be accepted to realize a small decrease in F_1, we say that there is a *strong trade-off* between the objectives, near the Pareto optimal value $(F_1(x), F_2(x))$. If, on the other hand, a large decrease in F_1 can be obtained with only a small increase in F_2, we say that the trade-off between the objectives is *weak* (near the Pareto optimal value $(F_1(x), F_2(x))$).

We can also consider the case in which we trade worse performance in the first objective for an improvement in the second. Here we find how much smaller $F_2(z)$

can be made, to obtain a feasible point z with $F_1(z) \leq F_1(x) + a$, where $a > 0$ is some constant. In this case we receive a benefit in the second objective, *i.e.*, a reduction in F_2 compared to $F_2(x)$. If this benefit is large (*i.e.*, by increasing F_1 a small amount we obtain a large reduction in F_2), we say the objectives exhibit a strong trade-off. If it is small, we say the objectives trade off weakly (near the Pareto optimal value $(F_1(x), F_2(x))$).

Optimal trade-off surface

The set of Pareto optimal values for a multicriterion problem is called the *optimal trade-off surface* (in general, when $q > 2$) or the *optimal trade-off curve* (when $q = 2$). (Since it would be foolish to accept any point that is not Pareto optimal, we can restrict our trade-off analysis to Pareto optimal points.) Trade-off analysis is also sometimes called *exploring the optimal trade-off surface*. (The optimal trade-off surface is usually, but not always, a surface in the usual sense. If the problem has an optimal point, for example, the optimal trade-off surface consists of a single point, the optimal value.)

An optimal trade-off curve is readily interpreted. An example is shown in figure 4.11, on page 185, for a (convex) bi-criterion problem. From this curve we can easily visualize and understand the trade-offs between the two objectives.

- The endpoint at the right shows the smallest possible value of F_2, without any consideration of F_1.
- The endpoint at the left shows the smallest possible value of F_1, without any consideration of F_2.
- By finding the intersection of the curve with a vertical line at $F_1 = \alpha$, we can see how large F_2 must be to achieve $F_1 \leq \alpha$.
- By finding the intersection of the curve with a horizontal line at $F_2 = \beta$, we can see how large F_1 must be to achieve $F_2 \leq \beta$.
- The slope of the optimal trade-off curve at a point on the curve (*i.e.*, a Pareto optimal value) shows the *local* optimal trade-off between the two objectives. Where the slope is steep, small changes in F_1 are accompanied by large changes in F_2.
- A point of large curvature is one where small decreases in one objective can only be accomplished by a large increase in the other. This is the proverbial *knee of the trade-off curve*, and in many applications represents a good compromise solution.

All of these have simple extensions to a trade-off surface, although visualizing a surface with more than three objectives is difficult.

Scalarizing multicriterion problems

When we scalarize a multicriterion problem by forming the weighted sum objective

$$\lambda^T f_0(x) = \sum_{i=1}^{q} \lambda_i F_i(x),$$

where $\lambda \succ 0$, we can interpret λ_i as the *weight* we attach to the ith objective. The weight λ_i can be thought of as quantifying our desire to make F_i small (or our objection to having F_i large). In particular, we should take λ_i large if we want F_i to be small; if we care much less about F_i, we can take λ_i small. We can interpret the ratio λ_i/λ_j as the *relative weight* or relative importance of the ith objective compared to the jth objective. Alternatively, we can think of λ_i/λ_j as *exchange rate* between the two objectives, since in the weighted sum objective a decrease (say) in F_i by α is considered the same as an increase in F_j in the amount $(\lambda_i/\lambda_j)\alpha$.

These interpretations give us some intuition about how to set or change the weights while exploring the optimal trade-off surface. Suppose, for example, that the weight vector $\lambda \succ 0$ yields the Pareto optimal point x^{po}, with objective values $F_1(x^{\mathrm{po}}), \ldots, F_q(x^{\mathrm{po}})$. To find a (possibly) new Pareto optimal point which trades off a better kth objective value (say), for (possibly) worse objective values for the other objectives, we form a new weight vector $\tilde{\lambda}$ with

$$\tilde{\lambda}_k > \lambda_k, \qquad \tilde{\lambda}_j = \lambda_j, \quad j \neq k, \quad j = 1, \ldots, q,$$

i.e., we increase the weight on the kth objective. This yields a new Pareto optimal point $\tilde{x}^{\mathrm{po}}$ with $F_k(\tilde{x}^{\mathrm{po}}) \leq F_k(x^{\mathrm{po}})$ (and usually, $F_k(\tilde{x}^{\mathrm{po}}) < F_k(x^{\mathrm{po}})$), *i.e.*, a new Pareto optimal point with an improved kth objective.

We can also see that at any point where the optimal trade-off surface is smooth, λ gives the inward normal to the surface at the associated Pareto optimal point. In particular, when we choose a weight vector λ and apply scalarization, we obtain a Pareto optimal point where λ gives the local trade-offs among objectives.

In practice, optimal trade-off surfaces are explored by ad hoc adjustment of the weights, based on the intuitive ideas above. We will see later (in chapter 5) that the basic idea of scalarization, *i.e.*, minimizing a weighted sum of objectives, and then adjusting the weights to obtain a suitable solution, is the essence of duality.

4.7.6 Examples

Regularized least-squares

We are given $A \in \mathbf{R}^{m \times n}$ and $b \in \mathbf{R}^m$, and want to choose $x \in \mathbf{R}^n$ taking into account two quadratic objectives:

- $F_1(x) = \|Ax - b\|_2^2 = x^T A^T A x - 2b^T A x + b^T b$ is a measure of the misfit between Ax and b,
- $F_2(x) = \|x\|_2^2 = x^T x$ is a measure of the size of x.

Our goal is to find x that gives a good fit (*i.e.*, small F_1) and that is not large (*i.e.*, small F_2). We can formulate this problem as a vector optimization problem with respect to the cone $\mathbf{R}^2_+$, *i.e.*, a bi-criterion problem (with no constraints):

$$\text{minimize (w.r.t. } \mathbf{R}^2_+) \quad f_0(x) = (F_1(x), F_2(x)).$$

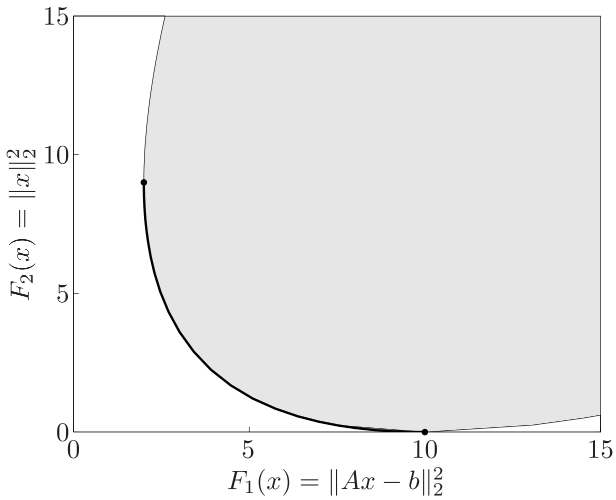


Figure 4.11 Optimal trade-off curve for a regularized least-squares problem. The shaded set is the set of achievable values $(\|Ax-b\|_2^2, \|x\|_2^2)$. The optimal trade-off curve, shown darker, is the lower left part of the boundary.

We can scalarize this problem by taking $\lambda_1 > 0$ and $\lambda_2 > 0$ and minimizing the scalar weighted sum objective

$$\begin{aligned} \lambda^T f_0(x) &= \lambda_1 F_1(x) + \lambda_2 F_2(x) \\ &= x^T(\lambda_1 A^T A + \lambda_2 I)x - 2\lambda_1 b^T Ax + \lambda_1 b^T b, \end{aligned}$$

which yields

$$x(\mu) = (\lambda_1 A^T A + \lambda_2 I)^{-1} \lambda_1 A^T b = (A^T A + \mu I)^{-1} A^T b,$$

where $\mu = \lambda_2/\lambda_1$. For any $\mu > 0$, this point is Pareto optimal for the bi-criterion problem. We can interpret $\mu = \lambda_2/\lambda_1$ as the relative weight we assign F_2 compared to F_1.

This method produces all Pareto optimal points, except two, associated with the extremes $\mu \to \infty$ and $\mu \to 0$. In the first case we have the Pareto optimal solution $x = 0$, which would be obtained by scalarization with $\lambda = (0, 1)$. At the other extreme we have the Pareto optimal solution $A^\dagger b$, where $A^\dagger$ is the pseudo-inverse of A. This Pareto optimal solution is obtained as the limit of the optimal solution of the scalarized problem as $\mu \to 0$, *i.e.*, as $\lambda \to (1, 0)$. (We will encounter the regularized least-squares problem again in §6.3.2.)

Figure 4.11 shows the optimal trade-off curve and the set of achievable values for a regularized least-squares problem with problem data $A \in \mathbf{R}^{100\times 10}$, $b \in \mathbf{R}^{100}$. (See exercise 4.50 for more discussion.)

Risk-return trade-off in portfolio optimization

The classical Markowitz portfolio optimization problem described on page 155 is naturally expressed as a bi-criterion problem, where the objectives are the negative mean return (since we wish to *maximize* mean return) and the variance of the return:

$$\begin{array}{ll} \text{minimize (w.r.t. } \mathbf{R}^2_+) & (F_1(x), F_2(x)) = (-\overline{p}^T x, x^T \Sigma x) \\ \text{subject to} & \mathbf{1}^T x = 1, \quad x \succeq 0. \end{array}$$

In forming the associated scalarized problem, we can (without loss of generality) take $\lambda_1 = 1$ and $\lambda_2 = \mu > 0$:

$$\begin{array}{ll} \text{minimize} & -\overline{p}^T x + \mu x^T \Sigma x \\ \text{subject to} & \mathbf{1}^T x = 1, \quad x \succeq 0, \end{array}$$

which is a QP. In this example too, we get all Pareto optimal portfolios except for the two limiting cases corresponding to $\mu \to 0$ and $\mu \to \infty$. Roughly speaking, in the first case we get a maximum mean return, without regard for return variance; in the second case we form a minimum variance return, without regard for mean return. Assuming that $\overline{p}_k > \overline{p}_i$ for $i \neq k$, *i.e.*, that asset k is the unique asset with maximum mean return, the portfolio allocation $x = e_k$ is the only one corresponding to $\mu \to 0$. (In other words, we concentrate the portfolio entirely in the asset that has maximum mean return.) In many portfolio problems asset n corresponds to a *risk-free* investment, with (deterministic) return $r_{\rm rf}$. Assuming that Σ, with its last row and column (which are zero) removed, is full rank, then the other extreme Pareto optimal portfolio is $x = e_n$, *i.e.*, the portfolio is concentrated entirely in the risk-free asset.

As a specific example, we consider a simple portfolio optimization problem with 4 assets, with price change mean and standard deviations given in the following table.

Asset	$\overline{p}_i$	$\Sigma_{ii}^{1/2}$
1	12%	20%
2	10%	10%
3	7%	5%
4	3%	0%

Asset 4 is a risk-free asset, with a (certain) 3% return. Assets 3, 2, and 1 have increasing mean returns, ranging from 7% to 12%, as well as increasing standard deviations, which range from 5% to 20%. The correlation coefficients between the assets are $\rho_{12} = 30\%$, $\rho_{13} = -40\%$, and $\rho_{23} = 0\%$.

Figure 4.12 shows the optimal trade-off curve for this portfolio optimization problem. The plot is given in the conventional way, with the horizontal axis showing standard deviation (*i.e.*, squareroot of variance) and the vertical axis showing expected return. The lower plot shows the optimal asset allocation vector x for each Pareto optimal point.

The results in this simple example agree with our intuition. For small risk, the optimal allocation consists mostly of the risk-free asset, with a mixture of the other assets in smaller quantities. Note that a mixture of asset 3 and asset 1, which are negatively correlated, gives some hedging, *i.e.*, lowers variance for a given level of mean return. At the other end of the trade-off curve, we see that aggressive growth portfolios (*i.e.*, those with large mean returns) concentrate the allocation in assets 1 and 2, the ones with the largest mean returns (and variances).

