Lecture Notes for ORIE 6300: Mathematical Programming I

Damek Davis[*]

December 13, 2023

Contents

[*] School of Operations Research and Information Engineering, Cornell University, Ithaca, NY 14850, USA; people.orie.cornell.edu/dsd95/.

1 This Course

Optimization is a broad mathematical discipline widely used in many applied sciences, including operations research, machine learning, signal processing, and statistics. The systematic study of optimization began with the calculus of variations. Since then, the subject has flourished: from the celebrated simplex method in linear programming to the mature theories of duality and algorithmic complexity in convex optimization, it has now enjoyed over 60 years of theoretical and computational advances. Key to these advances were the development of nonsmooth calculus and the recognition of its role in first-order optimality conditions. Based on duality, nonsmooth calculus, and numerical linear algebra techniques, the subject now enjoys an extensive algorithmic toolbox containing practical and provably "efficient" numerical methods. The purpose of this class is to give you a firm working knowledge of the techniques and results of modern optimization by developing the following set of core skills:

- Structure and special cases: recognize and exploit convexity and its special cases (linear and conic programming); recognize well-structured nonconvex problems (smoothness, composite structure, and regularity conditions).
- Duality: learn to "take a dual;" recognize when "strong duality" holds; exploit duality in algorithms.
- Nonsmooth calculus: compute first-order necessary optimality conditions with nonsmooth calculus (subdifferentials, normal cones, and the chain rule); compute the sensitivity of optimization problems with respect to perturbations of input data (value functions and Lagrange multipliers).
- Algorithms: learn a toolbox of algorithms (simplex, interior point, and first-order methods); choose appropriate algorithms by understanding tradeoffs induced by problem structure; characterize algorithmic complexity; numerically implement algorithms.

1.1 Prerequisites

These notes assume familiarity with (both are URLs)

- This linear algebra review.
- Chapter 1.1 of Borwein and Lewis

It would help if you read these before the first class.

1.2 Prior course website

You can view the course website at `https://people.orie.cornell.edu/dsd95/orie6300.html`.

1.3 Acknowledgements

Jim Renegar's ORIE 6300 course from Fall 2018 has influenced my presentation in these notes.

2 What is Optimization?

Skills. Recognize and exploit convexity and its special cases

2.1 A Mathematical Program

This course is on mathematical programming, a phrase synonymous with optimization. The term "mathematical programming" has (somewhat) fallen out of favor, but you will still hear the word "programming" when referring to concrete problem classes, e.g., "linear programming."

The overarching goal of this course is to minimize or maximize a given function over a constraint set. For us, the variables we optimize over will always live in $\mathbb{R}^d$ for some integer d. Setting the stage, we wish to minimize a function f over the feasible region $\mathcal{X}$. To state this formally, we would write

$$\begin{aligned} &\text{minimize} \quad && \underbrace{f(x)}_{\textbf{objective function}} \\ &\text{subject to}: x \in && \underbrace{\mathcal{X}}_{\substack{\textbf{feasible region/} \\ \textbf{constraint set}}} \end{aligned} \tag{$\mathcal{MP}$}$$

If $\bar{x}$ "solves" ($\mathcal{MP}$), we call it an *optimal solution.* Of course, this means that $f(\bar{x}) \leq f(x)$ for all other $x \in \mathcal{X}$. We often use the notation $\operatorname{argmin}_{x \in \mathcal{X}} f(x)$ to denote the set of optimal solutions to a mathematical program. Similarly, we say that $\bar{x}$ locally minimizes f over $\mathcal{X}$ there is some $\varepsilon > 0$ so that $f(x) \geq f(\bar{x})$ for all $x \in B_\varepsilon(\bar{x})$.

In modern optimization, functions are *extended valued* in the sense that they take values that may be real or infinite: $\mathbb{R} \cup \{-\infty, +\infty\}$. Thus it is common to move the constraint $\mathcal{X}$ from ($\mathcal{MP}$) into the objective by setting

$$F(x) = \begin{cases} f(x) & \text{if } x \in \mathcal{X} \\ +\infty & \text{otherwise.} \end{cases}$$

We then shift our attention to minimizing F. We almost exclusively require functions never to take the value $-\infty$. With this convention, you can check that a point $\bar{x}$ minimizes F if and only if it minimizes f over $\mathcal{X}$. To avoid thinking about infinite values, one might also write $f \colon \mathcal{X} \to \mathbb{R}$ for the function F. In modern optimization, however, one associates points in the *domain* of f with those taking finite values: $\operatorname{dom}(f) = \{x \in \mathbb{R}^d \mid f(x) < +\infty\}$. A function is called *proper* if it never takes value $-\infty$ and $\operatorname{dom}(f) \neq \emptyset$.

The most basic question one can ask about ($\mathcal{MP}$) is whether it has a minimizer. Weierstrass' famous theorem provides an answer to this question.

Theorem 2.1 (Weierstrass)**.** *Let $\mathcal{X}$ be a closed set, and suppose $f\colon \mathcal{X} \to \mathbb{R}$ is continuous and has bounded sublevel sets:*

$$\{x \in \mathcal{X}\colon\ f(x) \leq a\} \text{ is bounded for all } a \in \mathbb{R}.$$

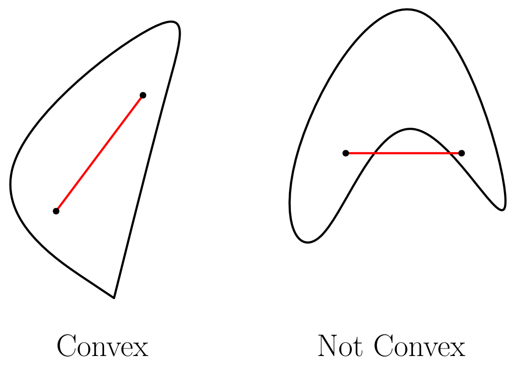
Then f has a minimizer.

Proof. Let $x_1, x_2, \ldots$ be a sequence in $\mathcal{X}$ with $f(x_k) \to \inf f$. The sequence $f(x_k)$ approaches a number less than $+\infty$ and is therefore upper bounded by a number a, meaning $x_k \in \{x \in \mathcal{X}\colon f(x) \leq a\}$. This a-sublevel set is compact: it is *bounded* by assumption and *closed* due to continuity. Hence, there exists a convergent subsequence $x_{i_1}, x_{i_2}, \ldots$ with $\bar{x} = \lim_{j\to\infty} x_{i_j}$, and since $\mathcal{X}$ is closed, we have $\bar{x} \in \mathcal{X}$. Therefore, by continuity $f(\bar{x}) = \lim_{j\to\infty} f(x_{i_j}) = \inf f$, meaning $\bar{x}$ minimizes f. □

Mere attainment of the minimal value will not be our only goal in this course. We're more broadly interested in elucidating useful *structures* that help us answer questions about the effort required to solve a problem or the behavior of solutions and optimal values under perturbations. Linearity and, more generally, convexity will supply us with such answers. We will focus most of our effort on these structures.

2.2 Fundamental Structures: Linearity and Convexity

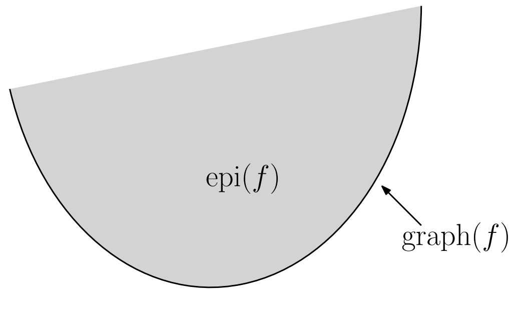
A *linear program* is a mathematical program with a linear objective $f(x) = c^T x = \langle c, x\rangle$ and a *polyhedral constraint* set, meaning $\mathcal{X}$ consists of points satisfying a finite list of linear equalities and inequalities. Such an $\mathcal{X}$ is called a *polyhedron.* Linear programs (LPs) are the most straightforward problem class in the course.



More generally, we will be interested in *convex optimization*: minimization of convex functions over convex sets. Convexity is a geometric property that is simple to state:

> $\mathcal{X}$ is *convex set* if for all $x, y \in \mathcal{X}$, the line segment between x and y lies in $\mathcal{X}$.

Algebraically, we would say: $\forall x, y \in \mathcal{X}, t \in [0,1]$, it holds $tx + (1-t)y \in \mathcal{X}$. Beyond sets, a function is convex if the region above its graph called the *epigraph*, is convex:



> a function f is *convex* if $\text{epi}(f) = \{(x,t)\colon f(x) \leq t\} \subseteq \mathbb{R}^{d+1}$ is convex.

Perhaps more familiar is the equivalent algebraic condition: the set $\mathrm{dom}(f)$ is convex and $\forall x, y \in \mathrm{dom}(f), t \in [0,1]$, it holds $f(tx + (1-t)y) \leq tf(x) + (1-t)f(y)$. With these definitions, we define a *convex program* as an $(\mathcal{MP})$ wherein f and $\mathcal{X}$ are convex.

It would help if you convinced yourself of the equivalences thus stated; we will use them freely throughout the course.

2.2.1 Aside: Ubiquity of Linear Objectives

Stepping back to $(\mathcal{MP})$, the nonlinear objective f can always be replaced by a linear one:

$$\begin{aligned} &\min_{x,t} t \\ \text{subject to} : &\; x \in \mathcal{X} \\ &\; t \geq f(x). \end{aligned} \tag{$\mathcal{EPI}$}$$

For this problem, the feasible region is a higher dimensional set $\hat{\mathcal{X}} = \{(x,t) \mid x \in \mathcal{X}, f(x) \leq t\} \subseteq \mathbb{R}^{d+1}$ and the objective is a linear function $\hat{f}(x,t) = \langle (0_d, 1), (x,t) \rangle$. Since $\hat{\mathcal{X}}$ is the epigraph of the $\mathbb{R} \cup \{+\infty\}$-valued function F, we call this transformed problem the *epigraphical form*. We will use this transformation to simplify our study of duality theory.

2.3 Basic Consequences of Convexity

We end this lecture with a few basic consequences of convexity (where A is a linear map):

- The set of optimal solutions to a convex program is convex.
- Intersections of convex sets are convex.
- Cartesian products of convex sets are convex.
- If $\mathcal{X}_1$ and $\mathcal{X}_2$ are convex, then so is $\mathcal{X}_1 + \mathcal{X}_2 = \{x_1 + x_2 \colon x_1 \in \mathcal{X}_1, x_2 \in \mathcal{X}_2\}$.
- If $\mathcal{X} \subseteq \mathbb{R}^d$ is a convex set, then $\{Ax \colon x \in \mathcal{X}\}$ is convex.
- If $\mathcal{Y} \subseteq \mathbb{R}^m$ is a convex set, then $\{x \in \mathbb{R}^d \colon Ax \in \mathcal{Y}\}$ is convex.
- The set $\{Ax \colon x \in \mathcal{X}\}$ is not necessarily closed, even when $\mathcal{X}$ is closed.

You should write a proof of the first six facts and provide a counterexample to the seventh one. Related to the 7th fact is the following positive result, which will reappear in our study of duality theory. (You will prove this fact in your first recitation.)

Theorem 2.2. *Let $\mathcal{X}$ be a polyhedron and let $A \in \mathbb{R}^{n \times d}$ be a matrix. Then the set $\{Ax \in \mathbb{R}^d \colon x \in \mathcal{X}\}$ is also a polyhedron.*

2.4 Exercises

Exercise 2.1. Prove the following basic consequences of convexity:

1. The set of optimal solutions to a convex program is convex.
2. Intersections of convex sets are convex.
3. Cartesian products of convex sets are convex.
4. If $\mathcal{X}_1$ and $\mathcal{X}_2$ are convex, then so is $\mathcal{X}_1 + \mathcal{X}_2 = \{x_1 + x_2 \colon x_1 \in \mathcal{X}_1, x_2 \in \mathcal{X}_2\}$.

5. If $\mathcal{X} \subseteq \mathbb{R}^d$ is a convex set and A is a matrix, then $\{Ax \colon x \in \mathcal{X}\}$ is convex.
6. If $\mathcal{Y} \subseteq \mathbb{R}^m$ is a convex set and A is a matrix, then $\{x \in \mathbb{R}^d \colon Ax \in \mathcal{Y}\}$ is convex.
7. The set $\{Ax \colon x \in \mathcal{X}\}$ is not necessarily closed, even when $\mathcal{X}$ is closed.
8. A convex set $\mathcal{X} \subseteq \mathbb{R}^d$ has a convex closure.
9. Let $\mathcal{X}$ be a closed convex set and let $x \in \mathcal{X}$. Show that $\mathcal{N}_{\mathcal{X}}(x)$ is a closed convex cone, meaning $\mathcal{N}_{\mathcal{X}}(x)$ is closed and convex and for all $v \in \mathcal{N}_{\mathcal{X}}(x)$ and $t \geq 0$, the inclusion $tv \in \mathcal{N}_{\mathcal{X}}(x)$ holds.

Exercise 2.2. Let $\mathcal{X} \subseteq \mathbb{R}^d$. We define the convex hull as the smallest convex set containing $\mathcal{X}$ and denote this set by $\mathrm{conv}(\mathcal{X})$. Here, the word "smallest" means that whenever a convex set $\mathcal{Y} \subseteq \mathbb{R}^d$ contains $\mathcal{X}$, it must be the case that $\mathcal{Y}$ contains $\mathrm{conv}(\mathcal{X})$ as well. Prove that

$$\mathrm{conv}(\mathcal{X}) = \left\{ x \in \mathbb{R}^d \colon x = \sum_{i=1}^{n_x} \alpha_i x_i \text{ for some } n_x > 0,\ x_i \in \mathcal{X}, \text{ and } \alpha_i \in [0,1] \text{ with } \sum_{i=1}^{n_x} \alpha_i = 1 \right\}.$$

Exercise 2.3. Consider the ℓ_1 ball:

$$\mathcal{X} := \left\{ x \in \mathbb{R}^d \colon \sum_{i=1}^{d} |x_i| \leq 1 \right\}.$$

1. Prove that $\mathcal{X}$ is a *polyhedron* (i.e., the intersection of finitely many linear inequalities, meaning $\mathcal{X} = \{x \in \mathbb{R}^d \colon a_i^T x \leq b_i \text{ for } i = 1, \ldots, n\}$ for a set of vectors a_i and scalars b_i). How many inequalities are needed to describe $\mathcal{X}$ (how large is n)?

2. A *lifting* of a polyhedron $\mathcal{P}_1 \subseteq \mathbb{R}^d$ is a description of the form $\mathcal{P}_1 = \{Ax \colon x \in \mathcal{P}_2\}$ where $\mathcal{P}_2 \subseteq \mathbb{R}^m$ is a polyhedron and $A \in \mathbb{R}^{d \times m}$ is a matrix.

   Find a lifting of $\mathcal{X}$ to $\mathbb{R}^{2d}$, where the associate polyhedron in $\mathbb{R}^{2d}$ is defined by at most $2d + 1$ inequalities.

Exercise 2.4. Let $f \colon \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be a convex function. Prove that any local minimum of f is a global minimum.

Exercise 2.5 (Weierstrass). Let $f \colon \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be a function with a closed epigraph and bounded sublevel sets. Show that f has a minimizer. (Hint: consider the epigraphical form)

Exercise 2.6 (Avoiding $-\infty$). Let $f : \mathbb{R}^d \to [-\infty, \infty]$ be a convex function. Suppose there is a point $x \in \mathrm{int}(\mathrm{dom}(f))$ and $f(x) \in \mathbb{R}$. Show that $f(x') > -\infty$ for all $x' \in \mathbb{R}^d$.

3 First-Order Optimality and Normal Cones

Skills. Recognize well-structured nonconvex problems (smoothness); compute first-order necessary optimality conditions with nonsmooth calculus (normal cones).

In every calculus course, we learn a basic technique for finding minimizers of differentiable functions $f \colon (a, b) \to \mathbb{R}$:

if f is minimized over (a, b) at a number t, then $f'(t) = 0$.

This condition is called *Fermat's rule* and encodes the intuition that a function must be "flat" at minimizers. Generalizing to higher dimensions, consider $(\mathcal{MP})$ where $f\colon \mathbb{R}^2 \to \mathbb{R}$ is differentiable and $\mathcal{X} = \{x \in \mathbb{R}^2 \colon g(x) = 0\}$ for a differentiable function $g\colon \mathbb{R}^2 \to \mathbb{R}$. For such problems, the Lagrange multiplier rule provides a similarly useful criterion: at minimizers $\bar{x}$, the gradient of f must be "normal" to $\mathcal{X}$ at $\bar{x}$, implying $\nabla f(\bar{x}) = \lambda \nabla g(\bar{x})$ for some $\lambda \in \mathbb{R}$. Generalizing these results to higher dimensions is the purpose of this lecture, focusing specifically on differentiable functions and closed convex constraint sets.

3.1 Differentiability and the Gradient

To generalize Fermat's rule to constrained optimization problems, we introduce a notion of smoothness called *Fréchet* differentiability. This notion makes precise the intuition that a function is differentiable if, at every point, we can approximate it by a linear function up to first order.

Definition 3.1. *A function $f\colon \mathcal{O} \to \mathbb{R}$ defined on an open set $\mathcal{O} \subseteq \mathbb{R}^d$ is differentiable at $x \in \mathcal{O}$ if there exists $v \in \mathbb{R}^d$ and $o_x\colon \mathbb{R}^d \to \mathbb{R}$ with*

$$f(y) = f(x) + \langle v, y - x\rangle + o_x(y) \qquad \textit{where} \qquad \lim_{y \to x} \frac{o_x(y)}{\|y - x\|} = 0. \tag{$\mathcal{DIFF}$}$$

One immediate consequence of this definition is that any such v is unique. Indeed, supposing v_1 and v_2 both satisfy the above definition (with a particular o_x and $\hat{o}_x$) and setting $y = x + \lambda(v_1 - v_2)$ for $\lambda > 0$, we have

$$\begin{cases} \langle v_1, y - x\rangle = f(y) - f(x) + o_x(y) \\ \langle v_2, y - x\rangle = f(y) - f(x) + \hat{o}_x(y) \end{cases} \implies \frac{\langle v_1 - v_2, y - x\rangle}{\|y - x\|} = \frac{o_x(y)}{\|y - x\|} - \frac{\hat{o}_x(y)}{\|y - x\|}.$$

We then find that $v_1 = v_2$ by letting $\lambda \searrow 0$ since the little-o terms tend to zero and the inner product is equal to $\|v_1 - v_2\|$ (check!). Since any such v is unique, we use the familiar notation $\nabla f(x)$ and call it the *(Fréchet) gradient* of f. To illustrate, fix a $z \in \mathbb{R}^d$ and define $f(x) = \frac{1}{2}\|x - z\|^2$. Then (check!)

$$\nabla f(x) = \nabla \left[\frac{1}{2}\| \cdot - z\|^2\right](x) = x - z \qquad \text{for all } x \in \mathbb{R}^d. \tag{3.1}$$

This gradient will reemerge in the next lecture.

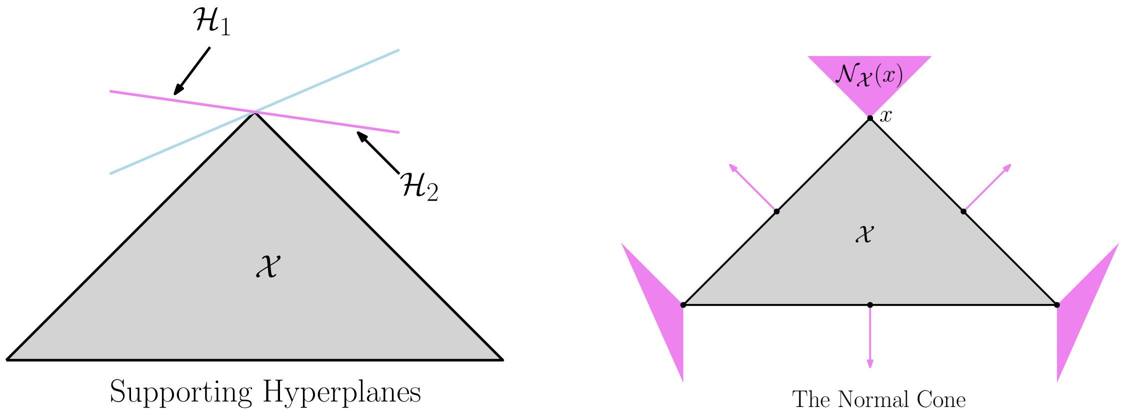
While we could now give a higher dimensional generalization of Fermat's rule (do this exercise!), we instead move to a more challenging and interesting issue: constrained minimization.

3.2 Normal Cones and First-Order Optimality: A First Look

From linear algebra, we know that there is a duality between hyperplanes and normal vectors: all points of a hyperplane are orthogonal to a subspace of *normal vectors* and conversely any vector is normal to the hyperplane of vectors orthogonal to it. A similar duality exists for closed convex sets, based on a fundamental property of such objects: every boundary point admits at least one supporting hyperplane (not obvious!). We then collect the normals to these hyperplanes into a single dual object called the *normal cone.*

Definition 3.2. *If $\mathcal{X} \subseteq \mathbb{R}^d$ is closed and convex, then at any $x \in \mathcal{X}$, define the* normal cone:

$$\mathcal{N}_{\mathcal{X}}(x) = \{v \in \mathbb{R}^d \mid \langle v, y - x\rangle \leq 0,\ \forall y \in \mathcal{X}\} \qquad \forall x \in \mathcal{X}.$$



Supporting Hyperplanes

The Normal Cone

Figure 1: Supporting Hyperplanes and Normal Cones

The normal cone is undefined at points $y \notin \mathcal{X}$. We adopt the convention for such points that $\mathcal{N}_{\mathcal{X}}(y) = \emptyset$. To get a basic understanding of normal cones, you should carefully prove the following identities:

- $\mathcal{N}_{\{x\}}(x) = \mathbb{R}^d$.
- $\mathcal{N}_{\mathbb{R}^d}(x) = \{0\}$
- For $\mathcal{X} = [0, 1]$, we have
$$\mathcal{N}_{\mathcal{X}}(t) = \begin{cases} \mathbb{R}_- & \text{if } t = 0 \\ \mathbb{R}_+ & \text{if } t = 1 \\ 0 & \text{if } t \in (0, 1). \end{cases}$$
- For $\mathcal{X} = B_1(0)$, we have
$$\mathcal{N}_{\mathcal{X}}(x) = \begin{cases} \{0\} & \text{if } \|x\| < 1 \\ \mathbb{R}_+ x & \text{if } \|x\| = 1. \end{cases}$$
- If $\mathcal{X}$ is a subspace of $\mathbb{R}^d$, we have $\mathcal{N}_{\mathcal{X}}(x) = \mathcal{X}^{\perp}$.

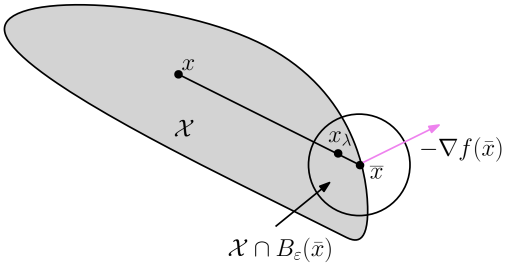
Finally, we note the following fact, which you should be able to prove:

Exercise 3.1. *Let $\mathcal{X}$ be a closed convex set and suppose that $x \in \mathcal{X}$. Then $\mathcal{N}_{\mathcal{X}}(x)$ is a closed convex set.*

Normal cones feature in the optimality conditions of constrained optimization, a result of the following general principle: if a point $\bar{x}$ is an optimal solution to $(\mathcal{MP})$, then $-\nabla f(\bar{x})$ *should* point outside $\mathcal{X}$. Why? Because the direction of maximum instantaneous decrease is parallel to the negative gradient. We now formalize this intuition by connecting normality and optimality.

Theorem 3.3 (First Order Optimality)**.** *Suppose $\bar{x}$ is a local minimizer of $f\colon \mathbb{R}^d \to \mathbb{R}$ on a closed convex set $\mathcal{X} \subseteq \mathbb{R}^d$. Then if f is differentiable $\bar{x}$, it holds:*

$$-\nabla f(\bar{x}) \in \mathcal{N}_{\mathcal{X}}(\bar{x}).$$



Proof. We want to show that $\langle -\nabla f(\bar{x}), x - \bar{x}\rangle \leq 0$ for all $x \in \mathcal{X}$. To that end, we fix a vector $x \in \mathcal{X}$ and a scalar $\lambda \in [0,1]$ and define $x_\lambda := (1-\lambda)x + \lambda\bar{x}$. In order to use the local optimality property of $\bar{x}$, we let $\varepsilon > 0$ be small enough that $\bar{x}$ minimizes f on $\mathcal{X} \cap B_\varepsilon(\bar{x})$. By the definition of x_λ, there must be a $\bar{\lambda} > 0$ such that for all $\lambda \geq \bar{\lambda}$, we have the inclusion $x_\lambda \in \mathcal{X} \cap B_\varepsilon(\bar{x})$. Consequently, we have $f(\bar{x}) \leq f(x_\lambda)$ for all $\lambda \in [\bar{\lambda}, 1]$.

Fix such a $\lambda \in [\bar{\lambda}, 1]$. Then as long as $x \neq \bar{x}$, it holds

$$\frac{x - \bar{x}}{\|x - \bar{x}\|} = \frac{x_\lambda - \bar{x}}{\|x_\lambda - \bar{x}\|} \qquad \text{(check!)}.$$

Therefore,

$$\frac{\langle -\nabla f(\bar{x}), x - \bar{x}\rangle}{\|x - \bar{x}\|} = \frac{\langle -\nabla f(\bar{x}), x_\lambda - \bar{x}\rangle}{\|x_\lambda - \bar{x}\|} = \frac{f(\bar{x}) - f(x_\lambda)}{\|x - x_\lambda\|} + \frac{o_{\bar{x}}(x_\lambda)}{\|x - x_\lambda\|} \leq \frac{o_{\bar{x}}(x_\lambda)}{\|x - x_\lambda\|}.$$

Letting $\lambda \to 1$, the left-hand side is constant, and the right-hand side tends to zero. This completes the proof. □

With this theorem, we can immediately deduce first-order optimality conditions for linearly constrained optimization problems.

Corollary 3.4. *Assume the setting of Theorem 3.3 and suppose that $\mathcal{X} = \{x \in \mathbb{R}^d \colon Ax = b\}$ for a matrix $A \in \mathbb{R}^{m \times d}$ and a vector $b \in \mathbb{R}^m$. Then there exists a vector $y \in \mathbb{R}^m$ so that*

$$\nabla f(\bar{x}) = A^T y.$$

Proof. First, assume $b = 0$. Then $\mathcal{X}$ is a subspace, so $\mathcal{N}_{\mathcal{X}}(\bar{x}) = \mathcal{X}^{\perp} = \ker(A)^{\perp} = \operatorname{range}(A^T)$.

You can see the general case in a few different ways:

- Show by direct computation that $\mathcal{N}_{\mathcal{X}}(x) = \operatorname{range}(A^T)$.
- In greater generality, show that if $\mathcal{X}$ is an affine space (i.e., a shift of a vector space), then $\mathcal{N}_{\mathcal{X}}(x) = (\mathcal{X} - \mathcal{X})^{\perp}$.
- In the greatest generality, you can proceed in two steps
  - First show that for any closed convex $\mathcal{X}$ and $y \in \mathbb{R}^d$, we have $N_{\mathcal{X}+y}(x+y) = N_{\mathcal{X}}(x)$ for all $x \in \mathcal{X}$.
  - Second, show that $\{x \in \mathbb{R}^d \colon Ax = b\} = \ker(A) + y$ for some $y \in \mathbb{R}^d$.

The third technique states that normals are *invariant under shifts.* □

Interpretation: Lagrange Multipliers. The components of the vector y in the above corollary are *Lagrange multipliers.* We can see this by considering the problem

$$\begin{aligned} &\text{minimize } f(x) \\ &\text{subject to : } \begin{cases} a_1^T x = b_1 \\ \quad\vdots \\ a_m^T x = b_m \end{cases}, \end{aligned}$$

where a_i^T are the rows of A. Then defining the functions $g_i(x) = a_i^T x - b$, the Lagrange multiplier rule states that at any local minimizer $\bar{x}$, there exists multipliers $y_1, \ldots, y_n$ such that $\nabla f(\bar{x}) = \sum_{i=1}^m y_i \nabla g_i(\bar{x}) = \sum_{i=1}^m y_i a_i = A^T y$.

3.3 Exercises

Exercise 3.2 (Normals at Interior Points). Suppose that $\mathcal{X}$ is a closed convex set and let $x \in \operatorname{int}(\mathcal{X})$. Prove that $\mathcal{N}_{\mathcal{X}}(x) = \{0\}$.

Exercise 3.3 (The Rayleigh Quotient; see Exercise 6 of Chapter 2.1 in Borwein and Lewis.)**.**

1. Let $f \colon \mathbb{R}^d \backslash \{0\} \to \mathbb{R} \cup \{+\infty\}$ be continuous, satisfying $f(\lambda x) = f(x)$ for all $\lambda > 0$ in $\mathbb{R}$ and nonzero x in $\mathbb{R}^d$. Prove f has a minimizer.

2. Given a symmetric matrix $A \in \mathbb{R}^{d \times d}$, define a function $g(x) = x^T Ax/\|x\|^2$ for nonzero $x \in \mathbb{R}^d$. Prove that g has a minimizer.

3. Calculate $\nabla g(x)$ for nonzero x.

4. Deduce that minimizers of g must be eigenvectors and calculate the minimum value.

4 Start of Duality: Projections and Hahn-Banach

Skills. Recognize and exploit convexity and its special cases; Compute first-order necessary optimality conditions with nonsmooth calculus.

In linear algebra, we learn that every subspace V of $\mathbb{R}^d$ can be *paired* with another subspace, denoted by $V^\perp$ and called the orthogonal complement, and that when the complement operation is applied to the orthogonal space, it returns $V^\perp$ to V: $(V^\perp)^\perp = V$. The orthogonal complement operation is an example of a *duality pairing*, and similar pairings are often available for objects that exhibit convexity. The pairing key to our study is the one between two convex optimization problems: the *primal* problem (the original problem of *primary* interest) and its associated *dual*. This course will show several duality theorems that connect these seemingly disparate problems and illuminate the behavior of both. For example, we will show that the primal and dual optimal values often coincide (strong duality) and that the size of dual optimal solutions elucidates how wildly the optimal value can vary as we change problem data (sensitivity). Duality also underlies some of the most powerful algorithms in convex optimization, namely the class of *primal-dual algorithms*, which place primal and dual on an equal footing by solving both problems in tandem.

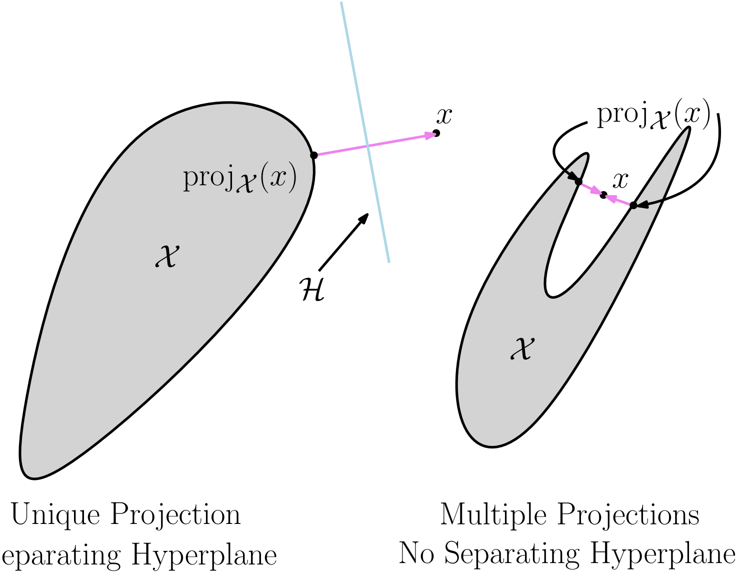


Figure 2: Projections and Separating Hyperplanes

The theory underlying duality is based on an elementary geometric consequence of convexity: if a point sits outside a closed convex set, then a hyperplane must separate it from the set. This observation is a geometric variant of the celebrated Hahn-Banach theorem from functional analysis, and its illuminating proof is summarized in Figure 2. Briefly (using the notation of the figure) we construct a hyperplane separating x from $\mathcal{X}$ in two steps: we first find the closest point to x in $\mathcal{X}$ and denote it by $\mathrm{proj}_{\mathcal{X}}(x)$; then we choose the separating hyperplane to be the one passing through $(1/2)x + (1/2)\mathrm{proj}_{\mathcal{X}}(x)$ with normal vector $x - \mathrm{proj}_{\mathcal{X}}(x)$. Rigorously grounding this argument will be the goal of this lecture.

