Convex Optimization

Lecture 6, Convex Optimization

National Taiwan University

April 8, 2020

Table of contents

Optimization Problems

- The notation

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

  is used to describe an optimization problem of finding an x that minimizes $f_0(x)$ among all x that satisfy the conditions $f_i(x) \leq 0, i = 1, \ldots, m$ and $h_i(x) = 0, i = 1, \ldots, p$.
- $x \in \mathbf{R}^n$: the **optimization variables**.
- $f_0 : \mathbf{R}^n \to \mathbf{R}$: the **objective function**.
- $f_i : \mathbf{R}^n \to \mathbf{R}$: the **inequality constraint functions**.
  - $f_i(x) \leq 0$: the **inequality constraints**.
- $h_i : \mathbf{R}^n \to \mathbf{R}$: the **equality constraint functions**.
  - $h_i(x) = 0$: the **equality constraints**.

Optimization Problems

Optimization Problems

Consider the problem

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

- The set
$$\mathcal{D} = \bigcap_{i=0}^{m} \textbf{dom } f_i \cap \bigcap_{i=1}^{p} \textbf{dom } h_i$$
is called the **domain** of the problem.
- A point $x \in \mathcal{D}$ is **feasible** if $f_i(x) \leq 0$ for all $i = 1, \ldots, m$ and $h_i(x) = 0$ for all $i = 1, \ldots, p$.
- The problem is called **feasible** if there exists $x \in \mathcal{D}$ that is feasible; the problem is called **infeasible** if there is no feasible point in $\mathcal{D}$.
- The set of all feasible points is called the **feasible set**.
- If there are no constraints (i.e., $m = p = 0$), then the feasible set equals $\mathcal{D} = \textbf{dom } f_0$, and the problem is called **unconstrained**.

Optimization Problems – Optimal Values

Optimal Values

- In the problem

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

the **optimal value** p^* is defined as

$$p^* = \inf \{ f_0(x) \mid f_i(x) \leq 0, i = 1, \ldots, m, h_i(x) = 0, i = 1, \ldots, p \} .$$

- If the problem is infeasible, we have $p^* = \infty$.
- If there are feasible points x_k with $f_0(x_k) \to -\infty$ as $k \to \infty$, then $p^* = -\infty$, and the problem is said to be **unbounded below**.

Optimization Problems – Optimal Points

Optimal Point

Suppose the optimal value of the problem

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

is p^*. Then we say x^* is an **optimal point** if

- x^* is feasible, and
- $f_0(x^*) = p^*$.

- The set of all optimal points is the **optimal set**, denoted

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

Optimization Problems – Optimal Points

- If there exists an optimal point for an optimization problem, 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.
  - e.g., this always occurs when the problem is unbounded below.
- A feasible point x with $f_0(x) \leq p^* + \epsilon$ (where $\epsilon > 0$) is called **ϵ-suboptimal**.
  - The set of all ϵ-suboptimal points is called the **ϵ-suboptimal set** for the optimization problem.

Optimization Problem

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

$$f_0(x) = \inf \{ f_0(z) \mid f_i(z) \leq 0, i = 1, \ldots, m, h_i(z) = 0, i = 1, \ldots, p, ||z - x||_2 \leq R \}.$$

  - This means x minimizes f_0 over nearby points in the feasible set.

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

Optimization Problems – Examples

We consider the following unconstrained problems as examples, with $f_0 : \mathbf{R} \to \mathbf{R}$ and $\mathbf{dom}\ f_0 = \mathbf{R}_{++}$. Recall that

$$p^* = \inf \{f_0(x) \mid x \text{ is feasible}\}.$$

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

Feasibility problems

- If the objective function is identically zero, the optimal value is either
  - 0, 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

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

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

Expressing Problems in Standard Forms

- An optimization problem in the form of

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

  is called in the **standard form**, i.e., the righthand side of the inequality and equality constraints are zeros.
- An equality constraint in a non-standard form $g_i(x) = \tilde{g}_i(x)$ can be reformulated as $h_i(x) = 0$ where $h_i(x) = g_i(x) - \tilde{g}_i(x)$.
- An inequality constraint of the form $f_i(x) \geq 0$ can be rewritten as $-f_i(x) \leq 0$.

Expressing Problems in Standard Forms – Examples

The optimization problem

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

can be expressed in standard form as

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

There are $2n$ inequality constraint functions:

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

and

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

Expressing Problems in Standard Forms – Examples

The **maximization problem**

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

can be solved by minimizing the function $-f_0(x)$ subject to the same constraints.