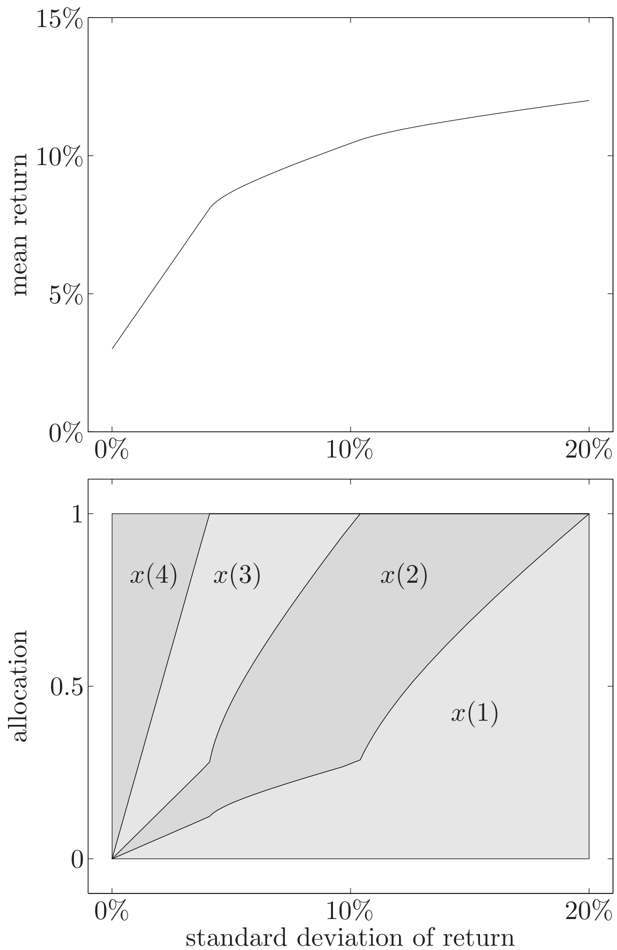


Figure 4.12 *Top.* Optimal risk-return trade-off curve for a simple portfolio optimization problem. The lefthand endpoint corresponds to putting all resources in the risk-free asset, and so has zero standard deviation. The righthand endpoint corresponds to putting all resources in asset 1, which has highest mean return. *Bottom.* Corresponding optimal allocations.

Bibliography

Linear programming has been studied extensively since the 1940s, and is the subject of many excellent books, including Dantzig [Dan63], Luenberger [Lue84], Schrijver [Sch86], Papadimitriou and Steiglitz [PS98], Bertsimas and Tsitsiklis [BT97], Vanderbei [Van96], and Roos, Terlaky, and Vial [RTV97]. Dantzig and Schrijver also provide detailed accounts of the history of linear programming. For a recent survey, see Todd [Tod02].

Schaible [Sch82, Sch83] gives an overview of fractional programming, which includes linear-fractional problems and extensions such as convex-concave fractional problems (see exercise 4.7). The model of a growing economy in example 4.7 appears in von Neumann [vN46].

Research on quadratic programming began in the 1950s (see, *e.g.*, Frank and Wolfe [FW56], Markowitz [Mar56], Hildreth [Hil57]), and was in part motivated by the portfolio optimization problem discussed on page 155 (Markowitz [Mar52]), and the LP with random cost discussed on page 154 (see Freund [Fre56]).

Interest in second-order cone programming is more recent, and started with Nesterov and Nemirovski [NN94, §6.2.3]. The theory and applications of SOCPs are surveyed by Alizadeh and Goldfarb [AG03], Ben-Tal and Nemirovski [BTN01, lecture 3] (where the problem is referred to as *conic quadratic programming*), and Lobo, Vandenberghe, Boyd, and Lebret [LVBL98].

Robust linear programming, and robust convex optimization in general, originated with Ben-Tal and Nemirovski [BTN98, BTN99] and El Ghaoui and Lebret [EL97]. Goldfarb and Iyengar [GI03a, GI03b] discuss robust QCQPs and applications in portfolio optimization. El Ghaoui, Oustry, and Lebret [EOL98] focus on robust semidefinite programming.

Geometric programming has been known since the 1960s. Its use in engineering design was first advocated by Duffin, Peterson, and Zener [DPZ67] and Zener [Zen71]. Peterson [Pet76] and Ecker [Eck80] describe the progress made during the 1970s. These articles and books also include examples of engineering applications, in particular in chemical and civil engineering. Fishburn and Dunlop [FD85], Sapatnekar, Rao, Vaidya, and Kang [SRVK93], and Hershenson, Boyd, and Lee [HBL01]) apply geometric programming to problems in integrated circuit design. The cantilever beam design example (page 163) is from Vanderplaats [Van84, page 147]. The variational characterization of the Perron-Frobenius eigenvalue (page 165) is proved in Berman and Plemmons [BP94, page 31].

Nesterov and Nemirovski [NN94, chapter 4] introduced the conic form problem (4.49) as a standard problem format in nonlinear convex optimization. The cone programming approach is further developed in Ben-Tal and Nemirovski [BTN01], who also describe numerous applications.

Alizadeh [Ali91] and Nesterov and Nemirovski [NN94, §6.4] were the first to make a systematic study of semidefinite programming, and to point out the wide variety of applications in convex optimization. Subsequent research in semidefinite programming during the 1990s was driven by applications in combinatorial optimization (Goemans and Williamson [GW95]), control (Boyd, El Ghaoui, Feron, and Balakrishnan [BEFB94], Scherer, Gahinet, and Chilali [SGC97], Dullerud and Paganini [DP00]), communications and signal processing (Luo [Luo03], Davidson, Luo, Wong, and Ma [DLW00, MDW$^+$02]), and other areas of engineering. The book edited by Wolkowicz, Saigal, and Vandenberghe [WSV00] and the articles by Todd [Tod01], Lewis and Overton [LO96], and Vandenberghe and Boyd [VB95] provide overviews and extensive bibliographies. Connections between SDP and moment problems, of which we give a simple example on page 170, are explored in detail by Bertsimas and Sethuraman [BS00], Nesterov [Nes00], and Lasserre [Las02]. The fastest mixing Markov chain problem is from Boyd, Diaconis, and Xiao [BDX04].

Multicriterion optimization and Pareto optimality are fundamental tools in economics; see Pareto [Par71], Debreu [Deb59] and Luenberger [Lue95]. The result in example 4.9 is known as the Gauss-Markov theorem (Kailath, Sayed, and Hassibi [KSH00, page 97]).

Exercises

Basic terminology and optimality conditions

4.1 Consider the optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x_1, x_2) \\ \text{subject to} & 2x_1 + x_2 \geq 1 \\ & x_1 + 3x_2 \geq 1 \\ & x_1 \geq 0, \quad x_2 \geq 0. \end{array}$$

Make a sketch of the feasible set. For each of the following objective functions, give the optimal set and the optimal value.

(a) $f_0(x_1, x_2) = x_1 + x_2$.

(b) $f_0(x_1, x_2) = -x_1 - x_2$.

(c) $f_0(x_1, x_2) = x_1$.

(d) $f_0(x_1, x_2) = \max\{x_1, x_2\}$.

(e) $f_0(x_1, x_2) = x_1^2 + 9x_2^2$.

4.2 Consider the optimization problem

$$\text{minimize} \quad f_0(x) = -\textstyle\sum_{i=1}^m \log(b_i - a_i^T x)$$

with domain $\mathbf{dom}\, f_0 = \{x \mid Ax \prec b\}$, where $A \in \mathbf{R}^{m \times n}$ (with rows a_i^T). We assume that $\mathbf{dom}\, f_0$ is nonempty.

Prove the following facts (which include the results quoted without proof on page 141).

(a) $\mathbf{dom}\, f_0$ is unbounded if and only if there exists a $v \neq 0$ with $Av \preceq 0$.

(b) f_0 is unbounded below if and only if there exists a v with $Av \preceq 0$, $Av \neq 0$. *Hint.* There exists a v such that $Av \preceq 0$, $Av \neq 0$ if and only if there exists no $z \succ 0$ such that $A^T z = 0$. This follows from the theorem of alternatives in example 2.21, page 50.

(c) If f_0 is bounded below then its minimum is attained, *i.e.*, there exists an x that satisfies the optimality condition (4.23).

(d) The optimal set is affine: $X_{\text{opt}} = \{x^\star + v \mid Av = 0\}$, where $x^\star$ is any optimal point.

4.3 Prove that $x^\star = (1, 1/2, -1)$ is optimal for the optimization problem

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & -1 \leq x_i \leq 1, \quad i = 1, 2, 3, \end{array}$$

where

$$P = \begin{bmatrix} 13 & 12 & -2 \\ 12 & 17 & 6 \\ -2 & 6 & 12 \end{bmatrix}, \qquad q = \begin{bmatrix} -22.0 \\ -14.5 \\ 13.0 \end{bmatrix}, \qquad r = 1.$$

4.4 [P. Parrilo] *Symmetries and convex optimization.* Suppose $\mathcal{G} = \{Q_1, \ldots, Q_k\} \subseteq \mathbf{R}^{n \times n}$ is a group, *i.e.*, closed under products and inverse. We say that the function $f : \mathbf{R}^n \to \mathbf{R}$ is $\mathcal{G}$-*invariant*, or *symmetric with respect to* $\mathcal{G}$, if $f(Q_i x) = f(x)$ holds for all x and $i = 1, \ldots, k$. We define $\overline{x} = (1/k)\sum_{i=1}^k Q_i x$, which is the average of x over its $\mathcal{G}$-orbit. We define the *fixed subspace* of $\mathcal{G}$ as

$$\mathcal{F} = \{x \mid Q_i x = x, \ i = 1, \ldots, k\}.$$

(a) Show that for any $x \in \mathbf{R}^n$, we have $\overline{x} \in \mathcal{F}$.

(b) Show that if $f : \mathbf{R}^n \to \mathbf{R}$ is convex and $\mathcal{G}$-invariant, then $f(\overline{x}) \leq f(x)$.

(c) We say the optimization problem

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

is $\mathcal{G}$-invariant if the objective f_0 is $\mathcal{G}$-invariant, and the feasible set is $\mathcal{G}$-invariant, which means

$$f_1(x) \leq 0, \ldots, f_m(x) \leq 0 \implies f_1(Q_i x) \leq 0, \ldots, f_m(Q_i x) \leq 0,$$

for $i = 1, \ldots, k$. Show that if the problem is convex and $\mathcal{G}$-invariant, and there exists an optimal point, then there exists an optimal point in $\mathcal{F}$. In other words, we can adjoin the equality constraints $x \in \mathcal{F}$ to the problem, without loss of generality.

(d) As an example, suppose f is convex and symmetric, *i.e.*, $f(Px) = f(x)$ for every permutation P. Show that if f has a minimizer, then it has a minimizer of the form $\alpha\mathbf{1}$. (This means to minimize f over $x \in \mathbf{R}^n$, we can just as well minimize $f(t\mathbf{1})$ over $t \in \mathbf{R}$.)

4.5 *Equivalent convex problems.* Show that the following three convex problems are equivalent. Carefully explain how the solution of each problem is obtained from the solution of the other problems. The problem data are the matrix $A \in \mathbf{R}^{m\times n}$ (with rows a_i^T), the vector $b \in \mathbf{R}^m$, and the constant $M > 0$.

(a) The *robust least-squares problem*

$$\text{minimize} \quad \sum_{i=1}^m \phi(a_i^T x - b_i),$$

with variable $x \in \mathbf{R}^n$, where $\phi : \mathbf{R} \to \mathbf{R}$ is defined as

$$\phi(u) = \begin{cases} u^2 & |u| \leq M \\ M(2|u| - M) & |u| > M. \end{cases}$$

(This function is known as the *Huber penalty function*; see §6.1.2.)

(b) The *least-squares problem with variable weights*

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^m (a_i^T x - b_i)^2/(w_i + 1) + M^2\mathbf{1}^T w \\ \text{subject to} & w \succeq 0, \end{array}$$

with variables $x \in \mathbf{R}^n$ and $w \in \mathbf{R}^m$, and domain $\mathcal{D} = \{(x, w) \in \mathbf{R}^n \times \mathbf{R}^m \mid w \succ -\mathbf{1}\}$. *Hint.* Optimize over w assuming x is fixed, to establish a relation with the problem in part (a).

(This problem can be interpreted as a weighted least-squares problem in which we are allowed to adjust the weight of the ith residual. The weight is one if $w_i = 0$, and decreases if we increase w_i. The second term in the objective penalizes large values of w, *i.e.*, large adjustments of the weights.)

(c) The *quadratic program*

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^m (u_i^2 + 2Mv_i) \\ \text{subject to} & -u - v \preceq Ax - b \preceq u + v \\ & 0 \preceq u \preceq M\mathbf{1} \\ & v \succeq 0. \end{array}$$

4.6 *Handling convex equality constraints.* A convex optimization problem can have only *linear* equality constraint functions. In some special cases, however, it is possible to handle convex equality constraint functions, *i.e.*, constraints of the form $h(x) = 0$, where h is convex. We explore this idea in this problem.