4.1 Projections: Existence, Uniqueness, and Characterization

Key to the proof outlined above was the existence of a projection. It turns out existence is insufficient to imply a separating hyperplane since, as Figure 2 illustrates, projections can exist when separating hyperplanes do not. In finite-dimensional spaces, like $\mathbb{R}^d$, if a set has the property that projections always exist and are unique, there is always a separating hyperplane (nontrivial!). Rather than pursue this challenging theorem, we will focus on closed convex sets, showing that projections exist, are unique, and are characterized by an *inclusion.*

Theorem 4.1 (Projections)**.** *Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a closed convex set and suppose $x \notin \mathcal{X}$. Then there is a unique point $\mathrm{proj}_{\mathcal{X}}(x) \in \mathcal{X}$ that is nearest to x. Moreover, $\mathrm{proj}_{\mathcal{X}}(x)$ is the unique point $y \in \mathcal{X}$ that satisfies the inclusion:*

$$x - y \in \mathcal{N}_{\mathcal{X}}(y). \tag{4.1}$$

Proof. **Existence.** Define $f(y) = \frac{1}{2}\|y - x\|^2$. Note that $f : \mathcal{X} \to \mathbb{R}$ has bounded sublevel sets since

$$\left\{ y \in \mathcal{X} \mid \frac{1}{2}\|y - x\|^2 \leq a \right\} \subseteq B_{\sqrt{2a}}(x) \qquad \text{for all } a \in \mathbb{R}.$$

Thus, by Weierstrass' theorem (Theorem 2.1), the function f has a minimizer $\bar{x} \in \mathcal{X}$. This point is the nearest point to x in $\mathcal{X}$. (Notice that we did not use convexity in this part.)

Uniqueness. By first-order optimality conditions (Theorem 3.3), any minimizer $\bar{x}$ of f over $\mathcal{X}$ must satisfy

$$x - \bar{x} = -\nabla f(\bar{x}) \in \mathcal{N}_{\mathcal{X}}(\bar{x}),$$

where the gradient identity follows by (3.1). Thus, it suffices to show that solutions to (4.1) are unique. To that end, suppose $y_1 \in \mathcal{X}$ and $y_2 \in \mathcal{X}$ satisfy (4.1). Then by definition of $\mathcal{N}_{\mathcal{X}}(\bar{x})$

$$\langle x - y_1, y_2 - y_1 \rangle \leq 0 \qquad \text{and} \qquad \langle x - y_2, y_1 - y_2 \rangle \leq 0.$$

Adding these inequalities, we get $\|y_2 - y_1\|^2 \leq 0$, as desired. □

4.2 Hahn-Banach: The Separating Hyperplane Theorem

With projections in hand, we can now prove the Hahn-Banach theorem.

Theorem 4.2 (Hahn-Banach)**.** *Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a closed convex set. For any point $x \notin \mathcal{X}$, the point x is strictly separated from $\mathcal{X}$ by the hyperplane that passes through $(1/2)x + (1/2)\mathrm{proj}_{\mathcal{X}}(x)$ and has normal vector $a := x - \mathrm{proj}_{\mathcal{X}}(x)$. More specifically,*

$$\max\{\langle a, y \rangle \colon y \in \mathcal{X}\} < \langle a, x \rangle. \tag{4.2}$$

where “max” *signifies the attainment of the maximal value.*

Proof. Denote $\bar{x} = \mathrm{proj}_{\mathcal{X}}(x)$ and notice that $a \in \mathcal{N}_{\mathcal{X}}(\bar{x})$ by (4.1). Thus, for any $y \in \mathcal{X}$

$$\langle a, y \rangle = \langle x - \bar{x}, y \rangle = \underbrace{\langle x - \bar{x}, y - \bar{x} \rangle}_{\leq 0} + \langle x - \bar{x}, \bar{x} \rangle \leq \underbrace{\langle x - \bar{x}, \bar{x} \rangle}_{\substack{= \langle a, \bar{x} \rangle \\ \text{max attained} \\ \text{at } \bar{x}}} = \underbrace{\langle x - \bar{x}, x - \bar{x} \rangle}_{= \|x - \bar{x}\|^2} + \langle x - \bar{x}, x \rangle < \langle a, x \rangle.$$

This completes the proof. □

A simple consequence of Hahn-Banach is the existence of separating hyperplanes in the sense of Figure 1. Note that by the duality between hyperplanes and their normals, a dual statement of this result is that normals exist at every boundary point of $\mathcal{X}$.

Corollary 4.3 (Existence of Supporting Hyperplanes)**.** *Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a closed convex set. Then for any point $x \in \operatorname{bdry} \mathcal{X}$, there exists a normal vector $v \in \mathcal{N}_{\mathcal{X}}(x)$ with $\|v\| \neq 0$.*

Proof. Since $x \in \operatorname{bdry} \mathcal{X}$, there exists a sequence $x_1, x_2, \ldots$ of points in the complement of $\mathcal{X}$ with the property that $\lim_{i\to\infty} x_i = x$. Let a_i be the normal vectors guaranteed to exist by Hahn-Banach (Theorem 4.2). Notice that Equation (4.2) is invariant to the scaling of a, so we can assume that $\|a_i\| = 1$ without loss of generality. By compactness of the unit sphere, we can also assume without loss of generality that the sequence a_i converges to a limit point a, which necessarily has unit norm. Then for all $y \in \mathcal{X}$, we have

$$\langle a, x\rangle = \lim_{i\to\infty} \langle a_i, x_i\rangle \geq \lim_{i\to\infty} \langle a_i, y\rangle = \langle a, y\rangle.$$

Thus, we have $\langle a, y - x\rangle \leq 0$ for all $y \in \mathcal{X}$, meaning $a \in \mathcal{N}_{\mathcal{X}}(x)$. □

A direct consequence of Corollary 4.3 and Hahn-Banach is the following representation theorem for convex sets: any closed convex set is the intersection of half-spaces.

Corollary 4.4. *Every closed convex set is the intersection of (possibly infinitely many) linear inequalities.*

Proof. The set of all normal vectors will supply the linear inequalities to $\mathcal{X}$:

$$\mathcal{P} := \{y \in \mathbb{R}^d \colon\ (\forall x \in \mathcal{X}, \forall v \in \mathcal{N}_{\mathcal{X}}(x))\ \langle v, y - x\rangle \leq 0\}.$$

We claim that $\mathcal{X} = \mathcal{P}$. Clearly $\mathcal{X} \subseteq \mathcal{P}$. To prove the opposite inclusion, we show that a point in the complement of $\mathcal{X}$ cannot be in $\mathcal{P}$. Indeed, if $z \in \mathbb{R}^d \backslash \mathcal{X}$, then Hahn-Banach supplies the normal $a := z - \operatorname{proj}_{\mathcal{X}}(z) \in \mathcal{N}_{\mathcal{X}}(z)$ and shows that the hyperplane normal to a separates z from $\operatorname{proj}_{\mathcal{X}}(z)$:

$$\langle a, z\rangle > \langle a, \operatorname{proj}_{\mathcal{X}}(z)\rangle.$$

Rearranging, we find $\langle a, z - \operatorname{proj}_{\mathcal{X}}(z)\rangle > 0$, implying $z \notin \mathcal{P}$. □

Combined with an exercise from the first lecture, we now have the following theorem:

Theorem 4.5 (Representation of Convex Sets)**.** *A set $\mathcal{X}$ is closed and convex if, and only if, it is the intersection of (possibly infinitely many) linear inequalities.*

4.3 Exercises

Exercise 4.1. Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a closed convex set. For any $x \in \mathcal{X}$, define the *proximal normal cone*

$$\mathcal{N}^P_{\mathcal{X}}(x) = \left\{v \in \mathbb{R}^d \colon x = \operatorname{proj}_{\mathcal{X}}(x + v)\right\}.$$

Prove that $\mathcal{N}_{\mathcal{X}}(x) = \mathcal{N}^P_{\mathcal{X}}(x)$.

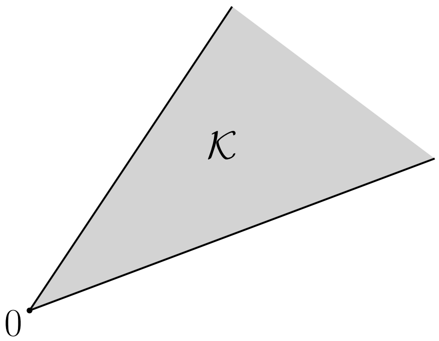
5 Conic Programs

Skills. Recognize and exploit convexity and its special cases (conic programming);

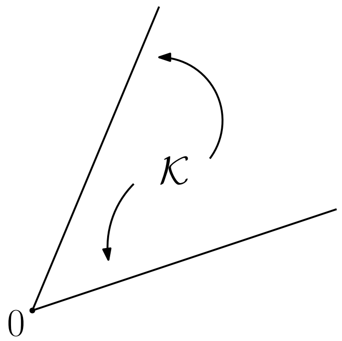
In the previous lecture, we began our search for duality: a pairing between primal and dual optimization problems. To help us find this pairing, we must first pick some particular ($\mathcal{MP}$) and then let it play the role of a primal or dual optimization problem. There is more than one way to do this, each with advantages and disadvantages. Nevertheless, we must choose a path, and the one we choose—conic duality—is motivated by geometric considerations. This path is taken without loss of generality since any convex program has an equivalent conic reformulation, a formulation we will arrive at towards the end of the lecture. We start with a more fundamental question: What is a cone?

5.1 Cones

A *cone* $\mathcal{K} \subseteq \mathbb{R}^d$ is a set with a simple geometric structure: If a cone contains a vector x, it must contain all positive scalar multiples of x. Thus, for every $x \in \mathcal{K}$ and $t > 0$, we have $tx \in \mathcal{K}$. A particularly uninteresting cone is the empty set. To eliminate this set from consideration, we always assume cones are nonempty. A more interesting cone is the normal cone of a closed convex set (you will have proved this on your first homework).



A Closed Convex Cone



A Closed Nonconvex Cone

A few immediate properties of cones follow:

- The closure of any cone must contain the origin.
- A cone $\mathcal{K} \subseteq \mathbb{R}^d$ is convex if, and only if, $\mathcal{K} + \mathcal{K} = \mathcal{K}$.
- Intersections of cones are cones.
- Cartesian products of cones are cones.
- If $\mathcal{K}_1, \mathcal{K}_2 \subseteq \mathbb{R}^d$ are cones, then $\mathcal{K}_1 + \mathcal{K}_2$ is a cone.

Suppose $A \in \mathbb{R}^{m \times d}$ is a matrix.

- If $\mathcal{K} \subseteq \mathbb{R}^d$ is a cone, then $\{Ax \colon x \in \mathcal{K}\}$ is a cone in $\mathbb{R}^m$.
- If $\mathcal{K}' \subseteq \mathbb{R}^m$ is a cone, then $\{x \colon Ax \in \mathcal{K}'\}$ is a cone in $\mathbb{R}^d$.

You should write a complete proof of these facts since we will take them for granted from now on. Having established these basic facts, let us turn our attention to a few examples.

5.1.1 Cones of Particular Significance

Historically, three types of cones have had the greatest practical significance:

Nonnegative Orthant. The cone underlying linear programming is known as the *nonnegative orthant*:

$$\mathbb{R}^d_+ = \{x \in \mathbb{R}^d \colon x_j \geq 0 \text{ for } j = 1, \ldots, d\}.$$

This cone is closed and convex. Its interior is the strictly positive orthant:

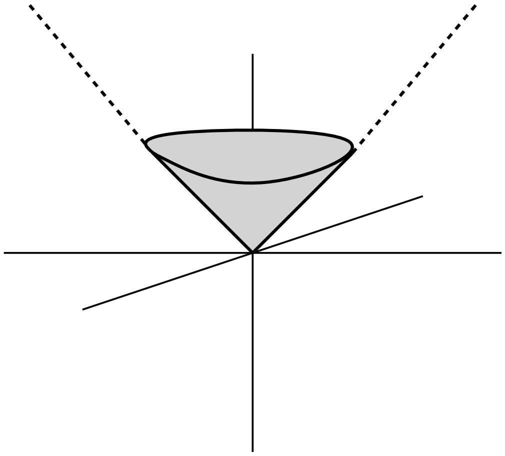
$$\mathbb{R}^d_{++} = \{x \in \mathbb{R}^d \colon x_j > 0 \text{ for } j = 1, \ldots, d\}.$$

This cone is not closed, but it is convex.

Second-Order Cone. The cone underlying *second order cone programming* is called (unsurprisingly) the *the second order cone:*

$$\mathrm{SOC}(d+1) = \{(x,t) \in \mathbb{R}^d \times \mathbb{R} \colon \|x\| \leq t\}.$$

In the literature, you will also find the names Lorentz or "ice cream cone."



A Second Order Cone

Positive Semi-Definite Cone. The cone underlying semidefinite programming is called the *positive semi-definite* (PSD) cone. It is contained in the vector space of symmetric matrices

$$\mathbb{S}^{d \times d} = \{X \in \mathbb{R}^{d \times d} \colon X^T = X\}.$$

Although this is a vector space of matrices, we can identify it with a subspace of $\mathbb{R}^{d^2}$ by "vectorizing" the matrix X.[1] The definition of this cone relies on a key fact from linear algebra: symmetric matrices only have real eigenvalues. In particular, the minimal eigenvalue $\lambda_{\min}(X)$ of a symmetric matrix X is real. Based on this fact, we define the PSD cone:

$$\mathbb{S}^{d \times d}_+ = \{X \in \mathbb{S}^{d \times d} \colon \lambda_{\min}(X) \geq 0\}.$$

[1] For example, $\begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \to \begin{bmatrix} X_{11} \\ X_{21} \\ X_{12} \\ X_{22} \end{bmatrix}$.

In the spirit of Theorem 4.5, we may alternatively write this cone as the intersection of infinitely many linear inequalities:

$$\mathbb{S}_+^{d\times d} = \{X \in \mathbb{S}^{d\times d} \colon v^T X v \geq 0 \quad \forall v \in \mathbb{R}^d\}.$$

To see the equivalence, recall from your first homework assignment that

$$\lambda_{\min}(X) = \min_{v \in \mathbb{R}^d \setminus \{0\}} \frac{v^T X v}{\|v\|^2},$$

for any symmetric $d \times d$ matrix X.

5.2 Conic Optimization Problems

To begin our discussion of conic optimization problems, we must narrow the class of objectives and feasible regions under consideration. As with the epigraphical form ($\mathcal{EPI}$), our objectives will be linear. Our constraints, on the other hand, will intermix linear and conic constraints. We consider two forms.

(First Form.) Given a matrix $A \in \mathbb{R}^{m\times d}$, a vector $b \in \mathbb{R}^m$, and a closed convex cone $\mathcal{K} \subseteq \mathbb{R}^m$, we say that

$$\mathcal{X} = \{x \in \mathbb{R}^d \colon Ax - b \in \mathcal{K}\}$$

a *conic constraint set.* A couple of basic examples follow:

- **(Affine)** If $\mathcal{K} = \{0\}$, then $\mathcal{X}$ is the set of solutions to $Ax = b$.
- **(Polyhedral)** If $\mathcal{K} = \mathbb{R}_+^d$, then $\mathcal{X} = \{x \in \mathbb{R}^d \colon Ax \geq b\}$ is a polyhedral set.[2]

Such conic constraints can be intersected, yielding new conic constraints. Indeed, suppose that for $i = 1, 2$, we have conic constraints $\mathcal{X}_i = \{x \in \mathbb{R}^d \colon A_i x - b_i \in \mathcal{K}_i\}$. Then

$$\mathcal{X}_1 \cap \mathcal{X}_2 = \left\{ x \colon \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} x - \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \in \mathcal{K}_1 \times \mathcal{K}_2 \right\}.$$

Similarly, products and Minkowski sums of conic constraint sets are also conic constraint sets (check!). Since it is closed under these basic operations, we say that this is a *universal form* for conic constraints.

(Second Form.) To illuminate the geometry of duality theory, we consider a second form of conic constraints:

$$\mathcal{X}' = \{x' \colon A'x' = b' \text{ and } x' \in \mathcal{K}'\},$$

where A' is a matrix, b' is a vector, and $\mathcal{K}'$ is a closed convex cone. This cone is the intersection of an affine set with a conic constraint. It is also universal since any

[2] When applied to vectors $z, y \in \mathbb{R}^m$, the comparison $z \geq y$, means $z_i \geq y_i$ for $i = 1, \ldots, m$.

constraint of the first form may be reduced to this form. Indeed, assuming $\mathcal{X}$ is a conic constraint of the first form, we may define a new variable $x' = \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^d \times \mathbb{R}^m$ and let

$$A' = \begin{bmatrix} A & | & -I \end{bmatrix}, \qquad b' = b, \qquad \text{and} \qquad \mathcal{K}' = \mathbb{R}^d \times \mathcal{K}.$$

Then, any point in $\mathcal{X}'$ gives rise to a point in $\mathcal{X}$ and vice versa (check!). In that sense, the first and second forms are equivalent.

In practice, one will encounter problems that fit neither the first nor the second form but are mixtures of both types of constraints. Since we must ultimately choose some problem to study, this discussion shows that we make this choice without loss of generality.

5.2.1 The Primal Conic Problem

We arrive at the primal conic problem, which will be the focus of the next few lectures:

$$\begin{aligned} \text{minimize } & c^T x \\ \text{subject to : } & Ax = b \\ & x \in \mathcal{K} \end{aligned} \tag{$\mathcal{PRIMAL}$}$$

where $c \in \mathbb{R}^d$ is a vector, $A \in \mathbb{R}^{m \times d}$ is a matrix, $b \in \mathbb{R}^m$ is a vector, and $\mathcal{K} \subseteq \mathbb{R}^d$ is a closed convex cone. Let us consider a few examples.

Linear programs. When $\mathcal{K} = \mathbb{R}^d_+$, this form is called a *linear program in standard equality form.* We will later see that linear programs have a particularly powerful duality theory.

Second Order Cone Programs. When $\mathcal{K}$ is the product of second order cones and copies of $\mathbb{R}_+$, this form is called a *second order cone program in standard equality form.* Of course, any linear program is also a second-order cone program.[3]

Semidefinite Programs. When $\mathcal{K} = \mathbb{S}^d_+$, this form is called a *semidefinite program in standard equality form.*

5.3 The Conical Form of a Convex Program

Recall the mathematical program ($\mathcal{MP}$). Building on this problem, we created the epigraphical form ($\mathcal{EPI}$), an equivalent problem that has a linear objective. A consequence of this form is that any convex program is equivalent to a convex program with a linear objective function. This section describes how such programs are equivalent to conic programs. This description relies on the following fact, which you will prove in your homework:

Exercise 5.1. *Let $\mathcal{X}$ be a closed convex set. Then*

$$\mathcal{K}_{\mathcal{X}} = \{(x,t) \colon t > 0 \text{ and } x/t \in \mathcal{X}\}$$

is a convex cone.

[3] This follows by considering zero second-order cones and using the identity $\mathbb{R}^d_+ = \mathbb{R}_+ \times \ldots \times \mathbb{R}_+$.

In general, $\mathcal{K}_{\mathcal{X}}$ is not closed since it does not contain the origin. Nonclosedness is not a problem since we will ultimately work with the closure of $\mathcal{K}_{\mathcal{X}}$, as we will see momentary. For now, we use this cone to turn any optimization problem with a linear objective into a conic optimization problem:

$$\left\{\begin{array}{c}\text{minimize } c^T x \\ \text{subject to : } x \in \mathcal{X}\end{array}\right\} \rightsquigarrow \left\{\begin{array}{c}\text{minimize } c^T x \\ \text{subject to : } t = 1 \\ \qquad\qquad (x,t) \in \mathcal{K}_{\mathcal{X}}\end{array}\right\}. \qquad (\mathcal{CONIC})$$

You should check that $\mathcal{K}_{\mathcal{X}}$ can be replaced by $\overline{\mathcal{K}}_{\mathcal{X}}$ without changing the optimal solution of the underlying problem (assuming it exists).

Disclaimer: When one takes the dual of a convex problem, it is usually unnecessary to convert the problem to conic form before doing so. Instead, there are often simpler techniques that apply in special cases. We only mention the conic form to illustrate that our restriction to conic programs is without loss of generality.

5.4 Exercises

Exercise 5.2. Prove the following:

1. The closure of any cone must contain the origin.
2. The intersection of two cones is a cone.
3. The Cartesian product of two cones is a cone.
4. If $\mathcal{K}_1, \mathcal{K}_2 \subseteq \mathbb{R}^d$ are cones, then $\mathcal{K}_1 + \mathcal{K}_2$ is a cone.
5. A cone $\mathcal{K} \subseteq \mathbb{R}^d$ is convex if and only if $\mathcal{K} + \mathcal{K} = \mathcal{K}$.

Suppose $A \in \mathbb{R}^{m \times d}$ is a matrix.

6. If $\mathcal{K} \subseteq \mathbb{R}^d$ is a cone, then $\{Ax \colon x \in \mathcal{K}\}$ is a cone in $\mathbb{R}^m$.
7. If $\mathcal{K}' \subseteq \mathbb{R}^m$ is a cone, then $\{x \colon Ax \in \mathcal{K}'\}$ is a cone in $\mathbb{R}^d$.
8. Give an example of a closed convex cone $\mathcal{K} \subseteq \mathbb{R}^d$ and a matrix $A \in \mathbb{R}^{m \times d}$ such that the set $\{Ax \colon x \in \mathcal{K}\}$ is not closed.

Exercise 5.3. Suppose $\mathcal{X}$ is a closed convex set.

1. If $\mathcal{X}$ is bounded, show that $\overline{\mathcal{K}}_{\mathcal{X}} = \mathcal{K}_{\mathcal{X}} \cup \{(0,0)\}$.
2. Give an example of a closed convex set $\mathcal{X}$ for which $\overline{\mathcal{K}}_{\mathcal{X}} \neq \mathcal{K}_{\mathcal{X}} \cup \{(0,0)\}$.

6 Farkas' Lemma

Skills. Recognize and exploit convexity and its special cases (conic programming); Learn to “take a dual;” Recognize when “strong duality” holds.

Suppose you and your friend are having a discussion about a polyhedral system:

$$\begin{aligned} Ax &= b \\ x &\geq 0 \end{aligned} \tag{6.1}$$

You claim that this system has no solution, but your friend does not believe you. Unfortunately, your friend has never taken an optimization course, and can do little math beyond arithmetic. How can you convince them of your claim?

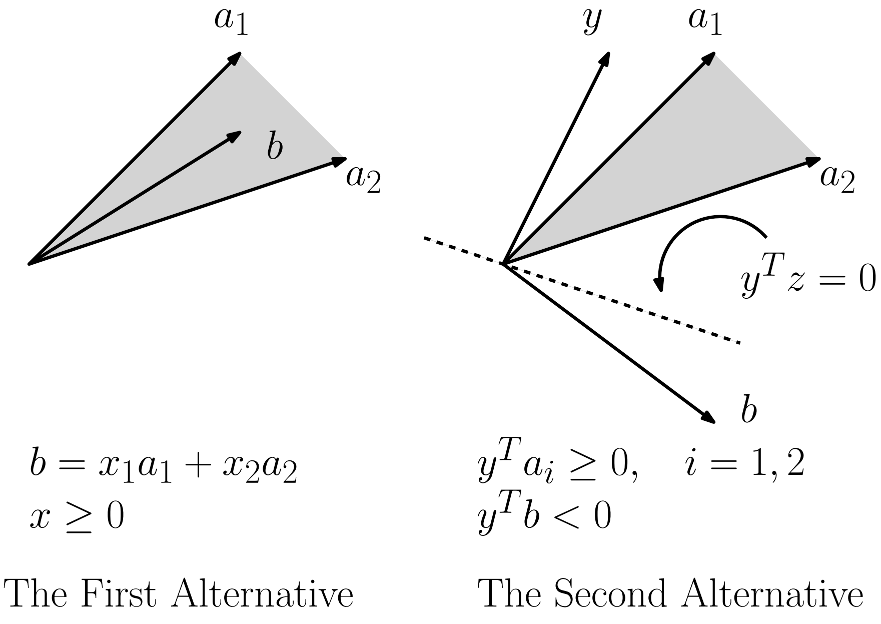
In this section we will show how you can provide your friend with a short "certificate of infeasibility"—a vector $y \in \mathbb{R}^m$—that proves the system (6.1) is infeasible.[4] This certificate is simply any solution to the alternative system

$$\begin{aligned} A^T y &\geq 0 \\ b^T y &< 0. \end{aligned} \tag{6.2}$$

Indeed, whenever x is a solution to (6.1), we have $Ax = b$ and $x \geq 0$. Thus,

$$0 > b^T y = (Ax)^T y = x^T (A^T y) \geq 0,$$

since both arguments of the final dot product are nonnegative vectors. This is clearly a contradiction, and your friend, who believes in arithmetic, is convinced.



The First Alternative

The Second Alternative

Figure 3: The two alternative systems

Figure 3 illustrates the geometry underlying this certificate, namely, y is the normal to a hyperplane that *passes through the origin* and separates the cone $A\mathbb{R}^d_+$ from the vector b.[5] The Hahn-Banach theorem (Theorem 4.2) guarantees a separating hyperplane exists[6]. This requires more argument. The goal of this lecture is to rigorously ground this argument and generalize this procedure to conic systems.

[4] Infeasible means not feasible, i.e., the system does not have a solution.

[5] Here, $A\mathbb{R}^d_+ = \{Ax \colon x \geq 0\}$.

[6] Strictly speaking, one must show that $A\mathbb{R}^d_+$ is a closed set. This was the subject of your first recitation.

6.1 Dual Cones

The system alternative to a conic system involves a related object, called the *dual cone.*

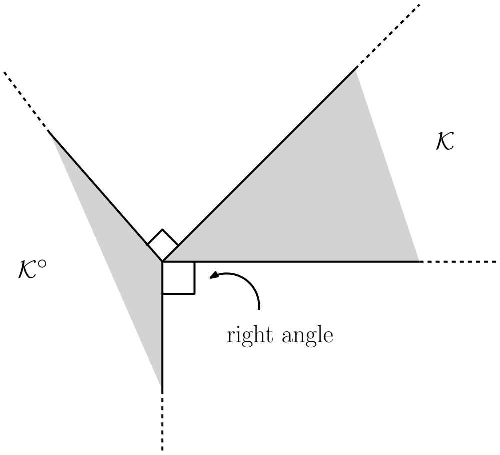
Definition 6.1 (Dual Cone)**.** *Let $\mathcal{K} \subseteq \mathbb{R}^d$ be a cone. Then the* dual cone *of $\mathcal{K}$ is the set*

$$\mathcal{K}^* := \{s \in \mathbb{R}^d \colon \langle x, s\rangle \geq 0 \quad \forall x \in \mathcal{K}\}.$$

The dual cone of any cone is always a cone, and moreover, it is always convex even if $\mathcal{K}$ is not convex (check!). Perhaps easier to visualize is the polar cone

$$\mathcal{K}^\circ := -\mathcal{K}^*.$$

If $\mathcal{K}$ is closed and convex, then the polar cone to $\mathcal{K}$ is just the normal cone: $\mathcal{K}^\circ = \mathcal{N}_{\mathcal{K}}(0)$ (check!).



The Polar Cone

Dual cones are sometimes easy to calculate as the following two examples show (check!):

- If $\mathcal{K}$ is a subspace, then $\mathcal{K}^* = \mathcal{K}^\perp$.
- If $\mathcal{K} = \mathbb{R}^d_+$ is the nonnegative orthant, then $\mathcal{K}^* = \mathbb{R}^d_+$.

The nonnegative orthant satisfies $\mathcal{K}^* = \mathcal{K}$, so it is called *self-dual.* Another self-dual cone is the second order cone $\text{SOC}(d + 1)$, a fact you will prove on your homework.

The product and closure operations can be applied before or after taking a dual. Indeed, it is easy to show that

$$(\mathcal{K}_1 \times \ldots \times \mathcal{K}_l)^* = \mathcal{K}_1^* \times \ldots \times \mathcal{K}_l^*,$$

where $\mathcal{K}_1, \ldots, \mathcal{K}_l$ are cones (check!). A simple but useful example of this fact is

$$(\mathbb{R}^d_+ \times \mathbb{R}^m)^* = \mathbb{R}^d_+ \times \{0\}.$$

Likewise, the closure satisfies

$$\left(\overline{\mathcal{K}}\right)^* = \overline{(\mathcal{K}^*)} = \mathcal{K}^*.$$

You will prove this fact on your homework. For now, we mention it since it plays a role in the following corollary of Hahn-Banach.

6.2 A Corollary to Hahn-Banach and Farkas Lemma

The connection between systems (6.1) and (6.2) is based on a simple corollary of Hahn-Banach. The corollary roughly states that the duality operation is invertible and in fact, it is its own inverse. You should view this invertibility as an instance of *strong duality*, a phenomenon we will return in the next lecture.

Corollary 6.2. *Assume $\mathcal{C}$ is a convex cone. Then $(\mathcal{C}^*)^* = \overline{\mathcal{C}}$.*

Equivalently, $b \notin \overline{\mathcal{C}}$ if and only if there exists $y \in \mathcal{C}^*$ satisfying $b^T y < 0$.

Proof. "$\subseteq$" We first show that $\overline{\mathcal{C}} \subseteq (\mathcal{C}^*)^*$. Since $(\mathcal{C}^*)^*$ is closed[7] and $\overline{\mathcal{C}}$ is the smallest closed set containing $\mathcal{C}$, we need only show that $\mathcal{C} \subseteq (\mathcal{C}^*)^*$. To that end, let $x \in \mathcal{C}$. To prove $x \in (\mathcal{C}^*)^*$, we must show that for all $y \in \mathcal{C}^*$, we have $\langle x, y\rangle \geq 0$. This is true by definition.

"$\supseteq$" We now show that $\overline{\mathcal{C}} \supseteq (\mathcal{C}^*)^*$. For the sake of contradiction, suppose that there exists $x \in (\mathcal{C}^*)^*$ such that $x \notin \overline{\mathcal{C}}$. By Hahn-Banach, there exists a vector $a \in \mathbb{R}^d$ such that

$$\max\{\langle y, a\rangle \colon y \in \overline{\mathcal{C}}\} < \langle x, a\rangle.$$

Clearly, the maximum is at least 0 since $0 \in \overline{\mathcal{C}}$. We now claim that this is indeed the maximum value, in particular, $\max\{\langle y, a\rangle \colon y \in \overline{\mathcal{C}}\} \leq 0$, implying $-a \in \overline{\mathcal{C}}^*$.

> Indeed, let $\overline{y}$ obtain the maximum and suppose that $\langle \overline{y}, a\rangle > 0$. Then $2\overline{y} \in \overline{\mathcal{C}}$, so $\langle 2\overline{y}, a\rangle > \langle \overline{y}, a\rangle$, which contradicts the definition of $\overline{y}$. Thus, for all $y \in \overline{\mathcal{C}}$, we have $\langle y, a\rangle \leq 0$, meaning $-a \in \overline{\mathcal{C}}^*$.

Now, since $-a \in \overline{\mathcal{C}}^*$, we have $\langle x, -a\rangle \geq 0$, which directly contradicts $\langle x, a\rangle > 0$. Therefore, $\overline{\mathcal{C}} \supseteq (\mathcal{C}^*)^*$, as desired. □

Before explaining how this theorem leads to certificates of infeasibility for conic systems, we must first decide upon the conic system we would like to analyze. To that end and in direct analogy with (6.1), we consider the following conic system:

$$\begin{aligned} Ax &= b \\ x &\in \mathcal{K} \end{aligned} \tag{6.3}$$

where $A \in \mathbb{R}^{m\times d}$ is a matrix, $b \in \mathbb{R}^m$ is a vector, and $\mathcal{K} \subseteq \mathbb{R}^d$ is a closed convex cone.

Let us return to Figure 3 and the proof strategy outlined in the introduction. In the event that (6.3) has no solution, we will provide a certificate—a vector $y \in \mathbb{R}^m$—which is normal to a particular hyperplane, one that passes through the origin and separates b from the *closure* of the following cone

$$\mathcal{C} := \{Ax \colon x \in \mathcal{K}\}.$$

The certificate will simply be the vector $y \in \mathcal{C}^*$ from the statement of Corollary 6.2, since $b \notin \overline{\mathcal{C}}$ if and only if $\exists y \in \mathcal{C}^*$ such that $b^T y < 0$. The attentive reader will notice a slight technical issue: Corollary (6.2) can only guarantee separation of b from $\overline{\mathcal{C}}$. Hence, we introduce a new

[7] Recall that all dual cones are closed.

concept of feasibility for the conic system, saying that it is *asymptotically feasible* if $b \in \overline{\mathcal{C}}$, and moreover, we call the vector y a certificate of asymptotic infeasibility. Intuitively, a conic system is asymptotically feasible if it becomes feasible with the help of a slight perturbation. Of course, there is no difference between feasibility and asymptotic feasibility when $\mathcal{C}$ is closed, a happy event that occurs if $\mathcal{K}$ is a polyhedral cone.

We are missing just one ingredient: a relationship between $\mathcal{C}^*$ and the alternative system (6.2). The next lemma supplies it.

