Contents

Chapter 1

Theory of Convex Functions

Contents

This chapter develops the basic theory of convex functions that we will need later. Much of the material is also covered in other courses, so we will refer to the literature for standard material and focus more on material that we feel is less standard (but important in our context).

1.1 Mathematical Background

1.1.1 Notation

For vectors in $\mathbb{R}^d$, we use bold font, and for their coordinates normal font, e.g. $\mathbf{x} = (x_1, \ldots, x_d) \in \mathbb{R}^d$. $\mathbf{x}_1, \mathbf{x}_2, \ldots$ denotes a sequence of vectors. Vectors are considered as column vectors, unless they are explicitly transposed. So $\mathbf{x}$ is a column vector, and $\mathbf{x}^\top$, its transpose, is a row vector. $\mathbf{x}^\top \mathbf{y}$ is the scalar product $\sum_{i=1}^d x_i y_i$ of two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.

$\|\mathbf{x}\|$ denotes the Euclidean norm (ℓ_2-norm or 2-norm) of vector $\mathbf{x}$,

$$\|\mathbf{x}\|^2 = \mathbf{x}^\top \mathbf{x} = \sum_{i=1}^d x_i^2.$$

We also use

$$\mathbb{N} = \{1, 2, \ldots\} \text{ and } \mathbb{R}_+ := \{x \in \mathbb{R} : x \geq 0\}$$

to denote the natural and non-negative real numbers, respectively. We are freely using basic notions and material from linear algebra and analysis, such as open and closed sets, vector spaces, matrices, continuity, convergence, limits, triangle inequality, among others.

1.1.2 The Cauchy-Schwarz inequality

Lemma 1.1 (Cauchy-Schwarz inequality)**.** *Let* $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$*. Then*

$$|\mathbf{u}^\top \mathbf{v}| \leq \|\mathbf{u}\| \, \|\mathbf{v}\| \, .$$

The inequality holds beyond the Euclidean norm; all we need is an inner product, and a norm induced by it. But here, we only discuss the Euclidean case.

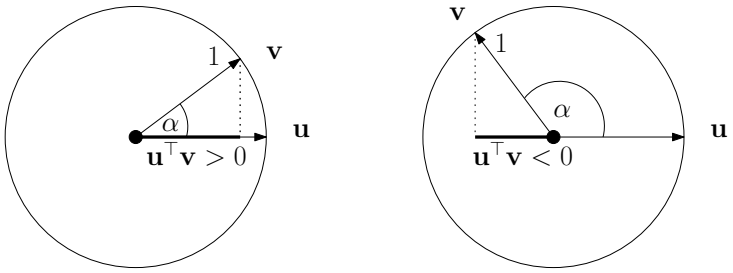
For nonzero vectors, the Cauchy-Schwarz inequality is equivalent to

$$-1 \leq \frac{\mathbf{u}^\top \mathbf{v}}{\|\mathbf{u}\| \, \|\mathbf{v}\|} \leq 1,$$

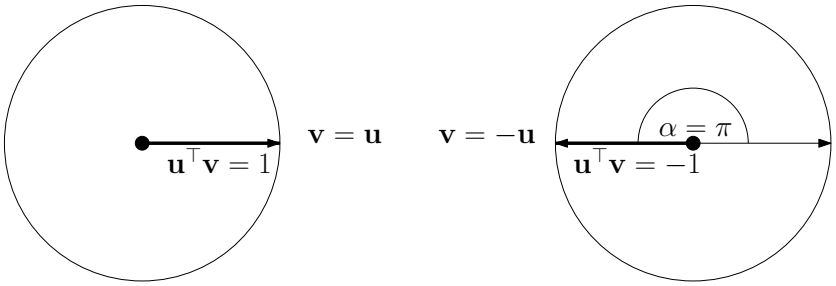
and this fraction can be used to define the angle α between $\mathbf{u}$ and $\mathbf{v}$:

$$\cos(\alpha) = \frac{\mathbf{u}^\top \mathbf{v}}{\|\mathbf{u}\| \, \|\mathbf{v}\|},$$

where $\alpha \in [0, \pi]$. The following shows the situation for two unit vectors ($\|\mathbf{u}\| = \|\mathbf{v}\| = 1$): The scalar product $\mathbf{u}^\top \mathbf{v}$ is the length of the projection of $\mathbf{v}$ onto $\mathbf{u}$ (which is considered to be negative when $\alpha > \pi/2$). This is just the highschool definition of the cosine.



Hence, equality in Cauchy-Schwarz is obtained if $\alpha = 0$ ($\mathbf{u}$ and $\mathbf{v}$ point into the same direction), or if $\alpha = \pi$ ($\mathbf{u}$ and $\mathbf{v}$ point into opposite directions):



Fix $\mathbf{u} \neq \mathbf{0}$. We see that the vector $\mathbf{v}$ maximizing the scalar product $\mathbf{u}^\top \mathbf{v}$ among all vectors $\mathbf{v}$ of some fixed length is a positive multiple of $\mathbf{u}$, while the scalar product is minimized by a negative multiple of $\mathbf{u}$.

Proof of the Cauchy-Schwarz inequality. There are many proof, but the authors particularly like this one: define the quadratic function

$$f(x) = \sum_{i=1}^{d} (u_i x + v_i)^2 = \left(\sum_{i=1}^{d} u_i^2 \right) x^2 + \left(2 \sum_{i=1}^{d} u_i v_i \right) x + \left(\sum_{i=1}^{d} v_i^2 \right) =: ax^2 + bx + c.$$

We know that $f(x) = ax^2 + bx + c = 0$ has the two solutions

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

This is known as the *Mitternachtsformel* in German-speaking countries, as you are supposed to know it even when you are asleep at midnight.

As by definition, $f(x) \geq 0$ for all x, $f(x) = 0$ has at most one real solution, and this is equivalent to having *discriminant* $b^2 - 4ac \leq 0$. Plugging in the definitions of a, b, c, we get

$$b^2 - 4ac = \left(2\sum_{i=1}^{d} u_i v_i\right)^2 - 4\left(\sum_{i=1}^{d} u_i^2\right)\left(\sum_{i=1}^{d} v_i^2\right) = 4(\mathbf{u}^\top \mathbf{v})^2 - 4\|\mathbf{u}\|^2 \|\mathbf{v}\|^2 \leq 0.$$

Dividing by 4 and taking square roots yields the Cauchy-Schwarz inequality.

1.1.3 The spectral norm

Definition 1.2 (Spectral norm)**.** *Let A be an $(m \times d)$-matrix. Then*

$$\|A\| := \max_{\mathbf{v} \in \mathbb{R}^d, \mathbf{v} \neq 0} \frac{\|A\mathbf{v}\|}{\|\mathbf{v}\|} = \max_{\|\mathbf{v}\|=1} \|A\mathbf{v}\|$$

is the 2-norm (or spectral norm) of A.

In words, the spectral norm is the largest factor by which a vector can be stretched in length under the mapping $\mathbf{v} \to A\mathbf{v}$. Note that as a simple consequence,

$$\|A\mathbf{v}\| \leq \|A\|\|\mathbf{v}\|$$

for all $\mathbf{v}$.

It is good to remind ourselves what a norm is, and why the spectral norm is actually a norm. We need that it is absolutely homegeneous: $\|\lambda A\| = |\lambda|\|A\|$ which follows from the fact that the Euclidean norm is absolutely homegeneous. Then we need the triangle inequality: $\|A + B\| \leq \|A\| + \|B\|$ for two matrices of the same dimensions. Again, this follows from the triangle inequality for the Euclidean norm. Finally, we need that $\|A\| = 0$ implies $A = 0$. Which is true, since for any nonzero matrix A, there is a vector $\mathbf{v}$ such that $A\mathbf{v}$ and hence the Euclidean norm of $A\mathbf{v}$ is nonzero.

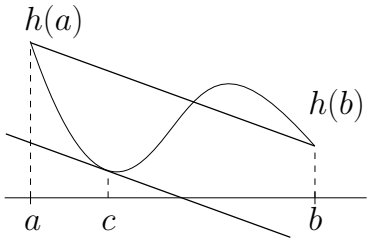
1.1.4 The mean value theorem

We also recall the *mean value theorem* that we will frequently need:

Theorem 1.3 (Mean value theorem)**.** *Let $a < b$ be real numbers, and let $h : [a, b] \to \mathbb{R}$ be a continuous function that is differentiable on (a, b); we denote the derivative by h'. Then there exists $c \in (a, b)$ such that*

$$h'(c) = \frac{h(b) - h(a)}{b - a}.$$

Geometrically, this means the following: We can interpret the value $(h(b)-h(a))/(b-a)$ as the slope of the line through the two points $(a, h(a))$ and $(b, h(b))$. Then the mean value theorem says that between a and b, we find a tangent to the graph of h that has the same slope:



1.1.5 The fundamental theorem of calculus

If a function h is *continuously* differentiable in an interval $[a, b]$, we have another way of expressing $h(b) - h(b)$ in terms of the derivative.

Theorem 1.4 (Fundamental theorem of calculus)**.** *Let $a < b$ be real numbers, and let $h : \mathbf{dom}(h) \to \mathbb{R}$ be a differentiable function on an open domain $\mathbf{dom}(h) \supset [a, b]$, and such that h' is continuous on $[a, b]$. Then*

$$h(b) - h(a) = \int_a^b h'(t)dt.$$

This theorem is the theoretical underpinning of typical definite integral computations in high school. For example, to evaluate $\int_2^4 x^2 dx$, we integrate x^2 (giving us $x^3/3$), and then compute

$$\int_2^4 x^2 dx = \frac{4^3}{3} - \frac{2^3}{3} = \frac{56}{3}.$$

1.1.6 Differentiability

For univariate functions $f : \mathbf{dom}(f) \to \mathbb{R}$ with $\mathbf{dom}(f) \subseteq \mathbb{R}$, differentiability is covered in high school. We will need the concept for multivariate and vector-valued functions $f : \mathbf{dom}(f) \to \mathbb{R}^m$ with $\mathbf{dom}(f) \subseteq \mathbb{R}^d$. Mostly, we deal with the case $m = 1$: real-valued functions in d variables. As we frequently need this material, we include a refresher here.

Definition 1.5. *Let $f : \mathbf{dom}(f) \to \mathbb{R}^m$ where $\mathbf{dom}(f) \subseteq \mathbb{R}^d$. Function f is called* differentiable *at $\mathbf{x}$ in the interior of $\mathbf{dom}(f)$ if there exists an $(m \times d)$-matrix A and an error function $r : \mathbb{R}^d \to \mathbb{R}^m$ defined in some neighborhood of $\mathbf{0} \in \mathbb{R}^d$ such that for all $\mathbf{y}$ in some neighborhood of $\mathbf{x}$,*

$$f(\mathbf{y}) = f(\mathbf{x}) + A(\mathbf{y} - \mathbf{x}) + r(\mathbf{y} - \mathbf{x}),$$

where

$$\lim_{\mathbf{v} \to \mathbf{0}} \frac{\|r(\mathbf{v})\|}{\|\mathbf{v}\|} = \mathbf{0}.$$

It then also follows that the matrix A is unique, and it is called the differential *or* Jacobian *of f at $\mathbf{x}$. We will denote it by $Df(\mathbf{x})$. More precisely, $Df(\mathbf{x})$ is the matrix of* partial derivatives *at the point $\mathbf{x}$,*

$$Df(\mathbf{x})_{ij} = \frac{\partial f_i}{\partial x_j}(\mathbf{x}).$$

f is called differentiable *if f is differentiable at all $\mathbf{x} \in \mathbf{dom}(f)$ (which implies that $\mathbf{dom}(f)$ is open).*

Differentiability at $\mathbf{x}$ means that in some neighborhood of $\mathbf{x}$, f is approximated by a (unique) affine function $f(\mathbf{x}) + Df(\mathbf{x})(\mathbf{y} - \mathbf{x})$, up to a sublinear error term. If $m = 1$, $Df(\mathbf{x})$ is a row vector typically denoted by $\nabla f(\mathbf{x})^\top$, where the (column) vector $\nabla f(\mathbf{x})$ is called the *gradient* of f at $\mathbf{x}$. Geometrically, this means that the graph of the affine function $f(\mathbf{x}) + \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x})$ is a *tangent hyperplane* to the graph of f at $(\mathbf{x}, f(\mathbf{x}))$; see Figure 1.1.

It also follows easily that a differentiable function is continuous, see Exercise 1.

Let us do a simple example to illustrate the concept of differentiability. Consider the function $f(x) = x^2$. We know that its derivative is $f'(x) = 2x$.

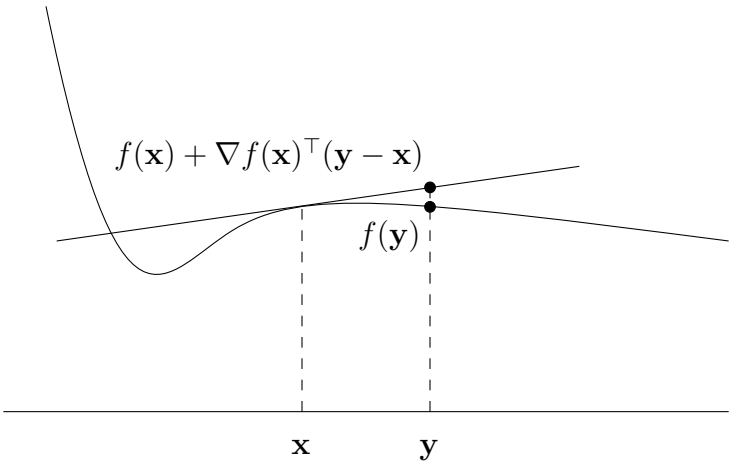


Figure 1.1: If f is differentiable at $\mathbf{x}$, the graph of f is locally (around $\mathbf{x}$) approximated by a tangent hyperplane

But why? For fixed x and $y = x + v$, we compute

$$\begin{aligned} f(y) = (x+v)^2 &= x^2 + 2vx + v^2 \\ &= f(x) + 2x \cdot v + v^2 \\ &= f(x) + A(y-x) + r(y-x), \end{aligned}$$

where $A := 2x, r(y-x) = r(v) := v^2$. We have $\lim_{v\to 0} \frac{|r(v)|}{|v|} = \lim_{v\to 0} |v| = 0$. Hence, $A = 2x$ is indeed the differential (a.k.a. derivative) of f at x.

In computing differentials, the *chain rule* is particularly useful.

Lemma 1.6 (Chain rule)**.** *Let $f : \mathbf{dom}(f) \to \mathbb{R}^m, \mathbf{dom}(f) \subseteq \mathbb{R}^d$ and $g : \mathbf{dom}(g) \to \mathbb{R}^d$. Suppose that $\mathbf{g}$ is differentiable at $\mathbf{x} \in \mathbf{dom}(g)$ and that f is differentiable at $g(\mathbf{x}) \in \mathbf{dom}(f)$. Then $f \circ g$ (the composition of f and g) is differentiable at $\mathbf{x}$, with the differential given by the matrix equation*

$$D(f \circ g)(\mathbf{x}) = Df(g(\mathbf{x}))Dg(\mathbf{x}).$$

Here is an application of the chain rule that we will use frequently. Let $f : \mathbf{dom}(f) \to \mathbb{R}^m$ be a differentiable function with (open) convex domain, and fix $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$. There is an open interval I containing $[0, 1]$ such that $\mathbf{x} + t(\mathbf{y} - \mathbf{x}) \in \mathbf{dom}(f)$ for all $t \in I$. Define $g : I \to \mathbb{R}^d$ by $g(t) =$

$\mathbf{x}+t(\mathbf{y}-\mathbf{x})$ and set $h = f \circ g$. Thus, $h : I \to \mathbb{R}^m$ with $h(t) = f(\mathbf{x}+t(\mathbf{y}-\mathbf{x}))$, and for all $t \in I$, we have

$$h'(t) = Dh(t) = Df(g(t))Dg(t) = Df(\mathbf{x} + t(\mathbf{y} - \mathbf{x}))(\mathbf{y} - \mathbf{x}). \tag{1.1}$$

1.2 Convex sets

Definition 1.7. *A set $C \subseteq \mathbb{R}^d$ is* convex *if for any two points $\mathbf{x}, \mathbf{y} \in C$, the connecting line segment is contained in C. In formulas, if for all $\lambda \in [0, 1]$, $\lambda\mathbf{x} + (1 - \lambda)\mathbf{y} \in C$; see Figure 1.2.*

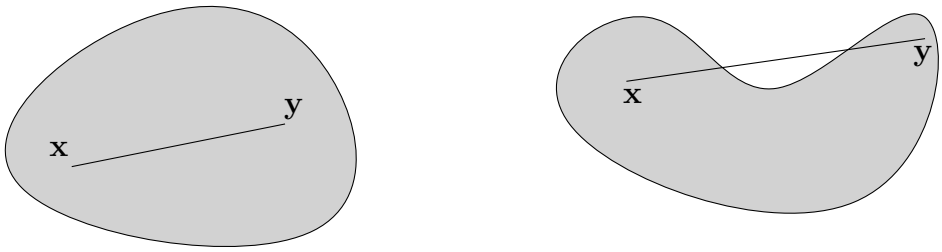


Figure 1.2: A convex set (left) and a non-convex set (right)

Observation 1.8. *Let $C_i, i \in I$ be convex sets, where I is a (possibly infinite) index set. Then $C = \bigcap_{i \in I} C_i$ is a convex set.*

For $d = 1$, convex sets are *intervals*.

1.2.1 The mean value inequality

The mean value inequality can be considered as as generalization of the mean value theorem to multivariate and vector-valued functions over convex sets (a "mean value equality" does not exist in this full generality).

To motivate it, let us consider the univariate and real-valued case first. Let $f : \mathbf{dom}(f) \to R$ be differentiable and suppose that f has bounded derivatives over an interval $X \subseteq \mathbf{dom}(f)$, meaning that for some real number B, we have $|f'(x)| \leq B$ for all $x \in X$. The mean value theorem then gives the *mean value inequality*

$$|f(y) - f(x)| = |f'(c)(y - x)| \leq B|y - x|$$

for all $x, y \in X$ and some in-between c. In other words, f is not only continuous but actually B-*Lipschitz* over X.

Vice versa, suppose that f is B-Lipschitz over a nonempty *open* interval X, then for all $c \in X$,

$$|f'(c)| = |\lim_{\delta \to 0} \frac{f(c+\delta) - f(c)}{\delta}| \leq B,$$

so f has bounded derivatives over X. Hence, over an open interval, Lipschitz functions are exactly the ones with bounded derivative. Even if the interval is not open, bounded derivatives still yield the Lipschitz property, but the other direction may fail. As a trivial example, the Lipschitz condition is always satisfied over a singleton interval $X = \{x\}$, but that does not say anything about the derivative at x. In any case, we need X to be an interval; if X has "holes", the previous arguments break down.

These considerations extend to multivariate and vector-valued functions over *convex* subsets of the domain.

Theorem 1.9. *Let $f : \mathbf{dom}(f) \to \mathbb{R}^m$ be differentiable, $X \subseteq \mathbf{dom}(f)$ a convex set, $B \in \mathbb{R}^+$. If $X \subseteq \mathbf{dom}(f)$ is nonemepty and open, the following two statements are equivalent.*

(i) f is B-Lipschitz, meaning that

$$\|f(\mathbf{x}) - f(\mathbf{y})\| \leq B \|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in X$$

(ii) f has differentials bounded by B (in spectral norm), meaning that

$$\|Df(\mathbf{x})\| \leq B, \quad \forall \mathbf{x} \in X.$$

Moreover, for every (not necessarily open) convex $X \subseteq \mathbf{dom}(f)$, (ii) implies (i), and this is the mean value inequality.