Consider the optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \dots, m \\ & h(x) = 0, \end{array} \tag{4.65}$$

where f_i and h are convex functions with domain $\mathbf{R}^n$. Unless h is affine, this is *not* a convex optimization problem. Consider the related problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \dots, m, \\ & h(x) \leq 0, \end{array} \tag{4.66}$$

where the convex equality constraint has been relaxed to a convex inequality. This problem is, of course, convex.

Now suppose we can guarantee that at any optimal solution $x^\star$ of the convex problem (4.66), we have $h(x^\star) = 0$, *i.e.*, the inequality $h(x) \leq 0$ is always active at the solution. Then we can solve the (nonconvex) problem (4.65) by solving the convex problem (4.66). Show that this is the case if there is an index r such that

- f_0 is monotonically increasing in x_r
- $f_1, \dots, f_m$ are nondecreasing in x_r
- h is monotonically decreasing in x_r.

We will see specific examples in exercises 4.31 and 4.58.

4.7 *Convex-concave fractional problems.* Consider a problem of the form

$$\begin{array}{ll} \text{minimize} & f_0(x)/(c^T x + d) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \dots, m \\ & Ax = b \end{array}$$

where $f_0, f_1, \dots, f_m$ are convex, and the domain of the objective function is defined as $\{x \in \mathbf{dom}\, f_0 \mid c^T x + d > 0\}$.

(a) Show that this is a quasiconvex optimization problem.

(b) Show that the problem is equivalent to

$$\begin{array}{ll} \text{minimize} & g_0(y, t) \\ \text{subject to} & g_i(y, t) \leq 0, \quad i = 1, \dots, m \\ & Ay = bt \\ & c^T y + dt = 1, \end{array}$$

where g_i is the perspective of f_i (see §3.2.6). The variables are $y \in \mathbf{R}^n$ and $t \in \mathbf{R}$. Show that this problem is convex.

(c) Following a similar argument, derive a convex formulation for the *convex-concave* fractional problem

$$\begin{array}{ll} \text{minimize} & f_0(x)/h(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \dots, m \\ & Ax = b \end{array}$$

where $f_0, f_1, \ldots, f_m$ are convex, h is concave, the domain of the objective function is defined as $\{x \in \mathbf{dom}\, f_0 \cap \mathbf{dom}\, h \mid h(x) > 0\}$ and $f_0(x) \geq 0$ everywhere.

As an example, apply your technique to the (unconstrained) problem with

$$f_0(x) = (\mathbf{tr}\, F(x))/m, \qquad h(x) = (\det(F(x))^{1/m},$$

with $\mathbf{dom}(f_0/h) = \{x \mid F(x) \succ 0\}$, where $F(x) = F_0 + x_1 F_1 + \cdots + x_n F_n$ for given $F_i \in \mathbf{S}^m$. In this problem, we minimize the ratio of the arithmetic mean over the geometric mean of the eigenvalues of an affine matrix function $F(x)$.

Linear optimization problems

4.8 *Some simple LPs.* Give an explicit solution of each of the following LPs.

(a) *Minimizing a linear function over an affine set.*

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

(b) *Minimizing a linear function over a halfspace.*

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

where $a \neq 0$.

(c) *Minimizing a linear function over a rectangle.*

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

where l and u satisfy $l \preceq u$.

(d) *Minimizing a linear function over the probability simplex.*

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{1}^T x = 1, \quad x \succeq 0. \end{array}$$

What happens if the equality constraint is replaced by an inequality $\mathbf{1}^T x \leq 1$?

We can interpret this LP as a simple portfolio optimization problem. The vector x represents the allocation of our total budget over different assets, with x_i the fraction invested in asset i. The return of each investment is fixed and given by $-c_i$, so our total return (which we want to maximize) is $-c^T x$. If we replace the budget constraint $\mathbf{1}^T x = 1$ with an inequality $\mathbf{1}^T x \leq 1$, we have the option of not investing a portion of the total budget.

(e) *Minimizing a linear function over a unit box with a total budget constraint.*

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{1}^T x = \alpha, \quad 0 \preceq x \preceq \mathbf{1}, \end{array}$$

where α is an integer between 0 and n. What happens if α is not an integer (but satisfies $0 \leq \alpha \leq n$)? What if we change the equality to an inequality $\mathbf{1}^T x \leq \alpha$?

(f) *Minimizing a linear function over a unit box with a weighted budget constraint.*

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & d^T x = \alpha, \quad 0 \preceq x \preceq \mathbf{1}, \end{array}$$

with $d \succ 0$, and $0 \leq \alpha \leq \mathbf{1}^T d$.

4.9 *Square LP.* Consider the LP

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

with A square and nonsingular. Show that the optimal value is given by

$$p^\star = \begin{cases} c^T A^{-1} b & A^{-T} c \preceq 0 \\ -\infty & \text{otherwise.} \end{cases}$$

4.10 *Converting general LP to standard form.* Work out the details on page 147 of §4.3. Explain in detail the relation between the feasible sets, the optimal solutions, and the optimal values of the standard form LP and the original LP.

4.11 *Problems involving ℓ_1- and ℓ_∞-norms.* Formulate the following problems as LPs. Explain in detail the relation between the optimal solution of each problem and the solution of its equivalent LP.

(a) Minimize $\|Ax - b\|_\infty$ (ℓ_∞-norm approximation).

(b) Minimize $\|Ax - b\|_1$ (ℓ_1-norm approximation).

(c) Minimize $\|Ax - b\|_1$ subject to $\|x\|_\infty \le 1$.

(d) Minimize $\|x\|_1$ subject to $\|Ax - b\|_\infty \le 1$.

(e) Minimize $\|Ax - b\|_1 + \|x\|_\infty$.

In each problem, $A \in \mathbf{R}^{m\times n}$ and $b \in \mathbf{R}^m$ are given. (See §6.1 for more problems involving approximation and constrained approximation.)

4.12 *Network flow problem.* Consider a network of n nodes, with directed links connecting each pair of nodes. The variables in the problem are the flows on each link: x_{ij} will denote the flow from node i to node j. The cost of the flow along the link from node i to node j is given by $c_{ij}x_{ij}$, where c_{ij} are given constants. The total cost across the network is

$$C = \sum_{i,j=1}^{n} c_{ij} x_{ij}.$$

Each link flow x_{ij} is also subject to a given lower bound l_{ij} (usually assumed to be nonnegative) and an upper bound u_{ij}.

The external supply at node i is given by b_i, where $b_i > 0$ means an external flow enters the network at node i, and $b_i < 0$ means that at node i, an amount $|b_i|$ flows out of the network. We assume that $\mathbf{1}^T b = 0$, *i.e.*, the total external supply equals total external demand. At each node we have conservation of flow: the total flow into node i along links and the external supply, minus the total flow out along the links, equals zero.

The problem is to minimize the total cost of flow through the network, subject to the constraints described above. Formulate this problem as an LP.

4.13 *Robust LP with interval coefficients.* Consider the problem, with variable $x \in \mathbf{R}^n$,

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

where $\mathcal{A} \subseteq \mathbf{R}^{m\times n}$ is the set

$$\mathcal{A} = \{A \in \mathbf{R}^{m\times n} \mid \bar{A}_{ij} - V_{ij} \le A_{ij} \le \bar{A}_{ij} + V_{ij},\ i = 1,\ldots,m,\ j = 1,\ldots,n\}.$$

(The matrices $\bar{A}$ and V are given.) This problem can be interpreted as an LP where each coefficient of A is only known to lie in an interval, and we require that x must satisfy the constraints for all possible values of the coefficients.

Express this problem as an LP. The LP you construct should be efficient, *i.e.*, it should not have dimensions that grow exponentially with n or m.

4.14 *Approximating a matrix in infinity norm.* The ℓ_∞-norm induced norm of a matrix $A \in \mathbf{R}^{m\times n}$, denoted $\|A\|_\infty$, is given by

$$\|A\|_\infty = \sup_{x\neq 0} \frac{\|Ax\|_\infty}{\|x\|_\infty} = \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}|.$$

This norm is sometimes called the max-row-sum norm, for obvious reasons (see §A.1.5).

Consider the problem of approximating a matrix, in the max-row-sum norm, by a linear combination of other matrices. That is, we are given $k+1$ matrices $A_0, \ldots, A_k \in \mathbf{R}^{m\times n}$, and need to find $x \in \mathbf{R}^k$ that minimizes

$$\|A_0 + x_1 A_1 + \cdots + x_k A_k\|_\infty.$$

Express this problem as a linear program. Explain the significance of any extra variables in your LP. Carefully explain how your LP formulation solves this problem, *e.g.*, what is the relation between the feasible set for your LP and this problem?

4.15 *Relaxation of Boolean LP.* In a *Boolean linear program*, the variable x is constrained to have components equal to zero or one:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \\ & x_i \in \{0,1\}, \quad i=1,\ldots,n. \end{array} \tag{4.67}$$

In general, such problems are very difficult to solve, even though the feasible set is finite (containing at most 2^n points).

In a general method called *relaxation*, the constraint that x_i be zero or one is replaced with the linear inequalities $0 \leq x_i \leq 1$:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \\ & 0 \leq x_i \leq 1, \quad i=1,\ldots,n. \end{array} \tag{4.68}$$

We refer to this problem as the *LP relaxation* of the Boolean LP (4.67). The LP relaxation is far easier to solve than the original Boolean LP.

(a) Show that the optimal value of the LP relaxation (4.68) is a lower bound on the optimal value of the Boolean LP (4.67). What can you say about the Boolean LP if the LP relaxation is infeasible?

(b) It sometimes happens that the LP relaxation has a solution with $x_i \in \{0,1\}$. What can you say in this case?

4.16 *Minimum fuel optimal control.* We consider a linear dynamical system with state $x(t) \in \mathbf{R}^n$, $t = 0, \ldots, N$, and actuator or input signal $u(t) \in \mathbf{R}$, for $t = 0, \ldots, N-1$. The dynamics of the system is given by the linear recurrence

$$x(t+1) = Ax(t) + bu(t), \quad t = 0, \ldots, N-1,$$

where $A \in \mathbf{R}^{n\times n}$ and $b \in \mathbf{R}^n$ are given. We assume that the initial state is zero, *i.e.*, $x(0) = 0$.

The *minimum fuel optimal control problem* is to choose the inputs $u(0), \ldots, u(N-1)$ so as to minimize the total fuel consumed, which is given by

$$F = \sum_{t=0}^{N-1} f(u(t)),$$

subject to the constraint that $x(N) = x_{\rm des}$, where N is the (given) time horizon, and $x_{\rm des} \in \mathbf{R}^n$ is the (given) desired final or target state. The function $f : \mathbf{R} \to \mathbf{R}$ is the *fuel use map* for the actuator, and gives the amount of fuel used as a function of the actuator signal amplitude. In this problem we use

$$f(a) = \begin{cases} |a| & |a| \leq 1 \\ 2|a| - 1 & |a| > 1. \end{cases}$$

This means that fuel use is proportional to the absolute value of the actuator signal, for actuator signals between -1 and 1; for larger actuator signals the marginal fuel efficiency is half.

Formulate the minimum fuel optimal control problem as an LP.

4.17 *Optimal activity levels.* We consider the selection of n nonnegative activity levels, denoted $x_1, \ldots, x_n$. These activities consume m resources, which are limited. Activity j consumes $A_{ij}x_j$ of resource i, where A_{ij} are given. The total resource consumption is additive, so the total of resource i consumed is $c_i = \sum_{j=1}^n A_{ij}x_j$. (Ordinarily we have $A_{ij} \geq 0$, *i.e.*, activity j consumes resource i. But we allow the possibility that $A_{ij} < 0$, which means that activity j actually *generates* resource i as a by-product.) Each resource consumption is limited: we must have $c_i \leq c_i^{\rm max}$, where $c_i^{\rm max}$ are given. Each activity generates revenue, which is a piecewise-linear concave function of the activity level:

$$r_j(x_j) = \begin{cases} p_j x_j & 0 \leq x_j \leq q_j \\ p_j q_j + p_j^{\rm disc}(x_j - q_j) & x_j \geq q_j. \end{cases}$$

Here $p_j > 0$ is the basic price, $q_j > 0$ is the quantity discount level, and $p_j^{\rm disc}$ is the quantity discount price, for (the product of) activity j. (We have $0 < p_j^{\rm disc} < p_j$.) The total revenue is the sum of the revenues associated with each activity, *i.e.*, $\sum_{j=1}^n r_j(x_j)$. The goal is to choose activity levels that maximize the total revenue while respecting the resource limits. Show how to formulate this problem as an LP.