Lemma 6.3. *The following relationship holds:*

$$\mathcal{C}^* = \{y\colon A^T y \in \mathcal{K}^*\}.$$

Proof. A point $y \in \mathbb{R}^d$ is contained in $\mathcal{C}^*$ if and only if $(\forall x \in \mathcal{K})\ 0 \leq \langle Ax, y\rangle = \langle x, A^T y\rangle$, which holds if and only if $A^T y \in \mathcal{K}^*$. □

With this Lemma, we arrive at Farkas' Lemma, a central statement in duality theory.[8] When $\mathcal{K}$ is polyhedral (e.g., $\mathcal{K} = \mathbb{R}^d_+$), the concepts of feasibility and asymptotic feasibility coincide. Thus, Farkas' Lemma shows that you can always provide your friend with a certificate of infeasibility for a set of linear inequalities.

Theorem 6.4 (Farkas' Lemma)**.** *Assume $\mathcal{K}$ is a convex cone and consider the following systems of constraints.*

$$\underbrace{\begin{Bmatrix} Ax = b \\ x \in \mathcal{K} \end{Bmatrix}}_{\text{(I)}} \qquad \textit{and} \qquad \underbrace{\begin{Bmatrix} A^T y \in \mathcal{K}^* \\ b^T y < 0 \end{Bmatrix}}_{\text{(II)}}$$

Then either (I) *is asymptotically feasible or* (II) *is feasible, but not both.*

In optimization, we typically avoid strict linear inequalities, such as $b^T y < 0$ from system (II). You should check that this constraint can be safely replaced by $b^T y = -1$ without changing the statement of the theorem.

6.3 Exercises

Exercise 6.1. Let $\mathcal{K}$ be a polyhedral cone.[9] Prove that $\mathcal{K}^*$ is also polyhedral.

Exercise 6.2 (Normal Cone to A Cone). Let $\mathcal{K} \subseteq \mathbb{R}^m$ be a convex cone. Prove that

$$\mathcal{N}_{\mathcal{K}}(x) = -\mathcal{K}^* \cap \{x\}^\perp \qquad \forall x \in \mathcal{K}.$$

Exercise 6.3. Prove that each of the following cones $\mathcal{K}$ are *self-dual*, meaning $\mathcal{K} = \mathcal{K}^*$.

1. $\mathbb{R}^d_+$
2. $\mathrm{SOC}(d+1)$
3. $\mathbb{S}^{d\times d}_+$

[8] Farkas' result originally appeared in a 1902 paper on polyhedral systems.

[9] The term polyhedral means the cone is defined by finitely many linear inequalities.

7 Strong Duality

Skills. Recognize and exploit convexity and its special cases (linear and conic programming); Learn to "take a dual;" Recognize when "strong duality" holds.

Farkas' Lemma (Theorem 6.4), though ostensibly about conic systems, has a related conic programming formulation (why?).

Theorem 7.1 (Farkas' Lemma, Optimization Form)**.** *Assume $\mathcal{K}$ is a convex cone and that $\{Ax\colon x \in \mathcal{K}\}$ is closed. Consider the following conic programming problems:*

$$\underbrace{\left\{\begin{array}{r} \text{minimize } 0^T x \\ \text{subject to : } Ax = b \\ x \in \mathcal{K} \end{array}\right\}}_{\text{(I)}} \qquad \textit{and} \qquad \underbrace{\left\{\begin{array}{c} \text{minimize } b^T y \\ \text{subject to : } A^T y \in \mathcal{K}^* \\ \end{array}\right\}}_{\text{(II)}}$$

Then either

- *both problems have optimal value* 0 *or*
- (I) *is infeasible and* $\inf\{b^T y\colon A^T y \in \mathcal{K}^*\} = -\infty$,

but not both.

Naming (I) the *primal problem* and (II) the *dual problem*, this pairing has all the features of the duality pairing that we described at the start of Section 4: the two problems have the same optimal value when (I) is feasible, and when (I) is infeasible, (II) takes on an infinite value. The problems are more pleasingly symmetric when we assign (I) the value $+\infty$, in case of infeasibility, and replace (II) with the equivalent maximization problem

$$\underbrace{\left\{\begin{array}{c} \text{maximize } b^T y \\ \text{subject to : } -A^T y \in \mathcal{K}^* \end{array}\right\}}_{\text{(II')}},$$

which then also takes on value $+\infty$ whenever (I) is infeasible. We moreover see that whenever x is feasible for (I), and y is feasible for (II'), we must have

$$b^T y = (Ax)^T y = x^T (A^T y) \leq 0 = 0^T x,$$

where the inequality follows since $-A^T y \in \mathcal{K}^*$. Thus the dual objective is *pushing the primal objective from below,* and when (I) is infeasible, the dual objective feels no resistance, so it grows unboundedly.

This primal problem above has cost vector $c = 0$, but the same idea—a dual objective pushing a primal objective from below—also suggests a way of thinking about the general primal problem ($\mathcal{PRIMAL}$):

$$\begin{array}{r} \text{minimize } c^T x \\ \text{subject to : } Ax = b \\ x \in \mathcal{K} \end{array} \tag{$\mathcal{P}$}$$

Let us create such a dual. In the interest of minimally adjusting (II'), we commit to the objective $b^T y$, but require that when x is feasible for $(\mathcal{P})$ and y is feasible for our yet-to-be-defined dual, the dual objective sits below the primal, namely $b^T y \leq c^T x$. Simplifying, we find

$$(Ax)^T y = x^T(A^T y) = b^T y \leq c^T x,$$

i.e., $(c - A^T y)^T x \geq 0$ for all $x \in \mathcal{K}$, meaning $c - A^T y \in \mathcal{K}^*$. Thus, we have arrived at the dual problem

$$\begin{aligned} &\text{maximize } b^T x \\ &\text{subject to} : c - A^T y \in \mathcal{K}^*. \end{aligned} \tag{$\mathcal{D}$}$$

Comparing $(\mathcal{D})$ with (II'), two crucial differences become apparent: the dual $(\mathcal{D})$ is not necessarily feasible and even if it is, it does not necessarily attain its minimal value. If it is infeasible, we give $(\mathcal{D})$ value $-\infty$. If it does not attain its minimal value, we should replace "maximize" with "sup," an issue we will return to later.

Let us summarize. The primal problem $(\mathcal{P})$ has optimal value $\texttt{val} \in [-\infty, +\infty]$. We let $\texttt{val} = +\infty$ if $(\mathcal{P})$ is infeasible. If $\texttt{val} = -\infty$, then primal problem is *unbounded*, meaning there is a sequence of feasible x_i with $c^T x_i \to -\infty$ as $i \to \infty$. If $\texttt{val} \in \mathbb{R}$, the primal problem is not unbounded, but the feasible set $\{x \colon Ax = b, x \in \mathcal{K}\}$ can still be unbounded—this is a slight peculiarity in terminology. We describe the dual problem $(\mathcal{D})$ with a similar nomenclature: The dual problem $(\mathcal{D})$ has optimal value $\texttt{val}^* \in [-\infty, +\infty]$. We let $\texttt{val}^* = -\infty$ if $(\mathcal{D})$ is infeasible. If $\texttt{val}^* = +\infty$, then dual problem is *unbounded*, meaning there is a sequence of feasible y_i with $b^T y_i \to +\infty$ as $i \to \infty$. If $\texttt{val}^* \in \mathbb{R}$, the dual problem is not unbounded, but the feasible set $\{y \colon c - A^T y \in \mathcal{K}^*\}$ can still be unbounded, again a slight peculiarity in terminology.

Based on our derivation of the dual problem, we always have $\texttt{val} \geq \texttt{val}^*$ whenever the primal and dual problems are feasible. If either problem is infeasible, this inequality still holds (trivially). Thus we have the following weak duality theorem

Theorem 7.2 (Weak Duality). *We have*

$$\texttt{val} \geq \texttt{val}^*.$$

7.1 Linear Programs

Even for linear programs, it is possible that both the primal and dual are infeasible ($\texttt{val} = +\infty, \texttt{val}^* = -\infty$) (check!). However, if either $(\mathcal{P})$ or $(\mathcal{D})$ is feasible, then $\texttt{val} = \texttt{val}^*$, a fact known as *strong duality.* Strong duality is not generally true, even when $\mathcal{K} = \mathrm{SOC}(3)$. As in Farkas' Lemma, the success or failure of strong duality hinges on whether or not the linear image of a certain cone is closed. Similar to Farkas' Lemma, we will give an asymptotic refinement of strong duality based on perturbations. Since linear images of polyhedral sets are polyhedral and closed, linear programs avoid this obstacle. Thus, we begin with linear programs, and prove the complete strong duality theorem.

Before proving the theorem, let us first resolve one mystery: whether optimal solutions to linear programs exist whenever $\texttt{val}$ or $\texttt{val}^*$ is finite.

Lemma 7.3 (Optimal Solutions of Linear Programs)**.** *Suppose $\mathcal{K}$ is polyhedral. If* $\mathtt{val}$ *is finite, then $(\mathcal{P})$ has an optimal solution. Likewise, if* $\mathtt{val}^*$ *is finite, then $(\mathcal{D})$ has an optimal solution.*

Proof. We only prove the first statement; the proof of the second is similar. Note that the set $\{c^T x \colon Ax = b, x \in \mathcal{K}\}$ is polyhedral, i.e., a closed interval with left endpoint $\mathtt{val}$. Thus, we may choose a primal feasible point $\bar{x}$ with $c^T \bar{x} = \mathtt{val}$. □

The proof of the following strong duality theorem is based on Farkas' Lemma. This is a standard approach for proving strong duality.

Theorem 7.4 (Strong Duality for Linear Programs)**.** *Suppose that $\mathcal{K}$ is polyhedral. If either $(\mathcal{P})$ or $(\mathcal{D})$ is feasible, then* $\mathtt{val} = \mathtt{val}^*$.

Proof. First, assume that $(\mathcal{P})$ is feasible. By weak duality, we know that $\mathtt{val} \geq \mathtt{val}^*$, so we can assume without loss of generality that $\mathtt{val} > -\infty$. For the sake of contradiction, we suppose that $\mathtt{val} > \mathtt{val}^*$. In particular there is a real number γ such that $\mathtt{val} > \gamma > \mathtt{val}^*$, and with such a γ, the system

$$\begin{aligned} c^T x &\leq \gamma \\ Ax &= b \\ x &\in \mathcal{K} \end{aligned}$$

is infeasible (check!). In the interest of applying Farkas' Lemma, we reformulate this system in standard equality form by adding an additional variable:

$$\begin{aligned} c^T x + s &= \gamma \\ Ax &= b \\ (x, s) &\in \mathcal{K} \times \mathbb{R}_+ \end{aligned} \tag{7.1}$$

This second system is also infeasible (check!), and it is, in fact, asymptotically infeasible since the cone $\mathcal{K} \times \mathbb{R}_+$ is polyhedral. Thus Farkas' Lemma implies there exists a pair $(\bar{y}, \bar{t}) \in \mathbb{R}^m \times \mathbb{R}_+$, that solves the following alternative system:

$$\begin{aligned} A^T y + tc &\in \mathcal{K}^* \\ 0^T y + t &\in \mathbb{R}_+ \\ b^T y + \gamma t &< 0, \end{aligned} \tag{7.2}$$

where we have used $(\mathcal{K} \times \mathbb{R}_+)^* = \mathcal{K}^* \times \mathbb{R}_+$. We claim that $\bar{t} > 0$: if not, $\bar{t} = 0$, so $A^T \bar{y} \in \mathcal{K}^*$ and $b^T \bar{y} < 0$, and so by Farkas' Lemma, there can be no x such that $Ax = b$ and $x \geq 0$—a contradiction, since $(\mathcal{P})$ is feasible. Therefore, $\bar{t} > 0$.

Now letting $\tilde{y} = \bar{y}/\bar{t}$, we get $A^T \tilde{y} + c \in \mathcal{K}^*$ and $b^T \tilde{y} < -\gamma$. Hence, the vector $\hat{y} = -\tilde{y}$ satisfies the system

$$\begin{aligned} c - A^T y &\in \mathcal{K}^* \\ b^T y &> \gamma \end{aligned} ,$$

In other words, the vector $\hat{y}$ is feasible for the dual problem, and $b^T \hat{y} > \gamma > \mathtt{val}^*$. This is a contradiction since the definition of $\mathtt{val}^*$ implies $\mathtt{val}^* \geq b^T \hat{y}$. Therefore, $\mathtt{val} = \mathtt{val}^*$.

On the other hand, suppose that $(\mathcal{D})$ is feasible. Observe that $\mathtt{val}^* = -\mathtt{val}'$, where $\mathtt{val}'$ is the optimal value to the following linear program:

$$\begin{aligned} \text{minimize } & -b^T y + 0^T s \\ \text{subject to : } & A^T y + s = c \\ & (y, s) \in \mathbb{R}^m \times \mathcal{K}^*. \end{aligned}$$

This is a standard equality form LP (since $\mathcal{K} \times \mathbb{R}^m$ is polyhedral). Therefore, the first part of the theorem applies, meaning $\mathtt{val}' = (\mathtt{val}')^*$, where $(\mathtt{val}')^*$ is the optimal value of the dual problem:

$$\begin{aligned} \text{maximize } & c^T x \\ \text{subject to : } & -b - Ax \in \{0\} \\ & 0 - I_d x \in \mathcal{K}. \end{aligned}$$

Clearly this problem is equivalent to $(\mathcal{P})$ and $\mathtt{val} = -(\mathtt{val}')^* = -\mathtt{val}' = \mathtt{val}^*$, which completes the proof. □

The following Corollary now immediately follows.

Corollary 7.5 (Strong Duality for LPs)**.** *Suppose that $\mathcal{K}$ is polyhedral. Then the following hold:*

1. *If $(\mathcal{P})$ and $(\mathcal{D})$ are feasible, then both have optimal solutions $\bar{x}$ and $\bar{y}$, respectively, and*

$$c^T \bar{x} = \mathtt{val} = \mathtt{val}^* = b^T \bar{y}.$$

2. *If $(\mathcal{P})$ is unbounded, then $(\mathcal{D})$ is infeasible.*
3. *If $(\mathcal{D})$ is unbounded, then $(\mathcal{P})$ is infeasible.*
4. *It is possible that $(\mathcal{P})$ and $(\mathcal{P})$ are both infeasible.*

7.2 Asymptotic Strong Duality

Looking back on the proof of Theorem 7.4, we used the polyhedrality of system (7.1) to ensure the notions of feasibility and asymptotic feasibility coincide. In this section, we look at asymptotic feasibility of the conic system

$$\begin{aligned} c^T x + s &= \gamma \\ Ax &= b \\ (x, s) &\in \mathcal{K} \times \mathbb{R}_+. \end{aligned} \tag{7.3}$$

for general closed convex cones $\mathcal{K}$. Recall that this system is asymptotically feasible if the closure of the cone

$$\mathcal{C} = \{(Ax, c^T x + s) \colon (x, s) \in \mathcal{K} \times \mathbb{R}_+\}$$

contains (b, γ). This set is, in turn, closely related to the value function $\mathtt{val} \colon \mathbb{R}^m \to [-\infty, \infty]$, defined by

$$\mathtt{val}(b') = \inf\{c^T x \colon Ax = b', x \in \mathcal{K}\},$$

where we let $\mathtt{val}(b') = +\infty$ if the set $\{x \colon Ax = b', x \in \mathcal{K}\}$ is empty. The following Lemma relates $\mathcal{C}$ to the epigraph of $\mathtt{val}$.

Lemma 7.6. *We have*

$$\mathcal{C} \subseteq \operatorname{epi}(\texttt{val}) \subseteq \overline{\mathcal{C}}.$$

Consequently, $\overline{\mathcal{C}} = \overline{\operatorname{epi}(\texttt{val})}$.

Proof. "$\mathcal{C} \subseteq \operatorname{epi}(\texttt{val})$" Suppose $(Ax, c^T x + s) \in \mathcal{C}$. Then clearly $\texttt{val}(Ax) = \inf\{c^T x' \colon Ax' = Ax, x \in \mathcal{K}\} \leq c^T x \leq c^T x + s$, meaning $(Ax, c^T x + s) \in \operatorname{epi}(\texttt{val})$. Therefore, $\mathcal{C} \subseteq \operatorname{epi}(\texttt{val})$.

"$\operatorname{epi}(\texttt{val}) \subseteq \overline{\mathcal{C}}$" Suppose that $(b', \gamma') \in \operatorname{epi}(\texttt{val})$, i.e., $\texttt{val}(b') \leq \gamma$. Suppose that $\gamma' > \texttt{val}(b')$, then by definition, there exists x such that $Ax = b'$ and $\texttt{val}(b') \leq c^T x \leq \gamma$. Consequently, $(b', \gamma') = (Ax, c^T x + (\gamma' - c^T x)) \in \mathcal{C} \subseteq \overline{\mathcal{C}}$. On the other had, if $\texttt{val}(b') = \gamma'$, then there exists a sequence of points $x_i \in \mathcal{K}$ so that $Ax_i = b'$ and $c^T x_i \to \texttt{val}(b')$ as $i \to \infty$. Then clearly $(Ax_i, c^T x_i) \in \mathcal{C}$ and $(Ax_i, c^T x_i) \to (b', \gamma')$, meaning $(b', \gamma') \in \overline{\mathcal{C}}$. Thus, $\operatorname{epi}(\texttt{val}) \subseteq \overline{\mathcal{C}}$. □

We can think of the set $\overline{\mathcal{C}}$ as a more "robust" version of $\mathcal{C}$ since we cannot escape it simply by walking towards its boundary. The value function also has a more "robust" and closely related function, called the *closure* of `val`, a concept we define in the following exercise.

Exercise 7.1. *Let* $f : \mathbb{R}^d \to [-\infty, +\infty]$ *be an extended valued function. Then there exists a unique function* $\operatorname{cl} f : \mathbb{R}^d \to [-\infty, \infty]$ *satisfying* $\operatorname{epi}(\operatorname{cl} f) = \overline{\operatorname{epi}(f)}$*. Moreover, it satisfies the following limiting formula:*

$$\operatorname{cl} f(x) = \lim_{\varepsilon \to 0} \inf_{y \in B_\varepsilon(x)} f(y) \tag{7.4}$$

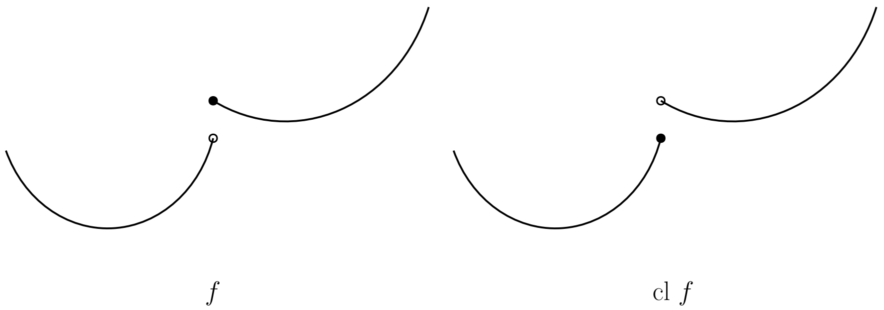


Figure 4: The Closure of a Function

Let us illustrate some basic properties of the closure. For example, any function dominates its closure: $\operatorname{cl} f(x) \leq f(x)$ for all $x \in \mathbb{R}^d$. Since any function is uniquely determined by its epigraph (check!), the double closure of a function is simply the closure: $\operatorname{cl}\operatorname{cl} f = \operatorname{cl} f$. Applying the exercise, we find that

$$\operatorname{cl} f(x) = \lim_{\varepsilon \to 0} \inf_{y \in B_\varepsilon(x)} \operatorname{cl} f(y).$$

This property implies that the value $\operatorname{cl} f(y)$ cannot suddenly shoot up as y approaches x—yet another kind of robustness.

If f satisfies $\operatorname{cl} f = f$, then we say f is *closed*. Any continuous function is closed (check!). A consequence of the exercise is that any closed function is *lower-semicontinuous*, meaning

$$f(x) = \lim_{\varepsilon \to 0} \inf_{y \in B_\varepsilon(x)} f(y).$$

Likewise, any lower-semicontinuous function is also closed (check!). A strange but important closed function is the indicator function $\delta_{\mathcal{X}}\colon \mathbb{R}^d \to \mathbb{R}$ associated to a set $\mathcal{X} \subseteq \mathbb{R}^d$. This function takes value 0 on $\mathcal{X}$ and $+\infty$ off of it. You can now easily show that $\mathcal{X}$ is closed if and only if $\delta_{\mathcal{X}}$ is closed.

Returning to our study of duality, we introduce the *asymptotic value function*: $\texttt{a-val} := \operatorname{cl}\texttt{val}$. From Lemma (7.6), we see that (b,γ) is asymptotically feasible for system (7.3) if and only if $(b,\gamma) \in \operatorname{epi}(\texttt{a-val})$, i.e., $\texttt{a-val}(b) \le \gamma$. From the exercise, we also have the expression

$$\texttt{a-val}(b) = \lim_{\varepsilon\to 0} \inf_{b' \in B_\varepsilon(b)} \texttt{val}(b'),$$

which we will return to at the end of the lecture. In the following proposition, we will see that it is not the value $\texttt{val}(b)$, but the asymptotic value $\texttt{a-val}(b)$ that is equal to $\texttt{val}^*$.

Theorem 7.7 (Asymptotic Strong Duality)**.** *If $(\mathcal{P})$ is asymptotically feasible (meaning the primal constraints are asymptotically feasible), then*

$$\texttt{a-val}(b) = \texttt{val}^*.$$

Proof. We do not know, whether $\texttt{a-val}(b) \ge \texttt{val}^*$. Thus, we first assume that there is a γ such that $\texttt{a-val}(b) < \gamma < \texttt{val}^*$ and derive a contradiction:

> Since $\texttt{a-val}(b) \le \gamma$, the pair (b,γ) is asymptotically feasible for (7.3). Hence, the alternative system
>
> $$\begin{aligned} A^T y + tc &\in \mathcal{K}^* \\ 0^T y + t &\in \mathbb{R}_+ \\ b^T y + \gamma t &< 0, \end{aligned} \tag{7.5}$$
>
> is infeasible. Since $\texttt{val}^* > \gamma > -\infty$, the dual $(\mathcal{D})$ is feasible. Thus, there exists a vector $\hat{y}$ satisfying $c - A^T\hat{y} \in \mathcal{K}^*$ and $\texttt{val}^* \ge b^T\hat{y} > \gamma$. Clearly $(-\hat{y}, 1)$ is feasible for (7.5), which is a contradiction.

Thus, $\texttt{a-val}(b) \ge \texttt{val}^*$. For the sake of contradiction, let us suppose a real γ with $\texttt{val}^* < \gamma < \texttt{a-val}(b)$ exists. Since $\texttt{a-val}(b) > \gamma$, we have $(b,\gamma) \notin \operatorname{epi}(\texttt{a-val})$, so the system (7.3) is asymptotically infeasible. Hence (7.5) is feasible by Farkas' Lemma. Let $(\overline{y}, \overline{t})$ satisfy (7.5). We claim that $\overline{t} > 0$:

> If not then, then $\overline{t} = 0$ and the system $\{y\colon A^T y \in \mathcal{K}^*, b^T y < 0\}$ is feasible. Thus, by Farkas' Lemma, the primal problem $(\mathcal{P})$ is not asymptotically feasible, contradicting the proposition's assumptions.

Therefore, $\overline{t} > 0$. Now letting $\tilde{y} = \overline{y}/\overline{t}$, we get $A^T\tilde{y} + c \in \mathcal{K}^*$ and $b^T\tilde{y} < -\gamma$. Hence, the vector $\hat{y} = -\tilde{y}$ satisfies the system

$$\begin{aligned} c - A^T y &\in \mathcal{K}^* \\ b^T y &> \gamma \end{aligned},$$

In other words, the vector $\tilde{y}$ is feasible for the dual problem, and $b^T\tilde{y} > \gamma > \mathtt{val}^*$. This is a contradiction since the definition of $\mathtt{val}^*$ implies $\mathtt{val}^* \geq b^T\tilde{y}$. Therefore, $\mathtt{a\text{-}val} = \mathtt{val}^*$. $\square$

The dual problem also has an asymptotic strong duality theory based on the dual value function $\mathtt{val}^*\colon \mathbb{R}^d \to [-\infty, \infty]$:

$$\mathtt{val}^*(c') = \sup\{b^Ty \colon c' - A^Ty \in \mathcal{K}^*\}.$$

Just as in the second half of Theorem 7.4's proof, we must look at the value of a related optimization problem in standard equation form:

$$\mathtt{val}'(c') = \inf\{-b^Ty + 0^Ts \colon A^Ty + s = c, (y, s) \in \mathbb{R}^m \times \mathcal{K}^*\}.$$

For this function, we likewise form an asymptotic value function: $\mathtt{a\text{-}val}' := \operatorname{cl}\mathtt{val}'$. Again, from the exercise, we have the expression:

$$\mathtt{a\text{-}val}'(c) = \lim_{\varepsilon\to 0} \inf_{c'\in B_\varepsilon(c)} \mathtt{val}'(c').$$

Applying Proposition 7.7, we thus find that $\mathtt{a\text{-}val}'(c) = (\mathtt{val}')^* = -\mathtt{val}(c)$ (check!).

By letting $\mathtt{a\text{-}val}^* = -\mathtt{a\text{-}val}'$ and noting that

$$\mathtt{a\text{-}val}^*(c) = \lim_{\varepsilon\to 0} \sup_{c'\in B_\varepsilon(c)} \mathtt{val}^*(c'),$$

we obtain the more meaningful result:

Corollary 7.8 (Dual Asymptotic Strong Duality). *If* $(\mathcal{D})$ *is asymptotically feasible, then*

$$\mathtt{a\text{-}val}^*(c) = \mathtt{val}.$$

By Proposition 7.7, strong duality ($\mathtt{val}(b) = \mathtt{val}^*$) holds whenever $\mathtt{a\text{-}val}(b) = \mathtt{val}(b) < \infty$. Thus, for feasible problems, the only obstacle to duality is the discrepancy $\mathtt{a\text{-}val}(b) \neq \mathtt{val}(b)$. In the literature, you will find a variety of conditions that ensure strong duality holds, but ultimately they exist only to ensure $\mathtt{a\text{-}val}(b) = \mathtt{val}(b)$. For example, in polyhedral problems the equality $\mathtt{a\text{-}val} \equiv \mathtt{val}$ always holds since

$$\operatorname{epi}(\mathtt{val}) \subseteq \operatorname{epi}(\mathtt{a\text{-}val}) = \overline{\operatorname{epi}(\mathtt{val})} = \overline{\mathcal{C}} = \mathcal{C} \subseteq \operatorname{epi}(\mathtt{val}) \implies \operatorname{epi}(\mathtt{a\text{-}val}) = \operatorname{epi}(\mathtt{val})$$

and functions are determined by their epigraphs. Another sufficient condition is that $\mathtt{val}$ is continuous at b. This a simple consequence of formula (7.6), and you will explore it in your homework. More stringent conditions also imply primal and dual optimal solutions exist, but to show this, we need the tools in the following section.

7.3 Exercises

Exercise 7.2 (A Compressive Sensing Problem). Consider the following optimization problem

$$\begin{gathered}\text{minimize } \|x\|_1 \\ \text{subject to : } Ax = b.\end{gathered}$$

(The symbol $\|x\|_1$ denotes the ℓ_1 *norm* on $\mathbb{R}^d$, a particular member of the the family of ℓ_p norms defined as follows: for any $p \in [1, \infty)$, we define

$$\|x\|_p^p := \sum_{i=1}^{d} |x_i|^p \qquad \forall x \in \mathbb{R}^d.$$

If $p = \infty$, we define $\|x\|_\infty := \max_{i=1,\ldots,d} |x_i|$ for all $x \in \mathbb{R}^d$.)

1. Write an equivalent linear programming formulation of this problem.
2. Take the dual of the linear program from part 1.
3. Prove that the linear program from part 2 is equivalent to the following problem

$$\begin{gathered}\text{maximize } \langle y, b\rangle \\ \text{subject to : } \|A^T y\|_\infty \leq 1.\end{gathered}$$

Exercise 7.3 (Failure Cases.)**.**

1. Give an example of a linear program where $\mathtt{val} = +\infty$ and $\mathtt{val}^* = -\infty$.
2. Give an example of a conic program where $\mathtt{val}$ is finite but not attained.
3. Give an example of a conic program where $\mathtt{val} = +\infty$, but $\mathtt{val}^*$ is finite.
4. Give an example of a conic program where $\mathtt{val}, \mathtt{val}^* \in \mathbb{R}$ and $\mathtt{val} \neq \mathtt{val}^*$.

Exercise 7.4 (Closed Functions). Let $f : \mathbb{R}^d \to [-\infty, +\infty]$ be an extended valued function.

1. Prove there exists a unique function $\operatorname{cl} f : \mathbb{R}^d \to [-\infty, +\infty]$, called the *closure* of f, satisfying

$$\operatorname{epi}(\operatorname{cl} f) = \overline{\operatorname{epi}(f)}.$$

   Moreover, prove the closure satisfies the following limiting formula:

$$\operatorname{cl} f(x) = \lim_{\varepsilon \to 0} \inf_{y \in B_\varepsilon(x)} f(y). \tag{7.6}$$

2. Suppose f is convex. Prove that $\operatorname{cl} f$ is convex.

Def. An extended-valued function is *closed* if $\operatorname{epi}(f)$ is closed.

3. Prove that $\operatorname{cl} f$ is closed.
4. Prove that $\operatorname{cl} f(x) \leq f(x)$ for all $x \in \mathbb{R}^d$.
5. Suppose f is continuous. Prove that f closed.

6. Suppose that f is continuous at a point $x \in \mathbb{R}^d$. Prove that $f(x) = \operatorname{cl} f(x)$. (In other words,
$$f(x) = \lim_{\varepsilon \to 0} \inf_{y \in B_\varepsilon(x)} f(y).)$$
7. Suppose that for all $x \in \mathbb{R}^d$, we have
$$f(x) = \lim_{\varepsilon \to 0} \inf_{y \in B_\varepsilon(x)} f(y).$$
Prove that f is closed. (Such functions are called lower semicontinuous.)
8. Prove that the sum of closed functions is closed.
9. Give an example of a closed extended valued function such that $\operatorname{dom}(f) = \{x \colon f(x) < +\infty\}$ is open.

Exercise 7.5. Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a set. Define the *indicator function* of $\mathcal{X}$ as follows:

$$\delta_{\mathcal{X}}(x) := \begin{cases} 0 & \text{if } x \in \mathcal{X} \\ +\infty & \text{otherwise.} \end{cases} \tag{7.7}$$

Prove that $\delta_{\mathcal{X}}$ is closed if and only if $\mathcal{X}$ is closed.

Exercise 7.6 (Asymptotic Feasibility)**.**

1. Suppose $\texttt{a-val}(b) < +\infty$. Show that $(\mathcal{P})$ is asymptotically feasible.
2. Give an asymptotically feasible conic program $(\mathcal{P})$ with $\texttt{a-val}(b) = +\infty$.

Exercise 7.7 (Strong Duality.). Let $A \in \mathbb{R}^{m \times d}$, let $c \in \mathbb{R}^d$, and let $\mathcal{K} \subseteq \mathbb{R}^d$ be a closed convex cone. Consider the family of primal and dual conic problems, which both depend on a parameter $b \in \mathbb{R}^m$:

$$\underbrace{\left\{\begin{array}{r} \text{minimize } c^T x \\ \text{subject to : } Ax = b \\ x \in \mathcal{K} \end{array}\right\}}_{\mathcal{P}(b)} \qquad \underbrace{\left\{\begin{array}{c} \text{maximize } b^T y \\ \text{subject to : } c - A^T y \in \mathcal{K}^* \\ \end{array}\right\}}_{\mathcal{D}(b)} \tag{7.8}$$

Recall the value function $\texttt{val} \colon \mathbb{R}^m \to [-\infty, \infty]$

$$\texttt{val}(b) = \inf\{c^T x \colon Ax = b, x \in \mathcal{K}\} \qquad \forall b \in \mathbb{R}^m,$$

and the asymptotic value function $\texttt{a-val} \colon \mathbb{R}^m \to [-\infty, \infty]$

$$\texttt{a-val} = \operatorname{cl} \texttt{val}.$$

1. Suppose there is a point $b \in \mathbb{R}^m$ such that $\texttt{val}(b) = \texttt{a-val}(b) \in \mathbb{R}$. Prove that $\texttt{val}(b') > -\infty$ for all $b' \in \mathbb{R}^m$.
2. Give an example of a conic program and a vector b such that the $\texttt{val}(b) = +\infty$ and $\texttt{a-val}(b) < +\infty$.