Proof. Suppose that f is B-Lipschitz over an open set X. For $\mathbf{v} \in \mathbb{R}^d$, $\mathbf{v} \to \mathbf{0}$, differentiability at $\mathbf{x} \in X$ yields for small $\mathbf{v} \in \mathbb{R}^d$ that $\mathbf{x} + \mathbf{v} \in X$ and therefore

$$B \|\mathbf{v}\| \geq \|f(\mathbf{x}+\mathbf{v}) - f(\mathbf{x})\| = \|Df(\mathbf{x})\mathbf{v} + r(\mathbf{v})\| \geq \|Df(\mathbf{x})\mathbf{v}\| - \|r(\mathbf{v})\|,$$

where $\|r(\mathbf{v})\| / \|\mathbf{v}\| \to 0$, the first inequality uses (i), and the last is the reverse triangle inequality. Rearranging and dividing by $\|\mathbf{v}\|$, we get

$$\frac{\|Df(\mathbf{x})\mathbf{v}\|}{\|\mathbf{v}\|} \leq B + \frac{\|r(\mathbf{v})\|}{\|\mathbf{v}\|}.$$

Let $\mathbf{v}^\star$ be a unit vector such that $\|Df(\mathbf{x})\| = \|Df(\mathbf{x})\mathbf{v}^\star\| / \|\mathbf{v}^\star\|$ and let $\mathbf{v} = t\mathbf{v}^\star$ for $t \to 0$. Then we further get

$$\|Df(\mathbf{x})\| \leq B + \frac{\|r(\mathbf{v})\|}{\|\mathbf{v}\|} \to B,$$

and $\|Df(\mathbf{x})\| \leq B$ follows, so differentials are bounded by B.

For the other direction, suppose that differentials are bounded by B over X (not necessarily open); we proceed as in [FM91].

For fixed $\mathbf{x}, \mathbf{y} \in X \subseteq \mathbf{dom}(f), \mathbf{x} \neq \mathbf{y}$, and $\mathbf{z} \in \mathbb{R}^m$ (to be determined later), we define

$$h(t) = \mathbf{z}^\top f(\mathbf{x} + t(\mathbf{y} - \mathbf{x}))$$

over $\mathbf{dom}(h) = [0, 1]$, in which case the chain rule yields

$$h'(t) = \mathbf{z}^\top Df(x + t(\mathbf{y} - \mathbf{x}))(\mathbf{y} - \mathbf{x}), \quad t \in (0, 1),$$

see also (1.1). Note that $\mathbf{x} + t(\mathbf{y} - \mathbf{x}) \in X$ for $t \in [0, 1]$ by convexity of X. The mean value theorem guarantees $c \in (0, 1)$ such that $h'(c) = h(1) - h(0)$. Now we compute

$$\begin{aligned}
\left\|\mathbf{z}^\top(f(\mathbf{y}) - f(\mathbf{x}))\right\| &= |h(1) - h(0)| = |h'(c)| \\
&= \mathbf{z}^\top Df(x + c(\mathbf{y} - \mathbf{x}))(\mathbf{y} - \mathbf{x}) \\
&\leq \|\mathbf{z}\|\|Df(x + c(\mathbf{y} - \mathbf{x}))(\mathbf{y} - \mathbf{x})\| \quad \text{(Cauchy-Schwarz)} \\
&\leq \|\mathbf{z}\|\|Df(x + c(\mathbf{y} - \mathbf{x}))\|\|(\mathbf{y} - \mathbf{x})\| \quad \text{(spectral norm)} \\
&\leq B\|\mathbf{z}\|\|(\mathbf{y} - \mathbf{x})\| \quad \text{(bounded differentials)}.
\end{aligned}$$

We assume w.l.o.g. that $f(\mathbf{x}) \neq f(\mathbf{x})$, as otherwise, (i) trivially holds; now we set

$$\mathbf{z} = \frac{f(\mathbf{y}) - f(\mathbf{x})}{\|f(\mathbf{y}) - f(\mathbf{x})\|.}$$

With this, the previous inequality reduces to (i), so f is indeed B-Lipschitz over X. □

1.3 Convex functions

We are considering real-valued functions $f : \mathbf{dom}(f) \to \mathbb{R}$, $\mathbf{dom}(f) \subseteq \mathbb{R}^d$.

Definition 1.10 ([BV04, 3.1.1]). *A function $f : \mathbf{dom}(f) \to \mathbb{R}$ is* convex *if (i) $\mathbf{dom}(f)$ is convex and (ii) for all $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$ and all $\lambda \in [0, 1]$, we have*

$$f(\lambda \mathbf{x} + (1 - \lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1 - \lambda) f(\mathbf{y}). \tag{1.2}$$

Geometrically, the condition means that the line segment connecting the two points $(\mathbf{x}, f(\mathbf{x})), (\mathbf{y}, f(\mathbf{y})) \in \mathbb{R}^{d+1}$ lies pointwise above the graph of f; see Figure 1.3. (Whenever we say "above", we mean "above or on".) An important special case arises when $f : \mathbb{R}^d \to \mathbb{R}$ is an affine function, i.e. $f(\mathbf{x}) = \mathbf{c}^\top \mathbf{x} + c_0$ for some vector $\mathbf{c} \in \mathbb{R}^d$ and scalar $c_0 \in \mathbb{R}$. In this case, (1.2) is always satisfied with equality, and line segments connecting points on the graph lie pointwise on the graph.

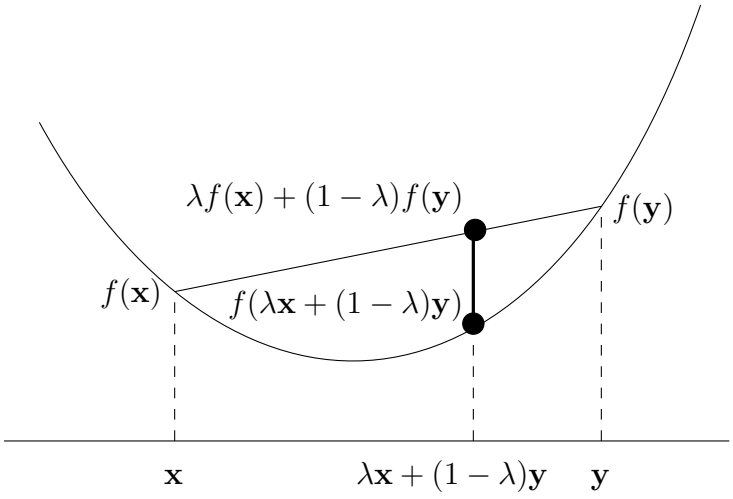


Figure 1.3: A convex function

While the graph of f is the set $\{(\mathbf{x}, f(\mathbf{x})) \in \mathbb{R}^{d+1} : \mathbf{x} \in \mathbf{dom}(f)\}$, the *epigraph* (Figure 1.4) is the set of points above the graph,

$$\mathbf{epi}(f) := \{(\mathbf{x}, \alpha) \in \mathbb{R}^{d+1} : \mathbf{x} \in \mathbf{dom}(f), \alpha \geq f(\mathbf{x})\}.$$

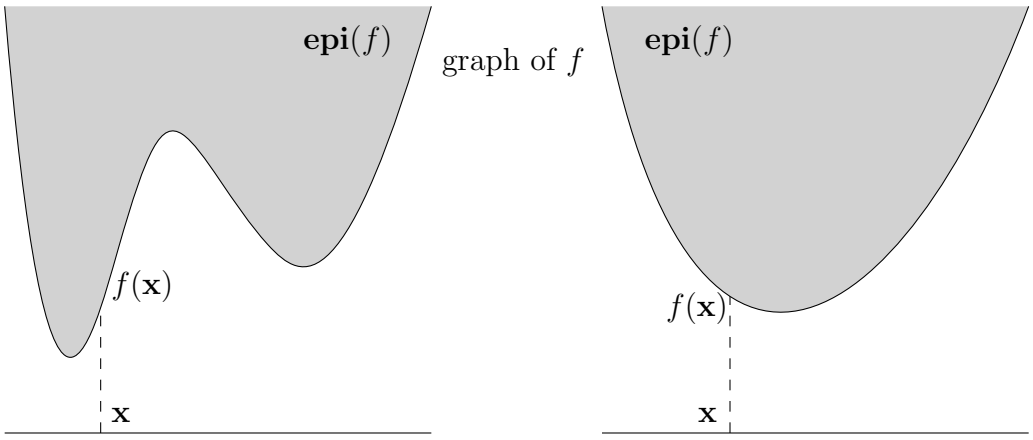


Figure 1.4: Graph and epigraph of a non-convex function (left) and a convex function (right)

Observation 1.11. *f is a convex function if and only if* $\mathbf{epi}(f)$ *is a convex set.*

Proof. This is easy but let us still do it to illustrate the concepts. Let f be a convex function and consider two points $(\mathbf{x}, \alpha), (\mathbf{y}, \beta) \in \mathbf{epi}(f)$, $\lambda \in [0, 1]$. This means, $f(\mathbf{x}) \leq \alpha, f(\mathbf{y}) \leq \beta$, hence by convexity of f,

$$f(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1-\lambda)f(\mathbf{y}) \leq \lambda\alpha + (1-\lambda)\beta.$$

Therefore, by definition of the epigraph,

$$\lambda(\mathbf{x}, \alpha) + (1-\lambda)(\mathbf{y}, \beta) = (\lambda\mathbf{x} + (1-\lambda)\mathbf{y}, \lambda\alpha + (1-\lambda)\beta) \in \mathbf{epi}(f),$$

so $\mathbf{epi}(f)$ is a convex set. In the other direction, let $\mathbf{epi}(f)$ be a convex set and consider two points $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$, $\lambda \in [0, 1]$. By convexity of $\mathbf{epi}(f)$, we have

$$\mathbf{epi}(f) \ni \lambda(\mathbf{x}, f(\mathbf{x})) + (1-\lambda)(\mathbf{y}, f(\mathbf{y})) = (\lambda\mathbf{x} + (1-\lambda)\mathbf{y}, \lambda f(\mathbf{x}) + (1-\lambda)f(\mathbf{y})),$$

and this is just a different way of writing (1.2). □

Lemma 1.12 (Jensen's inequality)**.** *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex function, $\mathbf{x}_1, \ldots, \mathbf{x}_m \in \mathbf{dom}(f)$, and $\lambda_1, \ldots, \lambda_m \in \mathbb{R}_+$ such that $\sum_{i=1}^m \lambda_i = 1$. Then*

$$f\left(\sum_{i=1}^m \lambda_i \mathbf{x}_i\right) \leq \sum_{i=1}^m \lambda_i f(\mathbf{x}_i).$$

For $m = 2$, this is (1.2). The proof of the general case is Exercise 2.

Lemma 1.13. *Let f be convex and suppose that $\mathbf{dom}(f)$ is open. Then f is continuous.*

This is not entirely obvious (see Exercise 3) and really needs $\mathbf{dom}(f) \subseteq \mathbb{R}^d$. It becomes false if we consider convex functions over vector spaces of infinite dimension. In fact, in this case, even linear functions (which are in particular convex) may fail to be continuous.

Lemma 1.14. *There exists an (infinite dimensional) vector space V and a linear function $f : V \to \mathbb{R}$ such that f is discontinuous at all $\mathbf{v} \in V$.*

Proof. This is a classical example. Let us consider the vector space V of all univariate polynomials; the vector space operations are addition of two polynomials, and multiplication of a polynomial with a scalar. We consider a polynomial such as $3x^5 + 2x^2 + 1$ as a function $x \mapsto 3x^5 + 2x^2 + 1$ over the domain $[-1, 1]$.

The standard norm in a function space such as V is the *supremum norm* $\|\cdot\|_\infty$, defined for any bounded function $h : [-1, 1] \to \mathbb{R}$ via $\|h\|_\infty := \sup_{x\in[-1,1]} |h(x)|$. Polynomials are continuous and as such bounded over $[-1, 1]$.

We now consider the linear function $f : V \to \mathbb{R}$ defined by $f(p) = p'(0)$, the derivative of p at 0. The function f is linear, simply because the derivative is a linear operator. As $\mathbf{dom}(f)$ is the whole space V, $\mathbf{dom}(f)$ is open. We claim that f is discontinuous at 0 (the zero polynomial). Since f is linear, this implies discontinuity at every polynomial $p \in V$. To prove discontinuity at 0, we first observe that $f(0) = 0$ and then show that there are polynomials p of arbitrarily small supremum norm with $f(p) = 1$. Indeed, for $n, k \in \mathbb{N}, n > 0$, consider the polynomial

$$p_{n,k}(x) = \frac{1}{n}\sum_{i=0}^{k}(-1)^i\frac{(nx)^{2i+1}}{(2i+1)!} = \frac{1}{n}\left(nx - \frac{(nx)^3}{3!} + \frac{(nx)^5}{5!} - \cdots \pm \frac{(nx)^{2k+1}}{(2k+1)!}\right)$$

which—for any fixed n and sufficiently large k—approximates the function

$$s_n(x) = \frac{1}{n}\sin(nx) = \frac{1}{n}\sum_{i=0}^{\infty}(-1)^i\frac{(nx)^{2i+1}}{(2i+1)!}$$

up to any desired precision over the whole interval $[-1, 1]$ (Taylor's theorem with remainder). In formulas, $\|p_{n,k} - s_n\|_\infty \to 0$ as $k \to \infty$. Moreover, $\|s_n\|_\infty \to 0$ as $n \to \infty$. Using the triangle inequality, this implies that $\|p_{n,k}\| \to 0$ as $n, k \to \infty$. On the other hand, $f(p_{n,k}) = p'_{n,k}(0) = 1$ for all n, k. □

1.3.1 First-order characterization of convexity

As an example of a convex function, let us consider $f(x_1, x_2) = x_1^2 + x_2^2$. The graph of f is the *unit paraboloid* in $\mathbb{R}^3$ which looks convex. However, to verify (1.2) directly is somewhat cumbersome. Next, we develop better ways to do this if the function under consideration is differentiable.

Lemma 1.15 ([BV04, 3.1.3]). *Suppose that* $\mathbf{dom}(f)$ *is open and that* f *is differentiable; in particular, the* gradient *(vector of partial derivatives)*

$$\nabla f(\mathbf{x}) := \left(\frac{\partial f}{\partial x_1}(\mathbf{x}), \ldots, \frac{\partial f}{\partial x_d}(\mathbf{x}) \right)$$

exists at every point $\mathbf{x} \in \mathbf{dom}(f)$. *Then* f *is convex if and only if* $\mathbf{dom}(f)$ *is convex and*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) \tag{1.3}$$

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

Geometrically, this means that for all $\mathbf{x} \in \mathbf{dom}(f)$, the graph of f lies above its tangent hyperplane at the point $(\mathbf{x}, f(\mathbf{x}))$; see Figure 1.5.

Proof. Suppose that f is convex, meaning that for $t \in (0, 1)$,

$$f(\mathbf{x}+t(\mathbf{y}-\mathbf{x})) = f((1-t)\mathbf{x}+t\mathbf{y}) \leq (1-t)f(\mathbf{x})+tf(\mathbf{y}) = f(\mathbf{x})+t(f(\mathbf{y})-f(\mathbf{x})).$$

Dividing by t and using differentiability at $\mathbf{x}$, we get

$$\begin{aligned} f(\mathbf{y}) &\geq f(\mathbf{x}) + \frac{f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) - f(\mathbf{x})}{t} \\ &= f(\mathbf{x}) + \frac{\nabla f(\mathbf{x})^\top t(\mathbf{y} - \mathbf{x}) + r(t(\mathbf{y} - \mathbf{x}))}{t} \\ &= f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{r(t(\mathbf{y} - \mathbf{x}))}{t}, \end{aligned}$$

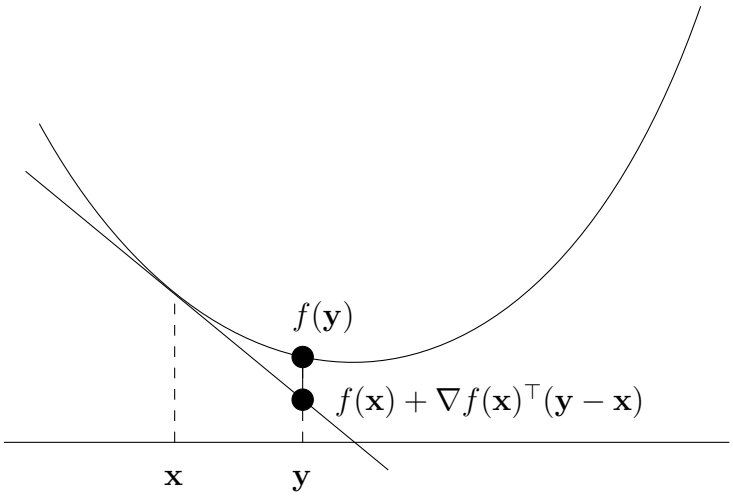


Figure 1.5: First-order characterization of convexity

where the error term $r(t(\mathbf{y} - \mathbf{x}))/t$ goes to 0 as $t \to 0$. The inequality $f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x})$ follows.

Now suppose this inequality holds for all $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$, let $\lambda \in [0, 1]$, and define $\mathbf{z} := \lambda\mathbf{x} + (1 - \lambda)\mathbf{y} \in \mathbf{dom}(f)$ (by convexity of $\mathbf{dom}(f)$). Then we have

$$\begin{aligned} f(\mathbf{x}) &\geq f(\mathbf{z}) + \nabla f(\mathbf{z})^\top(\mathbf{x} - \mathbf{z}), \\ f(\mathbf{y}) &\geq f(\mathbf{z}) + \nabla f(\mathbf{z})^\top(\mathbf{y} - \mathbf{z}). \end{aligned}$$

After multiplying the first inequality by λ and the second one by $(1 - \lambda)$, the gradient terms cancel in the sum of the two inequalities, and we get

$$\lambda f(\mathbf{x}) + (1 - \lambda) f(\mathbf{y}) \geq f(\mathbf{z}) = f(\lambda\mathbf{x} + (1 - \lambda)\mathbf{y}).$$

This is convexity. □

For $f(x_1, x_2) = x_1^2 + x_2^2$, we have $\nabla f(\mathbf{x}) = (2x_1, 2x_2)$, hence (1.3) boils down to

$$y_1^2 + y_2^2 \geq x_1^2 + x_2^2 + 2x_1(y_1 - x_1) + 2x_2(y_2 - x_2),$$

which after some rearranging of terms is equivalent to

$$(y_1 - x_1)^2 + (y_2 - x_2)^2 \geq 0,$$

hence true. There are relevant convex functions that are not differentiable, see Figure 1.6 for an example. More generally, Exercise 8 asks you to prove that the ℓ_1-norm (or 1-norm) $f(\mathbf{x}) = \|\mathbf{x}\|_1$ is convex.

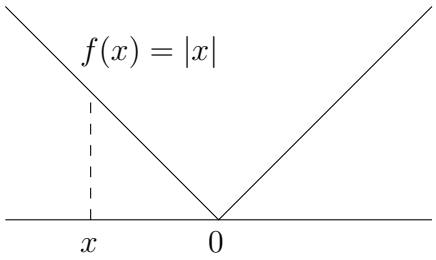


Figure 1.6: A non-differentiable convex function

There is another useful and less standard first-order characterization of convexity that we can easily derive from the standard one above.

Lemma 1.16. *Suppose that* $\mathbf{dom}(f)$ *is open and that* f *is differentiable. Then* f *is convex if and only if* $\mathbf{dom}(f)$ *is convex and*

$$(\nabla f(\mathbf{y}) - \nabla f(\mathbf{x}))^\top (\mathbf{y} - \mathbf{x}) \geq 0 \tag{1.4}$$

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

The inequality (1.4) is known as *monotonicity of the gradient*.

Proof. If f is convex, the first-order characterization in Lemma 1.15 yields

$$\begin{aligned} f(\mathbf{y}) &\geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}), \\ f(\mathbf{x}) &\geq f(\mathbf{y}) + \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}), \end{aligned}$$

for all $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$. After adding up these two inequalities, $f(\mathbf{x}) + f(\mathbf{y})$ appears on both sides and hence cancels, so that we get

$$0 \geq \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) = (\nabla f(\mathbf{y}) - \nabla f(\mathbf{x}))^\top (\mathbf{x} - \mathbf{y}).$$

Multiplying this by -1 yields (1.4).