4.18 *Separating hyperplanes and spheres.* Suppose you are given two sets of points in $\mathbf{R}^n$, $\{v^1, v^2, \ldots, v^K\}$ and $\{w^1, w^2, \ldots, w^L\}$. Formulate the following two problems as LP feasibility problems.

(a) Determine a hyperplane that separates the two sets, *i.e.*, find $a \in \mathbf{R}^n$ and $b \in \mathbf{R}$ with $a \neq 0$ such that

$$a^T v^i \leq b, \quad i = 1, \ldots, K, \qquad a^T w^i \geq b, \quad i = 1, \ldots, L.$$

Note that we require $a \neq 0$, so you have to make sure that your formulation excludes the trivial solution $a = 0$, $b = 0$. You can assume that

$$\mathbf{rank} \begin{bmatrix} v^1 & v^2 & \cdots & v^K & w^1 & w^2 & \cdots & w^L \\ 1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \end{bmatrix} = n + 1$$

(*i.e.*, the affine hull of the $K + L$ points has dimension n).

(b) Determine a sphere separating the two sets of points, *i.e.*, find $x_c \in \mathbf{R}^n$ and $R \geq 0$ such that

$$\|v^i - x_c\|_2 \leq R, \quad i = 1, \ldots, K, \qquad \|w^i - x_c\|_2 \geq R, \quad i = 1, \ldots, L.$$

(Here x_c is the center of the sphere; R is its radius.)

(See chapter 8 for more on separating hyperplanes, separating spheres, and related topics.)

4.19 Consider the problem

$$\begin{array}{ll} \text{minimize} & \|Ax-b\|_1/(c^Tx+d) \\ \text{subject to} & \|x\|_\infty \leq 1, \end{array}$$

where $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $c \in \mathbf{R}^n$, and $d \in \mathbf{R}$. We assume that $d > \|c\|_1$, which implies that $c^Tx + d > 0$ for all feasible x.

(a) Show that this is a quasiconvex optimization problem.

(b) Show that it is equivalent to the convex optimization problem

$$\begin{array}{ll} \text{minimize} & \|Ay-bt\|_1 \\ \text{subject to} & \|y\|_\infty \leq t \\ & c^Ty+dt = 1, \end{array}$$

with variables $y \in \mathbf{R}^n$, $t \in \mathbf{R}$.

4.20 *Power assignment in a wireless communication system.* We consider n transmitters with powers $p_1, \dots, p_n \geq 0$, transmitting to n receivers. These powers are the optimization variables in the problem. We let $G \in \mathbf{R}^{n\times n}$ denote the matrix of *path gains* from the transmitters to the receivers; $G_{ij} \geq 0$ is the path gain from transmitter j to receiver i. The *signal power* at receiver i is then $S_i = G_{ii}p_i$, and the *interference power* at receiver i is $I_i = \sum_{k\neq i} G_{ik}p_k$. The *signal to interference plus noise ratio*, denoted SINR, at receiver i, is given by $S_i/(I_i + \sigma_i)$, where $\sigma_i > 0$ is the (self-) noise power in receiver i. The objective in the problem is to maximize the minimum SINR ratio, over all receivers, *i.e.*, to maximize

$$\min_{i=1,\dots,n} \frac{S_i}{I_i+\sigma_i}.$$

There are a number of constraints on the powers that must be satisfied, in addition to the obvious one $p_i \geq 0$. The first is a maximum allowable power for each transmitter, *i.e.*, $p_i \leq P_i^{\max}$, where $P_i^{\max} > 0$ is given. In addition, the transmitters are partitioned into groups, with each group sharing the same power supply, so there is a total power constraint for each group of transmitter powers. More precisely, we have subsets $K_1, \dots, K_m$ of $\{1, \dots, n\}$ with $K_1 \cup \dots \cup K_m = \{1, \dots, n\}$, and $K_j \cap K_l = 0$ if $j \neq l$. For each group K_l, the total associated transmitter power cannot exceed $P_l^{\text{gp}} > 0$:

$$\sum_{k\in K_l} p_k \leq P_l^{\text{gp}}, \quad l = 1, \dots, m.$$

Finally, we have a limit $P_k^{\text{rc}} > 0$ on the total received power at each receiver:

$$\sum_{k=1}^n G_{ik}p_k \leq P_i^{\text{rc}}, \quad i = 1, \dots, n.$$

(This constraint reflects the fact that the receivers will saturate if the total received power is too large.)

Formulate the SINR maximization problem as a generalized linear-fractional program.

Quadratic optimization problems

4.21 *Some simple QCQPs.* Give an explicit solution of each of the following QCQPs.

(a) *Minimizing a linear function over an ellipsoid centered at the origin.*

$$\begin{array}{ll} \text{minimize} & c^Tx \\ \text{subject to} & x^TAx \leq 1, \end{array}$$

where $A \in \mathbf{S}^n_{++}$ and $c \neq 0$. What is the solution if the problem is not convex ($A \notin \mathbf{S}^n_+$)?

(b) *Minimizing a linear function over an ellipsoid.*

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & (x - x_c)^T A(x - x_c) \leq 1, \end{array}$$

where $A \in \mathbf{S}^n_{++}$ and $c \neq 0$.

(c) *Minimizing a quadratic form over an ellipsoid centered at the origin.*

$$\begin{array}{ll} \text{minimize} & x^T Bx \\ \text{subject to} & x^T Ax \leq 1, \end{array}$$

where $A \in \mathbf{S}^n_{++}$ and $B \in \mathbf{S}^n_{+}$. Also consider the nonconvex extension with $B \notin \mathbf{S}^n_{+}$. (See §B.1.)

4.22 Consider the QCQP

$$\begin{array}{ll} \text{minimize} & (1/2)x^T Px + q^T x + r \\ \text{subject to} & x^T x \leq 1, \end{array}$$

with $P \in \mathbf{S}^n_{++}$. Show that $x^\star = -(P + \lambda I)^{-1}q$ where $\lambda = \max\{0, \bar{\lambda}\}$ and $\bar{\lambda}$ is the largest solution of the nonlinear equation

$$q^T (P + \lambda I)^{-2} q = 1.$$

4.23 *ℓ_4-norm approximation via QCQP.* Formulate the ℓ_4-norm approximation problem

$$\text{minimize} \quad \|Ax - b\|_4 = (\textstyle\sum_{i=1}^m (a_i^T x - b_i)^4)^{1/4}$$

as a QCQP. The matrix $A \in \mathbf{R}^{m \times n}$ (with rows a_i^T) and the vector $b \in \mathbf{R}^m$ are given.

4.24 *Complex ℓ_1-, ℓ_2- and ℓ_∞-norm approximation.* Consider the problem

$$\text{minimize} \quad \|Ax - b\|_p,$$

where $A \in \mathbf{C}^{m \times n}$, $b \in \mathbf{C}^m$, and the variable is $x \in \mathbf{C}^n$. The complex ℓ_p-norm is defined by

$$\|y\|_p = \left(\sum_{i=1}^m |y_i|^p \right)^{1/p}$$

for $p \geq 1$, and $\|y\|_\infty = \max_{i=1,\ldots,m} |y_i|$. For $p = 1$, 2, and ∞, express the complex ℓ_p-norm approximation problem as a QCQP or SOCP with real variables and data.

4.25 *Linear separation of two sets of ellipsoids.* Suppose we are given $K + L$ ellipsoids

$$\mathcal{E}_i = \{P_i u + q_i \mid \|u\|_2 \leq 1\}, \quad i = 1, \ldots, K + L,$$

where $P_i \in \mathbf{S}^n$. We are interested in finding a hyperplane that strictly separates $\mathcal{E}_1, \ldots, \mathcal{E}_K$ from $\mathcal{E}_{K+1}, \ldots, \mathcal{E}_{K+L}$, *i.e.*, we want to compute $a \in \mathbf{R}^n$, $b \in \mathbf{R}$ such that

$$a^T x + b > 0 \text{ for } x \in \mathcal{E}_1 \cup \cdots \cup \mathcal{E}_K, \qquad a^T x + b < 0 \text{ for } x \in \mathcal{E}_{K+1} \cup \cdots \cup \mathcal{E}_{K+L},$$

or prove that no such hyperplane exists. Express this problem as an SOCP feasibility problem.

4.26 *Hyperbolic constraints as SOC constraints.* Verify that $x \in \mathbf{R}^n$, $y, z \in \mathbf{R}$ satisfy

$$x^T x \leq yz, \qquad y \geq 0, \qquad z \geq 0$$

if and only if

$$\left\| \begin{bmatrix} 2x \\ y - z \end{bmatrix} \right\|_2 \leq y + z, \qquad y \geq 0, \qquad z \geq 0.$$

Use this observation to cast the following problems as SOCPs.

(a) *Maximizing harmonic mean.*

$$\text{maximize} \quad \left(\sum_{i=1}^m 1/(a_i^T x - b_i)\right)^{-1},$$

with domain $\{x \mid Ax \succ b\}$, where a_i^T is the ith row of A.

(b) *Maximizing geometric mean.*

$$\text{maximize} \quad \left(\prod_{i=1}^m (a_i^T x - b_i)\right)^{1/m},$$

with domain $\{x \mid Ax \succeq b\}$, where a_i^T is the ith row of A.

4.27 *Matrix fractional minimization via SOCP.* Express the following problem as an SOCP:

$$\begin{array}{ll} \text{minimize} & (Ax+b)^T(I + B\,\mathbf{diag}(x)B^T)^{-1}(Ax+b) \\ \text{subject to} & x \succeq 0, \end{array}$$

with $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $B \in \mathbf{R}^{m\times n}$. The variable is $x \in \mathbf{R}^n$.
Hint. First show that the problem is equivalent to

$$\begin{array}{ll} \text{minimize} & v^T v + w^T \mathbf{diag}(x)^{-1} w \\ \text{subject to} & v + Bw = Ax + b \\ & x \succeq 0, \end{array}$$

with variables $v \in \mathbf{R}^m$, $w, x \in \mathbf{R}^n$. (If $x_i = 0$ we interpret w_i^2/x_i as zero if $w_i = 0$ and as ∞ otherwise.) Then use the results of exercise 4.26.

4.28 *Robust quadratic programming.* In §4.4.2 we discussed robust linear programming as an application of second-order cone programming. In this problem we consider a similar robust variation of the (convex) *quadratic* program

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & Ax \preceq b. \end{array}$$

For simplicity we assume that only the matrix P is subject to errors, and the other parameters (q, r, A, b) are exactly known. The robust quadratic program is defined as

$$\begin{array}{ll} \text{minimize} & \sup_{P\in\mathcal{E}}((1/2)x^T P x + q^T x + r) \\ \text{subject to} & Ax \preceq b \end{array}$$

where $\mathcal{E}$ is the set of possible matrices P.
For each of the following sets $\mathcal{E}$, express the robust QP as a convex problem. Be as specific as you can. If the problem can be expressed in a standard form (*e.g.*, QP, QCQP, SOCP, SDP), say so.

(a) A finite set of matrices: $\mathcal{E} = \{P_1, \dots, P_K\}$, where $P_i \in \mathbf{S}_+^n$, $i = 1, \dots, K$.

(b) A set specified by a nominal value $P_0 \in \mathbf{S}_+^n$ plus a bound on the eigenvalues of the deviation $P - P_0$:

$$\mathcal{E} = \{P \in \mathbf{S}^n \mid -\gamma I \preceq P - P_0 \preceq \gamma I\}$$

where $\gamma \in \mathbf{R}$ and $P_0 \in \mathbf{S}_+^n$,

(c) An ellipsoid of matrices:

$$\mathcal{E} = \left\{ P_0 + \sum_{i=1}^K P_i u_i \;\middle|\; \|u\|_2 \le 1 \right\}.$$

You can assume $P_i \in \mathbf{S}_+^n$, $i = 0, \dots, K$.