3. Suppose that `val` is continuous at a point $b \in \mathbb{R}^m$. Prove that strong duality holds:
$$\texttt{val}(b) = \sup\{b^T y \colon c - A^T y \in \mathcal{K}^*\}.$$

4. Prove that `val` and `a-val` are convex.

Consider the following basic property of convex functions:

Theorem 7.9 (Borwein and Lewis Theorem 4.1.3)**.** *Let $f \colon \mathbb{R}^d \to (-\infty, +\infty]$ be a convex function. Then f is continuous on the interior of its domain.*[10]

Notice that the function f in the above theorem never takes value $-\infty$. In the following two exercises, we use the concept of *strong feasibility.*

Definition 7.10 (Strong Feasibility)**.** *We say that $\mathcal{P}(b)$ is* strongly feasible *if there exists an $\varepsilon > 0$ such that for all $b' \in B_\varepsilon(b)$ the perturbed problem $\mathcal{P}(b')$ is feasible.*

5. Suppose that $\mathcal{P}(b)$ is strongly feasible. Then, show that strong duality holds:
$$\texttt{val}(b) = \sup\{b^T y \colon c - A^T y \in \mathcal{K}^*\}.$$

6. **(Slater's Condition.)** If $\mathcal{P}(b)$ has a feasible point x lying in the interior of $\mathcal{K}$ and if $\operatorname{rank}(A) = m$, prove that strong duality holds:
$$\texttt{val}(b) = \sup\{b^T y \colon c - A^T y \in \mathcal{K}^*\}.$$

   (Note that it is common to assume $\operatorname{rank}(A) = m$ in the optimization literature, and we can assume this without loss of generality. Indeed, if A is not full rank, we can row-reduce the system and form a new problem with the row-reduced matrix. Any solution to the new problem still solves the original one.)

8 Sensitivity: The Basics

Skills. Recognize and exploit convexity and its special cases (linear and conic programming); Learn to "take a dual;" Recognize when "strong duality" holds. Compute first-order necessary optimality conditions with nonsmooth calculus (subdifferentials, normal cones); Compute sensitivity of optimization problems with respect to perturbations of input data (value functions).

Consider the following story (it is less contrived than you may think):

> Although measles was thought to be mostly eradicated, cases have recently spiked in several areas of the world. In New York State, cases have been reported in Kings County (home to New York City), among other areas. In order to prevent a larger outbreak, pharmaceutical companies have ramped up production of the MMR vaccine throughout the state. These companies produce the vaccine at three facilities (A, B, and C) and then ship them to both Kings County and Tompkins County (home to

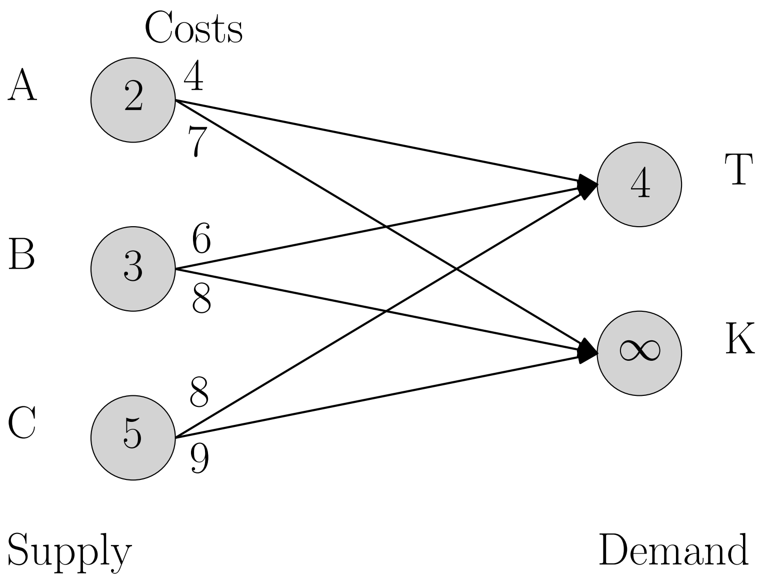


Figure 5: A Transportation Problem (cost and supply measured in 1000s))

Cornell). They have thus far manufactured 2000, 3000, and 5000 vaccines at facilities A, B, and C, respectively. Kings County will take as much of the vaccine as the companies can provide, while Tompkins County is only in need of 4000 vaccines. Figure 5 depicts the supply channels and the cost of transporting the vaccine from a given facility to either of the counties. As is standard procedure in any crisis, the governor has asked the students of ORIE 6300 to determine how much vaccine Tompkins County should purchase from each of the facilities in order to minimize the total cost of transportation.

A quick thought reveals that one may formulate this task as a linear program in the six variables: $x_{\mathrm{AT}}, x_{\mathrm{AK}}, x_{\mathrm{BT}}, x_{\mathrm{BK}}, x_{\mathrm{CT}}, x_{\mathrm{CK}}$, where for example x_{AT} represents how much vaccine is transported from facility A to Tompkins County,

$$\begin{array}{rccccccccccccl} \text{minimize} & 4x_{\mathrm{AT}} & + & 7x_{\mathrm{AK}} & + & 6x_{\mathrm{BT}} & + & 8x_{\mathrm{BK}} & + & 8x_{\mathrm{CT}} & + & 9x_{\mathrm{CK}} & & \\ \text{subject to :} & x_{\mathrm{AT}} & + & x_{\mathrm{AK}} & & & & & & & & & = & 2 \\ & & & & & x_{\mathrm{BT}} & + & x_{\mathrm{BK}} & & & & & = & 3 \\ & & & & & & & & & x_{\mathrm{CT}} & + & x_{\mathrm{CK}} & = & 5 \\ & x_{\mathrm{AT}} & & & + & x_{\mathrm{BT}} & & & + & x_{\mathrm{CT}} & & & = & 4 \\ & x_{\mathrm{AT}} & , & x_{\mathrm{AK}} & , & x_{\mathrm{BT}} & , & x_{\mathrm{BK}} & , & x_{\mathrm{CT}} & , & x_{\mathrm{CK}} & \geq & 0 \end{array} \tag{8.1}$$

One solution of this linear program has total cost \$73000: $x^*_{\mathrm{AT}} = 2, x^*_{\mathrm{AK}} = 0, x^*_{\mathrm{BT}} = 2, x^*_{\mathrm{BK}} = 1, x^*_{\mathrm{CT}} = 0, x^*_{\mathrm{CK}} = 5$.

At your next meeting, you present your plan to the governor, who looks at the \$73000 cost and notices that this is \$2000 less than the total budget. Afraid of the political consequences of coming in under budget, the governor refuses to spend less than \$75000 and requests facility A to increase its supply until an *optimal* budget of \$75000 is

[10] Recall that $\mathrm{dom}(f) = \{x \colon f(x) < +\infty\}$.

reached. In terms of Problem (8.1), this amounts to finding an appropriate $\varepsilon > 0$ so that the adjusted problem

$$\begin{array}{rcccccccccccccc}
\text{minimize} & 4x_{\text{AT}} & + & 7x_{\text{AK}} & + & 6x_{\text{BT}} & + & 8x_{\text{BK}} & + & 8x_{\text{CT}} & + & 9x_{\text{CK}} & & \\
\text{subject to :} & x_{\text{AT}} & + & x_{\text{AK}} & & & & & & & & & = & 2+\varepsilon \\
 & & & & & x_{\text{BT}} & + & x_{\text{BK}} & & & & & = & 3 \\
 & & & & & & & & & x_{\text{CT}} & + & x_{\text{CK}} & = & 5 \\
 & x_{\text{AT}} & & & + & x_{\text{BT}} & & & + & x_{\text{CT}} & & & = & 4 \\
 & x_{\text{AT}} & , & x_{\text{AK}} & , & x_{\text{BT}} & , & x_{\text{BK}} & , & x_{\text{CT}} & , & x_{\text{CK}} & \geq & 0
\end{array}$$

has optimal cost \$75000. Apart from brute force search, how might we find $\varepsilon > 0$?

Using the language of the previous section, the governor's demand is clear: find the value function. While there is in general no closed form expression for the value function, what is available is an understanding of how it changes under infinitesimal perturbations. To illustrate, consider the primal conic program $(\mathcal{P})$ for a fixed $b \in \mathbb{R}^m$, and as before let

$$\texttt{val}(b') = \inf\{c^T x \colon Ax = b', x \in \mathcal{K}\}, \qquad \forall b' \in \mathbb{R}^m.$$

Suppose the optimal value of $(\mathcal{P})$ is attained at $x_0 \in \mathcal{K}$, meaning $Ax_0 = b$ and

$$c^T x_0 - \texttt{val}(Ax_0) = 0.$$

Taking into account the definition of $\texttt{val}$, a more general inequality also holds:

$$c^T x - \texttt{val}(Ax) \geq 0, \qquad \forall x \in \mathcal{K}.$$

Taken together, these imply the point x_0 is optimal for the following program:

$$\begin{array}{c} \text{minimize } c^T x - \texttt{val}(Ax) \\ \text{subject to : } x \in \mathcal{K}. \end{array}$$

First order optimality conditions (Theorem 3.3) and the chain rule[11] then imply that

$$-\left(c - A^T \nabla \texttt{val}(Ax_0)\right) \in \mathcal{N}_{\mathcal{K}}(x_0). \tag{8.2}$$

Based on the elementary identity $\mathcal{N}_{\mathcal{K}}(x_0) = -\mathcal{K}^* \cap \{x_0\}^\perp$ (Exercise 6.2), we gain two insights:

(Dual Feasibility) The gradient $y_0 = \nabla \texttt{val}(Ax_0)$ is dual feasible: $c - A^T y_0 \in \mathcal{K}^*$.

(Dual Optimality) Multiplying both sides of (8.2) by x_0^T and noting $x_0^T \mathcal{N}_{\mathcal{K}}(x_0) = \{0\}$

$$0 = x_0^T(c - A^T \nabla \texttt{val}(Ax_0)) = c^T x_0 - b^T y_0 \implies \texttt{val}(b) = b^T y_0.$$

Thus by weak duality (Theorem 7.2), y_0 is dual optimal.

This argument is elegant and illuminating, but ultimately wrong since $\texttt{val}$ is not differentiable. It does, however, hint at the truth, namely that "differential" properties of $\texttt{val}$ are inherited from dual solutions. We will salvage this argument by relaxing the notion of Fréchet differentiability to (possibly) infinite-valued functions.

[11] It is an easy exercise to check that if a function f is differentiable at x, then $g(x) = f(Ax)$ is differentiable at x and $\nabla g(Ax) = A^T \nabla f(Ax)$.

8.1 The Fréchet Subdifferential

Looking back at the definition of the Fréchet gradient (Definition 3.1), we see that any such gradient provides a linear approximation of f up to first order. The Fréchet subgradient provides instead a linear *under approximation* up to first order.

Definition 8.1. *Let $f\colon \mathbb{R}^d \to (-\infty, \infty]$ and let $x \in \operatorname{dom}(f)$. Then v is a Fréchet subgradient of f at $x \in \mathbb{R}^d$ if there exists $o_x\colon \mathbb{R}^d \to \mathbb{R}$ with*

$$f(y) \geq f(x) + \langle v, y - x\rangle + o_x(y) \qquad \textit{where} \qquad \lim_{y \to x} \frac{o_x(y)}{\|y - x\|} = 0.$$

We let $\partial_F f(x)$ denote the set of Fréchet subgradients of f.

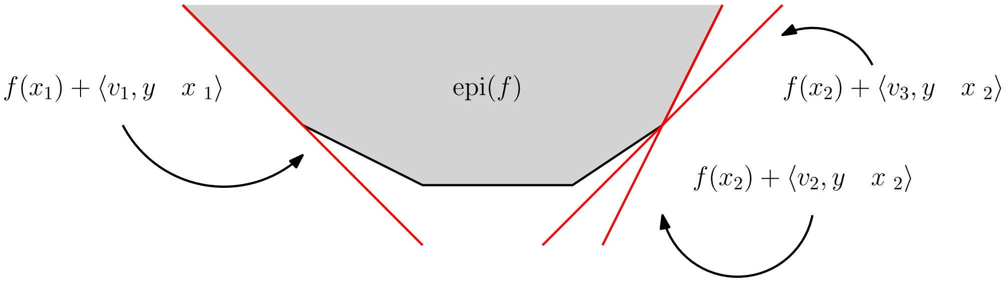

$f(x_1) + \langle v_1, y \quad x_1\rangle$
epi(f)
$f(x_2) + \langle v_3, y \quad x_2\rangle$
$f(x_2) + \langle v_2, y \quad x_2\rangle$

Figure 6: Linear under approximations of f. ($v_1 \in \partial f(x_1)$, $v_2, v_3 \in \partial f(x_2)$.)

Although we may compute the Fréchet subdifferential without convexity, with it the definition simplifies. Both Figure 7 and the following Lemma illustrate this fact.

Lemma 8.2 (Fréchet Subgradients of Convex Functions)**.** *Let $f\colon \mathbb{R}^d \to (-\infty, \infty]$ be a convex function. Then*

$$\partial_F f(x) = \left\{v\colon f(y) \geq f(x) + \langle v, y - x\rangle, \quad \forall y \in \mathbb{R}^d\right\}, \qquad \forall x \in \operatorname{dom}(f).$$

Equivalently, $v \in \partial_F f(x)$ if and only if $f(y) - \langle v, y\rangle$ is minimized at x.

A few facts and some notation follow. One need only check the under approximation property at $y \in \operatorname{dom}(f)$. If f takes the value $-\infty$, the subdifferential is an uninteresting object, so we simply define $\partial_F f(x) \equiv \emptyset$. The subdifferential can be empty in other cases; in the next section we will explain when it is not. For convex functions, it is common to simply define

$$\partial f(x) := \left\{v\colon f(y) \geq f(x) + \langle v, y - x\rangle, \quad \forall y \in \mathbb{R}^d\right\}$$

to be the *convex subdifferential* of f at x. We began with the Fréchet subdifferential since we will later consider nonconvex functions. We do, however, use the notation

$$\partial = \partial_F$$

whenever we deal with convex functions.

8.2 Subgradients and Dual Solutions

Returning to our study of the value function, we see that subgradients give us *global* lower approximations of `val`. In classical terminology, they help us measure the sensitivity of $(\mathcal{P})$. The following theorem relates dual solutions to subgradients of `val`, salvaging the argument of the introduction. A similar dual formulation of the theorem is also available by following the argument of Corollary 7.8. Both results relies both on Lemma 8.2 and on Part 4 of Exercise 7.7, where `val` and `a-val` were shown to be convex.

Theorem 8.3 (Sensitivity Analysis)**.** *Consider the primal problem $(\mathcal{P})$ with fixed vector $b \in \mathbb{R}^m$ and define $D := \{y \in \mathbb{R}^m : y$ optimal for $(\mathcal{D})\}$. Then the following hold:*

1. *If $(\mathcal{D})$ is feasible, then $\partial\texttt{val}(b) \subseteq D$.*
2. *If $(\mathcal{P})$ is asymptotically feasible, then $D \subseteq \partial\texttt{a-val}(b)$.*

Moreover, if $(\mathcal{D})$ is feasible and either (a) $\partial\texttt{val}(b) \neq \emptyset$ or (b) $(\mathcal{P})$ is feasible and $\texttt{val}(b) = \texttt{a-val}(b)$, then

$$\partial\texttt{val}(b) = D = \partial\texttt{a-val}(b).$$

Proof. Throughout the proof, we use dual feasibility to ensure $\texttt{val}(b') > -\infty$ for all b'.[12]

Beginning with Part 1, we may assume that $(\mathcal{P})$ is feasible and $\partial\texttt{val}(b) \neq \emptyset$, since in either case $\partial\texttt{val}(b) = \emptyset$ and there is nothing to prove. Thus, by definition there exists $y \in \partial\texttt{val}(b)$ satisfying:

$$\langle y, b' - b\rangle + \texttt{val}(b) \leq \texttt{val}(b') \qquad \forall b' \in \mathbb{R}^m.$$

Since $(\mathcal{P})$ is feasible, there is thus a feasible sequence $x_1, x_2, \ldots$ that has cost decreasing to $\texttt{val}(b)$: $c^T x_i \searrow \texttt{val}(b)$ as $i \to +\infty$. By definition, this sequence satisfies both $Ax_i = b$ and $x_i \in \mathcal{K}$, and the former yields $\texttt{val}(Ax_i) = \texttt{val}(b)$ for all i. We may learn about y by testing various b' in the inequality, and a wealth of candidates arise from setting $b' = Ax$ for $x \in \mathcal{K}$. Choosing any such candidate and substituting in $b = Ax_i$, we find that

$$\begin{aligned} 0 &\leq \texttt{val}(Ax) - \texttt{val}(Ax_i) - \langle y, A(x - x_i)\rangle \\ &\leq c^T x - c^T x_i - \langle A^T y, x - x_i\rangle + (c^T x_i - \texttt{val}(Ax_i)) \\ &= \langle c - A^T y, x - x_i\rangle + (c^T x_i - \texttt{val}(Ax_i)), \end{aligned}$$

where the first inequality comes from the trivial bound $\texttt{val}(Ax) \leq c^T x$. We test against two types of $x \in \mathcal{K}$, finding first that y is dual feasible and second that $b^T y = \texttt{val}(b)$:

(Dual Feasibility) For any $x' \in \mathcal{K}$, we set $x = x' + x_i$ and find that

$$\langle c - A^T y, x'\rangle \geq (\texttt{val}(Ax_i) - c^T x_i), \qquad \forall i \in \mathbb{N}.$$

The left hand side is constant, while the right hand side tends to zero, and thus $\langle c - A^T y, x'\rangle \geq 0$. Since $x' \in \mathcal{K}$ is arbitrary, it holds $c - A^T y \in \mathcal{K}^*$.

[12] This is a simple consequence of Weak Duality.

(Dual Optimality) Setting $x = 0$,

$$(c^T x_i - \texttt{val}(Ax_i)) \geq \langle c - A^T y, x_i\rangle = (c^T x_i - b^T y).$$

Both sides are nonnegative, the left hand side tends to zero, and the right hand side tends to $\texttt{val}(b) - b^T y$. Thus, $b^T y = \texttt{val}(b)$.

Since y is dual feasible and $b^T y = \texttt{val}(b)$, weak duality yields $y \in D$.

Turning to Part 2, we assume D is nonempty since otherwise there is nothing to prove. We will show any dual solution $\overline{y} \in D$ is a subgradient of $\texttt{a-val}$:

$$\langle \overline{y}, b' - b\rangle + \texttt{a-val}(b) \leq \texttt{a-val}(b') \qquad \forall b' \in \mathbb{R}^m.$$

Of course, we must show the inequality only when $\texttt{a-val}(b') < +\infty$. In turn, Exercise 7.6 shows that for such points, the perturbed system $\{x \colon Ax = b', x \in \mathcal{K}\}$ is asymptotically feasible. Thus, asymptotic strong duality yields $\texttt{a-val}(b') = \sup\{(b')^T y \colon c - A^T y \in \mathcal{K}^*\}$ (Theorem (7.7)). Similarly, asymptotic strong duality also yields $\texttt{a-val}(b) = b^T \overline{y}$. Putting these facts together, we see

$$\begin{aligned}\texttt{a-val}(b) = b^T\overline{y} = (b')^T\overline{y} + (b - b')^T\overline{y} &\leq \sup\{(b')^T y \colon c - A^T y \in \mathcal{K}^*\} + (b - b')^T\overline{y} \\ &= \texttt{a-val}(b') + (b - b')^T\overline{y},\end{aligned}$$

as desired.

The final statement follows from Exercise 8.3. Thus, the proof is complete. □

When $\mathcal{K}$ is polyhedral, equality $\texttt{val} \equiv \texttt{a-val}$ holds, so we obtain the following corollary:

Corollary 8.4 (Sensitivity Analysis of Linear Programs)**.** *Consider the primal problem* $(\mathcal{P})$ *with polyhedral cone* $\mathcal{K}$ *and fixed vector* $b \in \mathbb{R}^m$*. If* $(\mathcal{D})$ *is feasible, then*

$$\partial\texttt{val}(b) = \{y \in \mathbb{R}^m \colon y \text{ optimal for } (\mathcal{D})\}.$$

Proof. If $(\mathcal{P})$ is feasible, the result follows by the theorem. If $(\mathcal{P})$ is infeasible, it holds $\texttt{val}(b) = +\infty$ and $\partial\texttt{val}(b) = \emptyset$. Moreover, $(\mathcal{D})$ is unbounded, so it has no optimal solutions. □

Natural questions abound, such as when do subgradients exist and how do we compute them? These are the topics of the next section.

8.3 Exercises

Exercise 8.1. How can you convince the governor that the following solution is optimal: $x^*_{\mathrm{AT}} = 2, x^*_{\mathrm{AK}} = 0, x^*_{\mathrm{BT}} = 2, x^*_{\mathrm{BK}} = 1, x^*_{\mathrm{CT}} = 0, x^*_{\mathrm{CK}} = 5$. (Hint: use Weak Duality.)

Exercise 8.2. Prove Lemma 8.2.

Exercise 8.3. Let $f \colon \mathbb{R}^d \to (-\infty, \infty]$ be a convex function. Prove the following results:

1. If $\partial f(x) \neq \emptyset$, then $f(x) = \operatorname{cl} f(x)$.
2. If $f(x) = \operatorname{cl} f(x)$, then $\partial \operatorname{cl} f(x) \subseteq \partial f(x)$.

As a consequence, note that in the setting of Theorem 8.3, we have

$$\partial \texttt{val}(b) = \{y \in \mathbb{R}^m \colon y \text{ optimal for } (\mathcal{D})\}$$

whenever either $\partial \texttt{val}(b) \neq \emptyset$ or $(\mathcal{P})$ is feasible and $\texttt{val}(b) = \texttt{a-val}(b)$.

Exercise 8.4. We call a function $f : \mathbb{R}^d \to (-\infty, \infty]$ *polyhedral* if $\operatorname{epi}(f)$ is polyhedral. Prove that any polyhedral function f admits the representation:

$$f(x) = \max_{i=1,\ldots,n} \{a_i^T x + b_i\} + \delta_{\mathcal{X}}(x), \qquad \forall x \in \mathbb{R}^d$$

where $n \geq 0$, $\mathcal{X} \subseteq \mathbb{R}^d$ is a polyhedral set, and for $i = 1, \ldots, n$, we have $a_i \in \mathbb{R}^d$ and $b_i \in \mathbb{R}$. (See Exercise 7.5 for a definition of $\delta_{\mathcal{X}}$. Hint: write $f(x) = \inf\{t \colon (x,t) \in \operatorname{epi}(f)\}$.)

Does the value function of a polyhedral program admit such a representation?

Exercise 8.5. Use the argument in the introduction to write a proof of Theorem 8.3 Part 1 under the additional assumption that $(\mathcal{P})$ has an optimal solution. (Hint: Let $y \in \partial \texttt{val}(b)$ and consider the function $c^T x - \langle A^T y, x\rangle$ over the cone $\mathcal{K}$.)

Exercise 8.6. Prove that $\texttt{val} : \mathbb{R}^d \to (-\infty, +\infty]$ is closed if for every $\gamma, \tau \in \mathbb{R}$, the set

$$\{x \colon c^T x \leq \gamma, \|Ax\| \leq \tau, x \in \mathcal{K}\}$$

is bounded. Under this condition, prove that whenever $\texttt{val}(b)$ is finite, strong duality holds ($\texttt{val} = \texttt{val}^*$) and there exists a primal optimal solution.

9 Subgradients: Existence, Optimality, and Calculus

Skills. Recognize and exploit convexity and its special cases (linear and conic programming); Learn to "take a dual;" Recognize when "strong duality" holds. Compute first-order necessary optimality conditions with nonsmooth calculus (subdifferentials, normal cones, chain rule); Compute sensitivity of optimization problems with respect to perturbations of input data (value functions).

The results of the last section connect the set of dual solutions of a conic program to "differential" properties of the value function. The value function is not differentiable but it is at least convex, and so we began our study of its subdifferential and showed under mild conditions—mere nonemptyness—that the set of its subgradients and the set of dual solutions coincide. We now take a deeper look into the existence of subgradients, their role in first-order optimality conditions, and methods for computing them.

9.1 Existence of Subgradients

In some cases, subgradients exist throughout the domain of a function. Perhaps most strikingly, this is the case for indicator functions of closed convex sets.

Lemma 9.1. *Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a nonempty closed convex set. Then*

$$\partial \delta_{\mathcal{X}}(x) = \mathcal{N}_{\mathcal{X}}(x), \qquad \forall x \in \mathbb{R}^d.$$

On the other hand, consider the function $f(x) = -\sqrt{x} + \delta_{\mathbb{R}_+}(x)$: it is clearly closed, convex, and differentiable at any positive x, but its subdifferential is empty at zero. Looking at Figure 7, we see the reason: a subgradient v must give rise to a supporting hyperplane of the epigraph, but any supporting hyperplane at $(0, 0)$ is vertical and thus not a linear under approximator of f. The following theorem shows this is the only obstruction to the existence of subgradients. (The reader should recall that a *proper* function takes at least one finite value and never takes value $-\infty$.)

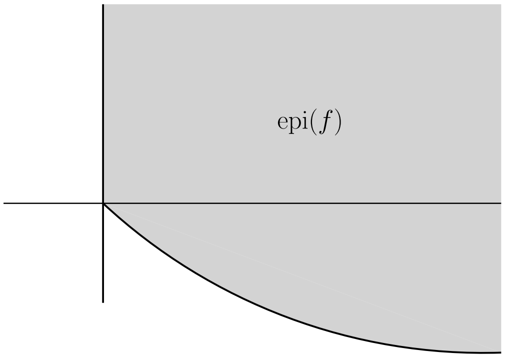


Figure 7: The epigraph of $f(x) = -\sqrt{x} + \delta_{\mathbb{R}_+}(x)$.

Theorem 9.2 (Characterization of Subgradients)**.** *Let $f : \mathbb{R}^d \to (-\infty, \infty]$ be a proper closed convex function. Then for all $x \in \operatorname{dom}(f)$ it holds*

$$\mathcal{N}_{\operatorname{epi}(f)}(x, f(x)) = \big(\mathcal{N}_{\operatorname{dom}(f)}(x) \times \{0\}\big) \cup \{\lambda(v, -1) \colon v \in \partial f(x), \lambda > 0\}. \tag{9.1}$$

Moreover, we have $\mathcal{N}_{\operatorname{epi}(f)}(x, f(x)) \neq \emptyset$ and

$$\partial f(x) \neq \emptyset, \qquad \forall x \in \operatorname{int}(\operatorname{dom}(f)).$$

Proof. We begin with the equality. To that end, let $x \in \operatorname{dom}(f)$. We show that any pair $(w, s) \in \mathbb{R}^{d+1}$ is contained in the left hand hand side if and only if it is contained in the right. To focus on a smaller set of s, we note that $s \leq 0$ for all pairs in the union. Likewise, for any normal $(w, s) \in \mathcal{N}_{\operatorname{epi}(f)}(x, f(x))$, it holds $s \leq 0$ since $(x, f(x) + 1) \in \operatorname{epi}(f)$ and hence

$$0 \geq \langle (w, s), (x, f(x) + 1) - (x, f(x)) \rangle = s.$$

Thus, we look at the cases $s = 0$ and $s < 0$ in turn.

First, a pair $(w, 0)$ is normal to $\operatorname{epi}(f)$ if and only if

$$0 \geq \langle (w,0), (y, f(y)+t) - (x, f(x)) \rangle = \langle w, y - x \rangle, \qquad \forall t \geq 0, \forall y \in \operatorname{dom}(f),$$

a condition equivalent to $w \in \mathcal{N}_{\operatorname{dom}(f)}(x)$. Second a pair (w, s) with $s < 0$ is normal to $\operatorname{epi}(f)$ if and only if

$$0 \geq \frac{1}{-s} \langle (w,s), (y, f(y)+t) - (x, f(x)) \rangle = \langle -(w/s), y - x \rangle + f(x) - f(y) - t, \qquad \forall t \geq 0, \forall y \in \operatorname{dom}(f),$$

a condition equivalent to $-(w/s) \in \partial f(x)$. Combining these equivalences, the equality holds.

To complete the proof, we show nonzero normals exist and moreover subgradients exist at interior points. Nonzero normals exist since $(x, f(x)) \in \operatorname{bdry}(\operatorname{epi}(f))$, a consequence of Corollary 4.3. Now let $(w, s) \in \mathcal{N}_{\operatorname{epi}(f)}(x, f(x))$ a nonzero normal and let x be interior to the domain of f. By rescaling, we may assume $s = 0$ or -1. Since x is interior to the domain, we have $s < 0$: if not, $s = 0$ and $w \in \mathcal{N}_{\operatorname{dom}(f)}(x) = \{0\}$, a contradiction.[13] Thus, we must have $s = -1$, implying $w \in \partial f(x)$. □

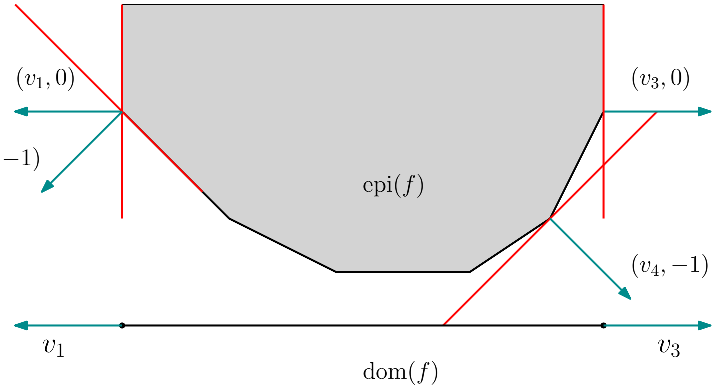


Figure 8: Normals to the Epigraph (compare with Figure 7)

Figure 8 illustrates the geometry of Theorem 9.2, showing that the normals to $\operatorname{epi}(f)$ either point horizontally or correspond to subgradients. Importantly, there can be no horizontal normals when x is interior to $\operatorname{dom}(f)$; trouble arises only at the boundary.

Corollary 9.3 (Existence of Subgradients without Closedness)**.** *Let $f : \mathbb{R}^d \to [-\infty, \infty]$ be a convex function, possibly not closed or proper. Suppose that $x \in \operatorname{int}(\operatorname{dom}(f))$ and $f(x)$ is finite. Then f is proper and*

$$\partial f(x) \neq \emptyset.$$

Proof. Since $f(x)$ is finite and $x \in \operatorname{int}(\operatorname{dom}(f))$, we know that f never takes value $-\infty$ (Exercise 2.6) and it is moreover continuous on the interior of its domain (Theorem 7.9). This continuity implies $f(x') = \operatorname{cl} f(x')$ for all $x' \in \operatorname{int}(\operatorname{dom}(f(x)))$ and hence $x \in \operatorname{int}(\operatorname{dom}(\operatorname{cl} f))$. Thus, since $\operatorname{cl} f(x)$ is finite and $x \in \operatorname{int}(\operatorname{dom}(\operatorname{cl} f))$, we know that $\operatorname{cl} f$ never takes value $-\infty$ (Exercise 2.6) and applying the theorem, we have $\partial \operatorname{cl} f(x) \neq \emptyset$. The proof is complete since $\partial f(x)$ contains $\partial \operatorname{cl} f(x)$ (Exercise 8.3). □

[13] See Exercise 3.2.

9.2 The Optimality Conditions of Conic Programming

Coupling Theorem 9.2 with Theorem 8.3, we give a simple sufficient condition, guaranteeing optimal solutions exist. The reader should recall Definition 7.10 in the following Proposition.

Corollary 9.4 (Existence of Optimal Solutions to Conic Programs)**.** *If* $(\mathcal{P})$ *is strongly feasible and* $(\mathcal{D})$ *is feasible, then* $\texttt{val} = \texttt{val}^*$ *and* $(\mathcal{D})$ *has an optimal solution. Likewise, if* $(\mathcal{D})$ *is strongly feasible and* $(\mathcal{P})$ *is feasible, then* $\texttt{val} = \texttt{val}^*$ *and* $(\mathcal{P})$ *has an optimal solution.*

Proof. Under the first condition, there is a neighborhood $\mathcal{U}$ of b such that for all

$$+\infty > \texttt{val}(b') \geq \texttt{a-val}(b') = \sup\{(b')^T y \colon c - A^T y\} > -\infty, \qquad \forall b' \in \mathcal{U},$$

where the equality is due to Theorem 7.7. From finiteness, both $\texttt{val}$ and $\texttt{a-val}$ are continuous on $\mathcal{U}$ (Theorem 7.9), implying both $\texttt{val}(b) = \texttt{a-val}(b) = \texttt{val}^*$ and $\partial\texttt{val}(b) \neq \emptyset$ (Corollary 9.3). To complete the proof, recall that $\partial\texttt{val}(b)$ comprises the set of dual solutions (Theorem 8.3). We leave the dual statement as an Exercise. □

Let us pause to reflect on this corollary and to connect it to *optimality conditions* in conic programming. More specifically, consider the following *primal-dual system* in the variables $x, s \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$:[14]

$$\begin{array}{rcll} Ax - b & = & 0 & \text{(primal feasibility)} \\ A^T y + s - c & = & 0 & \text{(dual feasibility)} \\ \langle s, x \rangle & = & 0 & \text{(complementary slackness)} \\ x \in \mathcal{K}, s \in \mathcal{K}^* & & & \text{(nonnegativity)} \end{array} \qquad (\mathcal{PDSYS})$$

We can relate solutions of this system to optimal solutions of the conic program:

> **(Sufficiency.)** Given any solution $(\overline{x}, \overline{s}, \overline{y})$ to this system, the vector $\overline{x}$ is primal feasible, the vector $\overline{y}$ is dual feasible, and
>
> $$0 = \langle \overline{s}, \overline{x} \rangle = \langle c - A^T \overline{y}, \overline{x} \rangle = c^T \overline{x} - (A\overline{x})^T y = c^T \overline{x} - b^T \overline{y}.$$
>
> Applying Weak Duality, we find that x and y are primal and dual optimal respectively:
>
> $$c^T \overline{x} \geq \texttt{val} \geq \texttt{val}^* \geq b^T \overline{y} \implies \texttt{val} = \texttt{val}^*.$$
>
> **(Necessity.)** On the other hand, suppose there exists primal optimal x^* and dual optimal y^* and $\texttt{val} = \texttt{val}^*$. Setting $s^* = c - A^T y^*$, we find that $s^* \in \mathcal{K}^*$ and complementary slackness holds. Hence, (x^*, s^*, y^*) solves the primal-dual system.