For the other direction, suppose that monotonicty of the gradient (1.4) holds. Then we in particular have

$$(\nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x}))^\top (t(\mathbf{y} - \mathbf{x})) \geq 0$$

for all $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$ and $t \in (0,1)$. Dividing by t, this yields

$$(\nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x}))^\top (\mathbf{y} - \mathbf{x})) \geq 0. \tag{1.5}$$

Fix $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$. For $t \in [0,1]$, let $h(t) := f(\mathbf{x} + t(\mathbf{y} - \mathbf{x}))$. In our case where f is real-valued, (1.1) yields $h'(t) = \nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x}))^\top (\mathbf{y} - \mathbf{x}), t \in (0,1)$. Hence, (1.5) can be rewritten as

$$h'(t) \geq \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}), \quad t \in (0,1).$$

By the mean value theorem, there is $c \in (0,1)$ such that $h'(c) = h(1) - h(0)$. Then

$$\begin{aligned} f(\mathbf{y}) = h(1) = h(0) + h'(c) &= f(\mathbf{x}) + h'(c) \\ &\geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}). \end{aligned}$$

This is the first-order characterization of convexity (Lemma 1.15). □

1.3.2 Second-order characterization of convexity

If $f : \mathbf{dom}(f) \to \mathbb{R}$ is twice differentiable (meaning that f is differentiable and the gradient function ∇f is also differentiable), convexity can be characterized as follows.

Lemma 1.17. *Suppose that $\mathbf{dom}(f)$ is open and that f is twice differentiable; in particular, the* Hessian *(matrix of second partial derivatives)*

$$\nabla^2 f(\mathbf{x}) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1}(\mathbf{x}) & \frac{\partial^2 f}{\partial x_1 \partial x_2}(\mathbf{x}) & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_d}(\mathbf{x}) \\ \frac{\partial^2 f}{\partial x_2 \partial x_1}(\mathbf{x}) & \frac{\partial^2 f}{\partial x_2 \partial x_2}(\mathbf{x}) & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_d}(\mathbf{x}) \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial^2 f}{\partial x_d \partial x_1}(\mathbf{x}) & \frac{\partial^2 f}{\partial x_d \partial x_2}(\mathbf{x}) & \cdots & \frac{\partial^2 f}{\partial x_d \partial x_d}(\mathbf{x}) \end{pmatrix}$$

exists at every point $\mathbf{x} \in \mathbf{dom}(f)$ and is symmetric. Then f is convex if and only if $\mathbf{dom}(f)$ is convex, and for all $\mathbf{x} \in \mathbf{dom}(f)$, we have

$$\nabla^2 f(\mathbf{x}) \succeq 0 \quad \textit{(i.e. } \nabla^2 f(\mathbf{x}) \textit{ is positive semidefinite).} \tag{1.6}$$

(A symmetric matrix M is positive semidefinite, *denoted by $M \succeq \mathbf{0}$, if $\mathbf{x}^\top M \mathbf{x} \geq 0$ for all $\mathbf{x}$, and* positive definite, *denoted by $M \succ \mathbf{0}$, if $\mathbf{x}^\top M \mathbf{x} > 0$ for all $\mathbf{x} \neq \mathbf{0}$.)*

The fact that the Hessians of a twice *continuously* differentiable function are symmetric is a classical result known as the Schwarz theorem [AE08, Corollary 5.5]. But symmetry in fact already holds if f is twice differentiable [Die69, (8.12.3)]. However, if f is only twice *partially* differentiable, we may get non-symmetric Hessians [AE08, Remark 5.6].

Proof. Once again, we employ our favorite univariate function $h(t) := f(\mathbf{x} + t(\mathbf{y} - \mathbf{x}))$, for fixed $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$ and $t \in I$ where $I \supset [0, 1]$ is a suitable open interval. But this time, we also need h's second derivative. For $t \in I$, $\mathbf{v} := \mathbf{y} - \mathbf{x}$, we have

$$\begin{aligned} h'(t) &= \nabla f(\mathbf{x} + t\mathbf{v})^\top \mathbf{v}, \\ h''(t) &= \mathbf{v}^\top \nabla^2 f(\mathbf{x} + t\mathbf{v})\mathbf{v}. \end{aligned}$$

The formula for $h'(t)$ has already been derived in the proof of Lemma 1.16, and the formula for $h''(t)$ is Exercise 9.

If f is convex, we always have $h''(0) \geq 0$, as we will show next. Given this, $\nabla^2 f(\mathbf{x}) \succeq 0$ follows for every $\mathbf{x} \in \mathbf{dom}(f)$: by openness of $\mathbf{dom}(f)$, for every $\mathbf{v} \in \mathbb{R}^d$ of sufficiently small norm, there is $\mathbf{y} \in \mathbf{dom}(f)$ such that $\mathbf{v} = \mathbf{y} - \mathbf{x}$, and then $\mathbf{v}^\top \nabla^2 f(\mathbf{x})\mathbf{v} = h''(0) \geq 0$. By scaling, this inequality extends to all $\mathbf{v} \in \mathbb{R}^d$.

To show $h''(0) \geq 0$, we observe that for all sufficiently small δ, $\mathbf{x} + \delta\mathbf{v} \in \mathbf{dom}(f)$ and hence

$$\frac{h'(\delta) - h'(0)}{\delta} = \frac{(\nabla f(\mathbf{x} + \delta\mathbf{v}) - \nabla f(\mathbf{x}))^\top \mathbf{v}}{\delta} = \frac{(\nabla f(\mathbf{x} + \delta\mathbf{v}) - \nabla f(\mathbf{x}))^\top \delta\mathbf{v}}{\delta^2} \geq 0,$$

by monotonicity of the gradient for convex f (Lemma 1.16). It follows that $h''(0) = \lim_{\delta \to 0}(h'(\delta) - h'(0))/\delta \geq 0$.

For the other direction, the mean value theorem applied to h' yields $c \in (0, 1)$ such that $h'(1) - h'(0) = h''(c)$, and spelled out, this is

$$\nabla f(\mathbf{y})^\top \mathbf{v} - \nabla f(\mathbf{x})^\top \mathbf{v} = \mathbf{v}^\top \nabla^2 f(\mathbf{x} + c\mathbf{v})\mathbf{v} \geq 0, \tag{1.7}$$

since $\nabla^2 f(\mathbf{z}) \succeq 0$ for all $\mathbf{z} \in \mathbf{dom}(f)$. Hence, we have proved monotonicity of the gradient which by Lemma 1.16 implies convexity of f. □

Geometrically, Lemma 1.17 means that the graph of f has non-negative curvature everywhere and hence "looks like a bowl". For $f(x_1, x_2) = x_1^2 + x_2^2$, we have

$$\nabla^2 f(\mathbf{x}) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix},$$

which is a positive definite matrix. In higher dimensions, the same argument can be used to show that the squared distance $d_{\mathbf{y}}(\mathbf{x}) = \|\mathbf{x} - \mathbf{y}\|^2$ to a fixed point $\mathbf{y}$ is a convex function; see Exercise 4. The non-squared Euclidean distance $\|\mathbf{x} - \mathbf{y}\|$ is also convex in $\mathbf{x}$, as a consequence of Lemma 1.18(ii) below and the fact that every seminorm (in particular the Euclidean norm $\|x\|$) is convex (Exercise 10). The squared Euclidean distance has the advantage that it is differentiable, while the Euclidean distance itself (whose graph is an "ice cream cone" for $d = 2$) is not.

1.3.3 Operations that preserve convexity

There are two important operations that preserve convexity.

Lemma 1.18 (Exercise 5)**.**

(i) Let $f_1, f_2, \ldots, f_m$ be convex functions, $\lambda_1, \lambda_2, \ldots, \lambda_m \in \mathbb{R}_+$. Then $f := \sum_{i=1}^m \lambda_i f_i$ is convex on $\mathbf{dom}(f) := \bigcap_{i=1}^m \mathbf{dom}(f_i)$.

(ii) Let f be a convex function with $\mathbf{dom}(f) \subseteq \mathbb{R}^d$, $g : \mathbb{R}^m \to \mathbb{R}^d$ an affine function, meaning that $g(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$, for some matrix $A \in \mathbb{R}^{d \times m}$ and some vector $\mathbf{b} \in \mathbb{R}^d$. Then the function $f \circ g$ (that maps $\mathbf{x}$ to $f(A\mathbf{x} + \mathbf{b})$) is convex on $\mathbf{dom}(f \circ g) := \{\mathbf{x} \in \mathbb{R}^m : g(\mathbf{x}) \in \mathbf{dom}(f)\}$.

1.4 Minimizing convex functions

The main feature that makes convex functions attractive in optimization is that every local minimum is a global one, so we cannot "get stuck" in local optima. This is quite intuitive if we think of the graph of a convex function as being bowl-shaped.

Definition 1.19. *A local minimum of $f : \mathbf{dom}(f) \to \mathbb{R}$ is a point $\mathbf{x}$ such that there exists $\varepsilon > 0$ with*

$$f(\mathbf{x}) \leq f(\mathbf{y}) \quad \forall \mathbf{y} \in \mathbf{dom}(f) \text{ satisfying } \|\mathbf{y} - \mathbf{x}\| < \varepsilon.$$

Lemma 1.20. *Let $\mathbf{x}^\star$ be a local minimum of a convex function $f : \mathbf{dom}(f) \to \mathbb{R}$. Then $\mathbf{x}^\star$ is a global minimum, meaning that*

$$f(\mathbf{x}^\star) \leq f(\mathbf{y}) \quad \forall \mathbf{y} \in \mathbf{dom}(f).$$

Proof. Suppose there exists $\mathbf{y} \in \mathbf{dom}(f)$ such that $f(\mathbf{y}) < f(\mathbf{x}^\star)$ and define $\mathbf{y}' := \lambda \mathbf{x}^\star + (1-\lambda)\mathbf{y}$ for $\lambda \in (0,1)$. From convexity (1.2), we get that that $f(\mathbf{y}') < f(\mathbf{x}^\star)$. Choosing λ so close to 1 that $\|\mathbf{y}' - \mathbf{x}^\star\| < \varepsilon$ yields a contradiction to $\mathbf{x}^\star$ being a local minimum. □

This does not mean that a convex function always has a global minimum. Think of $f(x) = x$ as a trivial example. But also if f is bounded from below over $\mathbf{dom}(f)$, it may fail to have a global minimum ($f(x) = e^x$). To ensure the existence of a global minimum, we need additional conditions. For example, it suffices if outside some ball B, all function values are larger than some value $f(\mathbf{x}), \mathbf{x} \in B$. In this case, we can restrict f to B, without changing the smallest attainable value. And on B (which is compact), f attains a minimum by continuity (Lemma 1.13). An easy example: for $f(x_1, x_2) = x_1^2 + x_2^2$, we know that outside any ball containing $\mathbf{0}$, $f(\mathbf{x}) > f(\mathbf{0}) = 0$.

Another easy condition in the differentiable case is given by the following result.

Lemma 1.21. *Suppose that $f : \mathbf{dom}(f) \to \mathbb{R}$ is convex and differentiable over an open domain $\mathbf{dom}(f) \subseteq \mathbb{R}^d$. Let $\mathbf{x} \in \mathbf{dom}(f)$. If $\nabla f(\mathbf{x}) = \mathbf{0}$, then $\mathbf{x}$ is a global minimum.*

Proof. Suppose that $\nabla f(\mathbf{x}) = \mathbf{0}$. According to Lemma 1.15, we have

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) = f(\mathbf{x})$$

for all $\mathbf{y} \in \mathbf{dom}(f)$, so $\mathbf{x}$ is a global minimum. □

The converse is also true and is a corollary of Lemma 1.27 [BV04, 4.2.3].

Lemma 1.22. *Suppose that $f : \mathbf{dom}(f) \to \mathbb{R}$ is convex and differentiable over an open domain $\mathbf{dom}(f) \subseteq \mathbb{R}^d$. Let $\mathbf{x} \in \mathbf{dom}(f)$. If $\mathbf{x}$ is a global minimum then $\nabla f(\mathbf{x}) = \mathbf{0}$.*

1.4.1 Strictly convex functions

In general, a global minimum of a convex function is not unique (think of $f(x) = 0$ as a trivial example). However, if we forbid "flat" parts of the graph of f, a global minimum becomes unique (if it exists at all).

Definition 1.23 ([BV04, 3.1.1]). *A function $f : \mathbf{dom}(f) \to \mathbb{R}$ is* strictly convex *if (i) $\mathbf{dom}(f)$ is convex and (ii) for all $\mathbf{x} \neq \mathbf{y} \in \mathbf{dom}(f)$ and all $\lambda \in (0,1)$, we have*

$$f(\lambda \mathbf{x} + (1-\lambda)\mathbf{y}) < \lambda f(\mathbf{x}) + (1-\lambda) f(\mathbf{y}). \tag{1.8}$$

This means that the open line segment connecting $(\mathbf{x}, f(\mathbf{x}))$ and $(\mathbf{y}, f(\mathbf{y}))$ is pointwise *strictly* above the graph of f. For example, $f(x) = x^2$ is strictly convex.

Lemma 1.24 ([BV04, 3.1.4]). *Suppose that $\mathbf{dom}(f)$ is open and that f is twice continuously differentiable. If the Hessian $\nabla^2 f(\mathbf{x}) \succ \mathbf{0}$ for every $x \in \mathbf{dom}(f)$ (i.e., $\mathbf{z}^\top \nabla^2 f(\mathbf{x}) \mathbf{z} > 0$ for any $\mathbf{z} \neq \mathbf{0}$), then f is strictly convex.*

The converse is false, though: $f(x) = x^4$ is strictly convex but has vanishing second derivative at $x = 0$.

Lemma 1.25. *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be strictly convex. Then f has at most one global minimum.*

Proof. Suppose $\mathbf{x}^\star \neq \mathbf{y}^\star$ are two global minima with $f_{\min} = f(\mathbf{x}^\star) = f(\mathbf{y}^\star)$, and let $\mathbf{z} = \frac{1}{2}\mathbf{x}^\star + \frac{1}{2}\mathbf{y}^\star$. By (1.8),

$$f(\mathbf{z}) < \frac{1}{2} f_{\min} + \frac{1}{2} f_{\min} = f_{\min},$$

a contradiction to $\mathbf{x}^\star$ and $\mathbf{y}^\star$ being global minima. □

1.4.2 Example: Least squares

Suppose we want to fit a hyperplane to a set of data points $\mathbf{x}_1, \ldots, \mathbf{x}_m$ in $\mathbb{R}^d$, based on the hypothesis that the points actually come (approximately) from a hyperplane. A classical method for this is *least squares*. For concreteness, let us do this in $\mathbb{R}^2$. Suppose that the data points are

$$(1,10), (2,11), (3,11), (4,10), (5,9), (6,10), (7,9), (8,10),$$

Figure 1.7 (left).

Also, for simplicity (and quite appropriately in this case), let us restrict to fitting a linear model, of more formally to fit non-vertical lines of the

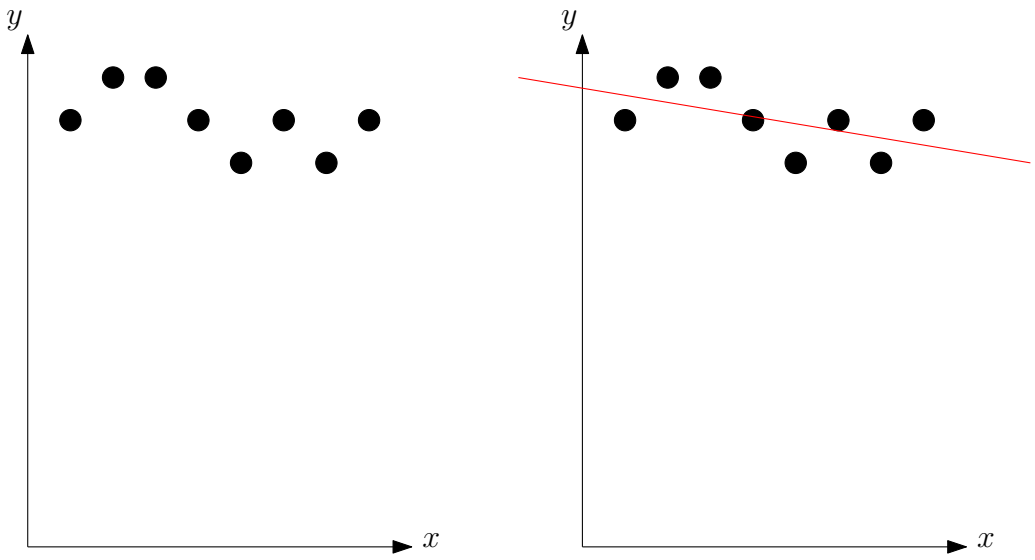


Figure 1.7: Data points in $\mathbb{R}^2$ (left) and least-squares fit (right)

form $y = w_0 + w_1 x$. If (x_i, y_i) is the i-th data point, the least squares fit chooses w_0, w_1 such that the *least squares objective*

$$f(w_0, w_1) = \sum_{i=1}^{8} (w_1 x_i + w_0 - y_i)^2$$

is minimized. It easily follows from Lemma 1.18 that f is convex. In fact,

$$f(w_0, w_1) = 204w_1^2 + 72w_1 w_0 - 706w_1 + 8w_0^2 - 160w_0 + 804, \tag{1.9}$$

so we can check convexity directly using the second order condition. We have gradient

$$\nabla f(w_0, w_1) = (72w_1 + 16w_0 - 160, 408w_1 + 72w_0 - 706)$$

and Hessian

$$\nabla^2(w_0, w_1) = \begin{pmatrix} 16 & 72 \\ 72 & 408 \end{pmatrix}.$$

A 2×2 matrix is positive semidefinite if the diagonal elements and the determinant are positive, which is the case here, so f is actually strictly

convex and has a unique global minimum. To find it, we solve the linear system $\nabla f(w_0, w_1) = (0, 0)$ of two equations in two unknowns and obtain the global minimum

$$(w_0^\star, w_1^\star) = \Big(\frac{43}{4}, -\frac{1}{6}\Big).$$

Hence, the "optimal" line is

$$y = -\frac{1}{6}x + \frac{43}{4},$$

see Figure 1.7 (right).

1.4.3 Constrained Minimization

Frequently, we are interested in minimizing a convex function only over a subset X of its domain.

Definition 1.26. *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be convex and let $X \subseteq \mathbf{dom}(f)$ be a convex set. A point $\mathbf{x} \in X$ is a* minimizer *of f over X if*

$$f(\mathbf{x}) \leq f(\mathbf{y}) \quad \forall \mathbf{y} \in X.$$

If f is differentiable, minimizers of f over X have a very useful characterization.

Lemma 1.27 ([BV04, 4.2.3])**.** *Suppose that $f : \mathbf{dom}(f) \to \mathbb{R}$ is convex and differentiable over an open domain $\mathbf{dom}(f) \subseteq \mathbb{R}^d$, and let $X \subseteq \mathbf{dom}(f)$ be a convex set. Point $\mathbf{x}^\star \in X$ is a minimizer of f over X if and only if*

$$\nabla f(\mathbf{x}^\star)^\top (\mathbf{x} - \mathbf{x}^\star) \geq 0 \quad \forall \mathbf{x} \in X.$$

Applying the this result with $X = \mathbf{dom}(f)$, we recover Lemma 1.21, and because $\mathbf{dom}(f)$ is open, its converse Lemma 1.22 follows [BV04, 4.2.3]. If X does not contain the global minimum, then Lemma 1.27 has a nice geometric interpretation. Namely, it means that X is contained in the halfspace $\{\mathbf{x} \in \mathbb{R}^d : \nabla f(\mathbf{x}^\star)^\top (\mathbf{x} - \mathbf{x}^\star) \geq 0\}$ (normal vector $\nabla f(\mathbf{x}^\star)$ pointing into the halfspace); see Figure 1.8. In still other words, $\mathbf{x} - \mathbf{x}^\star$ forms a non-obtuse angle with $\nabla f(\mathbf{x}^\star)$ for all $\mathbf{x} \in X$.