Equivalent Problems

We call two problems **equivalent** (informally) if from a solution of one, a solution of the other is readily found, and vice versa.

Example

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

(where $\alpha_i > 0, i = 0, \ldots, m,\ \beta_i \neq 0, i = 1, \ldots, p$) and

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

are equivalent problems.

Transformation of objective and constraint functions (1/2)

- Suppose that
  - $\phi_0 : \mathbf{R} \to \mathbf{R}$ is monotone increasing,
  - $\phi_1, \dots, \phi_m : \mathbf{R} \to \mathbf{R}$ satisfy $\phi_i(u) \leq 0$ if and only if $u \leq 0$, and
  - $\phi_{m+1}, \dots, \phi_{m+p} : \mathbf{R} \to \mathbf{R}$ satisfy $\phi_i(u) = 0$ if and only if $u = 0$.
- We define functions $\tilde{f}_i$ and $\tilde{h}_i$ as the compositions
  - $\tilde{f}_i(x) = \phi_i(f_i(x)), i = 0, \dots, m,$
  - $\tilde{h}_i(x) = \phi_{m+i}(h_i(x)), i = 1, \dots, p.$
- Then, the associated problem

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

and the standard form problem are equivalent.

Transformation of objective and constraint functions (2/2)

- Example: Consider the problem

$$\begin{array}{ll} \text{minimize} & ||Ax - b||_2 \\ \text{subject to} & ||x||_2 \leq 2 \end{array}$$

- It can be reformulated as

$$\begin{array}{ll} \text{minimize} & ||Ax - b||_2^2 \\ \text{subject to} & ||x||_2^2 - 4 \leq 0, \end{array}$$

and therefore is equivalent to

$$\begin{array}{ll} \text{minimize} & (Ax - b)^T(Ax - b) \\ \text{subject to} & x^T x - 4 \leq 0 \end{array}$$

Slack variables

- Observation: $f_i(x) \leq 0$ if and only if there is an $s_i \geq 0$ that satisfies $f_i(x) + s_i = 0$.
- Based on the observation, we obtain the transformed problem

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

  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 $f_i(x) \leq 0$.

Eliminating equality constraints

- Suppose the function $\phi : \mathbf{R}^k \to \mathbf{R}^n$ is such that x satisfies $h_i(x) = 0, i = 1, \ldots, p$ if and only if there is some $z \in \mathbf{R}^k$ such that $x = \phi(z)$.
- Then, the optimization problem

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

  is then equivalent to the original standard form problem.
- 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 any z that satisfies $x = \phi(z)$ is optimal for the transformed problem.

Eliminating linear equality constraints

- Consider the standard form problem with linear equality constraints

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

- Suppose $Ax = b$ is consistent. Then the solution set of $Ax = b$ can be parametrized as $\left\{Fz + x_0 \mid z \in \mathbf{R}^k\right\}$ where $F \in \mathbf{R}^{n \times k}$ is chosen to be any full rank matrix with $\mathcal{R}(F) = \mathcal{N}(A)$ (i.e., $k = n - \operatorname{rank} A$), and x_0 is any particular solution of $Ax = b$.
- Then we can eliminate these linear constraints and create an equivalent problem, as in

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

where we introduced new variables $z \in \mathbf{R}^k$.

Introducing equality constraints (1/2)

- We can also introduce equality constraints and new variables into a problem.
- As a typical example, consider the problem

$$\begin{array}{ll} \text{minimize} & f_0(A_0x + b_0) \\ \text{subject to} & f_i(A_ix + b_i) \leq 0, i = 1, \ldots, m \\ & h_i(x) = 0, i = 1, \ldots, p, \end{array}$$

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 $A_ix + b_i$.

Introducing equality constraints (2/2)

- 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 $i = 0, ..., m$, and form the equivalent problem

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

- This problem has $k_0 + ... + k_m$ new variables, $y_0 \in \mathbf{R}^{k_0}, ..., y_m \in \mathbf{R}^{k_m}$, and $k_0 + ... + k_m$ new equality constraints, $y_0 = A_0 x + b_0, ..., y_m = A_m x + b_m$.
- The objective and inequality constraints in this problem are independent, i.e., involve different optimization variables.