This argument and Corollary 9.4 yield the following theorem, underlying much algorithmic work in conic optimization.

[14] The vector s is often called a *slack variable*

Theorem 9.5 (Optimality Conditions of Conic Programming)**.** *Suppose $(\mathcal{P})$ is strongly feasible and $(\mathcal{D})$ is feasible. Then a vector x is primal optimal if and only if there exists $y \in \mathbb{R}^m$ so that $(x, c - A^T y, y)$ solves $(\mathcal{PDSYS})$. In this case, any such y is dual optimal.*

Similarly, suppose $(\mathcal{D})$ is strongly feasible and $(\mathcal{P})$ is feasible. Then a point $y \in \mathbb{R}^m$ is dual optimal if and only if there exists $x \in \mathbb{R}^d$ so that $(x, c - A^T y, y)$ solves $(\mathcal{PDSYS})$. In this case, any such x is primal optimal.

Complementary slackness is important for algorithm design since with it one can check whether a candidate solution is optimal, simply by solving a small linear equation, at least when $\mathcal{K} = \mathbb{R}^d$. This is so since the vectors x and s are nonnegative, and thus if $x_i > 0$, then $s_i = 0$ and $(A^T y)_i = c_i$. Linear programs enjoy an even stronger result: when optimal solutions exist, to some primal solution x there corresponds a dual solution y and slack variable $s = c - A^T y$ such that s is strictly complementary to x: $x_i > 0$ if and only if $s_i = 0$. You will prove this fact in the Exercises.

Theorem 9.6 (Optimality Conditions of Linear Programming)**.** *Suppose $\mathcal{K} = \mathbb{R}^d_+$ and both $(\mathcal{P})$ and $(\mathcal{D})$ are feasible. Then a vector x is primal optimal if and only if there exists $y \in \mathbb{R}^m$ so that $(x, c - A^T y, y)$ solves $(\mathcal{PDSYS})$. In this case, any such y is dual optimal. Moreover, there exists a primal optimal x, a dual optimal y, and slack variable $s = c - A^T y$ such that* strict complementary slackness *holds:*

$$x_i > 0 \textit{ if and only if } s_i = 0, \qquad \textit{for } i = 1, \ldots, d \qquad\qquad (\mathcal{SCS})$$

Likewise a point $y \in \mathbb{R}^m$ is dual optimal if and only if there exists $x \in \mathbb{R}^d$ so that $(x, c - A^T y, y)$ solves $(\mathcal{PDSYS})$. In this case, any such x is primal optimal. Finally, the pair (x, y) is primal dual optimal if and only if $(x, c - A^T y, y)$ solves $(\mathcal{PDSYS})$.

To see the algorithmic utility of complementary slackness, consider the primal-dual pair of linear programs:

$$\begin{array}{llll}
\text{minimize } 0x_1 + 3x_2 + 0x_3 + 3x_4 & & \qquad & \text{maximize } 4y_1 + 5y_2 \\
\text{subject to : } \begin{bmatrix} -1 & 2 & 0 & 1 \\ 0 & 1 & -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} & = \begin{bmatrix} 4 \\ 5 \end{bmatrix}, & \qquad & \text{subject to : } \begin{bmatrix} -1 & 0 \\ 2 & 1 \\ 0 & -1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 3 \\ 0 \\ 3 \end{bmatrix} \\
\qquad\qquad\quad \begin{bmatrix} x_1, & x_2, & x_3, & x_4 \end{bmatrix} & \geq 0 & &
\end{array}$$

Suppose we have the primal feasible point $x = (0, 1, 0, 2)$ and we want to know whether x is optimal. If x were optimal, there would exist a dual feasible y so that the second and fourth constraints of the dual linear program are active, meaning

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}.$$

The unique solution to this equation is $y = (1, 1)$. A quick check shows that this y is dual feasible. Thus the triple $(x, c - A^T y, y)$ is a solution to the primal-dual system and the pair (x, y) is primal-dual optimal.

9.3 Optimality Conditions in General

We have so far seen two sorts of optimality conditions: those arising from conic programs and those arising from certain mathematical programs. Both are united by the following far reaching generalization of Fermat's rule.

Theorem 9.7 (Fermat's Rule). *Let $f : \mathbb{R}^d \to (-\infty, \infty]$ be a proper function and suppose that $\bar{x}$ is a local minimizer of f. Then*

$$0 \in \partial_F f(\bar{x}).$$

If moreover f is convex, the condition $0 \in \partial f(x)$ is both necessary and sufficient for x to be a global minimum.

Let us illustrate Fermat's rule with the conic program $(\mathcal{P})$. To that end, define

$$f(x) := \langle c, x\rangle + \delta_{\mathcal{K}}(x) + \delta_{\{x\colon Ax=b\}}(x).$$

Then x is a minimizer of the conic program $(\mathcal{P})$ if and only if

$$0 \in \partial f(x) = \partial(\langle c, \cdot\rangle + \delta_{\mathcal{K}}(\cdot) + \delta_{\{x\colon Ax=b\}}(\cdot))(x) \overset{?}{=} c + \mathcal{N}_{\mathcal{K}}(x) + \mathcal{N}_{\{x\colon Ax=b\}}(x).$$

Using the identities $\mathcal{N}_{\mathcal{K}}(x) = -\mathcal{K}^* \cap \{x\}^\perp$ and $\mathcal{N}_{\{x\colon Ax=b\}}(x) = \{A^T y\colon y \in \mathbb{R}^m, Ax = b\}$ for feasible x, we find that x is a minimizer if and only if it is primal feasible and there is a $y \in \mathbb{R}^m$ with

$$c - A^T y \in \mathcal{K}^* \cap \{x\}^\perp.$$

Equivalently, the triple $(x, c - A^T y, y)$ satisfies the primal-dual system $(\mathcal{PDSYS})$, meaning the pair (x, y) is primal-dual optimal.

Though appealingly simple, this argument is not valid in general since the sum rule $\partial(f + g) \overset{?}{=} \partial f + \partial g$ can fail, even in $\mathbb{R}^2$. For example, consider the indicator functions of the unit balls centered at $(-1, 0)$ and $(1, 0)$:

$$f(x) = \delta_{B_1(-1,0)} \qquad \text{and} \qquad g = \delta_{B_1(1,0)}.$$

Then since $f + g = \delta_{\{0\}}$, we have

$$\partial(f + g)(0) = \mathbb{R}^2 \neq \mathbb{R} \times \{0\} = \partial f(0) + \partial g(0),$$

a failure. Nevertheless, the sum rule can succeed, and in the next section, we learn when it does.

9.4 Calculus

Nonsmooth calculus offers another perspective on duality, and leads to primal-dual systems for general convex programs. We have verified this claim for conic programs, at least when a "sum rule" holds. Even for conic programs, counterexamples show that the classical rules from multivariate calculus can fail. In this section, we determine when they succeed. Beginning with differentiable functions, we show that subgradients are full-fledged gradients.

Lemma 9.8 (Differentiable Functions)**.** *Let $f\colon \mathbb{R}^d \to (-\infty, \infty]$ be a proper convex function. Then if f is Fréchet differentiable at x, it holds*

$$\partial f(x) = \{\nabla f(x)\}.$$

Proof. The gradient $\nabla f(x)$ is a subgradient by Lemma 8.2. On the other hand, if $v \in \partial f(x)$, then x minimizes the differentiable function $f(x') - \langle v, x' \rangle$. By first-order optimality conditions $\nabla f(x) - v = 0$. This completes the proof. □

Perhaps less surprising is the following partial *sum rule,* saying if one is given two functions and a linear under approximator for each, one may add them together and get a linear under approximator of the sum.

Lemma 9.9 (Partial Sum Rule)**.** *Let $f, g\colon \mathbb{R}^d \to (-\infty, \infty]$ be proper convex functions. Then*

$$\partial f(x) + \partial g(x) \subseteq \partial (f+g)(x) \qquad \forall x \in \operatorname{dom}(f) \cap \operatorname{dom}(g).$$

A similar argument leads to the following *exact* sum rule for *separable functions.*

Lemma 9.10 (Separable Sum Rule)**.** *Let $d = d_1 + \ldots + d_n$ for integers d_i and let $f_i\colon \mathbb{R}^{d_i} \to (-\infty, +\infty]$ be proper convex functions. Then*

$$\partial (f_1 + \ldots + f_n)(x_1, \ldots, x_n) = \partial f_1(x_1) \times \ldots \times \partial f_n(x_n) \qquad \forall x_i \in \operatorname{dom}(f_i)$$

This rule is exact since one can vary the coordinates independently, an impossibility in Lemma 9.9.

Finally, a partial *chain rule* holds.

Lemma 9.11 (Partial Chain Rule)**.** *Let $f\colon \mathbb{R}^m \to (-\infty, \infty]$ be a proper convex function and let $A \in \mathbb{R}^{m \times d}$ be a matrix. Then*

$$A^T \partial f(Ax) \subseteq \partial (f \circ A)(x) \qquad \forall x \in A^{-1}(\operatorname{dom}(f)).$$

Proof. Given $v \in \partial f(Ax)$, one has $\langle v, Ax' - Ax \rangle + f(Ax) \leq f(Ax')$ for any $x' \in \operatorname{dom}(f \circ A)$. Since $\langle v, Ax' - Ax \rangle = \langle A^T v, x' - x \rangle$, the inclusion holds: $A^T v \in \partial (f \circ A)(x)$. □

We warn that equality can fail since the chain rule extends the sum rule. For example, stacking two identity matrices

$$A = \begin{bmatrix} I_d \\ I_d \end{bmatrix}$$

and defining the separable sum $h(y, z) = f(y) + g(z)$, we find two distinct sets, unequal in general:

$$A^T \partial h(Ax) = \partial f(x) + \partial g(x) \qquad \text{and} \qquad \partial h(Ax) = \partial (f+g)(x).$$

When then do nonsmooth calculus rules hold? The key enabling condition, used in the next theorem, is known as a *constraint qualification*, and as we will see, it generalizes the *strong feasibility* condition of conic programming.

Theorem 9.12 (Chain Rule)**.** *Let $f\colon \mathbb{R}^d \to (-\infty,\infty]$ and $g\colon \mathbb{R}^m \to (-\infty,\infty]$ be proper convex functions and let $A \in \mathbb{R}^{m\times d}$. Suppose the following regularity condition holds*

$$0 \in \operatorname{int}(\operatorname{dom}(g) - A\operatorname{dom}(f)).$$

Then equality holds

$$\partial(f + g\circ A)(x) = \partial f(x) + A^T\partial g(Ax), \qquad \forall x \in \operatorname{dom}(f + g\circ A).$$

In particular, equality holds if either (a) g is continuous at a point in $A\operatorname{dom}(f)$ or (b) $\operatorname{rank}(A) = m$ and f is continuous at a point in $A^{-1}(\operatorname{dom}(g))$.

Proof. Applying the earlier results, it holds $\partial f(x) + A^T\partial g(Ax) \subseteq \partial(f + g\circ A)(x)$. We may thus assume that $\partial(f + g\circ A)(x)$ is nonempty, since otherwise there is nothing to prove. Then letting $v \in \partial(f + g\circ A)(x)$, Lemma 8.2 shows

$$x \text{ minimizes } f(x') + g(Ax') - \langle v, x'\rangle.$$

To see how v relates to the subdifferentials of f and g, we place this problem within a larger family of perturbed programs and track the values of the perturbations with a value function $V : \mathbb{R}^m \to [-\infty,\infty]$:

$$V(b) := \inf_{x'\in\mathbb{R}^d}\{f(x') + g(Ax' + b) - \langle v, x'\rangle\}, \qquad \forall b \in \mathbb{R}^m.$$

Crucial to understanding v is the following claim: $\partial V(0) \neq \emptyset$. The claim follows since V is convex (check!) and V meets the criteria of Corollary 9.3: First $V(0) = f(x)+g(Ax)-\langle v, x\rangle$ is finite. Second the domain of V is simply $\operatorname{dom}(V) = \operatorname{dom}(g) - A\operatorname{dom}(f)$, so by assumption $0 \in \operatorname{int}(\operatorname{dom}(V))$.

Thus, letting $w \in \partial V(0)$, we have

$$\begin{aligned} 0 &\le V(b) - V(0) - \langle w, b\rangle \\ &\le f(x') + g(Ax' + b) - \langle v, x'\rangle - (f(x) + g(Ax) - \langle v, x\rangle) - \langle w, b\rangle, \qquad \forall b \in \mathbb{R}^m, \forall x' \in \mathbb{R}^d \end{aligned}$$

Let us test against various x' and b, showing that $v - A^Tw \in \partial f(x)$ and $w \in \partial g(Ax)$:

($v - A^Tw \in \partial f(x)$). Let $x' \in \operatorname{dom}(f)$ and $b = Ax - Ax'$. Then

$$\begin{aligned} 0 &\le f(x') - f(x) - \langle v, x' - x\rangle - \langle w, A(x' - x)\rangle \\ &= f(x') - f(x) - \langle v - A^Tw, x' - x\rangle, \qquad \forall x' \in \operatorname{dom}(f). \end{aligned}$$

Therefore, $v - A^Tw \in \partial f(x)$.

($w \in \partial g(Ax)$). Let $x' = x$, then

$$sss$$

Therefore, $w \in \partial g(Ax)$.

To complete the proof, write $v = (v - A^Tw) + A^Tw \in \partial f(x) + A^T\partial g(Ax)$, as desired. □

Let us return to the primal conic program ($\mathcal{P}$) and see the consequences of the chain rule. To that end, define

$$f(x) := \delta_{\mathcal{K}}(x) + c^T x \qquad \text{and} \qquad g(y) := \delta_{\{b\}}(y), \qquad \forall x \in \mathbb{R}^d, \forall y \in \mathbb{R}^m,$$

and observe that x minimizes ($\mathcal{P}$) if and only if x minimizes $f + g \circ A$. At any such optimal point, Fermat's rule says $0 \in \partial(f + g \circ A)(x)$. With the chain rule we can compute this subdifferential if the constraint qualification holds:

$$0 \in \text{int}(\text{dom}(g) - A\text{dom}(f)) = \text{int}(b - A\mathcal{K}).$$

This condition in turn is equivalent to strong feasibility of ($\mathcal{P}$). Thus if ($\mathcal{P}$) is strongly feasible, the chain rule shows

$$\begin{aligned} 0 &\in \partial(f + g \circ A)(x) \\ &= \partial f(x) + A^T \partial g(Ax) \\ &= c + \partial \delta_{\mathcal{K}}(x) + A^T \partial \delta_{\{b\}}(Ax) \\ &= c + \mathcal{N}_{\mathcal{K}}(x) + A^T \mathbb{R}^d, \end{aligned}$$

Equivalently, a feasible x solves ($\mathcal{P}$) if and only if there exists $y \in \mathbb{R}^d$ such that

$$(c - A^T y) \in \mathcal{K}^* \cap \{x\}^\perp,$$

where we simplified the inclusion with the help of the equality $\mathcal{N}_{\mathcal{K}}(x) = -\mathcal{K}^* \cap \{x\}^\perp$ (Exercise 6.2). In short, for any (x, y) that solves the inclusion, the triple $(x, c - A^T y, y)$ solves the primal-dual system ($\mathcal{PDSYS}$), meaning the pair (x, y) is primal-dual optimal and $\texttt{val} = \texttt{val}^*$.

Beyond their role in optimality conditions, subdifferentials feature in modern large scale optimization algorithms. We will explore this in the next section.

9.5 Exercises

Exercise 9.1 (Strict Complementary Slackness)**.** In this exercise, we examine the strict complementary slackness condition. To that end consider the following primal-dual pair of linear programs:

$$\begin{aligned} &\text{minimize } c^T x & \qquad & \text{maximize } b^T y \\ &\text{subject to : } Ax = b & & \text{subject to : } A^T y + s - c = 0 \\ &\qquad\qquad x \in \mathbb{R}_+ & & \qquad\qquad s \geq 0 \end{aligned} \tag{9.2}$$

Throughout this exercise, we suppose that optimal solutions exist. Consider the following condition.

Condition. Suppose that there is some $j \in \{1, \dots, d\}$ so that every optimal solution x^* satisfies $x_j^* = 0$.

In the next three parts, suppose the above condition holds. Under this condition, we will prove there is a dual optimal pair (y, s) with $s_j > 0$.

1. Consider the following linear program:
$$\begin{aligned} &\text{minimize } -x_j \\ &\text{subject to : } Ax = b \\ &\qquad\qquad\quad c^T x \leq \texttt{val} \\ &\qquad\qquad\quad x \geq 0. \end{aligned}$$
Show that its dual is
$$\begin{aligned} &\text{maximize } b^T y - t\texttt{val} \\ &\text{subject to : } A^T y - tc + s = -e_j \\ &\qquad\qquad\quad s, t \geq 0, \end{aligned}$$
where e_j denotes the jth standard basis vector. Prove that this dual has an optimal solution $(\bar{y}, \bar{t}, \bar{s})$ and show that $b^T \bar{y} = \bar{t}\texttt{val}$.
2. Suppose $\bar{t} > 0$ and let $y = \bar{y}/t$ and $s = (\bar{s} + e_j)/\bar{t}$. Prove that $s_j > 0$ (obvious) and (y, s) solves the original dual problem.
3. Suppose that $\bar{t} = 0$. Find an optimal solution (y, s) to the original dual problem with $s_j > 0$.

Using the above results, we can construct a primal-dual pair satisfying the strict complementary slackness condition. To that end, define a subset of indices $J \subseteq \{1, \dots, d\}$ by the following formula
$$J := \{j \colon \exists \text{ primal optimal } x \text{ with } x_j > 0\}.$$
Using J, we will construct a sequence $(x^1, y^1), \dots, (x^d, y^d)$ of primal-dual optimal pairs with the following properties: For each $j \in J$, we let y^j be an arbitrary dual optimal solution and let x^j be a primal optimal solution with $x_j^j > 0$. On the other hand, for each $j \notin J$, we let x^j be an arbitrary primal optimal solution and let y^j be a dual optimal optimal solution with $(c - A^T y^j)_j > 0$ (exists by Parts 1-3). Given these primal-dual optimal pairs, define
$$x^* := \frac{1}{d} \sum_{j=1}^{d} x^j \qquad \text{and} \qquad y^* := \frac{1}{d} \sum_{j=1}^{d} y^j.$$

4. **Bonus.** Show that the pair (x^*, y^*) is primal-dual optimal and in addition satisfies strict complementary slackness, namely,
$$x_j^* > 0 \text{ if and only if } (c - A^T y^*)_j = 0, \qquad \forall j$$

Exercise 9.2 (Existence of Optimal Solutions). Prove the second statement of Corollary 9.4. (Hint: Leverage the symmetry between primal and dual programs, using the discussion that closed Section 7.2.)

Exercise 9.3 (Fermat's Rule). Prove Theorem 9.7.

Exercise 9.4 (The Value Function is Convex). Show that the function V in Theorem 9.12 is convex.

Exercise 9.5 (Easy Subdifferential Facts)**.**

1. Let $f\colon \mathbb{R}^d \to (-\infty, \infty]$ be a closed, proper, convex function. Show that for all $x \in \operatorname{dom}(f)$, the set $\partial f(x)$ is closed and convex.
2. Let $d = d_1 + \ldots + d_n$ for integers d_i and let $f_i\colon \mathbb{R}^{d_i} \to (-\infty, +\infty]$ be proper convex functions. Then
$$\partial(f_1 + \ldots + f_n)(x_1, \ldots, x_n) = \partial f_1(x_1) \times \ldots \times \partial f_n(x_n) \qquad \forall x_i \in \operatorname{dom}(f_i)$$
3. Let $f\colon \mathbb{R}^d \to (-\infty, \infty]$ be a closed, proper, convex function and let $\lambda > 0$. Then prove that the function $g = \lambda f$ is satisfies
$$\partial g(x) = \lambda \partial f(x), \qquad \forall x \in \operatorname{dom}(f).$$
4. Let $f\colon \mathbb{R}^d \to (-\infty, \infty]$ be a closed, proper, convex function and let $b \in \mathbb{R}^d$. Then prove that the shifted function $g(\cdot) = f((\cdot) + b)$ satisfies
$$\partial g(x) = \partial f(x + b), \qquad \forall x \in \operatorname{dom}(f) - \{b\}.$$

.

Exercise 9.6 (Subdifferential of Scaling)**.** Let $f\colon \mathbb{R}^d \to (-\infty, \infty]$ be a closed, proper, convex function and let $\lambda > 0$. Then prove that the function $g = \lambda f$ is satisfies
$$\partial g(x) = \lambda \partial f(x), \qquad \forall x \in \mathbb{R}^d.$$

Exercise 9.7 (Subgradient Computations)**.** Compute the subdifferentials of the following functions on $\mathbb{R}^d$ (some are differentiable, others are easy applications of the chain rule):

1. **ℓ_1 norm.** $f(x) = \|x\|_1 = \sum_{i=1}^d |x_i|$.
2. **Hinge loss.** $f(x) = \max\{0, x\}$ (where $d = 1$).
3. **Hybrid Norm.** $f(x) = \sqrt{1 + x^2}$ (where $d = 1$).
4. **Logistic function.** $f(x) = \log(1 + \exp(x))$ (where $d = 1$).
5. **Indicator of ℓ_p ball.** $f(x) = \delta_{\mathcal{X}}(x)$ where for $p \in [1, \infty]$ and $\tau > 0$, we have $\mathcal{X} = \{x\colon \|x\|_p \leq \tau\}$.
6. **Max of coordinates.** $f(x) = \max\{x_1, \ldots, x_d\}$.
7. **Polyhedral Function.** $f(x) = \max_{i \leq m}\{\langle a_i, x\rangle + b_i\}$ where $a_1, \ldots, a_m \in \mathbb{R}^d$ are vectors and $b_1, \ldots, b_m \in \mathbb{R}$
8. **Quadratic.** $f(x) = \frac{1}{2}\langle Ax, x\rangle$ for some symmetric matrix $A \in \mathbb{R}^{d \times d}$.
9. **Least Squares.** $f(x) = \frac{1}{2}\|Ax - b\|_2^2$ where $A \in \mathbb{R}^{m \times d}$ and $b \in \mathbb{R}^m$.
10. **Least Absolute Deviations.** $f(x) = \|Ax - b\|_1$ where $A \in \mathbb{R}^{m \times d}$ and $b \in \mathbb{R}^m$.

Exercise 9.8 (Mean Value Theorem)**.** Suppose $f : \mathbb{R}^d \to \mathbb{R}$ is a closed convex function and let $x, y \in \mathbb{R}^d$. Show that there exists $t \in [0, 1]$ such that

$$f(x) - f(y) \in \langle x - y, \partial f((1 - t)x + ty)\rangle$$

(Hint: consider the convex function $t \mapsto f((1 - t)x + ty) + t(f(x) - f(y))$ on the compact interval $[0, 1]$.)

The next exercise relies on the following definition.

Definition 9.13 (Lipschitz Continuity)**.** *A function* $f : \mathbb{R}^d \to \mathbb{R}$ *is called* Lipschitz continuous *if*

$$|f(x) - f(y)| \le L\|x - y\|, \qquad \forall x, y \in \mathbb{R}^d.$$

for some $L > 0$*. The constant* L *is called a* Lipschitz constant *of* f*.*

Exercise 9.9 (Bounded Subgradients and Lipschitz Continuity)**.** Let $f : \mathbb{R}^d \to \mathbb{R}$ be a closed convex function. Show that f is Lipschitz continuous with Lipschitz constant L if and only if for any $x \in \mathbb{R}^d$, we have

$$v \in \partial f(x) \implies \|v\| \le L.$$

Exercise 9.10. Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a closed convex set. Define the distance function

$$\operatorname{dist}(x, \mathcal{X}) := \inf_{y \in \mathcal{X}} \|y - x\|, \qquad \forall x \in \mathbb{R}^d.$$

Prove that $\operatorname{dist}(x, \mathcal{X})$ is closed, convex, and 1-Lipschitz continuous. Show that

$$\partial \operatorname{dist}(x, \mathcal{X}) = \begin{cases} \mathcal{N}_{\mathcal{X}}(x) & \text{if } x \in \mathcal{X} \\ \left\{ \frac{1}{\operatorname{dist}(x,\mathcal{X})}(x - \operatorname{proj}_{\mathcal{X}}(x)) \right\} & \text{otherwise.} \end{cases}$$

Exercise 9.11. Let $f \colon \mathbb{R}^d \to (-\infty, \infty]$ be a closed, proper, and convex function. Prove that there exists $x \in \mathbb{R}^d$ such that $\partial f(x) \ne \emptyset$. (Hint: let $x_0 \in \mathbb{R}^d$ be a point where $f(x_0)$ is finite and consider the closed, proper, and convex function $g = f + \delta_{B_1(x_0)}$. Use Exercise 2.5 to show that g has a minimizer $\bar{x} \in B_1(x_0)$. Fermat's rule then shows that $0 \in \partial g(\bar{x})$. Conclude by using Theorem 9.12 to show $0 \in \partial f(\bar{x}) + \partial \delta_{B_1(x_0)}(\bar{x})$.)

10 First-Order Models and Algorithms

Skills. Learn a toolbox of algorithms (first order methods); Choose appropriate algorithms by understanding tradeoffs induced by problem structure; Characterize algorithmic complexity.

It is now a natural time to reflect on the aim of this course, which was for you to acquire a firm working knowledge of the techniques and results of modern optimization, focusing on structure, duality, nonsmooth calculus, and algorithms. We have so far developed skills in the first three areas, first emphasizing the role of convexity in the duality and sensitivity theory of conic programing and second establishing a rich calculus of necessary and sufficient optimality conditions for convex programs. The aim of this section is to bring these skills to bear on algorithm design in convex optimization.

To fix notation, we assume throughout this section that we wish to

$$\text{minimize } f(x)$$

where $f\colon \mathbb{R}^d \to (-\infty, \infty]$ is a proper closed convex function that has a nonempty set of minimizers $\mathcal{X}^* = \operatorname{argmin} f$. As we move through the section we will place further useful assumptions on f, for example, continuity or differentiability, but for now we consider this simple setting. We take the view that finding an exact minimizer of f may be impossible, which leads us to instead search for *approximate minimizers* $\hat{x}$ of f. We can measure how well $\hat{x}$ minimizes f in a couple of ways, for example, by measuring its *objective error* $f(\hat{x}) - \inf f$. Our goal is to design algorithms that take as input $\varepsilon > 0$ and output approximate minimizers $\hat{x}$ with objective error less than ε: $f(\hat{x}) - \inf f \leq \varepsilon$.

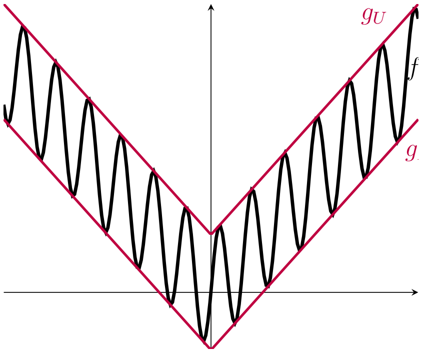


Figure 9: Tight upper and lower approximation of a function.

To find approximate minimizers of f we follow a well-worn path: we approximate f by a *model* function and then minimize this model. The hope is that minimizers of the model are approximate minimizers of f. For example, consider the lucky situation in Figure 9. We seek to minimize f, and we are in possession of both an upper model g_U and a lower model g_L, having the following properties:

$$g_L(x) \leq f(x) \leq g_U(x), \quad \forall x \qquad\qquad \text{and} \qquad\qquad |g_L(x) - g_U(x)| \leq \varepsilon, \quad \forall x.$$

Owing to their simplicity, we easily identify 0 to be the minimizers of g_L and g_U, and since g_L and g_U tightly approximate f, we expect 0 to nearly minimize f as well. Indeed, letting x^* minimize f, we can upper bound the objective error $f(0) - \inf f$ as follows:

$$f(0) \leq g_U(0) \leq g_L(0) + \varepsilon \leq g_L(x^*) + \varepsilon \leq \inf f + \varepsilon$$

Thus 0 nearly minimizes f, and the algorithm was successful.

Though successful in this example, we typically cannot implement algorithms based on tight global models, since in practice such models are hard to come by. Despite the impracticality of this idea, we can still salvage its overarching strategy by carrying it out iteratively using *local models* in the following way: we build a local model of f, minimize it in a small neighborhood, build a new model of f at this minimizer, and so on. Such algorithms form a sequence of *iterates*: $x_0, x_1, \ldots$ that we view as approximate minimizers of f. The goals of this section are to show one can in fact generate approximate minimizers of f using this strategy, and in addition that one can give estimates for how quickly the iterates approach a minimizer of f. To show this, we must first develop a proper notion of a local model, and then design algorithms that profitably use them.

10.1 From Global to Local Models

Let us give a preview of things to come with linear models and two classical algorithms: the *gradient descent algorithm* and the *subgradient method.*

10.1.1 Linear Models: Gradient Descent and the Subgradient Method

The gradient descent algorithm applies to differentiable f and is based on the following classical fact from calculus: the derivative of a function $\varphi : \mathbb{R} \to \mathbb{R}$ gives rise to a local linear approximation of φ: $\varphi(t) \approx \varphi(t_0) + \varphi'(t_0)(t - t_0)$. The Fréchet gradient (Definition 3.1) lifts this approximation to higher dimensions:

$$f(y) \approx f(x) + \langle \nabla f(x), y - x\rangle, \qquad \forall x, y \in \mathbb{R}^d.$$

To use this approximation in an algorithm, we define the following linear model of f at any x: $f_x(y) := f(x) + \langle \nabla f(x), y - x\rangle$ for all $y \in \mathbb{R}^d$.[15] Then we define a procedure that in some sense locally minimizes the models f_x: given $x_0, \ldots, x_k$, define the next iterate x_{k+1} by

$$x_{k+1} = \operatorname*{argmin}_y \left\{ f_{x_k}(y) + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}, \tag{10.1}$$

for some $\rho_k > 0$. Let us unpack this rule and develop intuition, especially for the scalar ρ_k. To that end, consider the limiting case where $\rho_k = 0$. In this case, the iterate x_{k+1} is the minimizer of the linear function $f_{x_k}(y)$ over all $y \in \mathbb{R}^d$. A little thought shows that unless f_{x_k} is constant (meaning $\nabla f(x_k) = 0$), x_{k+1} cannot exist, since linear functions are unbounded below. Keeping this in mind, we see the necessity of the *quadratic penalty* $\frac{\rho_k}{2}\|y - x_k\|^2$ with $\rho_k > 0$: the upward pull of the penalty not only forces the function $f_{x_k}(y) + \frac{\rho_k}{2}\|y - x_k\|^2$ to be bounded below, but also forces x_{k+1} to stay near x_k where the model is most accurate. All the while the update allows for some decrease in the linear model f_{x_k}, and if f_{x_k} approximates f well enough, it is possible that f decreases as well. We can see the role of ρ_k even more readily by directly computing x_{k+1} (check!):

$$x_{k+1} = x_k - \frac{1}{\rho_k}\nabla f(x_k). \tag{$\mathcal{GD}$}$$

From this form, we see that larger values of ρ_k lead x_{k+1} to be nearer to x_k, while in the case $\rho_k = 0$, the iterate x_{k+1} is ill-defined. This form also shows that the intuitive idea of locally

[15] We will use this notation throughout the section to denote models "centered" at a point x. For now we focus on linear models to establish the basic ideas.

minimizing linear models give rise to the classical *gradient descent* algorithm, originally introduced by Cauchy in 1847.

Moving beyond differentiable functions, the subgradient method builds local linear models not from gradients, but from subgradients: given $x \in \mathbb{R}^d$ and $v_x \in \partial f(x)$ we define

$$f_x(y) := f(x) + \langle v_x, y - x\rangle, \qquad \forall y \in \mathbb{R}^d.$$

Applying Algorithm (10.1) to these models, we arrive at the classical *subgradient method* of Shor: given $x_0, \ldots, x_k$, choose $v_k \in \partial f(x_k)$ and define the next iterate x_{k+1} by

$$x_{k+1} = x_k - \frac{1}{\rho_k} v_k. \tag{$\mathcal{SM}$}$$

Fixing these algorithms, we next seek to understand how quickly the iterates $x_0, x_1, \ldots$ approach an approximate minimizer. In seeking to quantify the the speed of an algorithm, we typically ask for lower and upper bounds on a certain function $K(\varepsilon)$ of the target accuracy $\varepsilon > 0$, defined as follows: for any $\varepsilon > 0$, we let $K = K(\varepsilon) > 0$ be the smallest integer with $f(x_K) - \inf f \leq \varepsilon$.[16] The function $K(\varepsilon)$ is called the *iteration complexity* of the algorithm. Intuitively, those algorithms with "smaller" complexity approach minimizers faster.