We typically write constrained minimization problems in the form

$$\operatorname{argmin}\{f(\mathbf{x}) : \mathbf{x} \in X\} \tag{1.10}$$

or

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{x} \in X. \end{array} \tag{1.11}$$

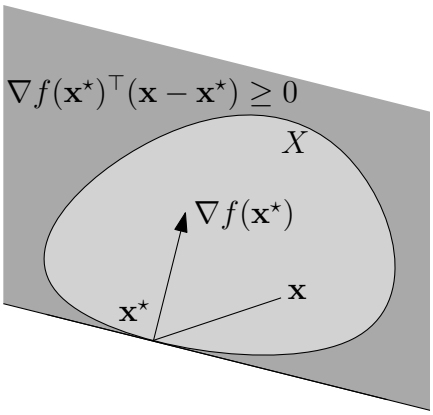


Figure 1.8: Optimality condition for constrained optimization

1.5 Existence of a minimizer

The existence of a minimizer (or a global minimum if $X = \mathbf{dom}(f)$) will be an assumption made by most minimization algorithms that we discuss later. In practice, such algorithms are being used (and often also work) if there is no minimizer. By "work", we mean in this case that they compute a point $\mathbf{x}$ such that $f(\mathbf{x})$ is close to $\inf_{\mathbf{y}\in X} f(\mathbf{y})$, assuming that the infimum is finite (as in $f(x) = e^x$). But a sound theoretical analysis usually requires the existence of a minimizer. Therefore, this section develops tools that may helps us in analyzing whether this is the case for a given convex function. To avoid technicalities, we restrict ourselves to the case $\mathbf{dom}(f) = \mathbb{R}^d$.

1.5.1 Sublevel sets and the Weierstrass Theorem

Definition 1.28. *Let $f : \mathbb{R}^d \to \mathbb{R}$, $\alpha \in \mathbb{R}$. The set*

$$f^{\leq\alpha} := \{\mathbf{x} \in \mathbb{R}^d : f(\mathbf{x}) \leq \alpha\}$$

is the α-sublevel set of f; see Figure 1.9

It is easy to see from the definition that every sublevel set of a convex function is convex. Moreover, as a consequence of continuity of f, sublevel sets are closed. The following (known as the Weierstrass Theorem) just formalizes an argument that we have made earlier.

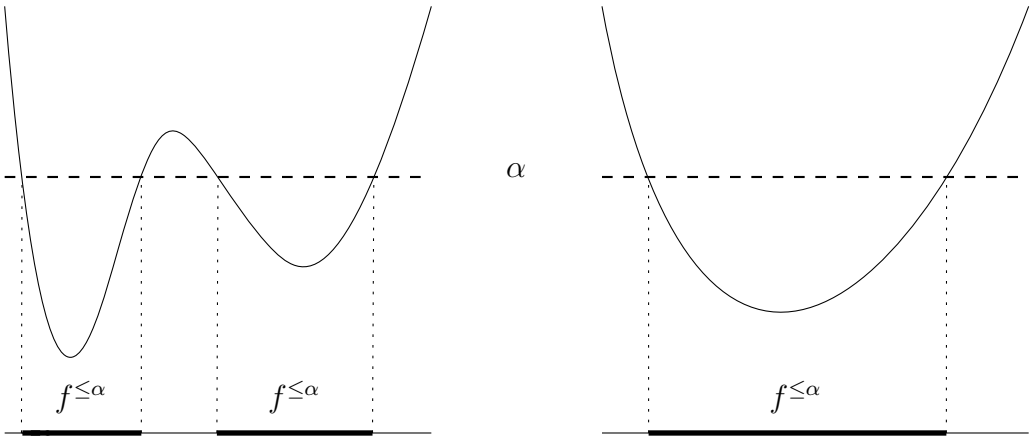


Figure 1.9: Sublevel set of a non-convex function (left) and a convex function (right)

Theorem 1.29. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex function, and suppose there is a nonempty and bounded sublevel set $f^{\leq\alpha}$. Then f has a global minimum.*

Proof. We know that f—as a continuous function—attains a minimum over the closed and bounded (= compact) set $f^{\leq\alpha}$ at some $\mathbf{x}^\star$. This $\mathbf{x}^\star$ is also a global minimum as it has value $f(\mathbf{x}^\star) \leq \alpha$, while any $\mathbf{x} \notin f^{\leq\alpha}$ has value $f(\mathbf{x}) > \alpha \geq f(\mathbf{x}^\star)$. □

1.6 Examples

In the following two sections, we give two examples of convex function minimization tasks that arise from machine learning applications.

1.6.1 Handwritten digit recognition

Suppose you want to write a program that recognizes handwritten decimal digits $0, 1, \ldots, 9$. You have a set P of grayscale images (28×28 pixels, say) that represent handwritten decimal digits, and for each image $\mathbf{x} \in P$, you know the digit $d(\mathbf{x}) \in \{0, \ldots, 9\}$ that it represents, see Figure 1.10. You want to train your program with the set P, and after that, use it to recognize handwritten digits in arbitrary 28×28 images.

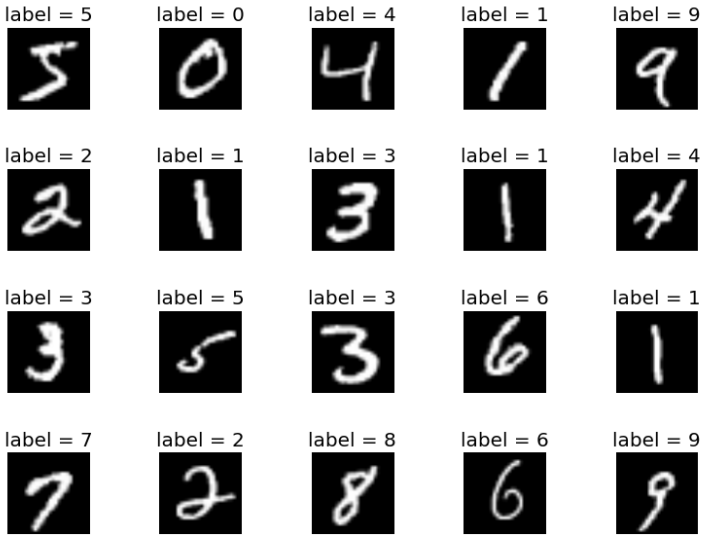


Figure 1.10: Some training images from the MNIST data set (picture from http://corochann.com/mnist-dataset-introduction-1138.html

The classical approach is the following. We represent an image as a *feature vector* $\mathbf{x} \in \mathbb{R}^{784}$, where x_i is the gray value of the i-th pixel (in some order). During the training phase, we compute a matrix $W \in \mathbb{R}^{10\times 784}$ and then use the vector $\mathbf{y} = W\mathbf{x} \in \mathbb{R}^{10}$ to predict the digit seen in an arbitrary image $\mathbf{x}$. The idea is that $y_j, j = 0, \ldots, 9$ corresponds to the probability of the digit being j. This does not work directly, since the entries of $\mathbf{y}$ may be negative and generally do not sum up to 1. But we can convert $\mathbf{y}$ to a vector $\mathbf{z}$ of actual probabilities, such that a small y_j leads to a small probability z_j and a large y_j to a large probability z_j. How to do this is not canonical, but here is a well-known formula that works:

$$z_j = z_j(\mathbf{y}) = \frac{e^{y_j}}{\sum_{k=0}^{9} e^{y_k}}. \tag{1.12}$$

The classification then simply outputs digit j with probability z_j. The matrix W is chosen such that it (approximately) minimizes the classification error on the training set P. Again, it is not canonical how we measure

classification error; here we use the following *loss function* to evaluate the error induced by a given matrix W.

$$\ell(W) = -\sum_{\mathbf{x}\in P} \ln\left(z_{d(\mathbf{x})}(W\mathbf{x})\right) = \sum_{\mathbf{x}\in P}\left(\ln\left(\sum_{k=0}^{9} e^{(W\mathbf{x})_k}\right) - (W\mathbf{x})_{d(\mathbf{x})}\right). \quad (1.13)$$

This function "punishes" images for which the correct digit j has low probability z_j (corresponding to a significantly negative value of $\log z_j$). In an ideal world, the correct digit would always have probability 1, resulting in $\ell(W) = 0$. But under (1.12), probabilities are always strictly between 0 and 1, so we have $\ell(W) > 0$ for all W.

Exercise 6 asks you to prove that ℓ is convex. In Exercise 7, you will characterize the situations in which ℓ has a global minimum.

1.6.2 Master's Admission

The computer science department of a well known Swiss university is admitting top international students to its MSc program, in a competitive application process. Applicants are submitting various documents (GPA, TOEFL test score, GRE test scores, reference letters,...). During the evaluation of an application, the admission committee would like to compute a (rough) forecast of the applicant's performance in the MSc program, based on the submitted documents.[1]

Data on the actual performance of students admitted in the past is available. To keep things simple in the following example, Let us base the forecast on GPA (grade point average) and TOEFL (Test of English as a Foreign Language) only. GPA scores are normalized to a scale with a minimum of 0.0 and a maximum of 4.0, where admission starts from 3.5. TOEFL scores are on an integer scale between 0 and 120, where admission starts from 100.

Table 1.1 contains the known data. GGPA (graduation grade point average on a Swiss grading scale) is the average grade obtained by an admitted student over all courses in the MSc program. The Swiss scale goes from 1 to 6 where 1 is the lowest grade, 6 is the highest, and 4 is the lowest passing grade.

[1] Any resemblance to real departments is purely coincidental. Also, no serious department will base performance forecasts on data from 10 students, as we will do it here.

GPA	TOEFL	GGPA
3.52	100	3.92
3.66	109	4.34
3.76	113	4.80
3.74	100	4.67
3.93	100	5.52
3.88	115	5.44
3.77	115	5.04
3.66	107	4.73
3.87	106	5.03
3.84	107	5.06

Table 1.1: Data for 10 admitted students: GPA and TOEFL scores (at time of application), GGPA (at time of graduation)

As in Section 1.4.2, we are attempting a linear regression with least squares fit, i.e. we are making the hypothesis that

$$\text{GGPA} \approx w_0 + w_1 \cdot \text{GPA} + w_2 \cdot \text{TOEFL}. \tag{1.14}$$

However, in our scenario, the relevant GPA scores span a range of only 0.5 while the relevant TOEFL scores span a range of 20. The resulting least squares objective would be somewhat ugly; we already saw this in our previous example (1.9), where the data points had large second coordinate, resulting in the w_1-scale being very different from the w_2-scale. This time, we normalize first, so that w_1 und w_2 become comparable and allow us to understand the relative influences of GPA and TOEFL.

The general setting is this: we have n *inputs* $\mathbf{x}_1, \ldots, \mathbf{x}_n$, where each vector $\mathbf{x}_i \in \mathbb{R}^d$ consists of d input variables; then we have n *outputs* $y_1, \ldots, y_n \in \mathbb{R}$. Each pair $(\mathbf{x}_i, y_i)$ is an *observation*. In our case, $d = 2, n = 10$, and for example, $((3.93, 100), 5.52)$ is an observation (of a student doing very well).

With variable *weights* $w_0, \mathbf{w} = (w_1, \ldots, w_d) \in \mathbb{R}^d$, we plan to minimize the least squares objective

$$f(w_0, \mathbf{w}) = \sum_{i=1}^{n} (w_0 + \mathbf{w}^\top \mathbf{x}_i - y_i)^2.$$

We first want to assume that the inputs and outputs are *centered*, mean-

ing that

$$\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i = \mathbf{0}, \quad \frac{1}{n}\sum_{i=1}^{n} y_i = 0.$$

This can be achieved by simply subtracting the mean $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$ from every input and the mean $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ from every output. In our example, this yields the numbers in Table 1.2 (left).

GPA	TOEFL	GGPA
-0.24	-7.2	-0.94
-0.10	1.8	-0.52
-0.01	5.8	-0.05
-0.02	-7.2	-0.18
0.17	-7.2	0.67
0.12	7.8	0.59
0.01	7.8	0.19
-0.10	-0.2	-0.12
0.11	-1.2	0.17
0.07	-0.2	0.21

GPA	TOEFL	GGPA
-2.04	-1.28	-0.94
-0.88	0.32	-0.52
-0.05	1.03	-0.05
-0.16	-1.28	-0.18
1.42	-1.28	0.67
1.02	1.39	0.59
0.06	1.39	0.19
-0.88	-0.04	-0.12
0.89	-0.21	0.17
0.62	-0.04	0.21

Table 1.2: Centered observations (left); normalized inputs (right)

After centering, the global minimum $(w_0^\star, \mathbf{w}^\star)$ of the least squares objective satisfies $w_0^\star = 0$ while $\mathbf{w}^\star$ is unaffected by centering (Exercise 11), so that we can simply omit the variable w_0 in the sequel.

Finally, we assume that all d input variables are on the same scale, meaning that

$$\frac{1}{n}\sum_{i=1}^{n} x_{ij}^2 = 1, \quad j = 1, \ldots, d.$$

To achieve this for fixed j (assuming that no variable is 0 in all inputs), we multiply all x_{ij} by $s(j) = \sqrt{n/\sum_{i=1}^{n} x_{ij}^2}$ (which, in the optimal solution $\mathbf{w}^\star$, just multiplies $w_j^\star$ by $1/s(j)$, an argument very similar to the one in Exercise 11). For our data set, the resulting normalized data are shown in

Table 1.2 (right). Now the least squares objective (after omitting w_0) is

$$f(w_1, w_2) = \sum_{i=1}^{10} (w_1 x_{i1} + w_2 x_{i2} - y_i)^2$$
$$\approx 10w_1^2 + 10w_2^2 + 1.99 w_1 w_2 - 8.7 w_1 - 2.79 w_2 + 2.09.$$

This is minimized at

$$\mathbf{w}^\star = (w_1^\star, w_2^\star) \approx (0.43, 0.097),$$

so if our initial hypothesis (1.14) is true, we should have

$$y_i \approx y_i^\star = 0.43 x_{i1} + 0.097 x_{i2} \tag{1.15}$$

in the normalized data. This can quickly be checked, and the results are not perfect, but not too bad, either; see Table 1.3 (ignore the last column for now).

x_{i1}	x_{i2}	y_i	$y_i^\star$	$z_i^\star$
-2.04	-1.28	-0.94	-1.00	-0.87
-0.88	0.32	-0.52	-0.35	-0.37
-0.05	1.03	-0.05	0.08	-0.02
-0.16	-1.28	-0.18	-0.19	-0.07
1.42	-1.28	0.67	0.49	0.61
1.02	1.39	0.59	0.57	0.44
0.06	1.39	0.19	0.16	0.03
-0.88	-0.04	-0.12	-0.38	-0.37
0.89	-0.21	0.17	0.36	0.38
0.62	-0.04	0.21	0.26	0.27

Table 1.3: Outputs $y_i^\star$ predicted by the linear model (1.15) and by the model $z_i^\star = 0.43 x_{i1}$ that simply ignores the second input variable

What we also see from (1.15) is that the first input variable (GPA) has a much higher influence on the output (GGPA) than the second one (TOEFL). In fact, if we drop the second one altogether, we obtain outputs $z_i^\star$ (last column in Table 1.3) that seem equivalent to the predicted outputs $y_i^\star$ within the level of noise that we have anyway.

We conclude that TOEFL scores are probably not indicative for the performance of admitted students, so the admission committee should not care too much about them. Requiring a minimum score of 100 might make sense, but whenever an applicant reaches at least this score, the actual value does not matter.

The LASSO. So far, we have computed linear functions $y = 0.43x_1 + 0.097x_2$ and $z = 0.43x_1$ that "explain" the historical data from Table 1.1. However, they are optimized to fit the historical data, not the future. We may have *overfitting*. This typyically leads to unrealiable predictions of high variance in the future. Also, ideally, we would like non-indicative variables (such as the TOEFL in our example) to actually have weight 0, so that the model "knows" the important variables and is therefore better to interpret.

The question is: how can we in general improve the quality of our forecast? There are various heuristics to identify the "important" variables' (subset selection). A very simple one is just to forget about weights close to 0 in the least squares solution. However, for this, we need to define what it means to be close to 0; and it may happen that small changes in the data lead to different variables being dropped if their weights are around the threshold. On the other end of the spectrum, there is *best subset selection* where we compute the least squares solution subject to the constraint that there are at most k nonzero weights, for some k that we believe is the right number of important variables. This is NP-hard, though.

A popular approach that in many cases improves forecasts and at the same time identifies important variables has been suggested by Tibshirani in 1996 [Tib96]. Instead of minimizing the least squares objective globally, it is minimized over a suitable ℓ_1-ball (ball in the 1-norm $\|\mathbf{w}\|_1 = \sum_{j=1}^{d} |w_j|$):

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{n} \|\mathbf{w}^\top \mathbf{x}_i - y_i\|^2 \\ \text{subject to} & \|\mathbf{w}\|_1 \leq R, \end{array} \tag{1.16}$$

where $R \in \mathbb{R}_+$ is some parameter. In our case, if we for example

$$\begin{array}{ll} \text{minimize} & f(w_1, w_2) = 10w_1^2 + 10w_2^2 + 1.99w_1w_2 - 8.7w_1 - 2.79w_2 + 2.09 \\ \text{subject to} & |w_1| + |w_2| \leq 0.2, \end{array} \tag{1.17}$$

we obtain weights $\mathbf{w}^\star = (w_1^\star, w_2^\star) = (0.2, 0)$: the non-indicative TOEFL score has disappeared automatically! For $R = 0.3$, the same happens (with $w_1^\star = 0.3$, respectively). For $R = 0.4$, the TOEFL score starts creeping back in: we get $(w_1^\star, w_2^\star) \approx (0.36, 0.036)$. For $R = 0.5$, we have $(w_1^\star, w_2^\star) \approx (0.41, 0.086)$, while for $R = 0.6$ (and all larger values of R), we recover the original solution $(w_1^\star, w_2^\star) = (0.43, 0.097)$.

It is important to understand that using the "fixed" weights (which may be significantly shrunken), we make predictions *worse* on the historical data (this must be so, since least squares was optimal for the historical data). But future predictions may benefit (a lot). To quantify this benefit, we need to make statistical assumptions about future observations; this is beyond the scope of our treatment here.

The phenomenon that adding a constraint on $\|\mathbf{w}\|_1$ tends to set weights to 0 is not restricted to $d = 2$. The constrained minimization problem (1.16) is called the *LASSO* (least absolute shrinkage and selection operator) and has the tendency to assign weights of 0 and thus to select a subset of input variables, where R controls how aggressive the selection is.

In our example, it is easy to get an intuition why this works. Let us look at the case $R = 0.2$. The smallest value attainable in (1.17) is the smallest α such that that the (elliptical) sublevel set $f^{\leq \alpha}$ of the least squares objective f still intersects the ℓ_1-ball $\{(w_1, w_2) : |w_1| + |w_2| \leq 0.2\}$. This smallest value turns out to be $\alpha = 0.75$, see Figure 1.11. For this value of α, the sublevel set intersects the ℓ_1-ball exactly in one point, namely $(0.2, 0)$.

At $(0.2, 0)$, the ellipse $\{(w_1, w_2) : f(w_1, w_2) = \alpha\}$ is "vertical enough" to just intersect the corner of the ℓ_1-ball. The reason is that the center of the ellipse is relatively close to the w_1-axis, when compared to its size. As R increases, the relevant value of α decreases, the ellipse gets smaller and less vertical around the w_1-axis; until it eventually stops intersecting the ℓ_1-ball $\{(w_1, w_2) : |w_1| + |w_2| \leq R\}$ in a corner (dashed situation in Figure 1.11, for $R = 0.4$).

Even though we have presented a toy example in this section, the background is real. The theory of admission and in particular performance forecasts has been developed in a recent PhD thesis by Zimmermann [Zim16].