4.29 *Maximizing probability of satisfying a linear inequality.* Let c be a random variable in $\mathbf{R}^n$, normally distributed with mean $\bar{c}$ and covariance matrix R. Consider the problem

$$\begin{array}{ll} \text{maximize} & \mathbf{prob}(c^T x \geq \alpha) \\ \text{subject to} & Fx \preceq g, \quad Ax = b. \end{array}$$

Assuming there exists a feasible point $\tilde{x}$ for which $\bar{c}^T \tilde{x} \geq \alpha$, show that this problem is equivalent to a convex or quasiconvex optimization problem. Formulate the problem as a QP, QCQP, or SOCP (if the problem is convex), or explain how you can solve it by solving a sequence of QP, QCQP, or SOCP feasibility problems (if the problem is quasiconvex).

Geometric programming

4.30 A heated fluid at temperature T (degrees above ambient temperature) flows in a pipe with fixed length and circular cross section with radius r. A layer of insulation, with thickness $w \ll r$, surrounds the pipe to reduce heat loss through the pipe walls. The design variables in this problem are T, r, and w.

The heat loss is (approximately) proportional to Tr/w, so over a fixed lifetime, the energy cost due to heat loss is given by $\alpha_1 Tr/w$. The cost of the pipe, which has a fixed wall thickness, is approximately proportional to the total material, *i.e.*, it is given by $\alpha_2 r$. The cost of the insulation is also approximately proportional to the total insulation material, *i.e.*, $\alpha_3 rw$ (using $w \ll r$). The total cost is the sum of these three costs.

The heat flow down the pipe is entirely due to the flow of the fluid, which has a fixed velocity, *i.e.*, it is given by $\alpha_4 Tr^2$. The constants α_i are all positive, as are the variables T, r, and w.

Now the problem: maximize the total heat flow down the pipe, subject to an upper limit $C_{\max}$ on total cost, and the constraints

$$T_{\min} \leq T \leq T_{\max}, \qquad r_{\min} \leq r \leq r_{\max}, \qquad w_{\min} \leq w \leq w_{\max}, \quad w \leq 0.1r.$$

Express this problem as a geometric program.

4.31 *Recursive formulation of optimal beam design problem.* Show that the GP (4.46) is equivalent to the GP

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^N w_i h_i \\ \text{subject to} & w_i/w_{\max} \leq 1, \quad w_{\min}/w_i \leq 1, \quad i = 1, \ldots, N \\ & h_i/h_{\max} \leq 1, \quad h_{\min}/h_i \leq 1, \quad i = 1, \ldots, N \\ & h_i/(w_i S_{\max}) \leq 1, \quad S_{\min} w_i/h_i \leq 1, \quad i = 1, \ldots, N \\ & 6iF/(\sigma_{\max} w_i h_i^2) \leq 1, \quad i = 1, \ldots, N \\ & (2i-1)d_i/v_i + v_{i+1}/v_i \leq 1, \quad i = 1, \ldots, N \\ & (i-1/3)d_i/y_i + v_{i+1}/y_i + y_{i+1}/y_i \leq 1, \quad i = 1, \ldots, N \\ & y_1/y_{\max} \leq 1 \\ & E w_i h_i^3 d_i/(6F) = 1, \quad i = 1, \ldots, N. \end{array}$$

The variables are w_i, h_i, v_i, d_i, y_i for $i = 1, \ldots, N$.

4.32 *Approximating a function as a monomial.* Suppose the function $f : \mathbf{R}^n \to \mathbf{R}$ is differentiable at a point $x_0 \succ 0$, with $f(x_0) > 0$. How would you find a monomial function $\hat{f} : \mathbf{R}^n \to \mathbf{R}$ such that $f(x_0) = \hat{f}(x_0)$ and for x near x_0, $\hat{f}(x)$ is very near $f(x)$?

4.33 Express the following problems as convex optimization problems.

(a) Minimize $\max\{p(x), q(x)\}$, where p and q are posynomials.

(b) Minimize $\exp(p(x)) + \exp(q(x))$, where p and q are posynomials.

(c) Minimize $p(x)/(r(x) - q(x))$, subject to $r(x) > q(x)$, where p, q are posynomials, and r is a monomial.

4.34 *Log-convexity of Perron-Frobenius eigenvalue.* Let $A \in \mathbf{R}^{n\times n}$ be an elementwise positive matrix, *i.e.*, $A_{ij} > 0$. (The results of this problem hold for irreducible nonnegative matrices as well.) Let $\lambda_{\rm pf}(A)$ denotes its Perron-Frobenius eigenvalue, *i.e.*, its eigenvalue of largest magnitude. (See the definition and the example on page 165.) Show that $\log \lambda_{\rm pf}(A)$ is a convex function of $\log A_{ij}$. This means, for example, that we have the inequality

$$\lambda_{\rm pf}(C) \leq (\lambda_{\rm pf}(A)\lambda_{\rm pf}(B))^{1/2},$$

where $C_{ij} = (A_{ij}B_{ij})^{1/2}$, and A and B are elementwise positive matrices.

Hint. Use the characterization of the Perron-Frobenius eigenvalue given in (4.47), or, alternatively, use the characterization

$$\log \lambda_{\rm pf}(A) = \lim_{k\to\infty} (1/k)\log(\mathbf{1}^T A^k \mathbf{1}).$$

4.35 *Signomial and geometric programs.* A *signomial* is a linear combination of monomials of some positive variables $x_1, \ldots, x_n$. Signomials are more general than posynomials, which are signomials with all positive coefficients. A *signomial program* is an optimization problem of the form

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

where $f_0, \ldots, f_m$ and $h_1, \ldots, h_p$ are signomials. In general, signomial programs are very difficult to solve.

Some signomial programs can be transformed to GPs, and therefore solved efficiently. Show how to do this for a signomial program of the following form:

- The objective signomial f_0 is a posynomial, *i.e.*, its terms have only positive coefficients.
- Each inequality constraint signomial $f_1, \ldots, f_m$ has exactly one term with a negative coefficient: $f_i = p_i - q_i$ where p_i is posynomial, and q_i is monomial.
- Each equality constraint signomial $h_1, \ldots, h_p$ has exactly one term with a positive coefficient and one term with a negative coefficient: $h_i = r_i - s_i$ where r_i and s_i are monomials.

4.36 Explain how to reformulate a general GP as an equivalent GP in which every posynomial (in the objective and constraints) has at most two monomial terms. *Hint.* Express each sum (of monomials) as a sum of sums, each with two terms.

4.37 *Generalized posynomials and geometric programming.* Let $x_1, \ldots, x_n$ be positive variables, and suppose the functions $f_i : \mathbf{R}^n \to \mathbf{R}$, $i = 1, \ldots, k$, are posynomials of $x_1, \ldots, x_n$. If $\phi : \mathbf{R}^k \to \mathbf{R}$ is a polynomial with nonnegative coefficients, then the composition

$$h(x) = \phi(f_1(x), \ldots, f_k(x)) \tag{4.69}$$

is a posynomial, since posynomials are closed under products, sums, and multiplication by nonnegative scalars. For example, suppose f_1 and f_2 are posynomials, and consider the polynomial $\phi(z_1, z_2) = 3z_1^2 z_2 + 2z_1 + 3z_2^3$ (which has nonnegative coefficients). Then $h = 3f_1^2 f_2 + 2f_1 + f_2^3$ is a posynomial.

In this problem we consider a generalization of this idea, in which ϕ is allowed to be a posynomial, *i.e.*, can have fractional exponents. Specifically, assume that $\phi : \mathbf{R}^k \to \mathbf{R}$ is a posynomial, with all its exponents nonnegative. In this case we will call the function h defined in (4.69) a *generalized posynomial.* As an example, suppose f_1 and f_2 are posynomials, and consider the posynomial (with nonnegative exponents) $\phi(z_1, z_2) = 2z_1^{0.3} z_2^{1.2} + z_1 z_2^{0.5} + 2$. Then the function

$$h(x) = 2f_1(x)^{0.3} f_2(x)^{1.2} + f_1(x) f_2(x)^{0.5} + 2$$

is a generalized posynomial. Note that it is *not* a posynomial, however (unless f_1 and f_2 are monomials or constants).

A *generalized geometric program* (GGP) is an optimization problem of the form

$$\begin{array}{ll} \text{minimize} & h_0(x) \\ \text{subject to} & h_i(x) \leq 1, \quad i = 1, \dots, m \\ & g_i(x) = 1, \quad i = 1, \dots, p, \end{array} \tag{4.70}$$

where $g_1, \dots, g_p$ are monomials, and $h_0, \dots, h_m$ are generalized posynomials.

Show how to express this generalized geometric program as an equivalent geometric program. Explain any new variables you introduce, and explain how your GP is equivalent to the GGP (4.70).

Semidefinite programming and conic form problems

4.38 *LMIs and SDPs with one variable.* The *generalized eigenvalues* of a matrix pair (A, B), where $A, B \in \mathbf{S}^n$, are defined as the roots of the polynomial $\det(\lambda B - A)$ (see §A.5.3).

Suppose B is nonsingular, and that A and B can be simultaneously diagonalized by a congruence, *i.e.*, there exists a nonsingular $R \in \mathbf{R}^{n \times n}$ such that

$$R^T A R = \mathbf{diag}(a), \qquad R^T B R = \mathbf{diag}(b),$$

where $a, b \in \mathbf{R}^n$. (A sufficient condition for this to hold is that there exists t_1, t_2 such that $t_1 A + t_2 B \succ 0$.)

(a) Show that the generalized eigenvalues of (A, B) are real, and given by $\lambda_i = a_i / b_i$, $i = 1, \dots, n$.

(b) Express the solution of the SDP

$$\begin{array}{ll} \text{minimize} & ct \\ \text{subject to} & tB \preceq A, \end{array}$$

with variable $t \in \mathbf{R}$, in terms of a and b.

4.39 *SDPs and congruence transformations.* Consider the SDP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x_1 F_1 + x_2 F_2 + \cdots + x_n F_n + G \preceq 0, \end{array}$$

with $F_i, G \in \mathbf{S}^k$, $c \in \mathbf{R}^n$.

(a) Suppose $R \in \mathbf{R}^{k \times k}$ is nonsingular. Show that the SDP is equivalent to the SDP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x_1 \tilde{F}_1 + x_2 \tilde{F}_2 + \cdots + x_n \tilde{F}_n + \tilde{G} \preceq 0, \end{array}$$

where $\tilde{F}_i = R^T F_i R$, $\tilde{G} = R^T G R$.

(b) Suppose there exists a nonsingular R such that $\tilde{F}_i$ and $\tilde{G}$ are diagonal. Show that the SDP is equivalent to an LP.

(c) Suppose there exists a nonsingular R such that $\tilde{F}_i$ and $\tilde{G}$ have the form

$$\tilde{F}_i = \begin{bmatrix} \alpha_i I & a_i \\ a_i^T & \alpha_i \end{bmatrix}, \quad i = 1, \dots, n, \qquad \tilde{G} = \begin{bmatrix} \beta I & b \\ b^T & \beta \end{bmatrix},$$

where $\alpha_i, \beta \in \mathbf{R}$, $a_i, b \in \mathbf{R}^{k-1}$. Show that the SDP is equivalent to an SOCP with a single second-order cone constraint.

4.40 *LPs, QPs, QCQPs, and SOCPs as SDPs.* Express the following problems as SDPs.

(a) The LP (4.27).

(b) The QP (4.34), the QCQP (4.35) and the SOCP (4.36). *Hint.* Suppose $A \in \mathbf{S}^r_{++}$, $C \in \mathbf{S}^s$, and $B \in \mathbf{R}^{r \times s}$. Then

$$\begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \succeq 0 \iff C - B^T A^{-1} B \succeq 0.$$

For a more complete statement, which applies also to singular A, and a proof, see §A.5.5.

(c) The matrix fractional optimization problem

$$\text{minimize} \quad (Ax+b)^T F(x)^{-1} (Ax+b)$$

where $A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$,

$$F(x) = F_0 + x_1 F_1 + \cdots + x_n F_n,$$

with $F_i \in \mathbf{S}^m$, and we take the domain of the objective to be $\{x \mid F(x) \succ 0\}$. You can assume the problem is feasible (there exists at least one x with $F(x) \succ 0$).

4.41 *LMI tests for copositive matrices and P_0-matrices.* A matrix $A \in \mathbf{S}^n$ is said to be *copositive* if $x^T A x \geq 0$ for all $x \succeq 0$ (see exercise 2.35). A matrix $A \in \mathbf{R}^{n \times n}$ is said to be a P_0-*matrix* if $\max_{i=1,\ldots,n} x_i (Ax)_i \geq 0$ for all x. Checking whether a matrix is copositive or a P_0-matrix is very difficult in general. However, there exist useful sufficient conditions that can be verified using semidefinite programming.