As one would suspect, useful bounds on complexity require the models f_x to be "close enough" to f, and models that poorly approximate f may lead to the estimate $K(\varepsilon) = +\infty$. For example, the loss $f(x) = \exp(x) + \exp(-x)$ is poorly approximated by its linear model, since it grows exponentially fast. To see the effect of this poor approximation, we apply the gradient descent algorithm ($\mathcal{GD}$) to f with starting point x_0, generating the following sequence:

$$x_{k+1} = x_k - (1/\rho) f'(x_k) = x_k - (1/\rho)(\exp(x_k) - \exp(-x_k)).$$

It is then a simple exercise to show that for any ρ there is an initial point x_0 for which the sequence x_k quickly diverges (check!).

Given that poor approximations can lead to divergence, we see that we must impose some conditions on how well f_x approximates f. Classically, two types of conditions have featured: fixing $q, l \geq 0$, we say a model is *q-quadratically accurate* at x if

$$f_x(y) \leq f(y) \leq f_x(y) + \frac{q}{2}\|y - x\|^2, \qquad \forall y \in \mathbb{R}^d. \tag{10.2}$$

and we say a model is *linearly accurate* at x if

$$f_x(y) \leq f(y) \leq f_x(y) + l\|y - x\|, \qquad \forall y \in \mathbb{R}^d. \tag{10.3}$$

See Figure 10 for an illustration.

To ensure models are quadratically or linearly accurate, the literature on gradient and subgradient methods typically refer to one of two sufficient conditions, both of which ensure different bounds on the complexity $K(\varepsilon)$. First in order to ensure a linear model is quadratically accurate, the function f is assumed to be differentiable and its gradient is assumed to be Lipschitz. We will prove this result in Section 10.6.

[16] Here we deliberately suppress the dependence of $K(\varepsilon)$ on other quantities, such as x_0 and f.

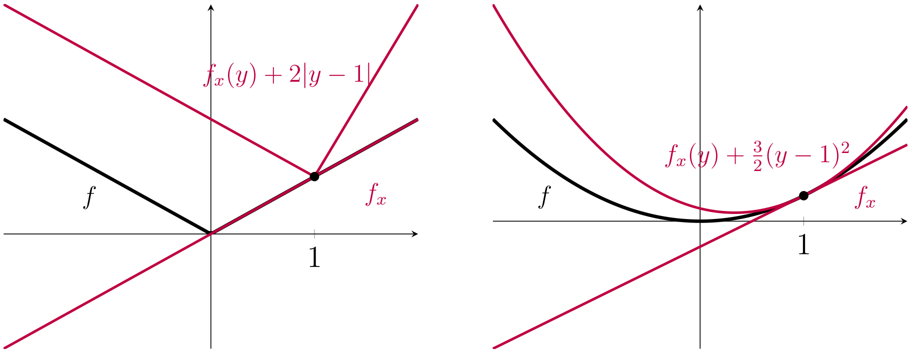


Figure 10: Linearly and quadratically accurate models.

Proposition 10.1 (Lipschitz Gradient and Quadratically Accurate Model)**.** *Suppose that f is differentiable and that ∇f is $\hat{q}$-Lipschitz for some $\hat{q} > 0$, meaning*

$$\|\nabla f(x) - \nabla f(y)\| \le \hat{q}\|x - y\|, \qquad x, y \in \mathbb{R}^d.$$

Then the linear model $f_x(y) = f(x) + \langle \nabla f(x), y - x\rangle$ $\hat{q}$-quadratically accurate.

Next in order to ensure a linear model is linearly accurate, the function f is assumed to be Lipschitz continuous. We will prove this result in Section 10.6.

Proposition 10.2 (Lipschitz Function and Linearly Accurate Model)**.** *Suppose that f is $\hat{l}$-Lipschitz for some $\hat{l} > 0$:*

$$|f(x) - f(y)| \le \hat{l}\|x - y\|, \qquad \forall x, y \in \mathbb{R}^d.$$

Then any linear model $f_x(y) = f(x) + \langle v_x, y - x\rangle$ with $v_x \in \partial f(x)$ is $(2\hat{l})$-linearly accurate.

Having seen that simple sufficient conditions lead to accurate linear models, we now discuss the complexity of the gradient and subgradient methods. First if f is Fréchet differentiable and ∇f is $\hat{q}$ Lipschitz continuous, we will later show in Corollary 10.13 that the gradient descent algorithm ($\mathcal{GD}$) with $\rho_k \equiv \hat{q}$ has complexity

$$K(\varepsilon) \le \left\lceil \hat{q} \cdot \frac{\operatorname{dist}^2(x_0, \mathcal{X}^*)}{2\varepsilon} \right\rceil. \tag{10.4}$$

Second if f is $\hat{l}$-Lipschitz, we will later show that the subgradient method ($\mathcal{SM}$) with appropriate ρ_k has complexity

$$K(\varepsilon) \le \left\lceil 4\hat{l}^2 \cdot \frac{\operatorname{dist}^2(x_0, \mathcal{X}^*)}{\varepsilon^2} \right\rceil. \tag{10.5}$$

Two important issues are how these complexities compare and whether they can be improved.

The complexities shown above differ from each other substantially in their dependence on ε as $\varepsilon \to 0$ ($\frac{1}{\varepsilon} \ll \frac{1}{\varepsilon^2}$), showing the subgradient method is much "slower" than the gradient method. We should intuitively expect this result, since quadratically accurate models give tighter approximations near the base point: $\frac{q}{2}\|x - y\|^2 \ll l\|y - x\|$ as $y \to x$. We will later verify this intuition and show it is impossible to improve the complexity of any natural variation of the basic "subgradient method"—a classical result of Nemirovski and Yudin [**?**]. This result implies that there is a complexity gap between those algorithms based on linearly accurate models and those based on quadratically accurate models. Surprisingly, we will show the gap $\frac{1}{\varepsilon} \ll \frac{1}{\varepsilon^2}$ can be further widened. In particular, we will also show that a suitable "acceleration" of the gradient method improves the dependence of the complexity on ε from $\frac{1}{\varepsilon}$ to $\frac{1}{\sqrt{\varepsilon}}$.[17] This $\frac{1}{\sqrt{\varepsilon}}$ complexity is then unimprovable in general, providing a useful goalpost for us to aim for.

10.1.2 Beyond Linear: Clipped, Aggregated, Projected, Proximal, and Max-linear Models

The above examples illustrate the ideas underlying the class of *first-order methods* in optimization. Such methods build local models f_x not from second or higher derivatives of f, but from zeroth or first-order characteristics of f, for example, from its function values, gradients, and subgradients. Given such models, most first-order methods use them in algorithms similar and often identical to (10.1), with the quadratic penalty $\frac{\rho_k}{2}\|y - x_k\|^2$ playing a central role. Departing from the setting of gradient and subgradient methods, we will show in this section that we need not restrict ourselves to linear or even differentiable models. Because of this flexibility, we will be able to design local models that more closely approximate the geometry of f, while being "simple enough" to locally minimize. Key to developing this theory is to view (10.2) and (10.3) not as conditions for linear models to satisfy, but as definitions in their own right, opening the door to a more expressive class of models. In keeping with theory of first order methods, however, we will not explicitly discuss algorithms based on higher order derivatives, since the complexity theory of such methods requires different techniques and tools than developed here.

A goal in the analysis of first-order methods is to give *dimension-free* results, where the number of variables d does not appear in complexity bounds. Looking back on (10.4) and (10.5), we see that both results are dimension-free, and when we later prove complexity results for more general classes of models they too will be independent of d. Since such bounds do not depend on d, first-order methods are thought to be uniquely scalable to large-scale problems in high-dimensions. Still the "work" or the per-iteration complexity of implementing a first-order method must in some way scale with d, since even updating x_k to x_{k+1} in the subgradient method takes d arithmetic operations in general. For example, the per iteration costs $O(d)$ or $O(d \log d)$ are "good," while the per iteration cost $O(d^2)$ is "bad."

Moving beyond the classical linear models considered above, we will find it helpful to combine the class of linearly and quadratically accurate models. To that end, we call a proper closed convex function $f_x \colon \mathbb{R}^d \to (-\infty, +\infty]$ an *(l, q)-model of f at x* if

$$f_x(y) \le f(y) \le f_x(y) + l\|x - y\| + \frac{q}{2}\|x - y\|^2, \qquad \forall y \in \mathbb{R}^d. \tag{$\mathcal{M}$}$$

We caution the reader that this is not standard terminology. We introduce this terminology because it simplifies the statements and proofs of various results in this section. The reader

[17] The first such accelerated gradient method that achieved it was developed in a famous 1983 paper of Nesterov.

should keep in mind the important classes of examples: quadratically accurate and linearly accurate models.

To further ground this discussion, we isolate the core algorithmic operation used throughout this section, namely for a fixed $x \in \mathbb{R}^d$ and $\rho > 0$ choose a model f_x and define x_+ to be the minimizer of the following problem, closely mirroring (10.1):

$$x_+ = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ f_x(y) + \frac{\rho}{2}\|y - x\|^2 \right\}. \tag{10.6}$$

We will later show that x_+ is in fact the unique minimizer of this problem, but let us delay this for a moment. While we will soon take up the complexity theory of algorithms based on repeatedly solving problems of the form (10.6), for now we mention only that the complexity of such algorithms closely mirrors that of the gradient and the subgradient method, which was discussed at the end of Section 10.1.1. Despite matching the complexity, the crucial difference is that the subproblems (10.6) may be much harder to solve. Nevertheless, in some cases nonlinear models are available when accurate linear models fail to exist, for example, when f takes value $+\infty$, as we will soon see.

In what follows, we introduce a few common "structures" that aid in building models of f and then discuss algorithms that arise from these models.

Clipped and Aggregated Methods. Suppose that we can create (l, q) models for a function f at any point x. In seeking to tighten these models, there are two natural strategies.

The first strategy—called *clipping*—assumes that we have access to a global lower bound $\texttt{lb}$ on f: $f(y) \geq \texttt{lb}$ for all $y \in \mathbb{R}^d$. If we do have such a lower bound, then the function

$$y \mapsto \max\{f_x(y), \texttt{lb}\}$$

is itself an (l, q) model of f at x (see Figure 11).

The second strategy—called *aggregation*—iteratively constructs tighter and tighter approximations of f by aggregating (l, q) models f_{x_i} of f, centered at points x_i. For example, x_i may be the iterates generated by an algorithm. If we have a sequence of such (l, q) models $f_{x_1}, \ldots, f_{x_k}$, then

$$y \mapsto \max\{f_{x_1}(y), \ldots, f_{x_k}(y)\}$$

is an (l, q) model of f at the point x_k (see Figure 11).

The two claims above follow by alternatively setting $g \equiv \texttt{lb}$ and $g = \max_{i \leq k-1} f_{x_i}$ in the following proposition (you will prove this in the exercises).

Proposition 10.3 (Clipping/Aggregation). *Let $x \in \mathbb{R}^d$ and suppose that f_x is an (l, q) model of f at x. Moreover, assume that $g \colon \mathbb{R}^d \to (-\infty, \infty]$ is closed, proper, convex, and dominated by f: $g(y) \leq f(y)$ for all $y \in \mathbb{R}^d$. Then*

$$\max\{f_x, g\}$$

is an (l, q)-model of f at x.

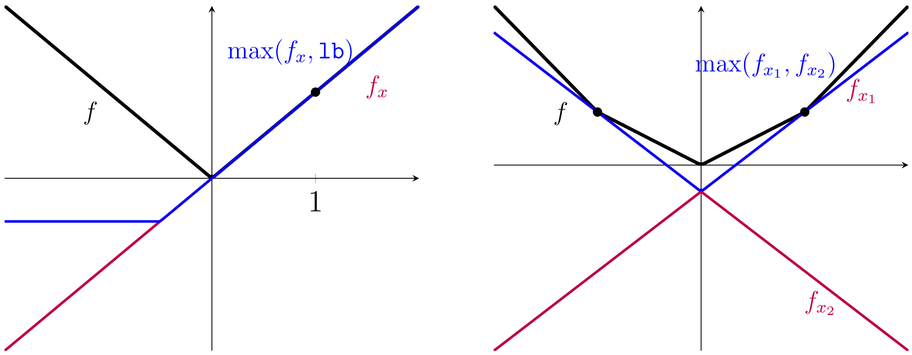


Figure 11: Clipped (left) and agreggated (right) models.

Turning to algorithms, the clipping strategy gives rise to the following method: given $x_0, \ldots, x_k$, choose an (l, q) model of f at x_k and define x_{k+1} by

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ \max\{f_{x_k}(y), \mathtt{lb}\} + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}, \tag{$\mathcal{CLIP}$}$$

While in general there is no closed form solution for x_k, if $f_{x_k}(y) = \langle a, y \rangle + b$ is an affine function, for example, a linear model arising from gradients or subgradients, then

$$x_{k+1} = x_k - \mathtt{clip}\left(\frac{\rho_k}{\|a\|^2}(\langle a, x_k \rangle + b - \mathtt{lb}) \right) \frac{a}{\rho_k} \qquad \text{where} \qquad \mathtt{clip}(t) = \max\{\min\{t, 1\}, 0\}. \tag{10.7}$$

(You will prove this in the exercises.) On the other hand, the aggregation strategy gives rise to the following algorithm: given $x_0, \ldots, x_k$, choose an (l, q) model of f at x_k and define x_{k+1} by

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ \max\{f_{x_1}(y), \ldots, f_{x_k}(y)\} + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}, \tag{$\mathcal{AGG}$}$$

for some $\rho_k > 0$. We caution the reader that each step of the algorithm becomes increasingly costly since the "max" function increases in complexity at each stage. In particular, there is no closed form formula for the steps of the algorithm.

Projected and Proximal Methods. Suppose that minimizing the function f amounts to minimizing a continuous convex function over a convex set. In other words, suppose that f admits a decomposition

$$f(y) = g(y) + \delta_{\mathcal{X}}(y), \qquad \forall y \in \mathcal{X},$$

where $g \colon \mathbb{R}^d \to (-\infty, \infty]$ is a continuous convex function and $\mathcal{X} \subseteq \mathbb{R}^d$ is a closed convex set. To construct a model for f, a natural strategy is to first approximate g with a model g_x and then form the sum $f_x := g_x + \delta_{\mathcal{X}}$.

For example, for every $x \in \mathbb{R}^d$ define the (sub)gradient model $g_x(y) := g(x) + \langle v_x, y - x \rangle$ for some $v_x \in \partial g(x)$. Then based on using the model $f_x := g_x + \delta_{\mathcal{X}}$ in (10.6), we derive the classical *projected subgradient method*: given $x_0, \ldots, x_k$ define x_{k+1} by

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ g_{x_k}(y) + \delta_{\mathcal{X}}(y) + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}, \qquad (\mathcal{PROJ})$$

for some $\rho_k > 0$. As a bit of algebra shows, every step of this algorithm admits a "closed form" solution (check!)

$$x_{k+1} = \operatorname{proj}_{\mathcal{X}}\left(x_k - \frac{1}{\rho_k} v_{x_k}\right).$$

The formula is of course not truly "closed form" unless projecting onto $\mathcal{X}$ is "simple enough." Nevertheless for a variety of important convex sets, there is a closed form solution to the projection. To support this claim, we will give a few examples in the exercises.

The following proposition shows that f_x is indeed a valid approximation (you will prove this in the exercises).

Proposition 10.4 (Projection/Proximal Models)**.** *Suppose that f admits the decomposition*

$$f = g + h,$$

where $g, h : \mathbb{R}^d \to (-\infty, \infty]$ are closed, proper, convex functions. Let $x \in \mathbb{R}^d$ and suppose that g_x is an (l, q) model of g at x. Then

$$g_x + h$$

is an (l, q)-model of f at x.

Let us look at another classical algorithm. As before, for every $x \in \mathbb{R}^d$ define the subgradient model $g_x(y) := g(x) + \langle v_x, y - x\rangle$ for some $v_x \in \partial g(x)$. Then based on using the approximation $f_x := g_x + h$, we derive the classical *proximal subgradient algorithm*: given $x_0, \ldots, x_k$, define x_{k+1} by

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ g_{x_k}(y) + h(y) + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}, \qquad (\mathcal{PROX})$$

for some $\rho_k > 0$. Again a bit of algebra shows every step of this algorithm admits a "closed form" solution (check!)

$$x_{k+1} = \operatorname{prox}_{(1/\rho_k)h}\left(x_k - \frac{1}{\rho_k} v_{x_k}\right),$$

where for any $\gamma > 0$, we define the *proximal operator* of h with parameter γ:

$$\operatorname{prox}_{\gamma h}(z) := \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ h(y) + \frac{1}{2\gamma}\|y - z\|^2 \right\}, \qquad \forall z \in \mathbb{R}^d.$$

As was the case for the projected subgradient method, the formula is not truly "closed form" unless computing the proximal operator is "simple enough." Nevertheless for a variety of important convex functions h, there is a closed form solution to the proximal operation. To support this claim, we will give a few examples in the exercises.

Max-linear Models. Suppose that we wish to optimize a convex function f, given in the explicit form

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

where $f_1 \colon \mathbb{R}^d \to \mathbb{R}$ and $f_2 \colon \mathbb{R}^d \to \mathbb{R}$ are convex and differentiable. Let us consider two natural models for f:

1. $f_x(y) = f(x) + \langle v_x, y - x\rangle$ for any $v_x \in \partial f(x)$.

This is the standard linear model built from subgradients. Notice that this model treats f as a "black box," ignoring the structure of f_1 and f_2. In contrast, the following alternative *max-linear* model

2. $f_x(y) = \max\{f_1(x) + \langle \nabla f_1(x), y - x\rangle, f_2(x) + \langle \nabla f_2(x), y - x\rangle\}$,

is nonlinear and nondifferentiable, but more closely approximates the geometry of f. For example, Figure 12 shows both models for the function $f(x) = \{(x+1)^2, (x-1)^2\}$, and verifies this claim. The figure also shows that the quadratic penalization strategy in (10.6) will perform better on the second model.

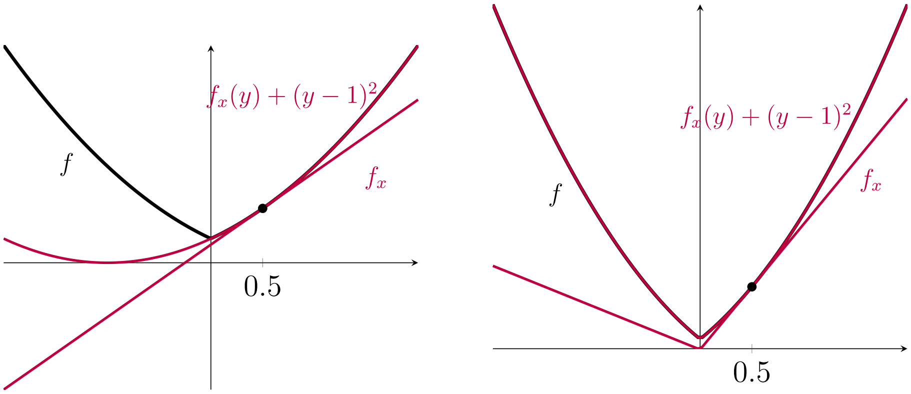


Figure 12: Linear model (left) and sublinear prox-linear model (right).

More generally, the following result holds (you will prove this in the exercises).

Proposition 10.5 (Max-Linear Models)**.** *Suppose that f admits the decomposition*

$$f = \max(f_1, \ldots, f_n),$$

where for each i, the function $f_i : \mathbb{R}^d \to (-\infty, \infty]$ is closed, proper, and convex. Let $x \in \mathbb{R}^d$ and suppose for each i, the function $(f_i)_x$ is an (l, q) model of f_i at x. Then

$$\max\{(f_1)_x, \ldots, (f_n)_x\}$$

is an (l, q)-model of f at x.

Turning to algorithms, the max-linear strategy gives rise to the following method (sometimes called *prox-linear* or *Gauss-Newton*): given $x_0, \ldots, x_k$, choose (l, q) models of $(f_1)_{x_k}$, $\ldots$, $(f_n)_{x_k}$ of $f_1, \ldots, f_n$ at x_k, and define x_{k+1} by

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ \max\{(f_1)_{x_k}(y), \ldots, (f_n)_{x_k}(y)\} + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}. \qquad (\mathcal{MAXL})$$

We caution the reader that each step of this algorithm might be as hard to solve as minimizing f itself (for example, consider again the example $\max\{(1+x)^2, (1-x)^2\}$). Nevertheless, when it is possible to implement this strategy, one should prefer it over the naive subgradient method. Indeed, in general the linear model built from subgradients is at best linearly accurate. On the other hand, if one knows the f_i have simple quadratically accurate models, then the max-linear model is quadratically accurate, resulting in improved complexity rates, as we will soon see.

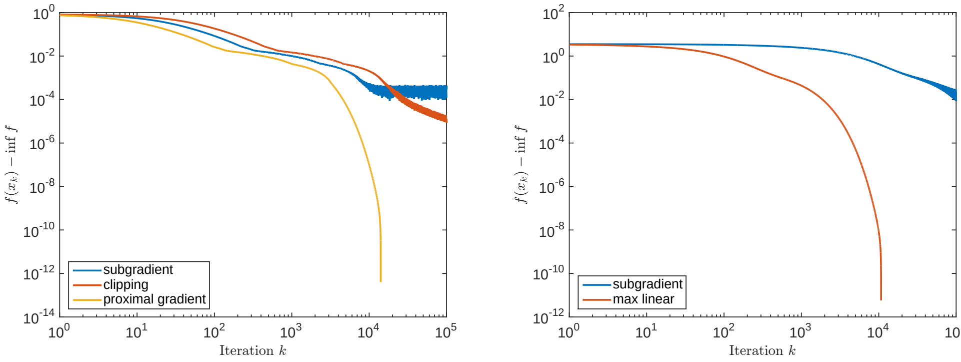
10.1.3 Two Small Examples

We now illustrate that a close look at problem structure can lead to algorithms that perform much better than those built from classical linear models. We illustrate this claim on two problems: the first, called *least absolute shrinkage and selection operator* (LASSO), is a common tool in statistical regression; the second is the minimization of the maximum of two quadratics, which serves as a prototypical example of a nonsmooth, non polyhedral convex function.

In Figure 13a, we compare the performance of the subgradient, clipped subgradient, and proximal gradient methods for the following LASSO problem:

$$\text{minimize}_{x\in\mathbb{R}^d}\ f(x) := \frac{1}{2}\|Ax-b\|^2 + \lambda\|x\|_1.$$

where $d = 20$, $m = 10$, $A \in \mathbb{R}^{m\times d}$, $b \in \mathbb{R}^m$, and λ is a small constant. The updates of the three algorithms are summarized below:



(a) LASSO

(b) Maximum of Quadratics

Figure 13: Convergence Plots for the Problems in Section 10.1.3

1. **Subgradient.** For every iterate x_k, we compute a subgradient[18] $v_{x_k} = A^T(Ax_k - b) + \lambda\text{sign}(x_k) \in \partial f(x_k)$ (check!) and then set

$$x_{k+1} = x_k - \frac{1}{\rho_k} v_{x_k}.$$

[18] Here, we let $\text{sign}(x) = (\text{sign}(x_1), \ldots, \text{sign}(x_d))$, with the convention that $\text{sign}(0) = 0$.

2. **Clipped.** For every iterate x_k, we compute a subgradient $v_{x_k} = A^T(Ax_k - b) + \lambda \text{sign}(x_k) \in \partial f(x_k)$ and then set

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ \max\{f(x_k) + \langle v_{x_k}, y - x_k\rangle, \inf f\} + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}$$

3. **Proximal Gradient.** For every iterate x_k, we compute a *gradient* $w_{x_k} = A^T(Ax - b)$ of the smooth function $g(x) = \frac{1}{2}\|Ax - b\|^2$ and then set

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ g(x_k) + \langle w_{x_k}, y - x_k\rangle + \lambda\|y\|_1 + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}.$$

We do not specify the control parameters ρ_k, but let us mention that they were tuned appropriately. At first glance, the subproblems may seem to be increasing in complexity. This is actually not the case since all have equally simple, closed form solutions. Indeed, the solution to the clipped problem may be gleaned from (10.7). Likewise, we will show in the exercises that the proximal gradient step for the ℓ_1 norm also has a simple closed form formula. Looking at Figure 13a, we conclude that "better models" lead to "faster algorithms," at least in this example.

Let us consider a second example. In Figure 13b, we compare the performance of subgradient and max-linear models for the following maximum of quadratics problem:

$$\text{minimize}_{x \in \mathbb{R}^d}\ f(x) := \max\left\{ \frac{1}{2}\langle A_1 x, x\rangle + \langle b_1, x\rangle, \frac{1}{2}\langle A_2 x, x\rangle + \langle b_2, x\rangle \right\}$$

where $d = 20$, $A_1, A_2 \in \mathbb{R}^{d\times d}$, and $b_1, b_2 \in \mathbb{R}^d$. The updates of the two algorithms are summarized below:

1. **Subgradient.** For every iterate x_k, we compute a subgradient $v_{x_k} = A^T(Ax_k - b) + \lambda \text{sign}(x_k) \in \partial f(x_k)$ (check!) and then set

$$x_{k+1} = x_k - \frac{1}{\rho_k} v_{x_k}.$$

2. **Max-linear.** For every iterate x_k, we compute a subgradient $\nabla f_i(x_k) = A_i x_k - b_i$ and of the quadratics $f_i(x) = \frac{1}{2}\langle A_i x, x\rangle + \langle b_i, x\rangle$ and then set.

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ \max\{f_1(x_k) + \langle \nabla f_1(x_k), y - x_k\rangle, f_2(x_k) + \langle \nabla f_2(x_k), y - x_k\rangle\} + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}.$$

Again, we mention that ρ_k was tuned appropriately. In addition, both algorithms have roughly the same computational cost, since the max-linear algorithm may be implemented with roughly three times the computational effort of a single subgradient update. Looking at Figure 13b, we again conclude that "better models" can lead to "faster algorithms."

Having developed a rich class of models we now turn our attention to a rigorous study of algorithms. In the next section, we look first at a first algorithm, based on repeatedly minimizing (10.6).

10.2 A First Algorithm

Returning to algorithms, we follow the path outlined in Section 10.1 and repeatedly minimize quadratically perturbed models of f. We call this a *First-Order Model Based Algorithm.* Concretely, we initialize the algorithm at a point $x_0 \in \mathrm{dom}(f)$ and then define a sequence of iterates $x_0, x_1, \ldots$ inductively: given iterates $x_0, \ldots, x_k$, we form an (l, q) model f_{x_k} of f at x_k and minimize

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ f_{x_k}(y) + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}, \qquad (\mathcal{MBA})$$

where the sequence of scalars $\rho_0, \rho_1, \ldots$ are again called *control parameters.*

For now we impose minimal assumptions on parameters and models. We only ask the control parameters to be strictly positive, which ensures the iterates exist (as we will see). As for the models f_{x_k}, they are indexed by x_k, but they may depend arbitrarily on the past iterates $x_0, \ldots, x_k$. And although we could allow their parameters to vary, we avoid this complication and instead ask each f_{x_k} to be an (l, q) approximation for fixed $l \geq 0$ and $q \geq 0$.

> **Disclaimer:** Below we will only provide estimates on algorithm speed in the two important cases of quadratically accurate models, where $l = 0$, and linearly accurate models, where $q = 0$. While we could now analyze $(\mathcal{MBA})$ for any pair (l, q), the "correct" control parameters and the rates of convergence change dramatically, depending on whether $l = 0$ or $l > 0$. Thus we only focus on these two classical cases.

Given this algorithm, a few natural questions arise:

1. **(Examples.)** How do we choose the models f_{x_k}?
2. **(Complexity.)** How quickly does $(\mathcal{MBA})$ approach a minimizer of f?
3. **(Accelerations.)** Once we understand its complexity, can we improve upon $(\mathcal{MBA})$?

In section 10.1, we looked at several classes of models and seen that $(\mathcal{MBA})$ recovers classical algorithms such as the gradient and subgradient method. Thus in this section, we beginning with complexity. Afterwards, we will improve upon $(\mathcal{MBA})$ in certain special cases.

10.2.1 Terminology: Iteration Complexity and Rates of Convergence

How quickly does x_k approach a minimizer of f? To answer this question we must first decide how to measure algorithm progress. Two common measures are the objective error and the distance to minimizers:

1. **(The Objective Error.)** $f(x_k) - \inf f$;
2. **(Distance to Minimizers.[19])** $\mathrm{dist}(x_k, \mathcal{X}^*)$.

[19] For a set $\mathcal{X} \subseteq \mathbb{R}^d$ and a point $x \in \mathbb{R}^d$, we define the *distance to* $\mathcal{X}$: $\mathrm{dist}(x, \mathcal{X}) := \inf_{y \in \mathcal{X}} \|y - x\|$.

Both quantities are nonnegative and take the value zero if and only if x_k is a minimizer. The distance may be expressed more simply with a projection operator: since $\mathcal{X}^* := \operatorname{argmin} f$ is closed and convex (check!) it holds $\operatorname{dist}(x, \mathcal{X}^*) = \|x - \operatorname{proj}_{\mathcal{X}^*}(x)\|$. Although both measures are evaluated at an iterate x_k, we will sometimes evaluate them at auxiliary, but still computable points $\hat{x}_k$ created from the past iterates $x_0, \ldots, x_k$. For example, we may set $\hat{x}_k$ to be an average of $x_0, \ldots, x_k$.

There are two common ways to discuss the "speed" of an algorithm. When measuring the *convergence rate* of the algorithm, we seek to bound either the objective error or the distance to minimizers by a decreasing function of k. For example, if the function is geometrically decreasing in k, we say the algorithm *converges linearly*, and if the function decreases polynomially in k, we say the algorithm *converges sublinearly.* More concretely, the two functions fit either form

1. **(Linear.)** cr^k for some $r \in (0, 1)$;

2. **(Sublinear.)** c/k^p for some $p > 0$,

for some $c > 0$. Equivalent to measuring the convergence rate of an algorithm is measuring its complexity: fixing a measure of progress and a target "accuracy" $\varepsilon > 0$, the *iteration complexity* provides a function $K(\varepsilon)$ so that the measure is less than ε when $k \geq K(\varepsilon)$. Clearly, linearly convergent algorithms have complexity bounded by $\log(c/\varepsilon)/\log(1/r)$, while a sublinearly convergent algorithm has complexity bounded by $(c/\varepsilon)^{1/p}$. Both the iteration complexity and the convergence rate count the number of times we minimize a model of f, but they ignore the cost of each such minimization.

Going forward we will always be able to measure the convergence rate or iteration complexity of the objective error. In contrast, it is only possible to measure the distance to the minimizer if f behaves well. For example, we will be able to measure $\operatorname{dist}(x_k, \mathcal{X}^*)$ if f grows rapidly away from minimizers:

$$\mu \cdot \operatorname{dist}(x, \mathcal{X}^*)^p \leq f(x) - \inf f, \qquad \forall x \in \mathbb{R}^d, \tag{10.8}$$

for some $p > 0$ and $\mu > 0$. In the exercises, you will look at the case where $p = 2$.

10.2.2 The Effect of Solving the Quadratically Penalized Subproblem

To understand quadratic penalization, we introduce the class of *α-strongly convex* functions: a proper function $h\colon \mathbb{R}^d \to (-\infty, \infty]$ is called α-strongly convex if $x \mapsto h(x) - \frac{\alpha}{2}\|x\|^2$ is convex. We need two properties of strongly convex functions: first they have unique minimizers and second they grow quadratically away from them. We prove this in the following lemma.

Lemma 10.6 (Strong Convexity and Quadratic Growth)**.** *Let $\alpha > 0$ and suppose that h is a closed, proper, and α-strongly convex. Then h has a unique minimizer $\bar{x}$ and*

$$\frac{\alpha}{2}\|y - \bar{x}\|^2 + h(\bar{x}) \leq h(y), \qquad \forall y \in \mathbb{R}^d.$$

Proof. Beginning with existence, we show h has a minimizer. This will follow from Weierstrauss' Theorem if h has bounded sublevel sets (Exercise 2.5). To show this, we will prove that a quadratic function of the form $c_0 + \langle v_0, x\rangle + \frac{\rho}{2}\|x\|^2$ globally lower bounds h, where the affine part comes from a linear under approximation of the proper, closed, convex function $g(\cdot) := (h(\cdot) - \frac{\rho}{2}\|\cdot\|^2)$. More precisely, since g is proper, closed, and convex, there is a point $x_0 \in \mathrm{dom}(g)$ with $\partial g(x_0) \neq \emptyset$ (Exercise 9.11). Choosing any $v \in \partial g(x_0)$ and any $a \in \mathbb{R}$, we thus have

$$\{x\colon h(x) \leq a\} = \left\{x\colon g(x) + \frac{\rho}{2}\|x\|^2 \leq a\right\} \subseteq \left\{x\colon g(x_0) + \langle x - x_0, v\rangle + \frac{\rho}{2}\|x\|^2 \leq a\right\}.$$

The right hand side is bounded, implying h has bounded sublevel sets.