Epigraph problem form (1/2)

- The epigraph form of the standard problem is the problem

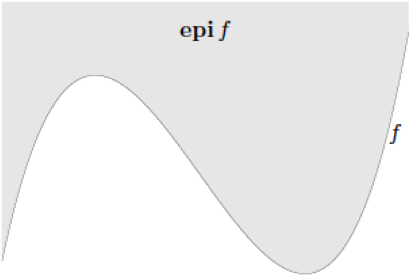
$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & f_0(x) - t \leq 0 \\ & f_i(x) \leq 0, i = 1, \dots, m \\ & h_i(x) = 0, i = 1, \dots, p, \end{array}$$

  with variables $x \in \mathbf{R}^n$ and $t \in \mathbf{R}$.

- It is equivalent to the original problem: (x, t) is optimal for the epigraph form problem if and only if x is optimal for the original problem and $t = f_0(x)$.

Epigraph problem form (2/2)

- Note that the objective function of the epigraph form problem is a linear function of the variables x, t.
- The epigraph form problem can be interpreted geometrically as an optimization problem in the 'graph space' (x, t).



Convex optimization problems in standard form (1/2)

A convex optimization problem is one of the form

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

where $f_0, \ldots, f_m$ are convex functions. Compared with the general standard form problem, 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.

Convex optimization problems in standard form (2/2)

- The feasible set of a convex optimization problem is convex, since it is the intersection of
  - the domain of the problem

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

    (which is a convex set),
  - m (convex) sublevel sets $\{x \mid f_i(x) \leq 0\}$, and
  - p hyperplanes $\{x \mid a_i^T x = b_i\}$.
    - W.l.o.g., we assume that $a_i \neq 0$.
- In a convex optimization problem, we minimize a convex objective function over a convex set.

Concave maximization problems

- We also refer to

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

  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.

Definition of Convex Optimization Problem

A closer look

- Consider the example with $x \in \mathbf{R}^2$,

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

  which is in the standard form.
- 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.
- The problem, although not in a form of convex optimization problem, can be easily transformed to, and be shown to be equivalent to, a convex optimization problem.

Local and global optima (1/2)

- As an important property of convex optimization problems, 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

$$f_0(x) = \inf \left\{ f_0(z) \;\middle|\; z \text{ feasible}, ||z - x||_2 \leq R \right\},$$

  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 $f_0(x) \leq f_0(y)$.

Local and global optima (2/2)

- Consider the point z given by

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

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

- By convexity of f_0 and the assumption that $f_0(y) < f_0(x)$, we have

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

  which leads to a contradiction. So, x is globally optimal.

An optimality criterion for differentiable f_0

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

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

- Let X denote the feasible set, i.e.,

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

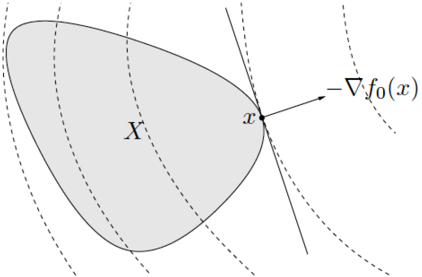
Then x is optimal if and only if $x \in X$ and

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

for all $y \in X$.

An optimality criterion for differentiable f_0

- The 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.



Proof of optimality condition

- The “if” part is obvious.
- For the “only if” part, suppose x is optimal, but the optimality condition $\nabla f_0(x)^T(y-x) \geq 0$ 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. Note that

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

so for small positive t, we have $f_0(z(t)) < f_0(x)$, which proves that x is not optimal.

Unconstrained problems

- For an unconstrained problem (i.e., $m = p = 0$), the optimality condition

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

  reduces to the well known necessary and sufficient condition

  $$\nabla f_0(x) = 0$$

  for x to be optimal.

Unconstrained problems

- To see this, 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 $y = x - t\nabla f_0(x)$. Then 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 $\nabla f_0(x) = 0$.
- If $\nabla f_0(x) = 0$ has no solutions, then there are no optimal points, possibly
  - the problem is unbounded below, or
  - the optimal value is finite, but not attained.
- On the other hand, $\nabla f_0(x) = 0$ can have multiple solutions.
  - In this case, each such solution is a minimizer of f_0.

Example – Unconstrained quadratic optimization.

- Consider the problem of minimizing the quadratic function

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

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, $x^* = -P^{-1}q$.
    - If P is singular, but $q \in \mathcal{R}(P)$, then the set of optimal points is the (affine) set $X_{opt} = -P^\dagger q + \mathcal{N}(P)$, where $P^\dagger$ denotes the pseudo-inverse of P.