1.7 Exercises

Exercise 1. *Prove that a differentiable function is continuous!*

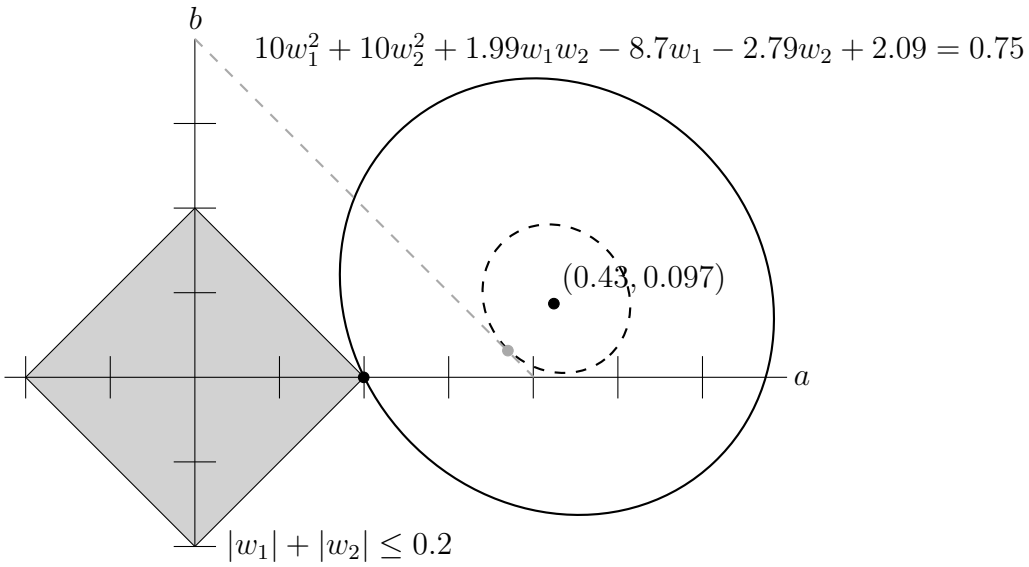


Figure 1.11: Lasso

Exercise 2. *Prove Jensen's inequality (Lemma 1.12)!*

Exercise 3. *Prove that a convex function (with $\mathbf{dom}(f)$ open) is continuous (Lemma 1.13)!*

Hint: *First prove that a convex function f is bounded on any cube $C = [l_1, u_1] \times [l_2, u_2] \times \cdots \times [l_d, u_d] \subseteq \mathbf{dom}(f)$, with the maximum value occurring on some corner of the cube (a point $\mathbf{z}$ such that $z_i \in \{l_i, u_i\}$ for all i). Then use this fact to show that—given $\mathbf{x} \in \mathbf{dom}(f)$ and $\varepsilon > 0$—all $\mathbf{y}$ in a sufficiently small ball around $\mathbf{x}$ satisfy $|f(\mathbf{y}) - f(\mathbf{x})| < \varepsilon$.*

Exercise 4. *Prove that the function $d_{\mathbf{y}} : \mathbb{R}^d \to \mathbb{R}$, $\mathbf{x} \mapsto \|\mathbf{x} - \mathbf{y}\|^2$ is strictly convex for any $\mathbf{y} \in \mathbb{R}^d$. (Use Lemma 1.24.)*

Exercise 5. *Prove Lemma 1.18! Can (ii) be generalized to show that for two convex functions f, g, the function $f \circ g$ is convex as well?*

Exercise 6. *Consider the function ℓ defined in (1.13). Prove that ℓ is convex!*

Exercise 7. *Consider the logistic regression problem with two classes. Given a training set P consisting of datapoint and label pairs $(\mathbf{x}, y)$ where $\mathbf{x} \in \mathbb{R}^d$ and*

$y \in \{-1, +1\}$, *we define our loss* ℓ *for weight vector* $\mathbf{w} \in \mathbb{R}^d$ *to be*

$$\ell(\mathbf{w}) = \sum_{(\mathbf{x},y)\in P} -\ln\left(z(y\mathbf{w}^\top \mathbf{x})\right) \,,$$

where $z(s) = 1/(1+\exp(-s))$. *This loss function is in fact a simplification of (1.13) when we only have two classes.*

We say that the weight vector $\mathbf{w}$ *is a* separator *for* P *if for all* $(\mathbf{x}, y) \in P$,

$$y(\mathbf{w}^\top \mathbf{x}) \geq 0\,.$$

A separator is said to be trivial *if for all* $(\mathbf{x}, y) \in P$,

$$y(\mathbf{w}^\top \mathbf{x}) = 0\,.$$

For example $\mathbf{w} = 0$ *is a trivial separator. Depending on the data* P, *there may be other trivial separators.*

Prove the following statement: the function ℓ *has a global minimum if and only if all separators are trivial.*

Exercise 8. *Prove that the function* $f(\mathbf{x}) = \|\mathbf{x}\|_1 = \sum_{i=1}^d |x_i|$ *(ℓ_1-norm) is convex!*

Exercise 9. *Let* $f : \mathbf{dom}(f) \to \mathbb{R}$ *be twice differentiable. For fixed* $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$, *consider the univariate function* $h(t) = f(\mathbf{x} + t(\mathbf{y} - \mathbf{x}))$ *over a suitable open interval* $\mathbf{dom}(h) \supseteq [0, 1]$ *such that* $\mathbf{x} + t(\mathbf{y} - \mathbf{x}) \in \mathbf{dom}(f)$ *for all* $t \in \mathbf{dom}(h)$. *Let us abbreviate* $\mathbf{v} = \mathbf{y} - \mathbf{x}$. *We already know that* $h'(t) = \nabla f(\mathbf{x} + t\mathbf{v})^\top \mathbf{v}$ *for* $t \in \mathbf{dom}(h)$. *Prove that*

$$h''(t) = \mathbf{v}^\top \nabla^2 f(\mathbf{x} + t\mathbf{v})\mathbf{v}, \quad t \in \mathbf{dom}(h).$$

Exercise 10. *A seminorm is a function* $f : \mathbb{R}^d \to \mathbb{R}$ *satisfying the following two properties for all* $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ *and all* $\lambda \in \mathbb{R}$.

(i) $f(\lambda\mathbf{x}) = |\lambda| f(\mathbf{x})$,

(ii) $f(\mathbf{x} + \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y})$ *(triangle inequality).*

Prove that every seminorm is convex!

Exercise 11. *Suppose that we have centered observations* $(\mathbf{x}_i, y_i)$ *such that* $\sum_{i=1}^n \mathbf{x}_i = \mathbf{0}, \sum_{i=1}^n y_i = 0$. *Let* $w_0^\star, \mathbf{w}^\star$ *be the global minimum of the least squares objective*

$$f(w_0, \mathbf{w}) = \sum_{i=1}^n (w_0 + \mathbf{w}^\top \mathbf{x}_i - y_i)^2.$$

Prove that $w_0^\star = 0$. *Also, suppose* $\mathbf{x}_i'$ *and* y_i' *are such that for all* i, $\mathbf{x}_i' = \mathbf{x}_i + \mathbf{q}$, $y_i' = y_i + r$. *Show that* $(w_0, \mathbf{w})$ *minimizes* f *if and only if* $(w_0 - \mathbf{w}^\top \mathbf{q} + r, \mathbf{w})$ *minimizes*

$$f'(w_o, \mathbf{w}) = \sum_{i=1}^n (w_0 + \mathbf{w}^\top \mathbf{x}_i' - y_i')^2.$$

Chapter 2

Gradient Descent

Contents

2.1 Overview

The gradient descent algorithm (including variants such as projected or stochastic gradient descent) is the most useful workhorse for minimizing loss functions in practice. The algorithm is extremely simple and surprisingly robust in the sense that it also works well for many loss functions that are not convex. While it is easy to construct (artificial) non-convex functions on which gradient descent goes completely astray, such functions do not seem to be typical in practice; however, understanding this on a theoretical level is an open problem, and only few results exist in this direction.

The vast majority of theoretical results concerning the performance of gradient descent hold for convex functions only. In this and the following chapters, we will present some of these results, but maybe more importantly, the main ideas behind them. As it turns out, the number of ideas that we need is rather small, and typically, they are shared between different results. Our approach is therefore to fully develop each idea once, in the context of a concrete result. If the idea reappears, we will typically only discuss the changes that are necessary in order to establish a new result from this idea. In order to avoid boredom from ideas that reappear too often, we omit other results and variants that one could also get along the lines of what we discuss.

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex and differentiable function. We also assume that f has a global minimum $\mathbf{x}^\star$, and the goal is to find (an approximation of) $\mathbf{x}^\star$. This usually means that for a given $\varepsilon > 0$, we want to find $\mathbf{x} \in \mathbb{R}^d$ such that

$$f(\mathbf{x}) - f(\mathbf{x}^\star) < \varepsilon.$$

Notice that we are not making an attempt to get near to $\mathbf{x}^\star$ itself — there can be several minima $\mathbf{x}_1^\star \neq \mathbf{x}^\star \neq \mathbf{x}_2^\star$ with $f(\mathbf{x}_1^\star) = f(\mathbf{x}_2^\star) = f(\mathbf{x}^\star)$.

Table 2.1 gives an overview of the results that we will prove. They concern several variants of gradient descent as well as several classes of functions. The significance of each algorithm and function class will briefly be discussed when it first appears.

In Chapter 6, we will also look at gradient descent on functions that are not convex. In this case, provably small approximation error can still be obtained for some particularly well-behaved functions (we will give an example). For smooth (but not necessarily convex) functions, we gener-

	Lipschitz convex functions	smooth convex functions	strongly convex functions	smooth & strongly convex functions
gradient descent	Thm. 2.1 $\mathcal{O}(1/\varepsilon^2)$	Thm. 2.7 $\mathcal{O}(1/\varepsilon)$		Thm. 2.12 $\mathcal{O}(\log(1/\varepsilon))$
accelerated gradient descent		Thm. 2.8 $\mathcal{O}(1/\sqrt{\varepsilon})$		
projected gradient descent	Thm. 3.2 $\mathcal{O}(1/\varepsilon^2)$	Thm. 3.4 $\mathcal{O}(1/\varepsilon)$		Thm. 3.5 $\mathcal{O}(\log(1/\varepsilon))$
proximal gradient descent		Thm. 3.14 $\mathcal{O}(1/\varepsilon)$		
subgradient descent	Thm. 4.7 $\mathcal{O}(1/\varepsilon^2)$		Thm. 4.11 $\mathcal{O}(1/\varepsilon)$	
stochastic gradient descent	Thm. 5.1 $\mathcal{O}(1/\varepsilon^2)$		Thm. 5.2 $\mathcal{O}(1/\varepsilon)$	

Table 2.1: Results on gradient descent. Below each theorem, the number of steps is given which the respective variant needs on the respective function class to achieve additive approximation error at most ε.

ally cannot show convergence in error, but a (much) weaker convergence property still holds.

2.2 The algorithm

Gradient descent is a very simple iterative algorithm for finding the desired approximation $\mathbf{x}$, under suitable conditions that we will get to. It computes a sequence $\mathbf{x}_0, \mathbf{x}_1, \ldots$ of vectors such that $\mathbf{x}_0$ is arbitrary, and for each $t \geq 0$, $\mathbf{x}_{t+1}$ is obtained from $\mathbf{x}_t$ by making a step of $\mathbf{v}_t \in \mathbb{R}^d$:

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{v}_t.$$

How do we choose $\mathbf{v}_t$ in order to get closer to optimality, meaning that $f(\mathbf{x}_{t+1}) < f(\mathbf{x}_t)$?

From differentiability of f at $\mathbf{x}_t$ (Definition 1.5), we know that for $\|\mathbf{v}_t\|$ tending to 0,

$$f(\mathbf{x}_t + \mathbf{v}_t) = f(\mathbf{x}_t) + \nabla f(\mathbf{x}_t)^\top \mathbf{v}_t + \underbrace{r(\mathbf{v}_t)}_{o(\|\mathbf{v}_t\|)} \approx f(\mathbf{x}_t) + \nabla f(\mathbf{x}_t)^\top \mathbf{v}_t.$$

To get any decrease in function value at all, we have to choose $\mathbf{v}_t$ such that $\nabla f(\mathbf{x}_t)^\top \mathbf{v}_t < 0$. But among all steps $\mathbf{v}_t$ of the same length, we should in fact choose the one with the most negative value of $\nabla f(\mathbf{x}_t)^\top \mathbf{v}_t$, so that we maximize our decrease in function value. This is achieved when $\mathbf{v}_t$ points into the direction of the negative gradient $-\nabla f(\mathbf{x}_t)$. But as differentiabilty guarantees decrease only for small steps, we also want to control how far we go along the direction of the negative gradient.

Therefore, the step of gradient descent is defined by

$$\mathbf{x}_{t+1} := \mathbf{x}_t - \gamma \nabla f(\mathbf{x}_t). \tag{2.1}$$

Here, γ is a fixed *stepsize*, but it may also make sense to have γ depend on t. For now, γ is fixed. We hope that for some reasonably small integer t, in the t-th iteration we get that $f(\mathbf{x}_t) - f(\mathbf{x}^\star) < \varepsilon$; see Figure 2.1 for an example.

Now it becomes clear why we are assuming that $\mathbf{dom}(f) = \mathbb{R}^d$: The update step (2.1) may in principle take us "anywhere", so in order to get a well-defined algorithm, we want to make sure that f is defined and differentiable everywhere.

The choice of γ is critical for the performance. If γ is too small, the process might take too long, and if γ is too large, we are in danger of overshooting. It is not clear at this point whether there is a "right" stepsize.

2.3 Vanilla analysis

Let $\mathbf{x}_t$ be some iterate in the sequence (2.1). We abbreviate $\mathbf{g}_t := \nabla f(\mathbf{x}_t)$, and will relate this vector to our current direction from an optimum $\mathbf{x}_t - \mathbf{x}^\star$. By definition of gradient descent (2.1), $\mathbf{g}_t = (\mathbf{x}_t - \mathbf{x}_{t+1})/\gamma$, hence

$$\mathbf{g}_t^\top (\mathbf{x}_t - \mathbf{x}^\star) = \frac{1}{\gamma} (\mathbf{x}_t - \mathbf{x}_{t+1})^\top (\mathbf{x}_t - \mathbf{x}^\star). \tag{2.2}$$

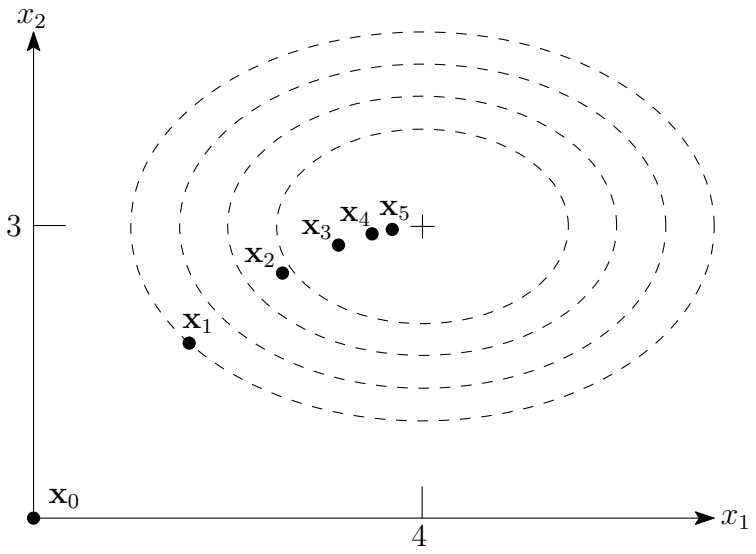


Figure 2.1: Example run of gradient descent on the quadratic function $f(x_1, x_2) = 2(x_1 - 4)^2 + 3(x_2 - 3)^2$ with global minimum $(4, 3)$; we have chosen $\mathbf{x}_0 = (0, 0), \gamma = 0.1$; dashed lines represent level sets of f (points of constant f-value)

Now we apply (somewhat out of the blue, but this will clear up in the next step) the basic vector equation $2\mathbf{v}^\top \mathbf{w} = \|\mathbf{v}\|^2 + \|\mathbf{w}\|^2 - \|\mathbf{v} - \mathbf{w}\|^2$ (a.k.a. the cosine theorem) to rewrite the same expression as

$$\begin{aligned} \mathbf{g}_t^\top (\mathbf{x}_t - \mathbf{x}^\star) &= \frac{1}{2\gamma} \left(\|\mathbf{x}_t - \mathbf{x}_{t+1}\|^2 + \|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \right) \\ &= \frac{1}{2\gamma} \left(\gamma^2 \|\mathbf{g}_t\|^2 + \|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \right) \\ &= \frac{\gamma}{2} \|\mathbf{g}_t\|^2 + \frac{1}{2\gamma} \left(\|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \right) \end{aligned} \tag{2.3}$$

Next we sum this up over the iterations t, so that the latter two terms in the bracket cancel in a telescoping sum.

$$\begin{aligned}\sum_{t=0}^{T-1} \mathbf{g}_t^\top (\mathbf{x}_t - \mathbf{x}^\star) &= \frac{\gamma}{2} \sum_{t=0}^{T-1} \|\mathbf{g}_t\|^2 + \frac{1}{2\gamma} \left(\|\mathbf{x}_0 - \mathbf{x}^\star\|^2 - \|\mathbf{x}_T - \mathbf{x}^\star\|^2 \right) \\ &\leq \frac{\gamma}{2} \sum_{t=0}^{T-1} \|\mathbf{g}_t\|^2 + \frac{1}{2\gamma} \|\mathbf{x}_0 - \mathbf{x}^\star\|^2 \end{aligned} \tag{2.4}$$

So far, we have not used any properties of the function f or its gradient $\mathbf{g}_t$, except the definition of the update step $\mathbf{x}_{t+1} = \mathbf{x}_t - \gamma \mathbf{g}_t$. Now we invoke convexity of f, or more precisely the first-order characterization of convexity (1.3) with $\mathbf{x} = \mathbf{x}_t, \mathbf{y} = \mathbf{x}^\star$:

$$f(\mathbf{x}_t) - f(\mathbf{x}^\star) \leq \mathbf{g}_t^\top (\mathbf{x}_t - \mathbf{x}^\star). \tag{2.5}$$

Hence we further obtain

$$\sum_{t=0}^{T-1} (f(\mathbf{x}_t) - f(\mathbf{x}^\star)) \leq \frac{\gamma}{2} \sum_{t=0}^{T-1} \|\mathbf{g}_t\|^2 + \frac{1}{2\gamma} \|\mathbf{x}_0 - \mathbf{x}^\star\|^2. \tag{2.6}$$

This gives us an upper bound for the *average* error $f(\mathbf{x}_t) - f(\mathbf{x}^\star)$, $t = 0, \ldots, T-1$, hence in particular for the error incurred by the iterate with the smallest function value. The last iterate is not necessarily the best one: gradient descent with fixed stepsize γ will in general also make steps that overshoot and actually increase the function value; see Exercise 14(i).

The question is of course: is this result any good? In general, the answer is no. A dependence on $\|\mathbf{x}_0 - \mathbf{x}^\star\|$ is to be expected (the further we start from $\mathbf{x}^\star$, the longer we will take); the dependence on the squared gradients $\|\mathbf{g}_t\|^2$ is more of an issue, and if we cannot control them, we cannot say much.

2.4 Lipschitz convex functions: $\mathcal{O}(1/\varepsilon^2)$ steps

Here is the cheapest "solution" to squeeze something out of the vanilla analysis (2.4): let us simply assume that all gradients of f are bounded in norm. Equivalently, such functions are Lipschitz continuous over $\mathbb{R}^d$

by Theorem 1.9. (A small subtetly here is that in the situation of real-valued functions, Theorem 1.9 is talking about the spectral norm of the $(1 \times d)$-matrix (or row vector) $\nabla f(\mathbf{x})^\top$, while below, we are talking about the Euclidean norm of the (column) vector $\nabla f(\mathbf{x})$; but these two norms are the same; see Exercise 12.)

Assuming bounded gradients rules out many interesting functions, though. For example, $f(x) = x^2$ (a supermodel in the world of convex functions) already doesn't qualify, as $\nabla f(x) = 2x$—and this is unbounded as x tends to infinity. But let's care about supermodels later.