(a) Show that A is copositive if it can be decomposed as a sum of a positive semidefinite and an elementwise nonnegative matrix:

$$A = B + C, \qquad B \succeq 0, \qquad C_{ij} \geq 0, \quad i,j = 1, \ldots, n. \tag{4.71}$$

Express the problem of finding B and C that satisfy (4.71) as an SDP feasibility problem.

(b) Show that A is a P_0-matrix if there exists a positive diagonal matrix D such that

$$DA + A^T D \succeq 0. \tag{4.72}$$

Express the problem of finding a D that satisfies (4.72) as an SDP feasibility problem.

4.42 *Complex LMIs and SDPs.* A complex LMI has the form

$$x_1 F_1 + \cdots + x_n F_n + G \preceq 0$$

where $F_1, \ldots, F_n$, G are complex $n \times n$ Hermitian matrices, *i.e.*, $F_i^H = F_i$, $G^H = G$, and $x \in \mathbf{R}^n$ is a real variable. A complex SDP is the problem of minimizing a (real) linear function of x subject to a complex LMI constraint.

Complex LMIs and SDPs can be transformed to real LMIs and SDPs, using the fact that

$$X \succeq 0 \iff \begin{bmatrix} \Re X & -\Im X \\ \Im X & \Re X \end{bmatrix} \succeq 0,$$

where $\Re X \in \mathbf{R}^{n \times n}$ is the real part of the complex Hermitian matrix X, and $\Im X \in \mathbf{R}^{n \times n}$ is the imaginary part of X.

Verify this result, and show how to pose a complex SDP as a real SDP.

4.43 *Eigenvalue optimization via SDP.* Suppose $A : \mathbf{R}^n \to \mathbf{S}^m$ is affine, *i.e.*,

$$A(x) = A_0 + x_1 A_1 + \cdots + x_n A_n$$

where $A_i \in \mathbf{S}^m$. Let $\lambda_1(x) \geq \lambda_2(x) \geq \cdots \geq \lambda_m(x)$ denote the eigenvalues of $A(x)$. Show how to pose the following problems as SDPs.

(a) Minimize the maximum eigenvalue $\lambda_1(x)$.

(b) Minimize the spread of the eigenvalues, $\lambda_1(x) - \lambda_m(x)$.

(c) Minimize the condition number of $A(x)$, subject to $A(x) \succ 0$. The condition number is defined as $\kappa(A(x)) = \lambda_1(x)/\lambda_m(x)$, with domain $\{x \mid A(x) \succ 0\}$. You may assume that $A(x) \succ 0$ for at least one x.
Hint. You need to minimize λ/γ, subject to

$$0 \prec \gamma I \preceq A(x) \preceq \lambda I.$$

Change variables to $y = x/\gamma$, $t = \lambda/\gamma$, $s = 1/\gamma$.

(d) Minimize the sum of the absolute values of the eigenvalues, $|\lambda_1(x)| + \cdots + |\lambda_m(x)|$.
Hint. Express $A(x)$ as $A(x) = A_+ - A_-$, where $A_+ \succeq 0$, $A_- \succeq 0$.

4.44 *Optimization over polynomials.* Pose the following problem as an SDP. Find the polynomial $p : \mathbf{R} \to \mathbf{R}$,

$$p(t) = x_1 + x_2 t + \cdots + x_{2k+1} t^{2k},$$

that satisfies given bounds $l_i \leq p(t_i) \leq u_i$, at m specified points t_i, and, of all the polynomials that satisfy these bounds, has the greatest minimum value:

$$\begin{array}{ll} \text{maximize} & \inf_t p(t) \\ \text{subject to} & l_i \leq p(t_i) \leq u_i, \quad i = 1, \ldots, m. \end{array}$$

The variables are $x \in \mathbf{R}^{2k+1}$.

Hint. Use the LMI characterization of nonnegative polynomials derived in exercise 2.37, part (b).

4.45 [Nes00, Par00] *Sum-of-squares representation via LMIs.* Consider a polynomial $p : \mathbf{R}^n \to \mathbf{R}$ of degree $2k$. The polynomial is said to be positive semidefinite (PSD) if $p(x) \geq 0$ for all $x \in \mathbf{R}^n$. Except for special cases (*e.g.*, $n = 1$ or $k = 1$), it is extremely difficult to determine whether or not a given polynomial is PSD, let alone solve an optimization problem, with the coefficients of p as variables, with the constraint that p be PSD.

A famous sufficient condition for a polynomial to be PSD is that it have the form

$$p(x) = \sum_{i=1}^{r} q_i(x)^2,$$

for some polynomials q_i, with degree no more than k. A polynomial p that has this sum-of-squares form is called SOS.

The condition that a polynomial p be SOS (viewed as a constraint on its coefficients) turns out to be equivalent to an LMI, and therefore a variety of optimization problems, with SOS constraints, can be posed as SDPs. You will explore these ideas in this problem.

(a) Let $f_1, \ldots, f_s$ be all monomials of degree k or less. (Here we mean monomial in the standard sense, *i.e.*, $x_1^{m_1} \cdots x_n^{m_n}$, where $m_i \in \mathbf{Z}_+$, and not in the sense used in geometric programming.) Show that if p can be expressed as a positive semidefinite quadratic form $p = f^T V f$, with $V \in \mathbf{S}_+^s$, then p is SOS. Conversely, show that if p is SOS, then it can be expressed as a positive semidefinite quadratic form in the monomials, *i.e.*, $p = f^T V f$, for some $V \in \mathbf{S}_+^s$.

(b) Show that the condition $p = f^T V f$ is a set of linear equality constraints relating the coefficients of p and the matrix V. Combined with part (a) above, this shows that the condition that p be SOS is equivalent to a set of linear equalities relating V and the coefficients of p, and the matrix inequality $V \succeq 0$.

(c) Work out the LMI conditions for SOS explicitly for the case where p is polynomial of degree four in two variables.

4.46 *Multidimensional moments.* The moments of a random variable t on $\mathbf{R}^2$ are defined as $\mu_{ij} = \mathbf{E}\, t_1^i t_2^j$, where i, j are nonnegative integers. In this problem we derive necessary conditions for a set of numbers μ_{ij}, $0 \leq i, j \leq 2k$, $i + j \leq 2k$, to be the moments of a distribution on $\mathbf{R}^2$.

Let $p : \mathbf{R}^2 \to \mathbf{R}$ be a polynomial of degree k with coefficients c_{ij},

$$p(t) = \sum_{i=0}^{k} \sum_{j=0}^{k-i} c_{ij} t_1^i t_2^j,$$

and let t be a random variable with moments μ_{ij}. Suppose $c \in \mathbf{R}^{(k+1)(k+2)/2}$ contains the coefficients c_{ij} in some specific order, and $\mu \in \mathbf{R}^{(k+1)(2k+1)}$ contains the moments μ_{ij} in the same order. Show that $\mathbf{E}\, p(t)^2$ can be expressed as a quadratic form in c:

$$\mathbf{E}\, p(t)^2 = c^T H(\mu) c,$$

where $H : \mathbf{R}^{(k+1)(2k+1)} \to \mathbf{S}^{(k+1)(k+2)/2}$ is a linear function of μ. From this, conclude that μ must satisfy the LMI $H(\mu) \succeq 0$.

Remark: For random variables on $\mathbf{R}$, the matrix H can be taken as the Hankel matrix defined in (4.52). In this case, $H(\mu) \succeq 0$ is a necessary and sufficient condition for μ to be the moments of a distribution, or the limit of a sequence of moments. On $\mathbf{R}^2$, however, the LMI is only a necessary condition.

4.47 *Maximum determinant positive semidefinite matrix completion.* We consider a matrix $A \in \mathbf{S}^n$, with some entries specified, and the others not specified. The *positive semidefinite matrix completion problem* is to determine values of the unspecified entries of the matrix so that $A \succeq 0$ (or to determine that such a completion does not exist).

(a) Explain why we can assume without loss of generality that the diagonal entries of A are specified.

(b) Show how to formulate the positive semidefinite completion problem as an SDP feasibility problem.

(c) Assume that A has at least one completion that is positive definite, and the diagonal entries of A are specified (*i.e.*, fixed). The positive definite completion with largest determinant is called the *maximum determinant completion*. Show that the maximum determinant completion is unique. Show that if $A^\star$ is the maximum determinant completion, then $(A^\star)^{-1}$ has zeros in all the entries of the original matrix that were not specified. *Hint.* The gradient of the function $f(X) = \log\det X$ is $\nabla f(X) = X^{-1}$ (see §A.4.1).

(d) Suppose A is specified on its tridiagonal part, *i.e.*, we are given $A_{11}, \ldots, A_{nn}$ and $A_{12}, \ldots, A_{n-1,n}$. Show that if there exists a positive definite completion of A, then there is a positive definite completion whose inverse is tridiagonal.

4.48 *Generalized eigenvalue minimization.* Recall (from example 3.37, or §A.5.3) that the largest generalized eigenvalue of a pair of matrices $(A, B) \in \mathbf{S}^k \times \mathbf{S}^k_{++}$ is given by

$$\lambda_{\max}(A, B) = \sup_{u \neq 0} \frac{u^T A u}{u^T B u} = \max\{\lambda \mid \det(\lambda B - A) = 0\}.$$

As we have seen, this function is quasiconvex (if we take $\mathbf{S}^k \times \mathbf{S}^k_{++}$ as its domain).

We consider the problem

$$\text{minimize} \quad \lambda_{\max}(A(x), B(x)) \tag{4.73}$$

where $A, B : \mathbf{R}^n \to \mathbf{S}^k$ are affine functions, defined as

$$A(x) = A_0 + x_1 A_1 + \cdots + x_n A_n, \qquad B(x) = B_0 + x_1 B_1 + \cdots + x_n B_n.$$

with $A_i, B_i \in \mathbf{S}^k$.

(a) Give a family of convex functions $\phi_t : \mathbf{S}^k \times \mathbf{S}^k \to \mathbf{R}$, that satisfy

$$\lambda_{\max}(A, B) \leq t \iff \phi_t(A, B) \leq 0$$

for all $(A, B) \in \mathbf{S}^k \times \mathbf{S}^k_{++}$. Show that this allows us to solve (4.73) by solving a sequence of convex feasibility problems.

(b) Give a family of matrix-convex functions $\Phi_t : \mathbf{S}^k \times \mathbf{S}^k \to \mathbf{S}^k$ that satisfy

$$\lambda_{\max}(A, B) \leq t \iff \Phi_t(A, B) \preceq 0$$

for all $(A, B) \in \mathbf{S}^k \times \mathbf{S}^k_{++}$. Show that this allows us to solve (4.73) by solving a sequence of convex feasibility problems with LMI constraints.

(c) Suppose $B(x) = (a^T x + b)I$, with $a \neq 0$. Show that (4.73) is equivalent to the convex problem

$$\begin{array}{ll} \text{minimize} & \lambda_{\max}(sA_0 + y_1 A_1 + \cdots + y_n A_n) \\ \text{subject to} & a^T y + bs = 1 \\ & s \geq 0, \end{array}$$

with variables $y \in \mathbf{R}^n$, $s \in \mathbf{R}$.

4.49 *Generalized fractional programming.* Let $K \in \mathbf{R}^m$ be a proper cone. Show that the function $f_0 : \mathbf{R}^n \to \mathbf{R}^m$, defined by

$$f_0(x) = \inf\{t \mid Cx + d \preceq_K t(Fx + g)\}, \qquad \mathbf{dom}\, f_0 = \{x \mid Fx + g \succ_K 0\},$$

with $C, F \in \mathbf{R}^{m \times n}$, $d, g \in \mathbf{R}^m$, is quasiconvex.

A quasiconvex optimization problem with objective function of this form is called a *generalized fractional program.* Express the generalized linear-fractional program of page 152 and the generalized eigenvalue minimization problem (4.73) as generalized fractional programs.

Vector and multicriterion optimization

4.50 *Bi-criterion optimization.* Figure 4.11 shows the optimal trade-off curve and the set of achievable values for the bi-criterion optimization problem

$$\text{minimize (w.r.t. } \mathbf{R}^2_+) \quad (\|Ax - b\|^2, \|x\|_2^2),$$

for some $A \in \mathbf{R}^{100 \times 10}$, $b \in \mathbf{R}^{100}$. Answer the following questions using information from the plot. We denote by x_{ls} the solution of the least-squares problem

$$\text{minimize} \quad \|Ax - b\|_2^2.$$

(a) What is $\|x_{\text{ls}}\|_2$?

(b) What is $\|Ax_{\text{ls}} - b\|_2$?