Problems with equality constraints only (1/2)

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

Problems with equality constraints only (2/2)

- 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.

Minimization over the nonnegative orthant (1/2)

- 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 is then

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

- 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 $\nabla f_0(x) \succeq 0$.

Minimization over the nonnegative orthant (2/2)

- 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.,

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

- Therefore, $[\nabla f_0(x)]_i x_i = 0$ for $i = 1, \ldots, n$. The optimality condition can therefore be expressed as

$$x \succeq 0, \quad \nabla f_0(x) \succeq 0, \quad x_i [\nabla f_0(x)]_i = 0, \quad i = 1, \ldots, n.$$

- The last condition is called **complementarity**, since it means that the set of indices corresponding to nonzero components of the vectors x and $\nabla f_0(x)$ are complementary (i.e., have empty intersection).

Quasiconvex Optimization Problems

- If f_0 is quasiconvex, but not necessarily convex, the problem

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

  is called 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.

Quasiconvex optimization

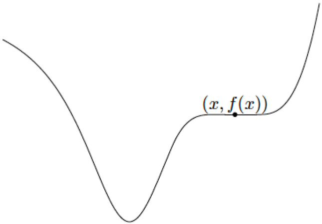
- Recall that a quasiconvex optimization problem has the standard form

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

  where the inequality constraint functions $f_1, \dots, f_m$ are convex, and the objective f_0 is quasiconvex (instead of convex).
- Some basic differences between convex and quasiconvex optimization problems will be studied.
  - It would be shown that 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 **R**.



Locally optimal solutions and optimality conditions

- Nevertheless, a variation of the optimality condition for convex problems ($\nabla f_0(x)^T(y-x) \geq 0$ for all $y \in X$) does hold for quasiconvex optimization problems with differentiable objective function.
- Let X denote the feasible set for the quasiconvex optimization problem described in a previous page.
- We first recognize that

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

for any quasiconvex differentiable function f.

- It then follows that x is optimal if

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

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.
- 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^* denote the optimal value of the quasiconvex optimization problem. If the feasibility problem

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

  is feasible, then we have $p^* \le t$. Otherwise, we have $p^* > t$.

Bisection for Quasiconvex Optimization (1/2)

Algorithm 4.1 Bisection method for quasiconvex optimization.

- given $l \leq p^*$, $u \geq p^*$, tolerance $\epsilon > 0$.
  repeat
  1. $t := (l + u)/2$.
  2. Solve the convex feasibility problem

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

  3. If the previous problem is feasible, $u := t$; else $l := t$.

  until $u - l \leq \epsilon$.

Bisection for Quasiconvex Optimization (2/2)

- The interval $[l, u]$ is guaranteed to contain p^*, i.e., we have $l \leq p^* \leq u$ 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

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

Quasiconvex Optimization Problem – An Example

- Consider the problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & ||Ax - b|| \leq \epsilon, \end{array}$$

  where $f_0(x) = \text{length}(x) = \min\{k \mid x_i = 0 \text{ for } i > k\}$. The problem variable is $x \in \mathbf{R}^n$; the problem parameters are $A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$, and $\epsilon > 0$.
- This is to find the minimum number of columns of A, taken in order, that can approximate the vector b within ϵ.
- It can be shown to be a quasiconvex optimization problem.
- The bisection algorithm can be applied by finding an appropriate family of functions $\phi_t(x)$ that satisfies

$$f_0(x) \leq t \iff \phi_t(x) \leq 0.$$

Introduction to CVX for Matlab

- CVX is a modeling system for constructing and solving disciplined convex programs (DCPs).
- CVX supports a number of standard problem types, including linear and quadratic programs (LPs/QPs), second-order cone programs (SOCPs), and semidefinite programs (SDPs).
- CVX can also solve much more complex convex optimization problems, including many involving nondifferentiable functions, such as ℓ_1 norms.
- You can use CVX to conveniently formulate and solve constrained norm minimization, entropy maximization, determinant maximization, and many other convex programs.

Introduction to CVX for Matlab

- CVX is implemented in Matlab, effectively turning Matlab into an optimization modeling language.
- Model specifications are constructed using common Matlab operations and functions, and standard Matlab code can be freely mixed with these specifications.
- This combination makes it simple to perform the calculations needed to form optimization problems, or to process the results obtained from their solution.

See http://web.cvxr.com/cvx/doc/index.html for more details.