Theorem 2.1. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable with a global minimum $\mathbf{x}^\star$; furthermore, suppose that $\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$ and $\|\nabla f(\mathbf{x})\| \leq B$ for all $\mathbf{x}$. Choosing the stepsize*

$$\gamma := \frac{R}{B\sqrt{T}},$$

gradient descent (2.1) yields

$$\frac{1}{T}\sum_{t=0}^{T-1}(f(\mathbf{x}_t) - f(\mathbf{x}^\star)) \leq \frac{RB}{\sqrt{T}}.$$

Proof. This is a simple calculation on top of (2.6): after plugging in the bounds $\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$ and $\|\mathbf{g}_t\| \leq B$, we get

$$\sum_{t=0}^{T-1}(f(\mathbf{x}_t) - f(\mathbf{x}^\star)) \leq \frac{\gamma}{2}B^2T + \frac{1}{2\gamma}R^2,$$

so want to choose γ such that

$$q(\gamma) = \frac{\gamma}{2}B^2T + \frac{R^2}{2\gamma}$$

is minimized. Setting the derivative to zero yields the above value of γ, and $q(R/(B\sqrt{T})) = RB\sqrt{T}$. Dividing by T, the result follows. □

This means that in order to achieve $\min_{t=0}^{T-1}(f(\mathbf{x}_t) - f(\mathbf{x}^\star)) \leq \varepsilon$, we need

$$T \geq \frac{R^2B^2}{\varepsilon^2}$$

many iterations. This is not particularly good when it comes to concrete numbers (think of desired error $\varepsilon = 10^{-6}$ when R, B are somewhat larger). On the other hand, the number of steps does not depend on d, the dimension of the space. This is very important since we often optimize in high-dimensional spaces. Of course, R and B may depend on d, but in many relevant cases, this dependence is mild.

What happens if we don't know R and/or B? An idea is to "guess" R and B, run gradient descent with T and γ resulting from the guess, check whether the result has absolute error at most ε, and repeat with a different guess otherwise. This fails, however, since in order to compute the absolute error, we need to know $f(\mathbf{x}^\star)$ which we typically don't. But Exercise 15 asks you to show that knowing R is sufficient.

2.5 Smooth convex functions: $\mathcal{O}(1/\varepsilon)$ steps

Our workhorse in the vanilla analysis was the first-order characterization of convexity: for all $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$, we have

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}). \tag{2.7}$$

Next we want to look at functions for which $f(\mathbf{y})$ can be bounded *from above* by $f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x})$, up to at most quadratic error. The following definition applies to all differentiable functions, convexity is not required.

Definition 2.2. *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be a differentiable function, $X \subseteq \mathbf{dom}(f)$ convex and $L \in \mathbb{R}_+$. Function f is called* smooth *(with parameter L) over X if*

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in X. \tag{2.8}$$

If $X = \mathbf{dom}(f)$, f is simply called smooth.

Recall that (2.7) says that for any $\mathbf{x}$, the graph of f is above its tangential hyperplane at $(\mathbf{x}, f(\mathbf{x}))$. In contrast, (2.8) says that for any $\mathbf{x} \in X$, the graph of f is below a not-too-steep tangential paraboloid at $(\mathbf{x}, f(\mathbf{x}))$; see Figure 2.2.

This notion of smoothness has become standard in convex optimization, but the naming is somewhat unfortunate, since there is an (older) definition of a smooth function in mathematical analysis where it means a function that is infinitely often differentiable.

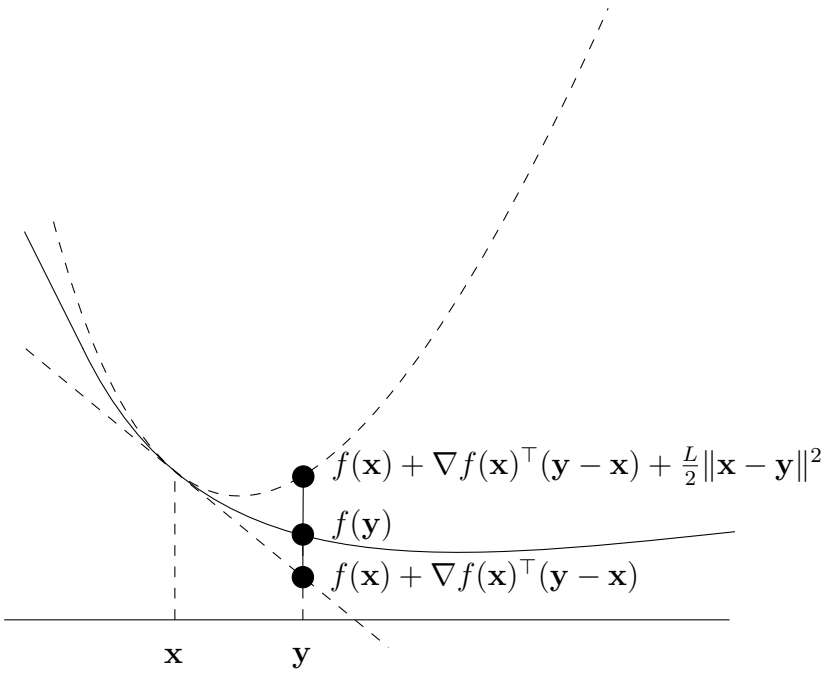


Figure 2.2: A smooth convex function

Let us discuss some cases. If $L = 0$, (2.7) and (2.8) together require that

$$f(\mathbf{y}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x}), \quad \forall \mathbf{x}, \mathbf{y} \in \mathbf{dom}(f),$$

meaning that f is an affine function. A simple calculation shows that our supermodel function $f(x) = x^2$ is smooth with parameter $L = 2$:

$$\begin{aligned} f(y) = y^2 &= x^2 + 2x(y - x) + (x - y)^2 \\ &= f(x) + f'(x)(y - x) + \frac{L}{2}(x - y)^2. \end{aligned}$$

More generally, we also claim that all quadratic functions of the form $f(\mathbf{x}) = \mathbf{x}^\top Q\mathbf{x} + \mathbf{b}^\top\mathbf{x} + c$ are smooth, where Q is a $(d \times d)$ matrix, $\mathbf{b} \in \mathbb{R}^d$ and $c \in \mathbb{R}$. Because $\mathbf{x}^\top Q\mathbf{x} = \mathbf{x}^\top Q^\top\mathbf{x}$, we get that $f(\mathbf{x}) = \mathbf{x}^\top Q\mathbf{x} = \frac{1}{2}\mathbf{x}^\top(Q + Q^\top)\mathbf{x}$, where $\frac{1}{2}(Q + Q^\top)$ is symmetric. Therefore, we can assume without loss of generality that Q is symmetric, i.e., it suffices to show that quadratic functions defined by symmetric functions are smooth.

Lemma 2.3 (Exercise 13)**.** *Let $f(\mathbf{x}) = \mathbf{x}^\top Q\mathbf{x} + \mathbf{b}^\top \mathbf{x} + c$, where Q is a symmetric $(d \times d)$ matrix, $\mathbf{b} \in \mathbb{R}^d, c \in \mathbb{R}$. Then f is smooth with parameter $2\|Q\|$, where $\|Q\|$ is the spectral norm of Q (Definition 1.2).*

The (univariate) convex function $f(x) = x^4$ is not smooth (over $\mathbb{R}$): at $x = 0$, condition (2.8) reads as

$$y^4 \leq \frac{L}{2} y^2,$$

and there is obviously no L that works for all y. The function is smooth, however, over any bounded set X (Exercise 18).

In general—and this is the important message here—only functions of asymptotically at most quadratic growth can be smooth. It is tempting to believe that any such "subquadratic" function is actually smooth, but this is not true. Exercise 14(iii) provides a counterexample.

While bounded gradients are equivalent to Lipschitz continuity of f (Theorem 1.9), smoothness turns out to be equivalent to Lipschitz continuity of ∇f—if f is convex over the whole space. In general, Lipschitz continuity of ∇f implies smoothness, but not the other way around.

Lemma 2.4. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable. The following two statements are equivalent.*

(i) f is smooth with parameter L.

(ii) $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.

We will derive the direction (ii)⇒(i) as Lemma 6.1 in Chapter 6 (which neither requires convexity nor domain $\mathbb{R}^d$). The other direction is a bit more involved. A proof of the equivalence can be found in the lecture slides of L. Vandenberghe, `http://www.seas.ucla.edu/~vandenbe/236C/lectures/gradient.pdf`.

The operations that we have shown to preserve convexity (Lemma 1.18) also preserve smoothness. This immediately gives us a rich collection of smooth functions.

Lemma 2.5 (Exercise 16)**.**

(i) Let $f_1, f_2, \ldots, f_m$ be smooth with parameters $L_1, L_2, \ldots, L_m$, and let $\lambda_1, \lambda_2, \ldots, \lambda_m \in \mathbb{R}_+$. Then the function $f := \sum_{i=1}^m \lambda_i f_i$ is smooth with parameter $\sum_{i=1}^m \lambda_i L_i$ over $\mathbf{dom}(f) := \bigcap_{i=1}^m \mathbf{dom}(f_i)$.

(ii) Let $f : \mathbf{dom}(f) \to \mathbb{R}$ with $\mathbf{dom}(f) \subseteq \mathbb{R}^d$ be smooth with parameter L, and let $g : \mathbb{R}^m \to \mathbb{R}^d$ be an affine function, meaning that $g(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$, for some matrix $A \in \mathbb{R}^{d \times m}$ and some vector $\mathbf{b} \in \mathbb{R}^d$. Then the function $f \circ g$ (that maps $\mathbf{x}$ to $f(A\mathbf{x} + \mathbf{b})$) is smooth with parameter $L\|A\|^2$ on $\mathbf{dom}(f \circ g) := \{\mathbf{x} \in \mathbb{R}^m : g(\mathbf{x}) \in \mathbf{dom}(f)\}$, where $\|A\|$ is the spectral norm of A (Definition 1.2).

We next show that for smooth convex functions, the vanilla analysis provides a better bound than it does under bounded gradients. In particular, we are now able to serve the supermodel $f(x) = x^2$.

We start with a preparatory lemma showing that gradient descent (with suitable stepsize γ) makes progress in function value on smooth functions in every step. We call this *sufficient decrease*, and maybe surprisingly, it does not require convexity.

Lemma 2.6. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable and smooth with parameter L according to (2.8). With*

$$\gamma := \frac{1}{L},$$

gradient descent (2.1) satisfies

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2, \quad t \geq 0.$$

More specifically, this already holds if f is smooth with parameter L over the line segment connecting $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$.

Proof. We apply the smoothness condition (2.8) and the definition of gradient descent that yields $\mathbf{x}_{t+1} - \mathbf{x}_t = -\nabla f(\mathbf{x}_t)/L$. We compute

$$\begin{aligned} f(\mathbf{x}_{t+1}) &\leq f(\mathbf{x}_t) + \nabla f(\mathbf{x}_t)^\top(\mathbf{x}_{t+1} - \mathbf{x}_t) + \frac{L}{2}\|\mathbf{x}_t - \mathbf{x}_{t+1}\|^2 \\ &= f(\mathbf{x}_t) - \frac{1}{L}\|\nabla f(\mathbf{x}_t)\|^2 + \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2 \\ &= f(\mathbf{x}_t) - \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2. \end{aligned}$$

□

Theorem 2.7. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable with a global minimum $\mathbf{x}^\star$; furthermore, suppose that f is smooth with parameter L according to (2.8). Choosing stepsize*

$$\gamma := \frac{1}{L},$$

gradient descent (2.1) yields

$$f(\mathbf{x}_T) - f(\mathbf{x}^\star) \leq \frac{L}{2T}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2, \quad T > 0.$$

Proof. We apply sufficient decrease (Lemma 2.6) to bound the sum of the $\|\mathbf{g}_t\|^2 = \|\nabla f(\mathbf{x}_t)\|^2$ after step (2.6) of the vanilla analysis as follows:

$$\frac{1}{2L}\sum_{t=0}^{T-1}\|\nabla f(\mathbf{x}_t)\|^2 \leq \sum_{t=0}^{T-1}(f(\mathbf{x}_t) - f(\mathbf{x}_{t+1})) = f(\mathbf{x}_0) - f(\mathbf{x}_T). \tag{2.9}$$

With $\gamma = 1/L$, (2.6) then yields

$$\begin{aligned}\sum_{t=0}^{T-1}(f(\mathbf{x}_t) - f(\mathbf{x}^\star)) &\leq \frac{1}{2L}\sum_{t=0}^{T-1}\|\nabla f(\mathbf{x}_t)\|^2 + \frac{L}{2}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2 \\ &\leq f(\mathbf{x}_0) - f(\mathbf{x}_T) + \frac{L}{2}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2,\end{aligned}$$

equivalently

$$\sum_{t=1}^{T}(f(\mathbf{x}_t) - f(\mathbf{x}^\star)) \leq \frac{L}{2}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2. \tag{2.10}$$

Because $f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t)$ for each $0 \leq t \leq T$ by Lemma 2.6, by taking the average we get that

$$f(\mathbf{x}_T) - f(\mathbf{x}^\star) \leq \frac{1}{T}\sum_{t=1}^{T}(f(\mathbf{x}_t) - f(\mathbf{x}^\star)) \leq \frac{L}{2T}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2.$$

□

This improves over the bounds of Theorem 2.1. With $R^2 := \|\mathbf{x}_0 - \mathbf{x}^\star\|^2$, we now only need

$$T \geq \frac{R^2 L}{2\varepsilon}$$

iterations instead of R^2B^2/ε^2 to achieve absolute error at most ε.

Exercise 17 shows that we do not need to know L to obtain the same asymptotic runtime.

Interestingly, the bound in Theorem 2.7 can be improved—but not by much. Fixing L and $R = \|\mathbf{x}_0 - \mathbf{x}^\star\|$, the bound is of the form $O(1/T)$. Lee and Wright have shown that a better upper bound of $o(1/T)$ holds, but that for any fixed $\delta > 0$, a lower bound of $\Omega(1/T^{1+\delta})$ also holds [LW19].

2.6 Acceleration for smooth convex functions: $\mathcal{O}(1/\sqrt{\varepsilon})$ steps

Let's take a step back, forget about gradient descent for a moment, and just think about what we actually use the algorithm for: we are minimizing a differentiable convex function $f : \mathbb{R}^d \to \mathbb{R}$, where we are assuming that we have acccess to the gradient vector $\nabla f(\mathbf{x})$ at any given point $\mathbf{x}$.

But is it clear that gradient descent is the best algorithm for this task? After all, it is just *some* algorithm that is using gradients to make progress locally, but there might be other (and better) such algorithms. Let us define a *first-order method* as an algorithm that only uses gradient information to minimize f. More precisely, we allow a first-order method to access f only via an oracle that is able to return values of f and ∇f at arbitrary points. Gradient descent is then just a specific first-order method.

For any class of convex functions, one can then ask a natural question: What is the best first-order method for the function class, the one that needs the smallest number of oracle calls in the worst case, as a function of the desired error ε? In particular, is there a method that asymptotically beats gradient descent?

There is an interesting history here: in 1979, Nemirovski and Yudin have shown that *every* first-order method needs in the worst case $\Omega(1/\sqrt{\varepsilon})$ steps (gradient evaluations) in order to achieve an additive error of ε on smooth functions [NY83]. Recall that we have seen an upper bound of $O(1/\varepsilon)$ for gradient descent in the previous section; in fact, this upper bound was known to Nemirovsky and Yudin already. Reformulated in the language of the previous section, there is a first-order method (gradient descent) that attains additive error $O(1/T)$ after T steps, and all first-order methods have additive error $\Omega(1/T^2)$ in the worst case.

The obvious question resulting from this was whether there actually exists a first-order method that has additive error $O(1/T^2)$ after T steps, on every smooth function. This was answered in the affirmative by Nesterov in 1983 when he proposed an algorithm that is now known as *(Nesterov's) accelerated gradient descent* [Nes83]. Nesterov's book (Sections 2.1 and 2.2) is a comprehensive source for both lower and upper bound [Nes18].

It is not easy to understand why the accelerated gradient descent algorithm is an optimal first-order method, and how Nesterov even arrived at it. A number of alternative derivations of optimal algorithms have been given by other authors, usually claiming that they provide a more natural or easier-to-grasp approach. However, each alternative approach requires some understanding of other things, and there is no well-established "simplest approach". Here, we simply throw the algorithm at the reader, without any attempt to motivate it beyond some obvious words. Then we present a short proof that the algorithm is indeed optimal.

Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex, differentiable, and smooth with parameter L. *Accelerated gradient descent* is the following algorithm: choose $\mathbf{z}_0 = \mathbf{y}_0 = \mathbf{x}_0$ arbitrary. For $t \geq 0$, set

$$\mathbf{y}_{t+1} := \mathbf{x}_t - \frac{1}{L}\nabla f(\mathbf{x}_t), \tag{2.11}$$

$$\mathbf{z}_{t+1} := \mathbf{z}_t - \frac{t+1}{2L}\nabla f(\mathbf{x}_t), \tag{2.12}$$

$$\mathbf{x}_{t+1} := \frac{t+1}{t+3}\mathbf{y}_{t+1} + \frac{2}{t+3}\mathbf{z}_{t+1}. \tag{2.13}$$

This means, we are performing a normal "smooth step" from $\mathbf{x}_t$ to obtain $\mathbf{y}_{t+1}$ and a more aggressive step from $\mathbf{z}_t$ to get $\mathbf{z}_{t+1}$. The next iterate $\mathbf{x}_{t+1}$ is a weighted average of $\mathbf{y}_{t+1}$ and $\mathbf{z}_{t+1}$, where we compensate for the more aggressive step by giving $\mathbf{z}_{t+1}$ a relatively low weight.

Theorem 2.8. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable with a global minimum $\mathbf{x}^\star$; furthermore, suppose that f is smooth with parameter L according to (2.8). Accelerated gradient descent (2.11), (2.12), and (2.13), yields*

$$f(\mathbf{y}_T) - f(\mathbf{x}^\star) \leq \frac{2L\left\|\mathbf{z}_0 - \mathbf{x}^\star\right\|^2}{T(T+1)}, \quad T > 0.$$

Comparing this bound with the one from Theorem 2.7, we see that the error is now indeed $O(1/T^2)$ instead of $O(1/T)$; to reach error at most ε,

accelerated gradient descent therefore only needs $O(1/\sqrt{\varepsilon})$ steps instead of $O(1/\varepsilon)$.