(c) What is $\|b\|_2$?

(d) Give the optimal value of the problem

$$\begin{array}{ll} \text{minimize} & \|Ax - b\|_2^2 \\ \text{subject to} & \|x\|_2^2 = 1. \end{array}$$

(e) Give the optimal value of the problem

$$\begin{array}{ll} \text{minimize} & \|Ax - b\|_2^2 \\ \text{subject to} & \|x\|_2^2 \le 1. \end{array}$$

(f) Give the optimal value of the problem

$$\text{minimize } \|Ax - b\|_2^2 + \|x\|_2^2.$$

(g) What is the rank of A?

4.51 *Monotone transformation of objective in vector optimization.* Consider the vector optimization problem (4.56). Suppose we form a new vector optimization problem by replacing the objective f_0 with $\phi \circ f_0$, where $\phi : \mathbf{R}^q \to \mathbf{R}^q$ satisfies

$$u \preceq_K v, \; u \neq v \Longrightarrow \phi(u) \preceq_K \phi(v), \; \phi(u) \neq \phi(v).$$

Show that a point x is Pareto optimal (or optimal) for one problem if and only if it is Pareto optimal (optimal) for the other, so the two problems are equivalent. In particular, composing each objective in a multicriterion problem with an increasing function does not affect the Pareto optimal points.

4.52 *Pareto optimal points and the boundary of the set of achievable values.* Consider a vector optimization problem with cone K. Let $\mathcal{P}$ denote the set of Pareto optimal values, and let $\mathcal{O}$ denote the set of achievable objective values. Show that $\mathcal{P} \subseteq \mathcal{O} \cap \mathbf{bd}\,\mathcal{O}$, *i.e.*, every Pareto optimal value is an achievable objective value that lies in the boundary of the set of achievable objective values.

4.53 Suppose the vector optimization problem (4.56) is convex. Show that the set

$$\mathcal{A} = \mathcal{O} + K = \{t \in \mathbf{R}^q \mid f_0(x) \preceq_K t \text{ for some feasible } x\},$$

is convex. Also show that the minimal elements of $\mathcal{A}$ are the same as the minimal points of $\mathcal{O}$.

4.54 *Scalarization and optimal points.* Suppose a (not necessarily convex) vector optimization problem has an optimal point $x^\star$. Show that $x^\star$ is a solution of the associated scalarized problem for any choice of $\lambda \succ_{K^*} 0$. Also show the converse: If a point x is a solution of the scalarized problem for any choice of $\lambda \succ_{K^*} 0$, then it is an optimal point for the (not necessarily convex) vector optimization problem.

4.55 *Generalization of weighted-sum scalarization.* In §4.7.4 we showed how to obtain Pareto optimal solutions of a vector optimization problem by replacing the vector objective $f_0 : \mathbf{R}^n \to \mathbf{R}^q$ with the scalar objective $\lambda^T f_0$, where $\lambda \succ_{K^*} 0$. Let $\psi : \mathbf{R}^q \to \mathbf{R}$ be a K-increasing function, *i.e.*, satisfying

$$u \preceq_K v, \; u \neq v \Longrightarrow \psi(u) < \psi(v).$$

Show that any solution of the problem

$$\begin{array}{lll} \text{minimize} & \psi(f_0(x)) & \\ \text{subject to} & f_i(x) \le 0, & i = 1, \ldots, m \\ & h_i(x) = 0, & i = 1, \ldots, p \end{array}$$

is Pareto optimal for the vector optimization problem

$$\begin{array}{ll} \text{minimize (w.r.t. } K) & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \dots, m \\ & h_i(x) = 0, \quad i = 1, \dots, p. \end{array}$$

Note that $\psi(u) = \lambda^T u$, where $\lambda \succ_{K^*} 0$, is a special case.

As a related example, show that in a multicriterion optimization problem (*i.e.*, a vector optimization problem with $f_0 = F : \mathbf{R}^n \to \mathbf{R}^q$, and $K = \mathbf{R}^q_+$), a *unique* solution of the scalar optimization problem

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

is Pareto optimal.

Miscellaneous problems

4.56 [P. Parrilo] We consider the problem of minimizing the convex function $f_0 : \mathbf{R}^n \to \mathbf{R}$ over the convex hull of the union of some convex sets, $\mathbf{conv}\left(\bigcup_{i=1}^q C_i\right)$. These sets are described via convex inequalities,

$$C_i = \{x \mid f_{ij}(x) \leq 0,\ j = 1, \dots, k_i\},$$

where $f_{ij} : \mathbf{R}^n \to \mathbf{R}$ are convex. Our goal is to formulate this problem as a convex optimization problem.

The obvious approach is to introduce variables $x_1, \dots, x_q \in \mathbf{R}^n$, with $x_i \in C_i$, $\theta \in \mathbf{R}^q$ with $\theta \succeq 0$, $\mathbf{1}^T\theta = 1$, and a variable $x \in \mathbf{R}^n$, with $x = \theta_1 x_1 + \cdots + \theta_q x_q$. This equality constraint is not affine in the variables, so this approach does not yield a convex problem.

A more sophisticated formulation is given by

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & s_i f_{ij}(z_i/s_i) \leq 0, \quad i = 1, \dots, q, \quad j = 1, \dots, k_i \\ & \mathbf{1}^T s = 1, \quad s \succeq 0 \\ & x = z_1 + \cdots + z_q, \end{array}$$

with variables $z_1, \dots, z_q \in \mathbf{R}^n$, $x \in \mathbf{R}^n$, and $s_1, \dots, s_q \in \mathbf{R}$. (When $s_i = 0$, we take $s_i f_{ij}(z_i/s_i)$ to be 0 if $z_i = 0$ and ∞ if $z_i \neq 0$.) Explain why this problem is convex, and equivalent to the original problem.

4.57 *Capacity of a communication channel.* We consider a communication channel, with input $X(t) \in \{1, \dots, n\}$, and output $Y(t) \in \{1, \dots, m\}$, for $t = 1, 2, \dots$ (in seconds, say). The relation between the input and the output is given statistically:

$$p_{ij} = \mathbf{prob}(Y(t) = i | X(t) = j), \quad i = 1, \dots, m, \quad j = 1, \dots, n.$$

The matrix $P \in \mathbf{R}^{m \times n}$ is called the *channel transition matrix*, and the channel is called a *discrete memoryless channel*.

A famous result of Shannon states that information can be sent over the communication channel, with arbitrarily small probability of error, at any rate less than a number C, called the *channel capacity*, in bits per second. Shannon also showed that the capacity of a discrete memoryless channel can be found by solving an optimization problem. Assume that X has a probability distribution denoted $x \in \mathbf{R}^n$, *i.e.*,

$$x_j = \mathbf{prob}(X = j), \quad j = 1, \dots, n.$$

The *mutual information* between X and Y is given by

$$I(X;Y) = \sum_{i=1}^{m}\sum_{j=1}^{n} x_j p_{ij} \log_2 \frac{p_{ij}}{\sum_{k=1}^{n} x_k p_{ik}}.$$

Then the channel capacity C is given by

$$C = \sup_x I(X;Y),$$

where the supremum is over all possible probability distributions for the input X, *i.e.*, over $x \succeq 0$, $\mathbf{1}^T x = 1$.

Show how the channel capacity can be computed using convex optimization.

Hint. Introduce the variable $y = Px$, which gives the probability distribution of the output Y, and show that the mutual information can be expressed as

$$I(X;Y) = c^T x - \sum_{i=1}^{m} y_i \log_2 y_i,$$

where $c_j = \sum_{i=1}^{m} p_{ij} \log_2 p_{ij}$, $j = 1, \ldots, n$.

4.58 *Optimal consumption.* In this problem we consider the optimal way to consume (or spend) an initial amount of money (or other asset) k_0 over time. The variables are $c_0, \ldots, c_T$, where $c_t \geq 0$ denotes the *consumption* in period t. The utility derived from a consumption level c is given by $u(c)$, where $u : \mathbf{R} \to \mathbf{R}$ is an increasing concave function. The present value of the utility derived from the consumption is given by

$$U = \sum_{t=0}^{T} \beta^t u(c_t),$$

where $0 < \beta < 1$ is a *discount factor*.

Let k_t denote the amount of money available for investment in period t. We assume that it earns an investment return given by $f(k_t)$, where $f : \mathbf{R} \to \mathbf{R}$ is an increasing, concave *investment return function*, which satisfies $f(0) = 0$. For example if the funds earn simple interest at rate R percent per period, we have $f(a) = (R/100)a$. The amount to be consumed, *i.e.*, c_t, is withdrawn at the end of the period, so we have the recursion

$$k_{t+1} = k_t + f(k_t) - c_t, \quad t = 0, \ldots, T.$$

The initial sum $k_0 > 0$ is given. We require $k_t \geq 0$, $t = 1, \ldots, T+1$ (but more sophisticated models, which allow $k_t < 0$, can be considered).

Show how to formulate the problem of maximizing U as a convex optimization problem. Explain how the problem you formulate is equivalent to this one, and exactly how the two are related.

Hint. Show that we can replace the recursion for k_t given above with the inequalities

$$k_{t+1} \leq k_t + f(k_t) - c_t, \quad t = 0, \ldots, T.$$

(Interpretation: the inequalities give you the option of throwing money away in each period.) For a more general version of this trick, see exercise 4.6.

4.59 *Robust optimization.* In some optimization problems there is uncertainty or variation in the objective and constraint functions, due to parameters or factors that are either beyond our control or unknown. We can model this situation by making the objective and constraint functions $f_0, \ldots, f_m$ functions of the optimization variable $x \in \mathbf{R}^n$ *and* a parameter vector $u \in \mathbf{R}^k$ that is unknown, or varies. In the *stochastic optimization* approach, the parameter vector u is modeled as a random variable with a known distribution, and we work with the expected values $\mathbf{E}_u\, f_i(x,u)$. In the *worst-case analysis* approach, we are given a set U that u is known to lie in, and we work with the maximum or worst-case values $\sup_{u\in U} f_i(x,u)$. To simplify the discussion, we assume there are no equality constraints.

(a) *Stochastic optimization.* We consider the problem

$$\begin{array}{ll} \text{minimize} & \mathbf{E}\, f_0(x,u) \\ \text{subject to} & \mathbf{E}\, f_i(x,u) \leq 0, \quad i=1,\ldots,m, \end{array}$$

where the expectation is with respect to u. Show that if f_i are convex in x for each u, then this stochastic optimization problem is convex.

(b) *Worst-case optimization.* We consider the problem

$$\begin{array}{ll} \text{minimize} & \sup_{u\in U} f_0(x,u) \\ \text{subject to} & \sup_{u\in U} f_i(x,u) \leq 0, \quad i=1,\ldots,m. \end{array}$$

Show that if f_i are convex in x for each u, then this worst-case optimization problem is convex.

(c) *Finite set of possible parameter values.* The observations made in parts (a) and (b) are most useful when we have analytical or easily evaluated expressions for the expected values $\mathbf{E}\, f_i(x,u)$ or the worst-case values $\sup_{u\in U} f_i(x,u)$.

Suppose we are given the set of possible values of the parameter is finite, *i.e.*, we have $u \in \{u_1,\ldots,u_N\}$. For the stochastic case, we are also given the probabilities of each value: $\mathbf{prob}(u=u_i)=p_i$, where $p\in\mathbf{R}^N$, $p\succeq 0$, $\mathbf{1}^T p = 1$. In the worst-case formulation, we simply take $U \in \{u_1,\ldots,u_N\}$.

Show how to set up the worst-case and stochastic optimization problems explicitly (*i.e.*, give explicit expressions for $\sup_{u\in U} f_i$ and $\mathbf{E}_u\, f_i$).

4.60 *Log-optimal investment strategy.* We consider a portfolio problem with n assets held over N periods. At the beginning of each period, we re-invest our total wealth, redistributing it over the n assets using a fixed, constant, allocation strategy $x\in\mathbf{R}^n$, where $x\succeq 0$, $\mathbf{1}^T x = 1$. In other words, if $W(t-1)$ is our wealth at the beginning of period t, then during period t we invest $x_i W(t-1)$ in asset i. We denote by $\lambda(t)$ the total return during period t, *i.e.*, $\lambda(t) = W(t)/W(t-1)$. At the end of the N periods our wealth has been multiplied by the factor $\prod_{t=1}^N \lambda(t)$. We call

$$\frac{1}{N}\sum_{t=1}^{N} \log\lambda(t)$$

the *growth rate* of the investment over the N periods. We are interested in determining an allocation strategy x that maximizes growth of our total wealth for large N.