Turning to quadratic growth, let $\bar{x}$ be a minimizer of h. Then by Fermat's rule and the sum rule

$$0 \in \partial h(\bar{x}) = \partial g(\bar{x}) + \rho\bar{x} \implies -\rho\bar{x} \in \partial g(\bar{x}).$$

Thus, by definition

$$\langle -\rho\bar{x}, y - \bar{x}\rangle + h(\bar{x}) - \frac{\rho}{2}\|\bar{x}\|^2 \leq h(y) - \frac{\rho}{2}\|y\|^2, \qquad \forall y \in \mathbb{R}^d.$$

Completing the square yields quadratic growth and shows $\bar{x}$ uniquely minimizes h. □

For us, the main consequence of this lemma is the next result, showing how $f(x_k)$ evolves in the $(\mathcal{MBA})$ algorithm. It relies on the following observation: h is α-strongly convex if and only if $y \mapsto h(y) - \frac{\alpha}{2}\|y - x\|^2$ is convex for all $x \in \mathbb{R}^d$ (check!). As we will later develop a second algorithm, one also based on repeatedly solving the quadratic subproblem (10.6), we answer this question more generally in the following Lemma.

Lemma 10.7 (Basic Lemma of (l, q) models)**.** *Fix $x \in \mathbb{R}^d$ and suppose that f_x is an (l, q) model of f at x. Let x_+ be the minimize of the following quadratic subproblem:*

$$x_+ = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{f_x(y) + \frac{\rho}{2}\|y - x\|^2\right\}.$$

Then the following bound holds:

$$\begin{aligned}\left\{f(x_+) + \frac{\rho}{2}\|y - x_+\|^2\right\} &+ \frac{\rho - q}{2}\|x_+ - x\|^2 - l\|x_+ - x\| \\ &\leq \left\{f(y) + \frac{\rho}{2}\|y - x\|^2\right\}, \qquad \forall y \in \mathbb{R}^d. \end{aligned} \tag{10.9}$$

Proof. Applying Lemma 10.6 to to $f_x + \frac{\rho}{2}\|y - x\|^2$, we find that

$$\begin{aligned}\frac{\rho}{2}\|x_+ - y\|^2 &+ \left\{f_x(x_+) + \frac{\rho}{2}\|x_+ - x\|^2\right\} \\ &\leq \left\{f_x(y) + \frac{\rho}{2}\|y - x\|^2\right\} \\ &\leq f(y) + \frac{\rho}{2}\|y - x\|^2,\end{aligned}$$

where the third inequality follows since f_x is an (l, q) model. To complete the proof, use the (l, q) property again to deduce $f_x(x_+) \geq f(x_+) - l\|x_+ - x\| - \frac{q}{2}\|x_+ - x\|^2$. □

Viewing x_+ as one "step" of an algorithm, the this lemma reveals algorithm progress toward an arbitrary point $y \in \mathbb{R}^d$. For example setting $\rho \geq q$, $l = 0$, and $y = x$, we find $f(x_+) \leq f(x)$, a useful fact in the next session.

10.2.3 Quadratically Accurate Models and Gradient Descent

Applying Lemma 10.7 to the iterates of ($\mathcal{MBA}$), we derive the following rate of convergence. In the proof, we will see that $f(x_1), f(x_2), \ldots$ is a decreasing sequence, showing that the iterates make monotonic progress to the minimizers.

Theorem 10.8 (Convergence Rate of ($\mathcal{MBA}$) Quadratically Accurate Models)**.** *Suppose that each f_{x_k} is a $(0,q)$ model of f at x_k for all $k \geq 0$. Let $\rho \geq q$ and suppose that $\rho_k = \rho$ for all $k \geq 0$. Then*

$$f(x_K) - \inf f \leq \frac{\rho \cdot \operatorname{dist}^2(x_0, \mathcal{X}^*)}{2K}, \qquad \forall K \geq 1.$$

Proof. Beginning with the sublinear case, we show $f(x_k) - \inf f$ is nonincreasing in k. To show this, we plug $y = x_k$ and $x = x_k$ in (10.9) and find $f(x_{k+1}) \leq f(x_k)$ for all $k \geq 0$, as desired. Combining with the telescoping upper bound

$$f(x_k) - \inf f \leq \frac{\rho}{2}\|x_{k-1} - y\|^2 - \frac{\rho}{2}\|x_k - y\|^2, \qquad \forall k \geq 1, \forall y \in \mathcal{X}^*,$$

given by (10.9), we see that

$$\begin{aligned} f(x_K) - \inf f \leq \frac{1}{K}\sum_{k=1}^{K}(f(x_k) - \inf f) &\leq \frac{1}{K}\sum_{k=1}^{K}\left(\frac{\rho}{2}\|x_{k-1} - y\|^2 - \frac{\rho}{2}\|x_k - y\|^2\right) \\ &\leq \rho \cdot \frac{\|x_0 - y\|^2}{2K}. \end{aligned}$$

Letting y be the projection of x_0 onto $\mathcal{X}^*$, the numerator becomes $\operatorname{dist}^2(y, \mathcal{X}^*)$, as desired. □

Returning to the classical example of the gradient descent algorithm, we have the following direct corollary of Theorem 10.8.

Corollary 10.9 (Convergence Rates of Gradient Descent)**.** *Suppose that f is differentiable and that ∇f is q-Lipschitz for some $q > 0$, meaning*

$$\|\nabla f(x) - \nabla f(y)\| \leq q\|x - y\|, \qquad x, y \in \mathbb{R}^d.$$

Let $\rho \geq q$ and Let $x_0, x_1, \ldots$ be the iterates of the gradient descent algorithm ($\mathcal{GD}$) with $\rho_k = \rho$ for all $k \geq 0$. Then

$$f(x_K) - \inf f \leq \frac{\rho \cdot \operatorname{dist}^2(x_0, \mathcal{X}^*)}{2K}, \qquad \forall K \geq 1.$$

Upper bounding the convergence rate by ε, we recover the classical complexity of the gradient descent algorithm, as claimed in Equation (10.4):

$$\frac{\rho \cdot \operatorname{dist}^2(x_0, \mathcal{X}^*)}{2K} \leq \varepsilon \iff K \geq \frac{\rho \cdot \operatorname{dist}^2(x_0, \mathcal{X}^*)}{2\varepsilon}.$$

We easily derive the same rate of convergence for the aggregated, projected gradient, proximal gradient, and max-linear algorithms as outlined in Section 10.1.2. As an example, we state the convergence rate of the proximal gradient method.

Corollary 10.10 (Convergence Rates of Proximal Gradient Descent). *Suppose that f admits the decomposition*

$$f = g + h,$$

where $g, h : \mathbb{R}^d \to (-\infty, \infty]$ are closed, proper, convex functions. In addition, suppose that g is differentiable and that ∇g is q-Lipschitz for some $q > 0$, meaning

$$\|\nabla g(x) - \nabla g(y)\| \leq q\|x - y\|, \qquad x, y \in \mathbb{R}^d.$$

Let $\rho \geq q$ and Let $x_0, x_1, \ldots$ be the iterates of the proximal gradient algorithm:

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ g(x_k) + \langle \nabla g(x_k), y - x_k \rangle + h(y) + \frac{\rho}{2}\|y - x_k\|^2 \right\}, \qquad \forall k \geq 0.$$

Then

$$f(x_K) - \inf f \leq \frac{\rho \cdot \operatorname{dist}^2(x_0, \mathcal{X}^*)}{2K}, \qquad \forall K \geq 1.$$

10.2.4 Linearly Accurate Models and the Subgradient Method

Moving to the case of linearly accurate models ($l > 0$), we will see a change not only in the rates of convergence, but also in the control parameters ρ_k. For example, we will find both that ($\mathcal{MBA}$) has worse rates and that to achieve these rates its control parameters must tend to infinity. To illustrate, we make the simplifying assumption that $q = 0$. In this setting, we may refine Lemma 10.7: Fixing a $k \geq 1$ and letting $\delta_k = \|x_k - x_{k-1}\|$, we have

$$f(x_k) - f(y) \leq \frac{\rho_{k-1}}{2}\|x_{k-1} - y\|^2 - \frac{\rho_{k-1}}{2}\|x_k - y\|^2 + l\delta_k - \frac{\rho_k}{2}\delta_k^2, \qquad \forall y \in \mathbb{R}^d$$

Noticing that $h(\delta) = l\delta - (\rho_k/2)\delta^2$ is a concave quadratic in δ, we may find its maximizer from first order optimality conditions: $h'(\delta) = 0$ if and only if $\delta = l/\rho_k$. Then from $h(\delta_k) \leq h(\delta) = \frac{l^2}{2\rho_k}$, we arrive at the following critical inequality:

$$f(x_k) - f(y) \leq \frac{\rho_{k-1}}{2}\|x_{k-1} - y\|^2 - \frac{\rho_{k-1}}{2}\|x_k - y\|^2 + \frac{l^2}{2\rho_{k-1}}, \qquad \forall y \in \mathbb{R}^d. \tag{10.10}$$

Plugging $y = x_{k-1}$ into this inequality, we see that

$$f(x_k) - f(x_{k-1}) \leq \frac{l^2}{2\rho_{k-1}} - \frac{\rho_{k-1}}{2}\|x_k - x_{k-1}\|^2,$$

showing that $f(x_k)$ does not necessarily decrease. Moreover, if $l^2/(2\rho_{k-1})$ dominates $((\rho_{k-1} + \alpha)/2)\|x_k - x_{k-1}\|^2$, the reason for increasing ρ_{k-1} becomes clear: if ρ_{k-1} is fixed, the iterates may oscillate.[20]

Beyond a change in rates and control parameters, we must also change what ($\mathcal{MBA}$) returns as an approximate minimizer. The reason is that $f(x_k)$ may increase over time, so last iterate x_K may not be the "best" one. In place of the last iterate we may instead return an average of the iterates or an iterate with minimal objective error. In what follows, we will return averages rather than the "best" iterate. To that end, we will need the following easy consequence of convexity (check!).

[20] In general, one expects $\|x_k - x_{k-1}\|$ to be less than or perhaps even substantially less than l/ρ_{k-1}.

Lemma 10.11 (Jensen's Inequality)**.** *Fix $K \geq 1$ and let $f\colon \mathbb{R}^d \to (-\infty, \infty]$ be a proper convex function. Suppose that $x_1, \ldots, x_K \in \operatorname{dom}(f)$ and $\lambda_1, \ldots, \lambda_n > 0$. Then*

$$f\left(\frac{1}{\sum_{i=1}^K \lambda_i} \sum_{i=1}^K \lambda_i x_i\right) \leq \frac{1}{\sum_{i=1}^K \lambda_i} \sum_{i=1}^K \lambda_i f(x_i).$$

With this Lemma and inequality 10.10 in hand, we are ready to derive the convergent rate of $(\mathcal{MBA})$ for $(l, 0)$ models. In what follows, we refer to the sequence of weighted averages

$$\overline{x}_K := \frac{1}{\sum_{k=1}^K \rho_{k-1}^{-1}} \sum_{k=1}^K \rho_{k-1}^{-1} x_k, \qquad \forall K \geq 1.$$

It is these points for which we obtain a rate of convergence.

Theorem 10.12 (Convergence Rate of $(\mathcal{MBA})$ with Linearly Accurate Models)**.** *Suppose that each f_{x_k} is an $(l, 0)$ model of f at x_k for all $k \geq 0$. Then*

$$f\left(\overline{x}_K\right) - \inf f \leq \frac{\operatorname{dist}^2(x_0, \operatorname{argmin} f) + l^2 \sum_{k=1}^K \rho_{k-1}^{-2}}{2\sum_{k=1}^K \rho_{k-1}^{-1}}, \qquad \forall K \geq 1. \tag{10.11}$$

Proof. Letting y be the projection of x_0 onto $\operatorname{argmin} f$ in (10.10), we find

$$f(x_k) - \inf f \leq \frac{\rho_{k-1}}{2}\|x_{k-1} - y\|^2 - \frac{\rho_{k-1}}{2}\|x_k - y\|^2 + \frac{l^2}{2\rho_{k-1}}.$$

Multiplying both sides by ρ_k^{-1} and summing from $k = 1$ to $k = K$, we have

$$\begin{aligned}\sum_{K=1}^k \rho_{k-1}^{-1}\left(f(x_k) - \inf f\right) &\leq \sum_{k=1}^K \left(\frac{1}{2}\|x_{k-1} - y\|^2 - \frac{1}{2}\|x_k - y\|^2 + \frac{l^2}{2\rho_k^2}\right) \\ &\leq \frac{1}{2}\|x_0 - y\|^2 + \frac{l^2}{2}\sum_{k=1}^K \rho_{k-1}^{-2}\end{aligned}$$

Dividing both sides by $\sum_{k=1}^K \rho_k^{-1}$ and applying Jensen's inequality to f, we find

$$f(\overline{x}_K) - \inf f \leq \frac{\|x_0 - y\|^2 + l^2 \sum_{k=1}^K \rho_{k-1}^{-2}}{2\sum_{k=1}^K \rho_{k-1}^{-1}}.$$

To complete the proof, note that $\operatorname{dist}(\overline{x}_k, \mathcal{X}^*) \leq \|\overline{x}_k - y\|$ and $\operatorname{dist}(x_0, \mathcal{X}^*) = \|x_0 - y\|$. □

The theorem allows for flexibility in the choice of control parameter ρ_k. There are two classical families, common in the literature:

1. **(Square Summable, but not Summable.)** Suppose that

$$\sum_{k=1}^{\infty} \rho_k^{-2} < +\infty \qquad \text{and} \qquad \sum_{k=1}^{\infty} \rho_k^{-1} = +\infty.$$

   Looking at (10.11), we see this choice guarantees $f(\bar{x}_K) \to f^*$ as $K \to \infty$. Common practical choices include

$$\rho_k = c_1 k^{\frac{1}{2}+c_2}$$

   for some positive c_1 and c_2. These stepsizes must be heavily "tuned" to achieve "good" performance.

2. **(Fixed Time Horizon).** When we only wish to run the algorithm for a fixed number of steps K, we can choose a constant stepsize $\rho_k = C\sqrt{K}$ with $C > 0$. With this choice, we get the bound

$$f(\overline{x}_K) - \inf f \leq \frac{C^2 \mathrm{dist}^2(x_0, \mathcal{X}^*) + l^2}{2C\sqrt{K}},$$

   yielding a convergence rate on the order of $1/\sqrt{K}$.

Returning to the classical example of the subgradient method, we have the following direct corollary of Theorem 10.8.

Corollary 10.13 (Convergence Rates of Subgradient Method)**.** *Suppose that f is $\hat{l}$-Lipschitz continuous, meaning*

$$|f(x) - f(y)| \leq \hat{l}\|x - y\|$$

Let $x_0, x_1, \ldots$ be the iterates of the subgradient method ($\mathcal{SM}$). Then

$$f(\overline{x}_K) - \inf f \leq \frac{\mathrm{dist}^2(x_0, \mathcal{X}^*) + 4\hat{l}^2 \sum_{k=1}^{K} \rho_{k-1}^{-2}}{2\sum_{k=1}^{K} \rho_{k-1}^{-1}}, \qquad \forall K \geq 1.$$

Fixing a $K \geq 0$ and choosing the fixed time horizon stepsize

$$\rho_k \equiv C\sqrt{K} \qquad \text{with } C := \frac{2\hat{l}}{\mathrm{dist}(x_0, \mathcal{X}^*)},$$

yields the convergence rate

$$f(\overline{x}_K) - \inf f \leq 2 \cdot \frac{\mathrm{dist}(x_0, \mathcal{X}^*)\hat{l}}{\sqrt{K}}.$$

Upper bounding the convergence rate by ε, we recover the classical complexity of the subgradient method, as claimed in Equation (10.5):

$$2 \cdot \frac{\mathrm{dist}(x_0, \mathcal{X}^*)\hat{l}}{\sqrt{K}} \leq \varepsilon \iff K \geq \frac{4\hat{l}^2 \cdot \mathrm{dist}^2(x_0, \mathcal{X}^*)}{\varepsilon^2}.$$

We easily derive the same rate of convergence for the aggregated, projected subgradient, proximal subgradient, and max-linear algorithms as outlined in Section 10.1.2. As an example, we state the convergence rate of the proximal subgradient method.

Corollary 10.14 (Convergence Rates of Proximal Subgradient)**.** *Suppose that f admits the decomposition*

$$f = g + h,$$

where $g, h : \mathbb{R}^d \to (-\infty, \infty]$ are closed, proper, convex functions. In addition, suppose that g is $\hat{l}$-Lipschitz continuous, meaning

$$|g(x) - g(y)| \leq \hat{l}\|x - y\|, \qquad x, y \in \mathbb{R}^d.$$

Let $x_0, x_1, \ldots$ be the iterates of the proximal subgradient method:

$$\begin{aligned} &\textit{Choose: } v_k \in \partial g(x_k) \\ &\textit{Set: } x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ g(x_k) + \langle v_k, y - x_k \rangle + h(y) + \frac{\rho}{2}\|y - x_k\|^2 \right\}, \qquad \forall k \geq 0. \end{aligned}$$

Then

$$f\left(\overline{x}_K\right) - \inf f \leq \frac{\operatorname{dist}^2(x_0, \mathcal{X}^*) + 4\hat{l}^2 \sum_{k=1}^K \rho_{k-1}^{-2}}{2\sum_{k=1}^K \rho_{k-1}^{-1}}, \qquad \forall K \geq 1.$$

10.3 An Acceleration for Quadratically Accurate Models

In our study of the model-based algorithm ($\mathcal{MBA}$), we have found a sharp division in complexity, one that favors quadratic accuracy over linear accuracy. Later we will look closely at linearly accurate models and show that the performance of ($\mathcal{MBA}$) in some sense matches the best possible complexity for any algorithm based on minimizing such models. On the other hand, for quadratically accurate models, there is still much room for improvement, as Yuri Nesterov found in 1983 with his so called *accelerated gradient method [?]*.

To illustrate, let us consider the special case where f is Fréchet differentiable and ∇f is q-Lipschitz. Then for a certain sequence of parameters $\gamma_k \geq 0$, Nesterov's accelerated method is given by the following recursion:

$$\begin{aligned} x_{k+1} &= y_k - \frac{1}{q}\nabla f(y_k) \\ y_{k+1} &= x_{k+1} + \gamma_k(x_{k+1} - x_k). \end{aligned}$$

Recalling the classical gradient method ($\mathcal{GD}$), we see the primary difference is the introduction of a new sequence of iterates y_k. Nesterov's algorithm is interpreted as a gradient step with a bit of "momentum" added, a term reserved for the differences $\gamma_k(y_{k+1} - y_k)$.

Nesterov's result states that the complexity of the accelerated gradient method (with properly chosen γ_k) is bounded by

$$K(\varepsilon) \leq \left\lceil \operatorname{dist}(x_0, \mathcal{X}^*)\sqrt{\frac{2q}{\varepsilon}} \right\rceil.$$

Comparing to the complexity bound of gradient descent in (10.4), we see a huge improvement in terms of dependence on ε. Moreover, in terms of any "reasonable" algorithm based on first-order information about f, this complexity is optimal (as shown by Nemirovski and Yudin [**?**]).

The goal of this section is to give a similar improvement for the entire class of quadratically accurate models. To do so, we introduce an "accelerated" version of ($\mathcal{MBA}$), one that was first proposed and analyzed in an influential 2008 paper of Paul Tseng [**?**]. Concretely, we initialize the algorithm at a pair of points $x_0, z_0 \in \operatorname{dom}(f)$ and then define two sequences of iterates: given iterates $x_0, \ldots, x_k$ and $z_0, \ldots, z_k$, we form a q-quadratically accurate model f_{x_k} of f at x_k and set

$$\begin{aligned} \theta_k &= \frac{2}{k+1}; \\ y_k &= (1-\theta_k)x_k + \theta_k z_k; \\ z_{k+1} &= \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ f_{y_k}(y) + \frac{\theta_k q}{2}\|y - z_k\|^2 \right\}; \\ x_{k+1} &= (1-\theta_k)x_k + \theta_k z_{k+1}. \end{aligned} \tag{$\mathcal{AMBA}$}$$

If you do not have intuition for this algorithm, you are not alone. As of late, a focus of much research has been to provide an intuitive explanation of accelerated algorithms. The interested reader should perform a quick internet search on "intuition for accelerated gradient methods," and read some of the "intuitive" explanations. You can judge for yourself how intuitive they really are.

The proof of acceleration for this algorithm follows two steps. First, we establish the following recursion, showing how the objective error evolves over time.

Proposition 10.15. *For any $k \geq 0$ and $y \in \mathcal{X}^*$, we have the following bound:*

$$\frac{1-\theta_{k+1}}{\theta_{k+1}^2}(f(x_{k+1}) - \inf f) + \frac{q}{2}\|y - z_{k+1}\|^2 \leq \frac{1-\theta_k}{\theta_k^2}(f(x_k) - \inf f) + \frac{q}{2}\|y - z_k\|^2. \tag{10.12}$$

This proposition is slightly cryptic, so let us defer its proof until Section 10.15 and instead look at its consequences. The following theorem shows shows that ($\mathcal{AMBA}$) achieves the desired $1/\sqrt{\varepsilon}$ complexity for the entire class of quadratically accurate models.

Theorem 10.16. *For every $k \geq 0$, it holds that*

$$f(x_k) - \inf f \leq 6q \cdot \frac{\operatorname{dist}^2(z_0, \mathcal{X}^*)}{(k+2)^2}.$$

Proof. Theorem 10.15 shows the sequence $\frac{1-\theta_{k+1}}{\theta_{k+1}^2}(f(x_{k+1}) - \inf f) + \frac{q}{2}\|y - z_{k+1}\|^2$ is nonincreasing. Therefore, we see that

$$\frac{1-\theta_k}{\theta_k^2}(f(x_k) - \inf f) \leq \frac{1-\theta_0}{\theta_0^2}(f(x_0) - \inf f) + \frac{q}{2}\|y - z_0\|^2 = \frac{q}{2}\|y - z_0\|^2,$$

since $\theta_0 = 1$. To complete the proof choose $y = \operatorname{proj}_{\mathcal{X}^*}(z_0)$, divide both sides of the equation by $\frac{1-\theta_k}{\theta_k^2}$, and use the bound: $\frac{\theta_k^2}{1-\theta_k} \leq \frac{12}{(k+2)^2}$. □

10.3.1 Proof of Proposition 10.15

We first upper bound $f(x_{k+1})$, using the quadratic accuracy of f_{y_k}:

$$\begin{aligned} f(x_{k+1}) &\le f_{y_k}(x_{k+1}) + \frac{q}{2}\|x_{k+1} - y_k\|^2 \\ &= f_{y_k}((1-\theta_k)x_k + \theta_k z_{k+1}) + \frac{q}{2}\|(1-\theta_k)x_k + \theta_k z_{k+1} - y_k\|^2. \end{aligned}$$

Next we simplify the expression in the norm to

$$(1-\theta_k)x_k + \theta_k z_{k+1} - y_k = (1-\theta_k)x_k + \theta_k z_{k+1} - ((1-\theta_k)x_k + \theta_k z_k) = \theta_k(z_{k+1} - z_k).$$

Using this simplification and the convexity of f_{y_k}, we thus have

$$\begin{aligned} f(x_{k+1}) &\le f_{y_k}((1-\theta_k)x_k + \theta_k z_{k+1}) + \frac{q}{2}\|(1-\theta_k)x_k + \theta_k z_{k+1} - y_k\|^2 \\ &\le (1-\theta_k)f_{y_k}(x_k) + \theta_k f_{y_k}(z_{k+1}) + \frac{q\theta_k^2}{2}\|z_{k+1} - z_k\|^2 \\ &\le (1-\theta_k)f(x_k) + \theta_k\left(f_{y_k}(z_{k+1}) + \frac{q\theta_k}{2}\|z_{k+1} - z_k\|^2\right), \end{aligned}$$

where the last line uses $f_{y_k}(x_k) \le f(x_k)$. Our aim is to simplify the term in parenthesis.

To that end, we recall that z_{k+1} is the minimizer of the $q\theta_k$-strongly convex function $z \mapsto f_{y_k}(z) + \frac{q\theta_k}{2}\|z - z_k\|^2$. Thus, by by the quadratic growth property in (10.6), we find that

$$\frac{q\theta_k}{2}\|y - z_{k+1}\|^2 + \left\{f_{y_k}(z_{k+1}) + \frac{q\theta_k}{2}\|z_{k+1} - z_k\|^2\right\} \le \left\{f_{y_k}(y) + \frac{q\theta_k}{2}\|y - z_k\|^2\right\}, \qquad \forall y \in \mathbb{R}^d.$$

Evaluating the above quadratic growth bound at any $y \in \mathcal{X}^*$, we find that

$$\begin{aligned} f(x_{k+1}) &\le (1-\theta_k)f(x_k) + \theta_k\left(f_{y_k}(z_{k+1}) + \frac{q\theta_k}{2}\|z_{k+1} - z_k\|^2\right) \\ &\le (1-\theta_k)f(x_k) + \theta_k\left(f_{y_k}(y) + \frac{q\theta_k}{2}\|y - z_k\|^2 - \frac{q\theta_k}{2}\|y - z_{k+1}\|^2\right) \\ &\le (1-\theta_k)f(x_k) + \theta_k\left(\inf f + \frac{q\theta_k}{2}\|y - z_k\|^2 - \frac{q\theta_k}{2}\|y - z_{k+1}\|^2\right), \end{aligned}$$

where the last line uses $f_{y_k}(y) \le f(y) = \inf f$. To conclude, subtract $\inf f$ from both sides and divide by θ_k^2:

$$\frac{1}{\theta_k^2}(f(x_{k+1}) - \inf f) + \frac{q}{2}\|y - z_{k+1}\|^2 \le \frac{1-\theta_k}{\theta_k^2}(f(x_k) - \inf f) + \frac{q}{2}\|y - z_k\|^2,$$

The proof is completed by using the lower bound: $\frac{1-\theta_{k+1}}{\theta_{k+1}^2} \le \frac{1}{\theta_k^2}$.

10.4 Lower Complexity Bounds

What is a "reasonable" algorithm and how might we lower bound such an algorithm's complexity? To gain some intuition, let us look at the subgradient method ($\mathcal{SM}$). Expanding its recursion at an iteration $k \ll d$, we see every iterate lies in a shift of a "small" subspace:

$$x_{k+1} = x_k - \frac{1}{\rho_k} v_k = x_0 - \sum_{i=0}^{k} \frac{1}{\rho_i} v_i \in x_0 + \operatorname{span}\{v_0, \ldots, v_k\}.$$

Without loss of generality, let us fix the initial iterate $x_0 = 0$. Then we see that the subgradient method is one example of a broader class of *conceptual algorithms*, having the following property: given iterates $x_0, \ldots, x_k$

1. an adversary chooses an *arbitrary* subgradient $v_k \in \partial f(x_k)$;
2. the algorithm may then choose the next iterate in

$$x_{k+1} \in \operatorname{span}\{v_0, \ldots, v_k\}.$$

While the subgradient method selects $x_{k+1} = \sum_{i=0}^{k} \frac{1}{\rho_i} v_i$, conceptual algorithms allow the optimizer to choose x_{k+1} arbitrarily. For example, the optimizer may choose x_{k+1} to be the minimizer of f over the subspace $\operatorname{span}\{v_0, \ldots, v_k\}$. Thus, the conceptual algorithm "should" perform at least as well the subgradient method. On the other hand,

the algorithm has no control over the choice of subgradients v_k.

Instead, subgradients are simply presented to the algorithm, and the adversary is free to choose the "worst-possible" ones.

Our goal in this section is to understand the worst case behavior of any conceptual algorithm. To understand this behavior we will introduce a single "worst-case" function, one that causes trouble for any such method. To gain some intuition for how we might do this, recall that the subgradient v_k may be chosen arbitrarily by an adversary. If the adversary chooses subgradients so that $\operatorname{span}\{v_0, \ldots, v_k\}$ is far from $\mathcal{X}^*$, then we expect such algorithms to perform poorly. To lower bound complexity we thus search for a function whose subgradients can reveal little about its minimizers.

For example fix $K < d$ and consider the function

$$f(x) := \max_{i \leq K+1} \{x(i)\} + \frac{1}{2}\|x\|^2 \qquad \forall x \in \mathbb{R}^d,$$

We seek a particular choice of subgradients $v_0, \ldots, v_k$ ensuring $f(x_k)$ remains large relative to $\inf f$. To achieve this goal, we must first compute both the subdifferential of f and its optimal value $\inf f$. For the first task, we define $I(x) = \{i \leq K+1 \colon x_i = \max_{j \leq K+1}\{x_j\}\}$ and compute

$$\partial f(x) = \operatorname{conv}\{e_i \colon i \in I(x)\} + x, \qquad \forall x \in \mathbb{R}^d,$$

where $e_1, \ldots, e_d$ are the canonical basis of $\mathbb{R}^d$. In what follows, we will be interested in a particular subgradient $v(x) \in \partial f(x)$:

$$v(x) := e_{\min I(x)} + x.$$

For the second task, we directly check that the point

$$x^* = \left(\underbrace{-\frac{1}{K+1},\dots,-\frac{1}{K+1}}_{K+1 \text{ times}},0,\dots 0\right)$$

satisfies $0 \in \partial f(x^*)$. Thus x^* is optimal and

$$\inf f = f(x^*) = -\frac{1}{K+1} + \frac{1}{2}\|x^*\|^2 = -\frac{1}{2(K+1)}.$$

Thus it remains to check the performance of the conceptual algorithm. We do so in the following proposition. (The reader should compare the bound in Proposition to the classical complexity bound for the subgradient method (10.5).)

Proposition 10.17 (Lower Complexity of "Subgradient Methods")**.** *There exists a choice of subgradients $v_0, v_1, \dots, v_K$ so that for all $k \leq K$, we have*

$$\min_{i \leq K} f(x_i) - f^* \geq \frac{\operatorname{dist}(x_0, \mathcal{X}^*)}{2\sqrt{K+1}}.$$

Proof. Before choosing the subgradients, we note that x^* is the unique minimizer of f (strong convexity) and conclude that

$$\operatorname{dist}(x_0, \mathcal{X}^*) = \|0 - x^*\| = \frac{1}{\sqrt{K+1}}.$$

Therefore, we can revise the desired bound to

$$\min_{i \leq K} f(x_i) - f^* \geq \frac{1}{2(K+1)}.$$

Since $f^* = \frac{1}{2(K+1)}$, it suffices to choose $v_0, \dots, v_K$ so that $f(x_k) \geq 0$ for all $k \leq K$. We will prove something stronger, namely, we will show that $x_k \in \operatorname{span}\{e_1, \dots, e_k\}$, yielding $f(x_k) = 0 + \frac{1}{2}\|x_k\|^2 \geq 0$, as desired.

To that end choose $v_i = v(x_i) \in \partial f(x_i)$. We claim that for all $k \leq K$, it holds that both x_k and v_{k-1} are contained in $\operatorname{span}\{e_1, \dots, e_k\}$. Let us begin with $k = 1$. In this case, $v(x_0) = v(0) = e_1 + 0 = e_1$, so

$$x_1 \in \operatorname{span}(v(x_0)) = \operatorname{span}(e_1).$$

Next suppose that $k + 1 \leq K$ and the inclusions hold:

$$x_i \in \operatorname{span}\{e_0, \dots, e_i\} \qquad \text{and} \qquad v(x_{i-1}) \in \operatorname{span}\{e_1, \dots, e_i\}, \qquad \forall i \leq k.$$

Then, since $\min I(x_k) \leq k + 1$ (check!), it holds that

$$v(x_k) = e_{\min I(x_k)} + x_k \in \operatorname{span}\{e_0, \dots, e_{k+1}\},$$

and consequently,

$$x_{k+1} \in \operatorname{span}\{v(x_0), \dots, v(x_k)\} \subseteq \operatorname{span}\{e_1, \dots, e_{k+1}\}.$$

Therefore, the claim follows by induction. □

An important feature of the above bound is it is valid only for a fixed dimension d and a fixed starting point $x_0 = 0$. This is not by accident: if we allow our bounds to depend on the dimension, we can no longer lower bound $\min_{i\leq k} f(x_i) - \inf f$ by a polynomial in $1/k$. The interested reader should consult Chapter 2 in Bubeck's excellent book [**?**]. In Chapter 3 of Bubeck's book, you will also find lower complexity bounds for smooth minimization methods, showing Nesterov's accelerated gradient method from Section 10.3 is "optimal."