Proof. The analysis uses a *potential function argument* [BG17]. We assign a potential $\Phi(t)$ to each time t and show that $\Phi(t+1) \leq \Phi(t)$. The potential is

$$\Phi(t) := t(t+1)\left(f(\mathbf{y}_t) - f(\mathbf{x}^\star)\right) + 2L\left\|\mathbf{z}_t - \mathbf{x}^\star\right\|^2.$$

If we can show that the potential always decreases, we get

$$\underbrace{T(T+1)\left(f(\mathbf{y}_T) - f(\mathbf{x}^\star)\right) + 2L\left\|\mathbf{z}_T - \mathbf{x}^\star\right\|^2}_{\Phi(T)} \leq \underbrace{2L\left\|\mathbf{z}_0 - \mathbf{x}^\star\right\|^2}_{\Phi(0)},$$

from which the statement immediately follows. For the argument, we need three well-known ingredients: (i) sufficient decrease (Lemma 2.6) for step (2.11) with $\gamma = 1/L$:

$$f(\mathbf{y}_{t+1}) \leq f(\mathbf{x}_t) - \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2; \tag{2.14}$$

(ii) the vanilla analysis (Section 2.3) for step (2.12) with $\gamma = \frac{t+1}{2L}$, $\mathbf{g}_t = \nabla f(\mathbf{x}_t)$:

$$\mathbf{g}_t^\top(\mathbf{z}_t - \mathbf{x}^\star) = \frac{t+1}{4L}\|\mathbf{g}_t\|^2 + \frac{L}{t+1}\left(\|\mathbf{z}_t - \mathbf{x}^\star\|^2 - \|\mathbf{z}_{t+1} - \mathbf{x}^\star\|^2\right); \tag{2.15}$$

(iii) convexity:

$$f(\mathbf{x}_t) - f(\mathbf{w}) \leq \mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{w}), \quad \mathbf{w} \in \mathbb{R}^d. \tag{2.16}$$

On top of this, we perform some simple calculations next. By definition, the potentials are

$$\begin{aligned}
\Phi(t+1) &= t(t+1)\left(f(\mathbf{y}_{t+1}) - f(\mathbf{x}^\star)\right) + 2(t+1)\left(f(\mathbf{y}_{t+1}) - f(\mathbf{x}^\star)\right) + 2L\left\|\mathbf{z}_{t+1} - \mathbf{x}^\star\right\|^2 \\
\Phi(t) &= t(t+1)\left(f(\mathbf{y}_t) - f(\mathbf{x}^\star)\right) + 2L\left\|\mathbf{z}_t - \mathbf{x}^\star\right\|^2
\end{aligned}$$

Now,

$$\Delta := \frac{\Phi(t+1) - \Phi(t)}{t+1}$$

can be bounded as follows.

$$\begin{aligned}
\Delta &= t\left(f(\mathbf{y}_{t+1})-f(\mathbf{y}_t)\right)+2\left(f(\mathbf{y}_{t+1})-f(\mathbf{x}^\star)\right)+\frac{2L}{t+1}\left(\|\mathbf{z}_{t+1}-\mathbf{x}^\star\|^2-\|\mathbf{z}_t-\mathbf{x}^\star\|^2\right)\\
&\overset{(2.15)}{=} t\left(f(\mathbf{y}_{t+1})-f(\mathbf{y}_t)\right)+2\left(f(\mathbf{y}_{t+1})-f(\mathbf{x}^\star)\right)+\frac{t+1}{2L}\|\mathbf{g}_t\|^2-2\mathbf{g}_t^\top(\mathbf{z}_t-\mathbf{x}^\star)\\
&\overset{(2.14)}{\leq} t\left(f(\mathbf{x}_t)-f(\mathbf{y}_t)\right)+2\left(f(\mathbf{x}_t)-f(\mathbf{x}^\star)\right)-\frac{1}{2L}\|\mathbf{g}_t\|^2-2\mathbf{g}_t^\top(\mathbf{z}_t-\mathbf{x}^\star)\\
&\leq t\left(f(\mathbf{x}_t)-f(\mathbf{y}_t)\right)+2\left(f(\mathbf{x}_t)-f(\mathbf{x}^\star)\right)-2\mathbf{g}_t^\top(\mathbf{z}_t-\mathbf{x}^\star)\\
&\overset{(2.16)}{\leq} t\mathbf{g}_t^\top(\mathbf{x}_t-\mathbf{y}_t)+2\mathbf{g}_t^\top(\mathbf{x}_t-\mathbf{x}^\star)-2\mathbf{g}_t^\top(\mathbf{z}_t-\mathbf{x}^\star)\\
&= \mathbf{g}_t^\top((t+2)\mathbf{x}_t-t\mathbf{y}_t-2\mathbf{z}_t)\\
&\overset{(2.13)}{=} \mathbf{g}_t^\top\mathbf{0}=0.
\end{aligned}$$

Hence, we indeed have $\Phi(t+1)\leq\Phi(t)$. □

2.7 Interlude

Let us get back to the supermodel $f(x)=x^2$ (that is smooth with parameter $L=2$, as we observed before). According to Theorem 2.7, gradient descent (2.1) with stepsize $\gamma=1/2$ satisfies

$$f(x_T)\leq\frac{1}{T}x_0^2. \tag{2.17}$$

Here we used that the minimizer is $x^\star=0$. Let us check how good this bound really is. For our concrete function and concrete stepsize, (2.1) reads as

$$x_{t+1}=x_t-\frac{1}{2}\nabla f(x_t)=x_t-x_t=0,$$

so we are always done after one step! But we will see in the next section that this is only because the function is particularly beautiful, and on top of that, we have picked the best possible smoothness parameter. To simulate a more realistic situation here, let us assume that we have not looked at the supermodel too closely and found it to be smooth with parameter $L=4$ only (which is a suboptimal but still valid parameter). In this case, $\gamma=1/4$ and (2.1) becomes

$$x_{t+1}=x_t-\frac{1}{4}\nabla f(x_t)=x_t-\frac{x_t}{2}=\frac{x_t}{2}.$$

So, we in fact have

$$f(x_T) = f\left(\frac{x_0}{2^T}\right) = \frac{1}{2^{2T}} x_0^2. \tag{2.18}$$

This is still vastly better than the bound of (2.17)! While (2.17) requires $T \approx x_0^2/\varepsilon$ to achieve $f(x_T) \leq \varepsilon$, (2.18) requires only

$$T \approx \frac{1}{2} \log\left(\frac{x_0^2}{\varepsilon}\right),$$

which is an exponential improvement in the number of steps.

2.8 Smooth and strongly convex functions: $\mathcal{O}(\log(1/\varepsilon))$ steps

The supermodel function $f(x) = x^2$ is not only smooth ("not too curved") but also *strongly convex* ("not too flat"). It will turn out that this is the crucial ingredient that makes gradient descent fast.

Definition 2.9. *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be a convex and differentiable function, $X \subseteq \mathbf{dom}(f)$ convex and $\mu \in \mathbb{R}_+, \mu > 0$. Function f is called* strongly convex *(with parameter μ) over X if*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{\mu}{2}\|\mathbf{x} - \mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in X. \tag{2.19}$$

If $X = \mathbf{dom}(f)$, f is simply called strongly convex.

While smoothness according to (2.8) says that for any $\mathbf{x} \in X$, the graph of f is *below* a *not-too-steep* tangential paraboloid at $(\mathbf{x}, f(\mathbf{x}))$, strong convexity means that the graph of f is *above* a *not-too-flat* tangential paraboloid at $(\mathbf{x}, f(\mathbf{x}))$. The graph of a smooth *and* strongly convex function is therefore at every point wedged between two paraboloids; see Figure 2.3.

We can also interpret (2.19) as a strengthening of convexity. In the form of (2.7), convexity reads as

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}), \quad \forall \mathbf{x}, \mathbf{y} \in \mathbf{dom}(f),$$

and therefore says that every convex function satisfies (2.19) with $\mu = 0$.

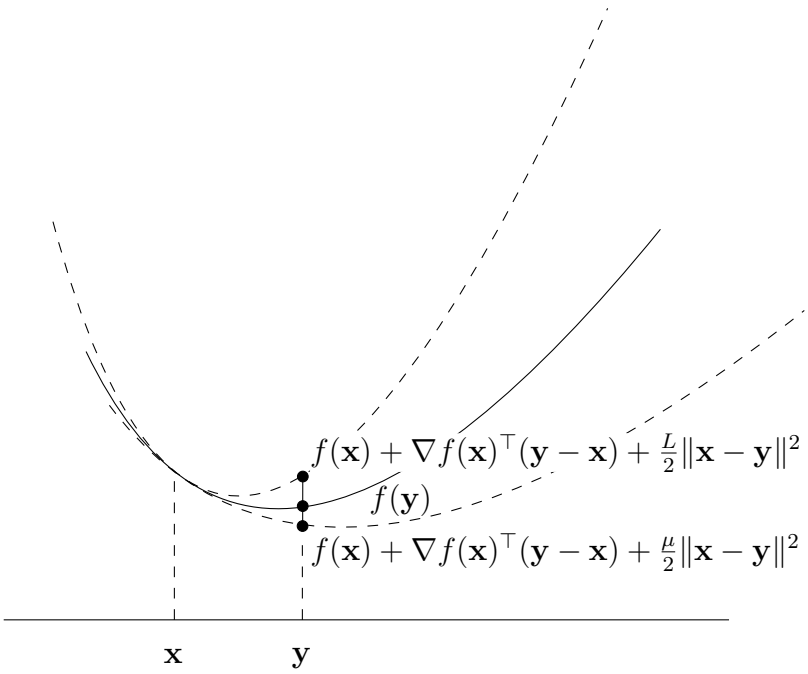


Figure 2.3: A smooth and strongly convex function

Lemma 2.10 (Exercise 19)**.** *If $f : \mathbb{R}^d \to \mathbb{R}$ is strongly convex with parameter $\mu > 0$, then f is strictly convex and has a unique global minimum.*

The supermodel $f(x) = x^2$ is particularly beautiful since it is both smooth and strongly convex with the same parameter $L = \mu = 2$ (going through the calculations in Exercise 13 will reveal this). We can easily characterize the class of particularly beautiful functions. These are exactly the ones whose sublevel sets are ℓ_2-balls.

Lemma 2.11 (Exercise 20)**.** *Let $f : \mathbb{R}^d \to \mathbb{R}$ be strongly convex with parameter $\mu > 0$ and smooth with parameter μ. Prove that f is of the form*

$$f(\mathbf{x}) = \frac{\mu}{2}\|\mathbf{x} - \mathbf{b}\|^2 + c,$$

where $\mathbf{b} \in \mathbb{R}^d, c \in \mathbb{R}$.

Once we have a unique global minimum $\mathbf{x}^\star$, we can attempt to prove that $\lim_{t\to\infty} \mathbf{x}_t = \mathbf{x}^\star$ in gradient descent. We start from the vanilla analysis

(2.3) and plug in the lower bound $\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star) = \nabla f(\mathbf{x}_t)^\top(\mathbf{x}_t - \mathbf{x}^\star) \geq f(\mathbf{x}_t) - f(\mathbf{x}^\star) + \frac{\mu}{2}\|\mathbf{x}_t - \mathbf{x}^\star\|^2$ resulting from strong convexity. We get

$$f(\mathbf{x}_t) - f(\mathbf{x}^\star) \leq \frac{1}{2\gamma}\left(\gamma^2\|\nabla f(\mathbf{x}_t)\|^2 + \|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2\right) - \frac{\mu}{2}\|\mathbf{x}_t - \mathbf{x}^\star\|^2. \tag{2.20}$$

Rewriting this yields a bound on $\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2$ in terms of $\|\mathbf{x}_t - \mathbf{x}^\star\|^2$, along with some "noise" that we still need to take care of:

$$\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \leq 2\gamma(f(\mathbf{x}^\star) - f(\mathbf{x}_t)) + \gamma^2\|\nabla f(\mathbf{x}_t)\|^2 + (1 - \mu\gamma)\|\mathbf{x}_t - \mathbf{x}^\star\|^2. \tag{2.21}$$

Theorem 2.12. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable. Suppose that f is smooth with parameter L according to (3.5) and strongly convex with parameter $\mu > 0$ according to (3.9). Exercise 23 asks you to prove that there is a unique global minimum $\mathbf{x}^\star$ of f. Choosing*

$$\gamma := \frac{1}{L},$$

gradient descent (2.1) with arbitrary $\mathbf{x}_0$ satisfies the following two properties.

(i) Squared distances to $\mathbf{x}^\star$ are geometrically decreasing:

$$\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \leq \left(1 - \frac{\mu}{L}\right)\|\mathbf{x}_t - \mathbf{x}^\star\|^2, \quad t \geq 0.$$

(ii) The absolute error after T iterations is exponentially small in T:

$$f(\mathbf{x}_T) - f(\mathbf{x}^\star) \leq \frac{L}{2}\left(1 - \frac{\mu}{L}\right)^T\|\mathbf{x}_0 - \mathbf{x}^\star\|^2, \quad T > 0.$$

Proof. For (i), we show that the noise in (2.21) disappears. By sufficient decrease (Lemma 2.6), we know that

$$f(\mathbf{x}^\star) - f(\mathbf{x}_t) \leq f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t) \leq -\frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2,$$

and hence the noise can be bounded as follows, using $\gamma = 1/L$, multiplying by 2γ and rearranging the terms, we get:

$$2\gamma\left(f(\mathbf{x}^\star) - f(\mathbf{x}_t)\right) + \gamma^2\|\nabla f(\mathbf{x}_t)\|^2 \leq 0,$$

Hence, (2.21) actually yields

$$\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \leq (1 - \mu\gamma)\|\mathbf{x}_t - \mathbf{x}^\star\|^2 = \left(1 - \frac{\mu}{L}\right)\|\mathbf{x}_t - \mathbf{x}^\star\|^2$$

and

$$\|\mathbf{x}_T - \mathbf{x}^\star\|^2 \leq \left(1 - \frac{\mu}{L}\right)^T \|\mathbf{x}_0 - \mathbf{x}^\star\|^2.$$

The bound in (ii) follows from smoothness (2.8), using $\nabla f(\mathbf{x}^\star) = \mathbf{0}$ (Lemma 1.22):

$$f(\mathbf{x}_T) - f(\mathbf{x}^\star) \leq \nabla f(\mathbf{x}^\star)^\top(\mathbf{x}_T - \mathbf{x}^\star) + \frac{L}{2}\|\mathbf{x}_T - \mathbf{x}^\star\|^2 = \frac{L}{2}\|\mathbf{x}_T - \mathbf{x}^\star\|^2.$$

□

From this, we can derivate a rate in terms of the number of steps required (T). Using the inequality $\ln(1 + x) \leq x$, it follows that after

$$T \geq \frac{L}{\mu} \ln\left(\frac{R^2 L}{2\varepsilon}\right),$$

iterations, we reach absolute error at most ε.

2.9 Exercises

Exercise 12. *Let $\mathbf{c} \in \mathbb{R}^d$. Prove that the spectral norm of $\mathbf{c}^\top$ equals the Euclidean norm of $\mathbf{c}$, meaning that*

$$\max_{\mathbf{x} \neq \mathbf{0}} \frac{|\mathbf{c}^\top \mathbf{x}|}{\|\mathbf{x}\|} = \|\mathbf{c}\|.$$

Exercise 13. *Prove Lemma 2.3: The quadratic function $f(\mathbf{x}) = \mathbf{x}^\top Q\mathbf{x} + \mathbf{b}^\top \mathbf{x} + c$, Q symmetric, is smooth with parameter $2\|Q\|$.*

Exercise 14. *Consider the function $f(x) = |x|^{3/2}$ for $x \in \mathbb{R}$.*

(i) Prove that f is strictly convex and differentiable, with a unique global minimum $x^\star = 0$.

(ii) Prove that for every fixed stepsize γ in gradient descent (2.1) applied to f, there exists x_0 for which $f(x_1) > f(x_0)$.

(iii) Prove that f is not smooth.

(iv) Let $X \subseteq \mathbb{R}$ be a closed convex set such that $0 \in X$ and $X \neq \{0\}$. Prove that f is not smooth over X.

Exercise 15. *In order to obtain average error at most ε in Theorem 2.1, we need to choose iteration number and stepsize as*

$$T \geq \left(\frac{RB}{\varepsilon}\right)^2, \quad \gamma := \frac{R}{B\sqrt{T}}.$$

If R or B are unknown, we cannot do this.

Suppose now that we know R but not B. This means, we know a concrete number R such that $\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$; we also know that there exists a number B such that $\|\nabla f(\mathbf{x})\| \leq B$ for all $\mathbf{x}$, but we don't know a concrete such number.

Develop an algorithm that—not knowing B—finds a vector $\mathbf{x}$ such that $f(\mathbf{x}) - f(\mathbf{x}^\star) < \varepsilon$, using at most

$$\mathcal{O}\left(\left(\frac{RB}{\varepsilon}\right)^2\right)$$

many gradient descent steps!

Exercise 16. *Prove Lemma 2.5! (Operations which preserve smoothness)*

Exercise 17. *In order to obtain average error at most ε in Theorem 2.7, we need to choose*

$$\gamma := \frac{1}{L}, \quad T \geq \frac{R^2 L}{2\varepsilon},$$

if $\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$. If L is unknown, we cannot do this.

Now suppose that we know R but not L. This means, we know a concrete number R such that $\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$; we also know that there exists a number L such that f is smooth with parameter L, but we don't know a concrete such number.

Develop an algorithm that—not knowing L—finds a vector $\mathbf{x}$ such that $f(\mathbf{x}) - f(\mathbf{x}^\star) < \varepsilon$, using at most

$$\mathcal{O}\left(\frac{R^2 L}{2\varepsilon}\right)$$

many gradient descent steps!

Exercise 18. *Let $a \in \mathbb{R}$. Prove that $f(x) = x^4$ is smooth over $X = (-a, a)$ and determine a concrete smoothness parameter L.*

Exercise 19. *Prove Lemma 2.10! (Strongly convex functions have unique global minimum)*

Exercise 20. *Prove Lemma 2.11! (Strongly convex and smooth functions)*

Chapter 3

Projected and Proximal Gradient Descent

Contents

3.1 The Algorithm

Another way to control gradients in (2.4) is to minimize f over a closed convex subset $X \subseteq \mathbb{R}^d$. For example, we may have a constrained optimization problem to begin with (for example the LASSO in Section 1.6.2), or we happen to know some region X containing a global minimum $\mathbf{x}^\star$, so that we can restrict our search to that region. In this case, gradient descent also works, but we need an additional *projection step*. After all, it can happen that some iteration of (2.1) takes us "into the wild" (out of X) where we have no business to do. *Projected* gradient descent is the following modification. We choose $\mathbf{x}_0 \in X$ arbitrary and for $t \geq 0$ define

$$\mathbf{y}_{t+1} := \mathbf{x}_t - \gamma \nabla f(\mathbf{x}_t), \tag{3.1}$$

$$\mathbf{x}_{t+1} := \Pi_X(\mathbf{y}_{t+1}) := \operatorname*{argmin}_{\mathbf{x} \in X} \|\mathbf{x} - \mathbf{y}_{t+1}\|^2. \tag{3.2}$$

This means, after each iteration, we project the obtained iterate $\mathbf{y}_{t+1}$ back to X. This may be very easy (think of X as the unit ball in which case we just have to scale $\mathbf{y}_{t+1}$ down to length 1 if it is longer). But it may also be very difficult. In general, computing $\Pi_X(\mathbf{y}_{t+1})$ means to solve an auxiliary convex constrained minimization problem in each step! Here, we are just assuming that we can do this. The projection is well-defined: the squared distance function $d_{\mathbf{y}}(\mathbf{x}) := \|\mathbf{x} - \mathbf{y}\|^2$ is strongly convex, and hence, a unique minimum over the nonempty closed and convex set X exists by Exercise 23.

We note that finding an initial $\mathbf{x}_0 \in X$ also reduces to projection (of $\mathbf{0}$, for example) onto X.

We will frequently need the following

Fact 3.1. *Let $X \subseteq \mathbb{R}^d$ be closed and convex, $\mathbf{x} \in X, \mathbf{y} \in \mathbb{R}^d$. Then*

(i) $(\mathbf{x} - \Pi_X(\mathbf{y}))^\top (\mathbf{y} - \Pi_X(\mathbf{y})) \leq 0$.

(ii) $\|\mathbf{x} - \Pi_X(\mathbf{y})\|^2 + \|\mathbf{y} - \Pi_X(\mathbf{y})\|^2 \leq \|\mathbf{x} - \mathbf{y}\|^2$.

Part (i) says that the vectors $\mathbf{x} - \Pi_X(\mathbf{y})$ and $\mathbf{y} - \Pi_X(\mathbf{y})$ form an obtuse angle, and (ii) equivalently says that the square of the long side $\mathbf{x} - \mathbf{y}$ in the triangle formed by the three points is at least the sum of squares of the two short sides; see Figure 3.1.

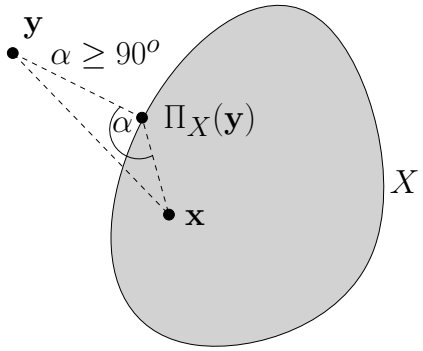


Figure 3.1: Illustration of Fact 3.1

Proof. $\Pi_X(\mathbf{y})$ is by definition a minimizer of the (differentiable) convex function $d_{\mathbf{y}}(\mathbf{x}) = \|\mathbf{x} - \mathbf{y}\|^2$ over X, and (i) is just the equivalent optimality condition of Lemma 1.27. We need X to be closed in the first place in order to ensure that we can project onto X (see Exercise 23 applied with $d_{\mathbf{y}}(\mathbf{x})$). Indeed, for example, the number 1 has no closest point in the set $[-\infty, 0) \in \mathbb{R}$. Part (ii) follows from (i) via the (by now well-known) equation $2\mathbf{v}^\top \mathbf{w} = \|\mathbf{v}\|^2 + \|\mathbf{w}\|^2 - \|\mathbf{v} - \mathbf{w}\|^2$. □

Exercise 21 asks you to prove that if $\mathbf{x}_{t+1} = \mathbf{x}_t$ in projected gradient descent (i.e. we project back to the previous iterate), then $\mathbf{x}_t$ is a minimizer of f over X.