We use a discrete stochastic model to account for the uncertainty in the returns. We assume that during each period there are m possible scenarios, with probabilities π_j, $j=1,\ldots,m$. In scenario j, the return for asset i over one period is given by p_{ij}. Therefore, the return $\lambda(t)$ of our portfolio during period t is a random variable, with m possible values $p_1^T x,\ldots,p_m^T x$, and distribution

$$\pi_j = \mathbf{prob}(\lambda(t) = p_j^T x), \quad j=1,\ldots,m.$$

We assume the same scenarios for each period, with (identical) independent distributions. Using the law of large numbers, we have

$$\lim_{N\to\infty}\frac{1}{N}\log\left(\frac{W(N)}{W(0)}\right) = \lim_{N\to\infty}\frac{1}{N}\sum_{t=1}^{N}\log\lambda(t) = \mathbf{E}\log\lambda(t) = \sum_{j=1}^{m}\pi_j\log(p_j^T x).$$

In other words, with investment strategy x, the long term growth rate is given by

$$R_{\rm lt} = \sum_{j=1}^{m} \pi_j \log(p_j^T x).$$

The investment strategy x that maximizes this quantity is called the *log-optimal investment strategy*, and can be found by solving the optimization problem

$$\begin{array}{ll} \mbox{maximize} & \sum_{j=1}^m \pi_j \log(p_j^T x) \\ \mbox{subject to} & x \succeq 0, \quad \mathbf{1}^T x = 1, \end{array}$$

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

Show that this is a convex optimization problem.

4.61 *Optimization with logistic model.* A random variable $X \in \{0,1\}$ satisfies

$$\mathbf{prob}(X=1) = p = \frac{\exp(a^T x + b)}{1 + \exp(a^T x + b)},$$

where $x \in \mathbf{R}^n$ is a vector of variables that affect the probability, and a and b are known parameters. We can think of $X = 1$ as the event that a consumer buys a product, and x as a vector of variables that affect the probability, *e.g.*, advertising effort, retail price, discounted price, packaging expense, and other factors. The variable x, which we are to optimize over, is subject to a set of linear constraints, $Fx \preceq g$.

Formulate the following problems as convex optimization problems.

(a) *Maximizing buying probability.* The goal is to choose x to maximize p.

(b) *Maximizing expected profit.* Let $c^T x + d$ be the profit derived from selling the product, which we assume is positive for all feasible x. The goal is to maximize the expected profit, which is $p(c^T x + d)$.

4.62 *Optimal power and bandwidth allocation in a Gaussian broadcast channel.* We consider a communication system in which a central node transmits messages to n receivers. ('Gaussian' refers to the type of noise that corrupts the transmissions.) Each receiver channel is characterized by its (transmit) power level $P_i \geq 0$ and its bandwidth $W_i \geq 0$. The power and bandwidth of a receiver channel determine its *bit rate* R_i (the rate at which information can be sent) via

$$R_i = \alpha_i W_i \log(1 + \beta_i P_i / W_i),$$

where α_i and β_i are known positive constants. For $W_i = 0$, we take $R_i = 0$ (which is what you get if you take the limit as $W_i \to 0$).

The powers must satisfy a total power constraint, which has the form

$$P_1 + \cdots + P_n = P_{\rm tot},$$

where $P_{\rm tot} > 0$ is a given total power available to allocate among the channels. Similarly, the bandwidths must satisfy

$$W_1 + \cdots + W_n = W_{\rm tot},$$

where $W_{\rm tot} > 0$ is the (given) total available bandwidth. The optimization variables in this problem are the powers and bandwidths, *i.e.*, $P_1, \ldots, P_n$, $W_1, \ldots, W_n$.

The objective is to maximize the total utility,

$$\sum_{i=1}^{n} u_i(R_i),$$

where $u_i : \mathbf{R} \to \mathbf{R}$ is the utility function associated with the ith receiver. (You can think of $u_i(R_i)$ as the revenue obtained for providing a bit rate R_i to receiver i, so the objective is to maximize the total revenue.) You can assume that the utility functions u_i are nondecreasing and concave.

Pose this problem as a convex optimization problem.

4.63 *Optimally balancing manufacturing cost and yield.* The vector $x \in \mathbf{R}^n$ denotes the nominal parameters in a manufacturing process. The yield of the process, *i.e.*, the fraction of manufactured goods that is acceptable, is given by $Y(x)$. We assume that Y is log-concave (which is often the case; see example 3.43). The cost per unit to manufacture the product is given by $c^T x$, where $c \in \mathbf{R}^n$. The cost per acceptable unit is $c^T x / Y(x)$. We want to minimize $c^T x / Y(x)$, subject to some convex constraints on x such as a linear inequalities $Ax \preceq b$. (You can assume that over the feasible set we have $c^T x > 0$ and $Y(x) > 0$.)

This problem is *not* a convex or quasiconvex optimization problem, but it can be solved using convex optimization and a one-dimensional search. The basic ideas are given below; you must supply all details and justification.

(a) Show that the function $f : \mathbf{R} \to \mathbf{R}$ given by

$$f(a) = \sup\{Y(x) \mid Ax \preceq b,\ c^T x = a\},$$

which gives the maximum yield versus cost, is log-concave. This means that by solving a convex optimization problem (in x) we can evaluate the function f.

(b) Suppose that we evaluate the function f for enough values of a to give a good approximation over the range of interest. Explain how to use these data to (approximately) solve the problem of minimizing cost per good product.

4.64 *Optimization with recourse.* In an optimization problem with recourse, also called *two-stage optimization*, the cost function and constraints depend not only on our choice of variables, but also on a discrete random variable $s \in \{1, \ldots, S\}$, which is interpreted as specifying which of S *scenarios* occurred. The scenario random variable s has known probability distribution π, with $\pi_i = \mathbf{prob}(s = i)$, $i = 1, \ldots, S$.

In two-stage optimization, we are to choose the values of two variables, $x \in \mathbf{R}^n$ and $z \in \mathbf{R}^q$. The variable x must be chosen *before* the particular scenario s is known; the variable z, however, is chosen *after* the value of the scenario random variable is known. In other words, z is a function of the scenario random variable s. To describe our choice z, we list the values we would choose under the different scenarios, *i.e.*, we list the vectors

$$z_1, \ldots, z_S \in \mathbf{R}^q.$$

Here z_3 is our choice of z when $s = 3$ occurs, and so on. The set of values

$$x \in \mathbf{R}^n, \qquad z_1, \ldots, z_S \in \mathbf{R}^q$$

is called the *policy*, since it tells us what choice to make for x (independent of which scenario occurs), and also, what choice to make for z in each possible scenario.

The variable z is called the *recourse variable* (or *second-stage variable*), since it allows us to take some action or make a choice after we know which scenario occurred. In contrast, our choice of x (which is called the *first-stage variable*) must be made without any knowledge of the scenario.

For simplicity we will consider the case with no constraints. The cost function is given by

$$f : \mathbf{R}^n \times \mathbf{R}^q \times \{1, \ldots, S\} \to \mathbf{R},$$

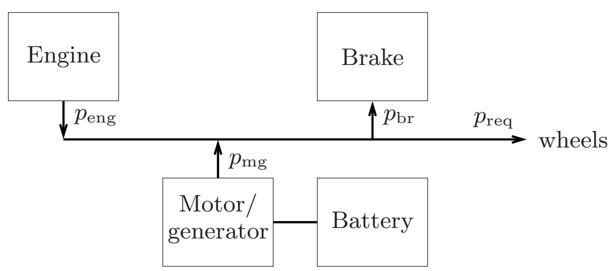
where $f(x, z, i)$ gives the cost when the first-stage choice x is made, second-stage choice z is made, and scenario i occurs. We will take as the overall objective, to be minimized over all policies, the expected cost

$$\mathbf{E}\, f(x, z_s, s) = \sum_{i=1}^{S} \pi_i f(x, z_i, i).$$

Suppose that f is a convex function of (x, z), for each scenario $i = 1, \ldots, S$. Explain how to find an optimal policy, *i.e.*, one that minimizes the expected cost over all possible policies, using convex optimization.

4.65 *Optimal operation of a hybrid vehicle.* A hybrid vehicle has an internal combustion engine, a motor/generator connected to a storage battery, and a conventional (friction) brake. In this exercise we consider a (highly simplified) model of a *parallel hybrid vehicle*, in which both the motor/generator and the engine are directly connected to the drive wheels. The engine can provide power to the wheels, and the brake can take power from the wheels, turning it into heat. The motor/generator can act as a motor, when it uses energy stored in the battery to deliver power to the wheels, or as a generator, when it takes power from the wheels or engine, and uses the power to charge the battery. When the generator takes power from the wheels and charges the battery, it is called *regenerative braking*; unlike ordinary friction braking, the energy taken from the wheels is *stored*, and can be used later. The vehicle is judged by driving it over a known, fixed test track to evaluate its fuel efficiency.

A diagram illustrating the power flow in the hybrid vehicle is shown below. The arrows indicate the direction in which the power flow is considered positive. The engine power $p_{\rm eng}$, for example, is positive when it is delivering power; the brake power $p_{\rm br}$ is positive when it is taking power from the wheels. The power $p_{\rm req}$ is the required power at the wheels. It is positive when the wheels require power (*e.g.*, when the vehicle accelerates, climbs a hill, or cruises on level terrain). The required wheel power is negative when the vehicle must decelerate rapidly, or descend a hill.



All of these powers are functions of time, which we discretize in one second intervals, with $t = 1, 2, \ldots, T$. The required wheel power $p_{\rm req}(1), \ldots, p_{\rm req}(T)$ is given. (The speed of the vehicle on the track is specified, so together with known road slope information, and known aerodynamic and other losses, the power required at the wheels can be calculated.)

Power is conserved, which means we have

$$p_{\rm req}(t) = p_{\rm eng}(t) + p_{\rm mg}(t) - p_{\rm br}(t), \quad t = 1, \ldots, T.$$

The brake can only dissipate power, so we have $p_{\rm br}(t) \geq 0$ for each t. The engine can only provide power, and only up to a given limit $P_{\rm eng}^{\rm max}$, *i.e.*, we have

$$0 \leq p_{\rm eng}(t) \leq P_{\rm eng}^{\rm max}, \quad t = 1, \ldots, T.$$

The motor/generator power is also limited: $p_{\rm mg}$ must satisfy

$$P_{\rm mg}^{\rm min} \leq p_{\rm mg}(t) \leq P_{\rm mg}^{\rm max}, \quad t = 1, \ldots, T.$$

Here $P_{\rm mg}^{\rm max} > 0$ is the maximum motor power, and $-P_{\rm mg}^{\rm min} > 0$ is the maximum generator power.

The battery charge or energy at time t is denoted $E(t)$, $t = 1, \ldots, T+1$. The battery energy satisfies

$$E(t+1) = E(t) - p_{\rm mg}(t) - \eta |p_{\rm mg}(t)|, \quad t = 1, \ldots, T,$$

where $\eta > 0$ is a known parameter. (The term $-p_{\mathrm{mg}}(t)$ represents the energy removed or added the battery by the motor/generator, ignoring any losses. The term $-\eta|p_{\mathrm{mg}}(t)|$ represents energy lost through inefficiencies in the battery or motor/generator.)

The battery charge must be between 0 (empty) and its limit $E_{\mathrm{batt}}^{\mathrm{max}}$ (full), at all times. (If $E(t) = 0$, the battery is fully discharged, and no more energy can be extracted from it; when $E(t) = E_{\mathrm{batt}}^{\mathrm{max}}$, the battery is full and cannot be charged.) To make the comparison with non-hybrid vehicles fair, we fix the initial battery charge to equal the final battery charge, so the net energy change is zero over the track: $E(1) = E(T+1)$. We do not specify the value of the initial (and final) energy.

The objective in the problem (to be minimized) is the total fuel consumed by the engine, which is

$$F_{\mathrm{total}} = \sum_{t=1}^{T} F(p_{\mathrm{eng}}(t)),$$

where $F : \mathbf{R} \to \mathbf{R}$ is the *fuel use characteristic* of the engine. We assume that F is positive, increasing, and convex.

Formulate this problem as a convex optimization problem, with variables $p_{\mathrm{eng}}(t)$, $p_{\mathrm{mg}}(t)$, and $p_{\mathrm{br}}(t)$ for $t = 1, \ldots, T$, and $E(t)$ for $t = 1, \ldots, T+1$. Explain why your formulation is equivalent to the problem described above.