10.5 Stochastic Methods

It is common in statistics and machine learning to encounter problems of the form

$$\text{minimize}_{x\in\mathbb{R}^d}\ f(x) = \frac{1}{m}\sum_{i=1}^m f_i(x),$$

where m is large. For example each term may take the form

$$f_i(x) = g_i(x, (a_i, b_i)),$$

where a_i is a training datum, b_i is the label of a_i, the vector $x \in \mathbb{R}^d$ encodes a classifier, and the loss g_i penalizes misclassification of a_i by x, i.e., it encourages $x(a_i) = b_i$, if we interpret x as a mapping from a-space to b-space.[21] Much research in modern machine learning is devoted to such problems, but there the losses f_i often lack convexity. When the loss is nonconvex, we cannot expect to minimize it, and the focus of current research is to quickly find a "critical" point: $\nabla f(x) = 0$. Nevertheless, whether the loss is convex or nonconvex the algorithm of choice is the following *stochastic gradient method (SGM)*:[22]

$$\begin{aligned}\textbf{Sample: }& i_k \text{ uniformly from } \{1,\dots,m\}\\ \textbf{Update: }& x_{k+1} = x_k - \alpha_k \nabla f_{i_k}(x_k).\end{aligned}$$

for a deterministic sequence $\alpha_0, \alpha_1, \ldots$ of "stepsizes," determined before running the algorithm. To gain some intuition on why this algorithm works, we take the expectation of x_{k+1} over the randomness in the kth iteration

$$\mathbb{E}\left[x_{k+1} \mid x_0, \ldots, x_k\right] = x_k - \alpha_k \frac{1}{m}\sum_{i=1}^m \nabla f_i(x_k).$$

This shows the algorithm is an *unbiased* estimator of the classical (nonstochastic) gradient method:

$$x_{k+1} = x_k - \alpha_k \frac{1}{m}\sum_{i=1}^m \nabla f_i(x_k).$$

It also illustrates the large computational savings afforded by SGM, since the stochastic method computes just a single gradient $\nabla f_{i_k}(x_k)$ per iteration.

Perhaps the biggest mystery in machine learning is why SGM performs so well in practice, for example, SGM often finds global minima of highly nonconvex and nonsmooth problems

[21] For example, x could encode the "weights" of a neural network.

[22] It is even a standard option in the industry backed solvers Facebook's PyTorch and Google's Tensorflow.

(e.g., ReLU neural networks). When f is nonconvex, little theory is known. A particularly challenging case is when f lacks smoothness and convexity; the interested reader should consult my website for recent progress on this front.

In contrast, when f is convex, there is an extensive supporting theory, originating from the pioneering 1951 work of Robbins and Monro [?]. This work has a counterintuitive conclusion:

the complexity of the stochastic gradient algorithm does not scale with m.

In fact one can prove similar complexity guarantees even in the "infinite sum" case, where

$$f(x) = \mathbb{E}_{z\sim P}\left[f(x,z)\right].$$

Here, z encodes "population" data, assumed to follow a fixed but unknown probability distribution P, and for each z, the loss $f(\cdot, z)\colon \mathbb{R}^d \to \mathbb{R}$ is convex. While P is unknown, we assume that we can sample from it. In this setting, the stochastic gradient method takes the form

$$\begin{aligned} &\textbf{Sample: } z_k \sim P \\ &\textbf{Update: } x_{k+1} = x_k - \alpha_k \nabla_x f(x_k, z_k). \end{aligned}$$

Although the stochastic gradient method sees only the "training" sample $f(\cdot, z_1), \ldots, f(\cdot, z_k)$, the algorithm minimizes the expectation $f(x)$, as if it had truly seen infinitely many samples. In machine learning, this fact is summarized:

The stochastic gradient method minimizes the testing error.

Underlying this result is a crucial assumption: the data $z_1, \ldots, z_k$ are *i.i.d.*, meaning each sample is touched just once.

Since these properties seem almost magical, the reader may wonder whether there is a catch? If there were no difference between the iteration complexity of stochastic and deterministic gradients method, then one would always choose a stochastic method. A quick experiment, however, shows the "vanilla" stochastic gradient method performs worse than the gradient method (see Figure 14.) Later, we will provide some formal evidence to illuminate this behavior. For now we mention that more "practical" variants of the stochastic gradient method abound, for example, variants that use "adaptive stepsizes" α_k, leverage second order information, or incorporate algorithm history or "minibatches" of gradients to "reduce the variance" of "gradient estimators;" a quick internet search will reveal much work in this vein, so we will not dwell on it here. Instead for simplicity, we will focus on the "vanilla" method, providing a useful stepping stone for further study.

Keeping in line with our earlier focus on models, we will analyze a stochastic variant of $(\mathcal{MBA})$. To make $(\mathcal{MBA})$ stochastic, we make an analogy with the stochastic gradient method. For this classical method, each new iterate is the unique minimizer of

$$x_{k+1} = \operatorname*{argmin}_{y\in\mathbb{R}^d} \left\{ f_{i_k}(x) + \langle \nabla f_{i_k}(x), y - x\rangle + \frac{1}{2\alpha_k}\|y - x_k\|^2 \right\}.$$

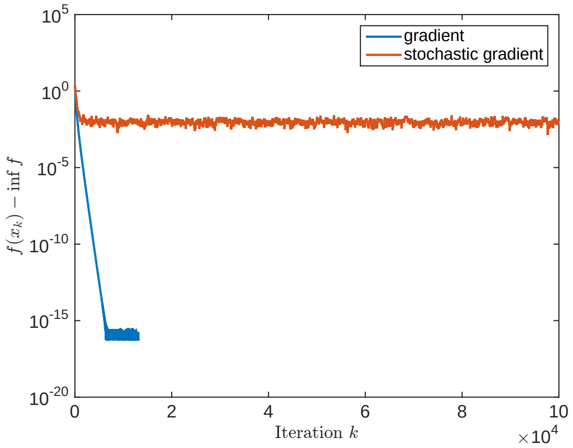


Figure 14: (Stochastic) Gradient Methods on a Least Squares Problem: $f(x) = \frac{1}{2m}\sum_{i=1}^m (a_i^T x - b_i)^2$.

Thus the stochastic gradient method doubly approximates f at every step: First it builds the linear model-based approximation: $f_{x_k}(y) = \frac{1}{m}\sum_{i=1}^m (f_i(x_k) + \langle \nabla f_i(x_k), y - x_k\rangle)$. Then it minimizes a stochastic approximation of the model: $(f_{i_k})_{x_k}(y) = f_{i_k}(x_k) + \langle \nabla f_{i_k}(x_k), y - x_k\rangle$. Viewing the stochastic gradient method through this lens, we see there is a clear stochastic generalization of ($\mathcal{MBA}$).

Namely, assume the following holds:

Assumption A (Stochastic Model-Based Algorithm Assumptions)**.**

1. **(Convexity.)** The functions $f_1, \ldots, f_m$ are closed, proper, and convex.
2. **(Common Domain.)** All terms have the same domain: $\mathrm{dom}(f) = \mathrm{dom}(f_1) = \ldots = \mathrm{dom}(f_m)$.
3. **(Lipschitz Continuity.)** The functions $f_1, \ldots, f_n$ are l-Lipschitz continuous on their domains.
4. **(Linearly Accurate Models.)** At every $x \in \mathrm{dom}(f)$ and for every $i = 1, \ldots m$, the function $(f_i)_x \colon \mathbb{R}^d \to (-\infty, \infty]$ is an l-linearly accurate model of f_i at x:

$$(f_i)_x(y) \leq f_i(y) \leq (f_i)_x(y) + l\|y - x\|, \qquad \forall y \in \mathbb{R}^d.$$

With these assumptions in hand, we introduce the *stochastic model-based algorithm:* given $x_0, \ldots, x_k$, define the next iterate by

$$\begin{aligned} &\textbf{Sample: } i_k \text{ uniformly from } \{1, \ldots, m\} \\ &\textbf{Update: } x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ (f_{i_k})_{x_k}(y) + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}, \end{aligned} \tag{$\mathcal{SMBA}$}$$

for a deterministic sequence of control parameters $\rho_0, \rho_1, \ldots$, meaning one choses all parameters before the algorithm begins.

Let us comment on Assumption A. Part 1 reflects our focus on convex optimization. Part 2 ensures $f(x_k) < +\infty$ for every iterate of ($\mathcal{SMBA}$). Without this assumption could not prove complexity bounds on the sequence x_k. Part 3 asks for Lipschitz continuity of f_i only on $\mathrm{dom}(f)$. This allows for infinite valued functions, for example, the term f_i could decompose into a sum $f_i = g_i + \delta_{\mathcal{X}}$, where g_i is l-Lipschitz and $\mathcal{X}$ is closed and convex. Such Lipschitz assumptions are common in stochastic optimization and can sometimes be relaxed. Finally, Part 4 asks each model to be "accurate enough." Recall linearly accurate models featured in convergence of "subgradient methods," rather than "gradient methods."" Thus, the results we prove below specialize classical results for the *stochastic subgradient method.* There, each f_i is l-Lipschitz and one builds linear models from subgradients $v_{i,x} \in \partial f(x)$, meaning

$$(f_i)_x(y) = f(x) + \langle v_{i,x}, y - x\rangle, \qquad \forall x, y \in \mathbb{R}^d.$$

For such problems, the algorithm ($\mathcal{SMBA}$) specializes to:

$$\begin{aligned} &\textbf{Sample: } i_k \text{ uniformly from } \{1, \ldots, m\} \\ &\textbf{Update: } x_{k+1} = x_k - \alpha_k v_{i_k, x_k}. \end{aligned}$$

This is classical stochastic subgradient method, applicable to nondifferentiable losses. While we will look only at linearly accurate models, it is possible to give similar guarantees for quadratically accurate ones. We work with linearly accurate models for reasons of simplicity and generality.

Departing from classical algorithms, one may use any of the models in Section 10.1.2. For example, if each f_i is simple, a natural model is simply the function itself: $(f_i)_x = f_i$. This choice gives us the *stochastic proximal algorithm:*

$$\begin{aligned} &\textbf{Sample: } i_k \text{ uniformly from } \{1, \ldots, m\} \\ &\textbf{Update: } x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ f_{i_k}(y) + \frac{\rho_k}{2}\|y - x_k\|^2 \right\}, \end{aligned}$$

which in some cases outperforms subgradient methods. For example consider the *least absolute deviations problem*:

$$\text{minimize}_{x \in \mathbb{R}^d} \ f(x) = \frac{1}{m}\sum_{i=1}^{m} |a_i^T x - b|$$

where $a_1, \ldots, a_m \in \mathbb{R}^d$ and we define $f_i(x) := |a_i^T x - b|$. In the exercises, you will show each step of the proximal algorithm has a simple closed form solution, computable with $O(d)$ operations. More importantly the stochastic proximal method dramatically outperforms the (stochastic) subgradient method, as shown in Figure 15. Despite each method having the same parameters, the proximal method converges linearly, unlike the deterministic or stochastic subgradient method. The reason is we chose $b \in \mathrm{range}(A)$, meaning one can perfectly fit the data. In machine learning, this is called the "interpolation" phenomenon. While the proximal method converges linearly for any natural step size it is possible to tune the stepsizes of the subgradient methods, in a way that leads to linear convergence; for more information, see my paper [**?**].

Moving to complexity, we state the following theorem, characterizing the complexity of ($\mathcal{SMBA}$).

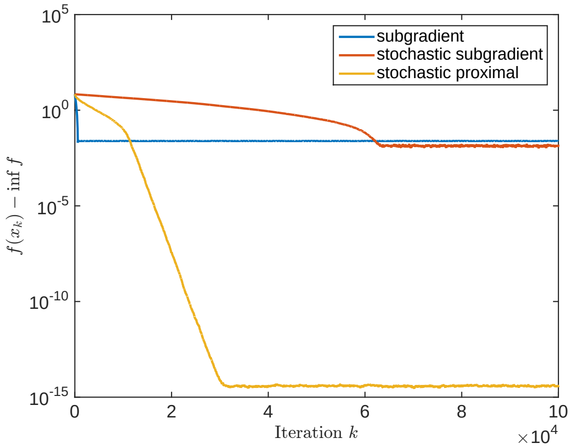


Figure 15: (Stochastic) Gradient Methods on a Least Squares Problem: $f(x) = \frac{1}{m}\sum_{i=1}^m |a_i^T x - b_i|_1$.

Theorem 10.18. *Fix a constant $C > 0$, a time horizon $K \geq 0$, and set $\rho_k \equiv \rho = C\sqrt{K+1}$. Let K^* be a uniform random variable on the set $\{0, \ldots, K\}$. Then*

$$\mathbb{E}\left[f(x_{K^*}) - \inf f\right] \leq \frac{\frac{1}{2}C^2 \mathrm{dist}^2(x_0, \mathcal{X}^*) + 2l^2}{C\sqrt{K+1}}$$

Setting $C := \frac{2l}{\mathrm{dist}(x_0, \mathcal{X}^*)}$, we have

$$\mathbb{E}\left[f(x_{K^*}) - \inf f\right] \leq \frac{2\mathrm{dist}(x_0, \mathcal{X}^*)l}{\sqrt{K+1}}$$

Before we prove the theorem, we make a few comments.

1. Similar to the results of Section 10.4, the convergence rate in this bound is the best possible for any "reasonable algorithm." This was proved by Nemirovski and Yudin; see [**?**] for another proof.

2. The expectation symbol denotes integration with respect to two sources of randomness: the random gradient index i_k and the random iterate choice K^*. In particular, we may simplify the expectation as follows

$$\mathbb{E}\left[f(x_{K^*}) - \inf f\right] = \frac{1}{K+1}\sum_{k=0}^{K} \mathbb{E}_{i_1,\ldots,i_K}\left[f(x_k) - \inf f\right].$$

   Thus the theorem states that the average functional error is small. Applying Jensen's inequality (Lemma 10.11), we also deduce that the expected functional error at $(K+1)^{-1}\sum_{k=0}^{K} x_k$ is small.

3. On the surface, the bound of Theorem 10.18 appears to match the bound of Theorem 10.12. This counterintuitive conclusion suggests that one should always prefer stochastic algorithms over deterministic ones. Indeed, their convergence rates are

nearly the same but the work per iteration of a deterministic method is m times that of a stochastic one. Here is the catch: while both methods have the same dependence on K, the problem parameter l can be dramatically different. For example, let $\bar{x} \in \mathbb{R}^d$ and let $a_1, \ldots, a_m \sim N(0, I_d)$ be sampled from a Gaussian distribution. For $i = 1, \ldots, m$, define $f_i(x) = |a_i^T x - a_i^T \bar{x}|$. Then provided $m \geq \Omega(d)$, for each i

$$\text{the Lip. constant of } f_i \text{ is } \Omega(\sqrt{d}) \text{ while the Lip. constant of } f \text{ is } O(1)$$

with high probability. We will not prove these facts, but we mention that they follow from standard concentration inequalities for Gaussian vectors. Let us use these computations to compare the complexity of the stochastic and deterministic subgradient methods, working under the additional assumption that $m = O(d)$. In this case each algorithm reaches ε accuracy in the following number of iterations:

$$\underset{\text{stochastic subgradient method}}{O\left(\frac{d \cdot \operatorname{dist}^2(x_0, \mathcal{X}^*)}{\varepsilon^2}\right)} \qquad \text{and} \qquad \underset{\text{subgradient method}}{O\left(\frac{\operatorname{dist}^2(x_0, \mathcal{X}^*)}{\varepsilon^2}\right)}$$

Taking into account the number of subgradients computed in each iteration, we find that both methods require at most

$$O\left(\frac{d \cdot \operatorname{dist}^2(x_0, \mathcal{X}^*)}{\varepsilon^2}\right)$$

subgradient evaluations to reach an ε accurate solution (in expected objective value). In short, there is no free lunch.

Turning to the proof, we will see it is a consequence of the next lemma, which the reader should compare to Lemma 10.9.

Lemma 10.19. *For all $k \geq 0$, we have*

$$\mathbb{E}\left[f(x_k) - \inf f\right] \leq \mathbb{E}\left[\frac{\rho}{2}\|x_k - y\|^2 - \frac{\rho}{2}\|x_{k+1} - y\|^2\right] + \frac{2l^2}{\rho}, \qquad \forall y \in \mathcal{X}^*.$$

Proof. We use throughout the proof that $f(y) = \inf f$ for all $y \in \mathcal{X}^*$. We will also use the notation $\mathbb{E}_k\left[\cdot\right]$ to denote the conditional expectation $\mathbb{E}\left[\cdot \mid x_0, x_1, \ldots, x_k\right]$, where we condition on the entire history of the algorithm up until time k. Thus

$$f(x_k) - f(y) = \mathbb{E}_k\left[f(x_k) - f(y)\right] = \mathbb{E}_k\left[f_{i_k}(x_k) - f(y)\right]$$

where the second equality follows since i_k is uniformly sampled from $1, \ldots, m$. Continuing, we see that

$$\begin{aligned} f(x_k) - f(y) &= \mathbb{E}_k\left[f_{i_k}(x_k) - f(y)\right] \\ &\leq \mathbb{E}_k\left[f_{i_k}(x_{k+1}) - f(y) + l\|x_k - x_{k+1}\|\right] \\ &\leq \mathbb{E}_k\left[(f_{i_k})_{x_k}(x_{k+1}) - f(y) + 2l\|x_k - x_{k+1}\|\right], \end{aligned}$$

where the first inequality follows from Lipschitz continuity of f_i and the second follows because $(f_{i_k})_{x_k}$ is l-linearly accurate. We have put ourselves in favorable place, since x_{k+1} is the unique minimizer of the ρ-strongly convex function $(f_{i_k})_{x_k}(y)+\frac{\rho}{2}\|y-x_k\|^2$. In particular, quadratic growth away from minimizers implies

$$\left\{(f_{i_k})_{x_k}(x_{k+1})+\frac{\rho}{2}\|x_{k+1}-x_k\|^2\right\}+\frac{\rho}{2}\|y-x_k\|^2 \leq \left\{(f_{i_k})_{x_k}(y)+\frac{\rho}{2}\|y-x_{k+1}\|^2\right\}.$$

Letting $\Delta_k=\|y-x_k\|^2$ and $\delta_k=\|x_{k+1}-x_k\|$ for all k, we find the inequality

$$\begin{aligned} f(x_k)-f(y) &\leq \mathbb{E}_k\left[(f_{i_k})_{x_k}(x_{k+1})-f(y)+2l\|x_k-x_{k+1}\|\right] \\ &\leq \mathbb{E}_k\left[(f_{i_k})_{x_k}(y)-f(y)+\frac{\rho}{2}\Delta_k-\frac{\rho}{2}\Delta_{k+1}+2l\delta_k-\frac{\rho}{2}\delta_k^2\right]. \end{aligned}$$

Two observations simplify this inequality:

1. First, since i_k is a uniform random variable, we have $\mathbb{E}_k\left[(f_{i_k})_{x_k}(y)-f(y)\right]=0$.

2. Second, we have $2l\delta_k-2l\delta_k^2 \leq \max_{\delta\in\mathbb{R}}(2l\delta-(\rho/2)\delta^2) \leq \frac{2l^2}{\rho}$.

Putting these together, we find

$$\begin{aligned} f(x_k)-f(y) &\leq \mathbb{E}_k\left[(f_{i_k})_{x_k}(y)-f(y)+\frac{\rho}{2}\Delta_k-\frac{\rho}{2}\Delta_{k+1}+2l\delta_k-\frac{\rho}{2}\delta_k^2\right] \\ &\leq \mathbb{E}_k\left[\frac{\rho}{2}\Delta_k-\frac{\rho}{2}\Delta_{k+1}\right]+\frac{2l^2}{\rho}. \end{aligned}$$

To complete the proof, take expectations of both sides and use the law of total expectations to simplify $\mathbb{E}\left[\mathbb{E}_k\left[\cdot\right]\right]=\mathbb{E}\left[\cdot\right]$. □

We now conclude with the proof of Theorem 10.18.

Proof of Theorem 10.18. We apply the lemma (and the notation of the proof of the lemma) in a straightforward manner:

$$\begin{aligned} \mathbb{E}\left[f(x_{K^*})-\inf f\right] &= \mathbb{E}\left[\frac{1}{K+1}\sum_{k=0}^{K}(f(x_k)-\inf f)\right] \\ &\leq \mathbb{E}\left[\frac{1}{K+1}\sum_{k=0}^{K}\left(\frac{\rho}{2}\Delta_k-\frac{\rho}{2}\Delta_{k+1}+\frac{2l^2}{\rho}\right)\right] \\ &\leq \mathbb{E}\left[\frac{\frac{\rho}{2}\Delta_0}{K+1}\right]+\frac{2l^2}{\rho} \\ &= \frac{\frac{C^2}{2}\|x_0-y\|^2+2l^2}{C\sqrt{K+1}}. \end{aligned}$$

To complete the proof, choose $y=\operatorname{prox}_{\mathcal{X}^*}(x_0)$. □

10.6 Appendix: Proofs of Propositions 10.1 and 10.2

We first prove that the linear models of functions with Lipschitz gradients are quadratically accurate.

Proof of Proposition 10.1. Fix $y \in \mathbb{R}^d$. We first see that the lower bound $f_x(y) \leq f(y)$ follows since $\nabla f(x) \in \partial f(x) = \{\nabla f(x)\}$.

Moving to upper bound, we define a path $z(t) = (1-t)x + ty$ and let $g(t) = f(z(t))$. Since $g(1) = f(y)$, $g(0) = f(x)$, and $g'(s) = \langle \nabla f(z(t)), \dot{z}(t)\rangle = \langle \nabla f(z(t)), y - x\rangle$, the inequality we wish to to prove is equivalent to

$$g(1) - g(0) - g'(0) \leq \frac{q}{2}\|x - y\|^2.$$

To that end we use the fundamental theorem of calculus to show

$$g(1) - g(0) = \int_0^1 g'(s)ds = g'(0) + \int_0^1 (g'(s) - g'(0))ds.$$

The proof will be complete if we can bound the integral by $\frac{q}{2}\|x - y\|^2$. We prove this bound in two steps. First we upper bound the integrand:

$$\begin{aligned}|g'(s) - g'(0)| &= |\langle \nabla f(z(s)) - \nabla f(z(0)), y - x\rangle| \\ &\leq \|\nabla f((1-s)x + sy) - \nabla f(0)\|\|y - x\| \\ &\leq qs\|x - y\|^2, \qquad\qquad \forall s \in [0, 1],\end{aligned}$$

Then we integrate the upper bound on the integrand:

$$\int_0^1 (g'(s) - g'(0))ds \leq q\|x - y\|^2 \cdot \int_0^1 sds \leq \frac{q}{2}\|x - y\|^2.$$

This completes the proof. □

Next we show that the linear models of Lipschitz continuous functions are linearly accurate.

Proof of Proposition 10.2. Fix $y \in \mathbb{R}^d$. We first see that the lower bound $f_x(y) \leq f(y)$ follows since $v_x \in \partial f(x)$.

Moving to the upper bound, we recall a useful fact from Exercise 9.9: the subgradients of f are bounded, namely $\|v_x\| \leq \hat{l}$ holds. In particular, we have $|\langle v_x, y - x\rangle| \leq \hat{l}\|y - x\|$. Using this bound and Lipschitz continuity, we find that

$$\begin{aligned}f(y) &\leq f(x) + \hat{l}\|x - y\| \\ &\leq f(x) + \langle v_x, y - x\rangle - |\langle v_x, y - x\rangle| + \hat{l}\|x - y\| \\ &\leq f_x(y) + 2\hat{l}\|x - y\|,\end{aligned}$$

completing the proof. □

10.7 Exercises

Exercise 10.1 (Descent Directions)**.**

1. Suppose that f is Fréchet differentiable on $\mathbb{R}^d$ and that $\nabla f(x)$ is a continuous function of x. Show that for all $x \in \mathbb{R}^d$ with $\nabla f(x) \neq 0$, there exists $\gamma > 0$ such that
$$f(x - \gamma \nabla f(x)) < f(x).$$
(**Hint:** Consider the derivative of the one variable function $g(\gamma) = f(x - \gamma \nabla f(x))$.)

2. Consider a convex function $f(x, y) = a|x| + b|y|$ for scalars $a, b > 0$. Find a point $(x_0, y_0) \in \mathbb{R}^2$, coefficients $a, b > 0$, and a subgradient $v \in \partial f(x, y)$ so that
$$f((x_0, y_0) - \gamma v) > f(x_0, y_0) \qquad \forall \gamma > 0.$$

3. Let f be a continuous convex function. Let $x \in \mathbb{R}^d$ and suppose that $0 \notin \partial f(x)$. In this exercise, we will show that the minimal norm subgradient of f at x
$$v := \mathrm{proj}_{\partial f(x)}(0).$$
is a descent direction.

   (a) Show that
   $$\langle w, -v \rangle \leq -\|v\|^2 \qquad \forall w \in \partial f(x).$$

   (b) Next, define the one variable continuous convex function $g(\gamma) = f(x - \gamma v)$. Show that
   $$\eta \in \partial g(0) \implies \eta < -\|v\|^2.$$
   Can 0 be a minimizer of g?

   (c) Show that for all $\gamma < 0$, we have $g(\gamma) > g(0)$.

   (d) Use parts (b) and (c) to show that for $g(\gamma) < g(0)$ for all sufficiently small $\gamma > 0$.

Exercise 10.2 (Proofs of Approximation Quality)**.** Prove the following propositions.

1. **Clipped/Aggregated Models.** Let $x \in \mathbb{R}^d$ and suppose that f_x is an (l, q) model of f at x. Moreover, assume that $g \colon \mathbb{R}^d \to (-\infty, \infty]$ is closed, proper, convex, and dominated by f: $g(y) \leq f(y)$ for all $y \in \mathbb{R}^d$. Then
$$\max\{f_x, g\}$$
is an (l, q)-model of f at x.

2. **Projected/Proximal Models.** Suppose that f admits the decomposition
$$f = g + h,$$
where $g, h : \mathbb{R}^d \to (-\infty, \infty]$ are closed, proper, convex functions. Let $x \in \mathbb{R}^d$ and suppose that g_x is an (l, q) model of g at x. Then
$$g_x + h$$
is an (l, q)-model of f at x.

3. **Max-Linear Models.** Suppose that f admits the decomposition

   $$f = \max(f_1, \ldots, f_n),$$

   where for each i, the function $f_i : \mathbb{R}^d \to (-\infty, \infty]$ is closed, proper, and convex. Let $x \in \mathbb{R}^d$ and suppose for each i, the function $(f_i)_x$ is an (l, q) model of f_i at x. Then

   $$\max\{(f_1)_x, \ldots, (f_n)_x\}$$

   is an (l, q)-model of f at x.

Exercise 10.3 (Clipping Subproblem)**.** Let $a, x \in \mathbb{R}^d$, let $\mathtt{lb} \in \mathbb{R}$, let $\rho > 0$, and let $b \in \mathbb{R}$. Prove that the point

$$x_+ = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ \max\{\langle a, x\rangle + b, \mathtt{lb}\} + \frac{\rho}{2}\|y - x\|^2 \right\}$$

satisfies

$$x_+ = x - \mathtt{clip}\left(\frac{\rho}{\|a\|^2}(\langle a, x\rangle + b - \mathtt{lb})\right)\frac{a}{\rho} \qquad \text{where} \qquad \mathtt{clip}(t) = \max\{\min\{t, 1\}, 0\}.$$

(**Hint:** use first order optimality conditions.)

In the the following exercises we study the core algorithmic subproblem in *proximal algorithms.* For motivation recall the *proximal subgradient method* from above. This is perhaps the most common algorithm one encounters in first-order methods, so you should at least have a working knowledge of how to implement its steps, when possible. In general it can be quite hard to implement these steps. Indeed, the subproblem includes as a special case the projection of a vector onto a convex set, a generally difficult task. Still for a few useful functions we can implement these steps, even with simple closed form expressions.

Exercise 10.4. Let $f\colon \mathbb{R}^d \to (-\infty, \infty]$ be a closed, proper, convex function. Let $\gamma > 0$ and define the *proximal operator* $\operatorname{prox}_{\gamma f}\colon \mathbb{R}^d \to \mathbb{R}^d$:

$$\operatorname{prox}_{\gamma f}(x) = \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ f(y) + \frac{1}{2\gamma}\|y - x\|^2 \right\}.$$

1. Prove that for all $x \in \mathbb{R}^d$, we have

   $$x_+ = \operatorname{prox}_{\gamma f}(x) \iff (x - x_+) \in \gamma \partial f(x_+)$$

   (**Hint:** use strong convexity.)

2. Prove that $x \in \mathbb{R}^d$ is minimizes f if and only if $x = \operatorname{prox}_{\gamma f}(x)$.

3. **(Minty's Theorem.)** Prove that

   $$\operatorname{range}(I + \partial f) = \{x + v \colon v \in \partial f(x)\} = \mathbb{R}^d.$$

   (**Hint:** use part (a).)

4. Prove that $\mathrm{prox}_{\gamma f}$ is 1-Lipschitz, i.e.,

   $$\|\mathrm{prox}_{\gamma f}(x) - \mathrm{prox}_{\gamma f}(y)\| \leq \|x - y\|, \qquad \forall x, y \in \mathbb{R}^d.$$

   (**Hint:** use strong convexity.)

Notice the relation between proximal and projection operators: If $f(x) = \delta_{\mathcal{X}}$ for a closed convex set $\mathcal{X}$, then $\mathrm{prox}_{\gamma f} = \mathrm{proj}_{\mathcal{X}}$ for all $\gamma > 0$.

Exercise 10.5 (**Calculus of Proximal Operators.**).

1. **(Linear Perturbation.)** Suppose that $f \colon \mathbb{R}^d \to (-\infty, \infty]$ is closed, proper, and convex, let $\gamma > 0$, and let $b, v \in \mathbb{R}^d$. Define a function

   $$g(x) = f(x + b) + v^T x, \qquad \forall x \in \mathbb{R}^d$$

   Prove that

   $$\mathrm{prox}_{\gamma g}(x) = \mathrm{prox}_{\gamma f}(x - \gamma v + b) - b, \qquad \forall x \in \mathbb{R}^d$$

   (**Hint:** First try the cases where $b = 0$ or $v = 0$.)

2. **(Separability.)** Let $d = d_1 + \ldots + d_n$ for integers d_i and let $f_i \colon \mathbb{R}^{d_i} \to (-\infty, +\infty]$ be proper convex functions. Let $\gamma > 0$ and for all $x = (x_1, \ldots, x_n) \in \mathbb{R}^d$, define $f(x_1, \ldots, x_n) := \sum_{i=1}^n f(x_i)$. Prove that

   $$\mathrm{prox}_{\gamma f}(x_1, \ldots, x_n) = (\mathrm{prox}_{\gamma f}(x_1), \ldots, \mathrm{prox}_{\gamma f_n}(x_n)), \qquad \forall x \in \mathbb{R}^d.$$

3. **(Scalarization.)** Let $f \colon \mathbb{R} \to (-\infty, \infty]$ be a scalar function, let $\gamma > 0$, and let $a \in \mathbb{R}^d \backslash \{0\}$. Define

   $$g(x) = f(a^T x), \qquad \forall x \in \mathbb{R}^d$$

   Prove that for all $x \in \mathbb{R}^d$, we have

   $$\mathrm{prox}_{\gamma g}(x) = x - \rho a \qquad \text{where } \rho = \frac{1}{\|a\|^2}(a^T x - \mathrm{prox}_{(\gamma \|a\|^2) f}(a^T x)).$$

   (**Hint:** Be careful: the chain rule $\partial g(y) = a \partial f(a^T y)$ may not hold. Instead, use the inclusion $a \partial f(a^T y) \subseteq \partial g(y)$.)

Exercise 10.6 (**Proximal Operator Examples.**). Compute the proximal operators of the following functions

1. $f(x) := \|x\|_1 = \sum_{i=1}^d |x_i|$.

2. $f(x) = \max\{0, x\}$ for a scalar variable $x \in \mathbb{R}$.

3. $f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle$, where $b \in \mathbb{R}^d$ and $A \in \mathbb{R}^{d \times d}$ is a symmetric positive semidefinite matrix.

4. $f(x) = \|x\|_2$.
   (**Hint:** First compute the subdifferential of f, keeping in mind that f is differentiable everywhere except the origin.)

5. $f(x) = \delta_{\mathcal{X}}$, where $\mathcal{X} = \{x \in \mathbb{R}^d \colon x \geq 0\}$ is the nonnegative orthant.

6. $f(x) = \delta_{\mathcal{X}}$, where $\mathcal{X} = \{x \colon Ax = b\}$ is an affine space defined by matrix $A \in \mathbb{R}^{m \times d}$ and $b \in \mathbb{R}^m$.

7. $f(x) = \delta_{\mathcal{X}}$, where $\mathcal{X} = \{x \colon \|x\|_\infty \leq 1\}$
(**Hint:** You already computed $\partial f(x)$ on a previous homework assignment.)

Exercise 10.7 ((Projection onto $\mathbb{S}_+^{d\times d}$).). Recall that any symmetric matrix $A \in \mathbb{S}^{d\times d}$ (not necessarily positive semidefinite) has an eigenvalue decomposition

$$A = Q\Lambda Q^T \qquad \text{where} \left\{\begin{aligned} Q^T Q &= I \\ \Lambda &= \operatorname{diag}(\lambda_1, \ldots, \lambda_d) \\ \lambda_1 &\geq \ldots \geq \lambda_d. \end{aligned}\right\}.$$

For any such matrix, prove that

$$\operatorname{proj}_{\mathbb{S}_+^{d\times d}}(A) = Q \max\{\Lambda, 0\} Q^T.$$

(**Hint:** Verify the first order optimality conditions $A - \operatorname{proj}_{\mathbb{S}_+^{d\times d}}(A) \in \mathcal{N}_{\mathbb{S}_+^{d\times d}}(\operatorname{proj}_{\mathbb{S}_+^{d\times d}}(A))$)