Example – Least Square Problems

- Consider the least square problem described as

$$\text{minimize} \qquad ||Ax - b||_2^2$$

where $A \in \mathbf{R}^{m \times n}$ and $b \in \mathbf{R}^m$, $m > n$, rank $A = n$.

- It has an analytic solution $x_{opt} = (A^T A)^{-1} A^T b$.
- In CVX for Matlab, it can be written as follows.

```
cvx_begin
  variable x(n)
  minimize norm(A * x - b)
cvx_end
```

Example – Linear Programming

- Consider the linear problem described as

$$\begin{aligned} \text{minimize} \quad & -2x - 3y - 4z \\ \text{subject to} \quad & 3x + 2y + z \leq 10 \\ & 2x + 5y + 3z \leq 15 \\ & x, y, z \geq 0 \end{aligned}$$

- In CVX for Matlab, it can be written as follows.

```
cvx_begin
    variables x y z
    minimize -2 * x - 3 * y - 4 * z
    subject to
    3 * x + 2 * y + z <= 10
    2 * x + 5 * y + 3 * z <= 15
    x >= 0
    y >= 0
    z >= 0
cvx_end
```

Example – Unconstrained Quadratic Optimization

- Consider the problem of minimizing the quadratic function

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

where $P \in \mathbf{S}^n_{++}$ (which makes f_0 convex).

- In CVX for Matlab, it can be written as follows.

```
cvx_begin
    variables x(n)
    minimize quad_form(x, P/2) + sum(x .* q) + r
cvx_end
```

The DCP ruleset

- CVX enforces the conventions dictated by the disciplined convex programming (DCP) ruleset.
- CVX will issue an error message whenever it encounters a violation of any of the rules, so it is important to understand them before beginning to build models.
- The rules are drawn from basic principles of convex analysis, and are easy to learn, once you've had an exposure to convex analysis and convex optimization.
- The DCP ruleset is a set of sufficient, but not necessary, conditions for convexity.
- So it is possible to construct expressions that violate the ruleset but are in fact convex.

The DCP ruleset

- As an example consider the entropy function, $-\sum_{i=1}^{n} x_i \log x_i$, defined for $x > 0$, which is concave. If it is expressed as

```
- sum( x .* log( x ) )
```

  CVX will reject it, because its concavity does not follow from any of the composition rules.

- Problems involving entropy, however, can be solved, by explicitly using the entropy function,

```
sum( entr( x ) )
```

  which is in the base CVX library, and thus recognized as concave by CVX.

- If a convex (or concave) function is not recognized as convex or concave by CVX, it can be added as a new atom; see Adding new functions to the atom library.

The DCP ruleset

- As another example consider the function
$$\sqrt{x^2+1} = \left\| \begin{bmatrix} x \\ 1 \end{bmatrix} \right\|_2,$$
which is convex.
- If it is written as
```
norm( [ x 1 ] )
```
it will be recognized by CVX as a convex expression, and therefore can be used in (appropriate) constraints and objectives. But if it is written as
```
sqrt( x^2 + 1 )
```
CVX will reject it, since convexity of this function does not follow from the CVX ruleset.

See http://web.cvxr.com/cvx/doc/dcp.html for more details.

Linear Optimization Problems (1/2)

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

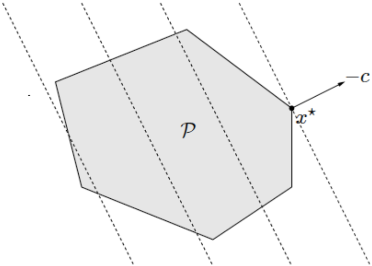
$$\begin{aligned} \text{minimize} \quad & c^T x + d \\ \text{subject to} \quad & Gx \preceq h \\ & Ax = b, \end{aligned}$$

  where $G \in \mathbf{R}^{m \times n}$ and $A \in \mathbf{R}^{p \times n}$.

- Linear programs are a special case of convex optimization problems.
- It is common to omit the constant d in the objective function.

Linear Optimization Problems (2/2)

- We also refer to a maximization problem with affine objective and constraint functions as an LP since we can maximize an affine objective $c^T x + d$, by minimizing $-c^T x - d$ (which is still convex).
- The feasible set of an LP is a polyhedron $\mathcal{P}$; the problem is to minimize the affine function $c^T x + d$ over $\mathcal{P}$.



Standard and inequality forms of linear programs

- In a standard form LP the only inequalities are componentwise nonnegativity constraints $x \succeq 0$:

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

  - Some LP algorithms are developed specifically for standard form LP.
- If the LP has no equality constraints, it is called an inequality form LP, usually written as

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

Converting LPs to standard form

- In order to transform a general LP to a standard form LP, 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 $x = x^+ - x^-$, where $x^+, x^- \succeq 0$.
- This yields the problem

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

  which is an LP in standard form, with variables x^+, x^-, and s.
- How to convert LPs into an inequality form?

Examples of Linear Programming – 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.

Piecewise-linear minimization

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

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

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

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

- Then, the inequality can be expressed as a set of m separate inequalities:

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

- This is an inequality-form LP, with variables x and t.

Solving Linear Programming Problems

- The Simplex method.
  - Developed by Dantzig in 1947.
  - One of the top 10 algorithms of the 20th century.
  - Usually very efficient for practical applications. Average-case performance: $\mathcal{O}(n^3)$.
  - Worst-case performance (though rarely happens): $\mathcal{O}(2^n)$.

Simplex Method (1/3)

We briefly describe the simplex method using an example.

- Consider the linear programming problem:

$$\begin{array}{ll} \text{maximize} & x_1 + x_2 \\ \text{subject to} & 2x_1 + x_2 \leq 12 \\ & x_1 + 2x_2 \leq 9 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \end{array}$$

- The simplex method first introduce two slack variables x_3, x_4, making it

$$\begin{array}{ll} \text{maximize} & x_1 + x_2 \\ \text{subject to} & 2x_1 + x_2 + x_3 = 12 \\ & x_1 + 2x_2 + x_4 = 9 \\ & x_1, x_2, x_3, x_4 \geq 0 \end{array}$$

Simplex Method (2/3)

- The problem

$$\begin{array}{ll} \text{maximize} & x_1 + x_2 \\ \text{subject to} & 2x_1 + x_2 + x_3 = 12 \\ & x_1 + 2x_2 + x_4 = 9 \\ & x_1, x_2, x_3, x_4 \geq 0 \end{array}$$

can be represented by a matrix

$$\left[\begin{array}{c|cccc|c} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 1 & 0 & 12 \\ 0 & 1 & 2 & 0 & 1 & 9 \end{array}\right]$$

with initial feasible point $x = \left[\begin{array}{cccc} 0 & 0 & 12 & 9 \end{array}\right]^T$

Simplex Method (3/3)

- While the first row has positive entries, select the column with the largest value in the first row as the "pivot column" and then perform elementary row operations. Repeat until the first row does not have any positive entry.

$$\left[\begin{array}{c|cccc|c} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 1 & 0 & 12 \\ 0 & 1 & 2 & 0 & 1 & 9 \end{array}\right] \rightarrow \left[\begin{array}{c|cccc|c} 1 & 0 & \frac{1}{2} & \frac{-1}{2} & 0 & -6 \\ 0 & 1 & \frac{1}{2} & \frac{1}{2} & 0 & 6 \\ 0 & 0 & \frac{3}{2} & \frac{-1}{2} & 1 & 3 \end{array}\right]$$
$$\rightarrow \left[\begin{array}{c|cccc|c} 1 & 0 & 0 & \frac{-1}{3} & \frac{-1}{3} & -7 \\ 0 & 1 & 0 & \frac{2}{3} & \frac{-1}{3} & 5 \\ 0 & 0 & 1 & \frac{-1}{3} & \frac{2}{3} & 2 \end{array}\right]$$

- The feasible point $x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T = \begin{bmatrix} 0 & 0 & 12 & 9 \end{bmatrix}^T$ changes to $x = \begin{bmatrix} 6 & 0 & 0 & 3 \end{bmatrix}^T$ and finally $x = \begin{bmatrix} 5 & 2 & 0 & 0 \end{bmatrix}^T$.

Solving Linear Programming Problems

- The Simplex method.
  - Developed by Dantzig in 1947.
  - One of the top 10 algorithms of the 20th century.
  - Usually very efficient for practical applications. Average-case performance: $\mathcal{O}(n^3)$.
  - Worst-case performance (though rarely happens): $\mathcal{O}(2^n)$.
- Interior-point methods
  - Developed since the late 70s' [Khachiyan1979, Karmarkar1984] with worst-case performance $\mathcal{O}(n^4)$, $\mathcal{O}(n^{3.5})$, respectively.
  - The average-case performance is still not better than the Simplex method.