3.2 Bounded gradients: $\mathcal{O}(1/\varepsilon^2)$ steps

As in the unconstrained case, let us first assume that gradients are bounded by a constant B—this time over X. This implies that f is B-Lipschitz over X (see Theorem 1.9), but the converse may not hold.

If we minimize f over a closed and *bounded* (= compact) convex set X, we get the existence of a minimizer and a bound R for the initial distance to it for free; assuming that f is *continuously* differentiable, we also have a bound B for the gradient norms over X. This is because then $\mathbf{x} \mapsto \|\nabla f(\mathbf{x})\|$ is a continuous function that attains a maximum over X. In this case, our vanilla analysis yields a much more useful result than the one in Theorem 2.1, with the same stepsize and the same number of steps.

Theorem 3.2. *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be convex and differentiable, $X \subseteq \mathbf{dom}(f)$ closed and convex, $\mathbf{x}^\star$ a minimizer of f over X; furthermore, suppose that $\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$, and that $\|\nabla f(\mathbf{x})\| \leq B$ for all $\mathbf{x} \in X$. Choosing the constant stepsize*

$$\gamma := \frac{R}{B\sqrt{T}},$$

projected gradient descent (3.1) with $\mathbf{x}_0 \in X$ yields

$$\frac{1}{T}\sum_{t=0}^{T-1}(f(\mathbf{x}_t) - f(\mathbf{x}^\star)) \leq \frac{RB}{\sqrt{T}}.$$

Proof. The only required changes to the vanilla analysis are that in steps (2.2) and (2.3), $\mathbf{x}_{t+1}$ needs to be replaced by $\mathbf{y}_{t+1}$ as this is the real next (non-projected) gradient descent iterate after these steps; we therefore get

$$\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star) = \frac{1}{2\gamma}\left(\gamma^2\|\mathbf{g}_t\|^2 + \|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{y}_{t+1} - \mathbf{x}^\star\|^2\right). \tag{3.3}$$

From Fact 3.1 (ii) (with $\mathbf{x} = \mathbf{x}^\star, \mathbf{y} = \mathbf{y}_{t+1}$), we obtain $\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \leq \|\mathbf{y}_{t+1} - \mathbf{x}^\star\|^2$, hence we get

$$\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star) \leq \frac{1}{2\gamma}\left(\gamma^2\|\mathbf{g}_t\|^2 + \|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2\right) \tag{3.4}$$

and return to the previous vanilla analysis for the remainder of the proof. □

3.3 Smooth convex functions: $\mathcal{O}(1/\varepsilon)$ steps

We recall from Definition 2.2 that f that is smooth over X if

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x}) + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in X. \tag{3.5}$$

To minimize f over X, we use projected gradient descent again. The runtime turns out to be the same as in the unconstrained case. Again, we have sufficient decrease. This is not obvious from the following lemma, but you are asked to prove it in Exercise 22.

Lemma 3.3. *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be differentiable and smooth with parameter L over a closed and convex set $X \subseteq \mathbf{dom}(f)$, according to (3.5). Choosing stepsize*

$$\gamma := \frac{1}{L},$$

projected gradient descent (3.1) with arbitrary $\mathbf{x}_0 \in X$ satisfies

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) - \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2 + \frac{L}{2}\|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2, \quad t \geq 0.$$

More specifically, this already holds if f is smooth with parameter L over the line segment connecting $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$.

Proof. We proceed similar to the proof of the "unconstrained" sufficient decrease Lemma 2.6, except that we now need to deal with projected gradient descent. We again start from smoothness but then use $\mathbf{y}_{t+1} = \mathbf{x}_t - \nabla f(\mathbf{x}_t)/L$, followed by the usual equation $2\mathbf{v}^\top\mathbf{w} = \|\mathbf{v}\|^2 + \|\mathbf{w}\|^2 - \|\mathbf{v}-\mathbf{w}\|^2$:

$$\begin{aligned}
f(\mathbf{x}_{t+1}) &\leq f(\mathbf{x}_t) + \nabla f(\mathbf{x}_t)^\top(\mathbf{x}_{t+1} - \mathbf{x}_t) + \frac{L}{2}\|\mathbf{x}_t - \mathbf{x}_{t+1}\|^2 \\
&= f(\mathbf{x}_t) - L(\mathbf{y}_{t+1} - \mathbf{x}_t)^\top(\mathbf{x}_{t+1} - \mathbf{x}_t) + \frac{L}{2}\|\mathbf{x}_t - \mathbf{x}_{t+1}\|^2 \\
&= f(\mathbf{x}_t) - \frac{L}{2}\left(\|\mathbf{y}_{t+1} - \mathbf{x}_t\|^2 + \|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 - \|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2\right) \\
&\quad + \frac{L}{2}\|\mathbf{x}_t - \mathbf{x}_{t+1}\|^2 \\
&= f(\mathbf{x}_t) - \frac{L}{2}\|\mathbf{y}_{t+1} - \mathbf{x}_t\|^2 + \frac{L}{2}\|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2 \\
&= f(\mathbf{x}_t) - \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2 + \frac{L}{2}\|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2.
\end{aligned}$$

□

Theorem 3.4. *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be convex and differentiable. Let $X \subseteq \mathbf{dom}(f)$ be a closed convex set, and assume that there is a minimizer $\mathbf{x}^\star$ of f over X; furthermore, suppose that f is smooth over X with parameter L according to (3.5). Choosing stepsize*

$$\gamma := \frac{1}{L},$$

projected gradient descent (3.1) with $\mathbf{x}_0 \in X$ satisfies

$$f(\mathbf{x}_T) - f(\mathbf{x}^\star) \leq \frac{L}{2T}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2, \quad T > 0.$$

Proof. The plan is as in the proof of Theorem 2.7 to use the inequality

$$\frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2 \leq f(\mathbf{x}_t) - f(\mathbf{x}_{t+1}) + \frac{L}{2}\|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2 \tag{3.6}$$

resulting from sufficient decrease (Lemma 3.3) to bound the squared gradient $\|\mathbf{g}_t\|^2 = \|\nabla f(\mathbf{x}_t)\|^2$ in the vanilla analysis. Unfortunately, (3.6) has an extra term compared to what we got in the unconstrained case. But we can compensate for this in the vanilla analysis itself. Let us go back to its "constrained" version (3.3), featuring $\mathbf{y}_{t+1}$ instead of $\mathbf{x}_{t+1}$:

$$\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star) = \frac{1}{2\gamma}\left(\gamma^2\|\mathbf{g}_t\|^2 + \|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{y}_{t+1} - \mathbf{x}^\star\|^2\right).$$

Previously, we applied $\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \leq \|\mathbf{y}_{t+1} - \mathbf{x}^\star\|^2$ (Fact 3.1(ii)) to get back on the unconstrained vanilla track. But in doing so, we dropped a term that now becomes useful. Indeed, Fact 3.1(ii) actually yields $\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 + \|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2 \leq \|\mathbf{y}_{t+1} - \mathbf{x}^\star\|^2$, so that we get the following upper bound for $\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)$:

$$\frac{1}{2\gamma}\left(\gamma^2\|\mathbf{g}_t\|^2 + \|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 - \|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2\right). \tag{3.7}$$

Using $f(\mathbf{x}_t) - f(\mathbf{x}^\star) \leq \mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)$ from convexity, we have (with $\gamma = 1/L$) that

$$\begin{aligned}\sum_{t=0}^{T-1}(f(\mathbf{x}_t) - f(\mathbf{x}^\star)) &\leq \sum_{t=0}^{T-1}\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star) \qquad (3.8)\\ &\leq \frac{1}{2L}\sum_{t=0}^{T-1}\|\mathbf{g}_t\|^2 + \frac{L}{2}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2 - \frac{L}{2}\sum_{t=0}^{T-1}\|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2.\end{aligned}$$

To bound the sum of the squared gradients, we use (3.6):

$$\begin{aligned}\frac{1}{2L}\sum_{t=0}^{T-1}\|\mathbf{g}_t\|^2 &\leq \sum_{t=0}^{T-1}\left(f(\mathbf{x}_t) - f(\mathbf{x}_{t+1}) + \frac{L}{2}\|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2\right)\\ &= f(\mathbf{x}_0) - f(\mathbf{x}_T) + \frac{L}{2}\sum_{t=0}^{T-1}\|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2.\end{aligned}$$

Plugging this into (3.8), the extra terms cancel, and we arrive—as in the unconstrained case—at

$$\sum_{t=1}^{T} (f(\mathbf{x}_t) - f(\mathbf{x}^\star)) \leq \frac{L}{2} \|\mathbf{x}_0 - \mathbf{x}^\star\|^2.$$

The statement follows as in the proof of Theorem 2.7 from the fact that due to sufficient decrease (Exercise 22), the last iterate is the best one. □

3.4 Smooth and strongly convex functions: $\mathcal{O}(\log(1/\varepsilon))$ steps

Assuming that f is smooth *and* strongly convex over a set X, we can also prove fast convergence of projected gradient descent. This does not require any new ideas, we have seen all the ingredients before.

We recall from Definition 2.9 that f is strongly convex with parameter $\mu > 0$ over X if

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{\mu}{2} \|\mathbf{x} - \mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in X. \tag{3.9}$$

Theorem 3.5. *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be convex and differentiable. Let $X \subseteq \mathbf{dom}(f)$ be a nonempty closed and convex set and suppose that f is smooth over X with parameter L according to (3.5) and strongly convex over X with parameter $\mu > 0$ according to (3.9). Exercise 23 asks you to prove that there is a unique minimizer $\mathbf{x}^\star$ of f over X. Choosing*

$$\gamma := \frac{1}{L},$$

projected gradient descent (3.1) with arbitrary $\mathbf{x}_0$ satisfies the following two properties.

(i) Squared distances to $\mathbf{x}^\star$ are geometrically decreasing:

$$\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \leq \left(1 - \frac{\mu}{L}\right) \|\mathbf{x}_t - \mathbf{x}^\star\|^2, \quad t \geq 0.$$

(ii) The absolute error after T iterations is exponentially small in T:

$$\begin{aligned} f(\mathbf{x}_T) - f(\mathbf{x}^\star) &\leq \|\nabla f(\mathbf{x}^\star)\| \left(1 - \frac{\mu}{L}\right)^{T/2} \|\mathbf{x}_0 - \mathbf{x}^\star\| \\ &+ \frac{L}{2} \left(1 - \frac{\mu}{L}\right)^{T} \|\mathbf{x}_0 - \mathbf{x}^\star\|^2, \quad T > 0. \end{aligned}$$

We note that this is *almost* the same result as in Theorem 2.12 for the unconstrained case; in fact, the result in part (i) is identical, but in part (ii), we get an additional term. This is due to the fact that in the constrained case, we cannot argue that $\nabla f(\mathbf{x}^\star) = \mathbf{0}$. In fact, this additional term is the dominating one, once the error becomes small. It has the effect that the required number of steps to reach error at most ε will roughly double, in comparison to the bound of Theorem 2.12.

Proof. In the strongly convex case, the "constrained" vanilla bound (3.7)

$$\frac{1}{2\gamma}\left(\gamma^2\|\nabla f(\mathbf{x}_t)\|^2 + \|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 - \|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2\right)$$

on $f(\mathbf{x}_t) - f(\mathbf{x}^\star)$ can be strengthened to

$$\frac{1}{2\gamma}\left(\gamma^2\|\nabla f(\mathbf{x}_t)\|^2 + \|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 - \|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2\right) - \frac{\mu}{2}\|\mathbf{x}_t - \mathbf{x}^\star\|^2 \tag{3.10}$$

Now we proceed as in the proof of Theorem 2.12 and rewrite the latter bound into a bound on $\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2$ that is

$$2\gamma(f(\mathbf{x}^\star) - f(\mathbf{x}_t)) + \gamma^2\|\nabla f(\mathbf{x}_t)\|^2 - \|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2 + (1 - \mu\gamma)\|\mathbf{x}_t - \mathbf{x}^\star\|^2,$$

so we have geometric decrease in squared distance to $\mathbf{x}^\star$, up to some noise. Again, we show that by sufficient decrease, the noise in this bound disappears. From Lemma 3.3, we know that

$$f(\mathbf{x}^\star) - f(\mathbf{x}_t) \leq f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t) \leq -\frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|^2 + \frac{L}{2}\|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2,$$

and using this, the noise can be bounded. Multiplying the previous inequality by $2/L$, and rearranging the terms we get:

$$\frac{2}{L}\left(f(\mathbf{x}^\star) - f(\mathbf{x}_t)\right) + \frac{1}{L^2}\|\nabla f(\mathbf{x}_t)\|^2 - \|\mathbf{y}_{t+1} - \mathbf{x}_{t+1}\|^2 \leq 0.$$

With $\gamma = 1/L$, this exactly shows that the noise is nonpositive. This yields (i). The bound in (ii) follows from smoothness (2.8):

$$\begin{aligned}
f(\mathbf{x}_T) - f(\mathbf{x}^\star) &\leq \nabla f(\mathbf{x}^\star)^\top(\mathbf{x}_T - \mathbf{x}^\star) + \frac{L}{2}\|\mathbf{x}^\star - \mathbf{x}_T\|^2 \\
&\leq \|\nabla f(\mathbf{x}^\star)\|\,\|\mathbf{x}_T - \mathbf{x}^\star\| + \frac{L}{2}\|\mathbf{x}^\star - \mathbf{x}_T\|^2 \text{ (Cauchy-Schwarz)} \\
&\leq \|\nabla f(\mathbf{x}^\star)\|\left(1 - \frac{\mu}{L}\right)^{T/2}\|\mathbf{x}_0 - \mathbf{x}^\star\| + \frac{L}{2}\left(1 - \frac{\mu}{L}\right)^{T}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2.
\end{aligned}$$

□

3.5 Projecting onto ℓ_1-balls

Problems that are ℓ_1-regularized appear among the most commonly used models in machine learning and signal processing, and we have already discussed the Lasso as an important example of that class. We will now address how to perform projected gradient as an efficient optimization for ℓ_1-constrained problems. Let

$$X = B_1(R) := \left\{ \mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\|_1 = \sum_{i=1}^{d} |x_i| \leq R \right\}$$

be the ℓ_1-ball of radius $R > 0$ around $\mathbf{0}$, i.e., the set of all points with 1-norm at most R. Our goal is to compute $\Pi_X(\mathbf{v})$ for a given vector $\mathbf{v}$, i.e. the projection of $\mathbf{v}$ onto X; see Figure 3.2.

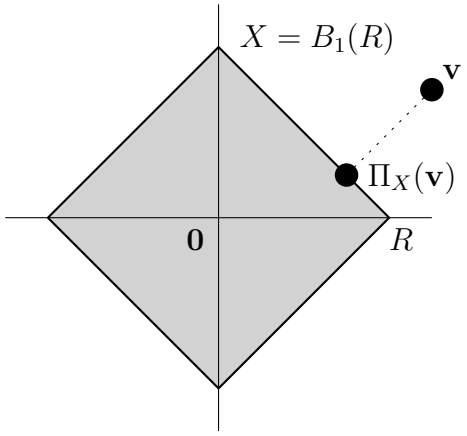


Figure 3.2: Projecting onto an ℓ_1-ball

At first sight, this may look like a rather complicated task. Geometrically, X is a *cross polytope* (square for $d = 2$, octahedron for $d = 3$), and as such it has 2^d many facets. But we can start with some basic simplifying observations.

Fact 3.6. *We may assume without loss of generality that (i) $R = 1$, (ii) $v_i \geq 0$ for all i, and (iii) $\sum_{i=1}^{d} v_i > 1$.*

Proof. If we project $\mathbf{v}/R$ onto $B_1(1)$, we obtain $\Pi_X(\mathbf{v})/R$ (just scale Figure 3.2), so we can restrict to the case $R = 1$. For (ii), we observe that simultaneously flipping the signs of a fixed subset of coordinates in both $\mathbf{v}$ and $\mathbf{x} \in X$ yields vectors $\mathbf{v}'$ and $\mathbf{x}' \in X$ such that $\|\mathbf{x} - \mathbf{v}\| = \|\mathbf{x}' - \mathbf{v}'\|$; thus, $\mathbf{x}$ minimizes the distance to $\mathbf{v}$ if and only if $\mathbf{x}'$ minimizes the distance to $\mathbf{v}'$. Hence, it suffices to compute $\Pi_X(\mathbf{v})$ for vectors with nonnegative entries. If $\sum_{i=1}^d v_i \leq 1$, we have $\Pi_X(\mathbf{v}) = \mathbf{v}$ and are done, so the interesting case is (iii). □

Fact 3.7. *Under the assumptions of Fact 3.6, $\mathbf{x} = \Pi_X(\mathbf{v})$ satisfies $x_i \geq 0$ for all i and $\sum_{i=1}^d x_i = 1$.*

Proof. If $x_i < 0$ for some i, then $(-x_i - v_i)^2 \leq (x_i - v_i)^2$ (since $v_i \geq 0$), so flipping the i-th sign in $\mathbf{x}$ would yield another vector in X at least as close to $\mathbf{v}$ as $\mathbf{x}$, but such a vector cannot exist by strict convexity of the squared distance. And if $\sum_{i=1}^d x_i < 1$, then $\mathbf{x}' = \mathbf{x} + \lambda(\mathbf{v} - \mathbf{x}) \in X$ for some small positive λ, with $\|\mathbf{x}' - \mathbf{v}\| = (1 - \lambda)\|\mathbf{x} - \mathbf{v}\|$, again contradicting the optimality of $\mathbf{x}$. □

Corollary 3.8. *Under the assumptions of Fact 3.6,*

$$\Pi_X(\mathbf{v}) = \operatorname*{argmin}_{\mathbf{x} \in \Delta_d} \|\mathbf{x} - \mathbf{v}\|^2,$$

where

$$\Delta_d := \left\{ \mathbf{x} \in \mathbb{R}^d : \sum_{i=1}^d x_i = 1, x_i \geq 0 \ \forall i \right\}$$

is the standard simplex.

This means, we have reduced the projection onto an ℓ_1-ball to the projection onto the standard simplex; see Figure 3.3.

To address the latter task, we make another assumption that can be established by suitably permuting the entries of $\mathbf{v}$ (which just permutes the entries of its projection onto Δ_d in the same way).

Fact 3.9. *We may assume without loss of generality that $v_1 \geq v_2 \geq \cdots \geq v_d$.*

Lemma 3.10. *Let $\mathbf{x}^\star := \operatorname{argmin}_{\mathbf{x} \in \Delta_d} \|\mathbf{x} - \mathbf{v}\|^2$. Under the assumption of Fact 3.9, there exists (a unique) $p \in \{1, \ldots, d\}$ such that*

$$\begin{aligned} x_i^\star &> 0, \quad i \leq p, \\ x_i^\star &= 0, \quad i > p. \end{aligned}$$

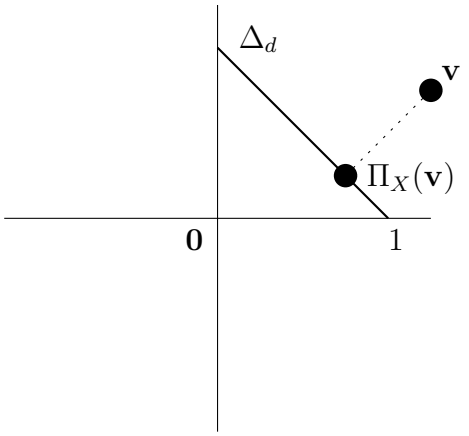


Figure 3.3: Projecting onto the standard simplex

Proof. We are using the optimality criterion of Lemma 1.27:

$$\nabla d_{\mathbf{v}}(\mathbf{x}^\star)^\top(\mathbf{x}-\mathbf{x}^\star) = 2(\mathbf{x}^\star-\mathbf{v})^\top(\mathbf{x}-\mathbf{x}^\star) \geq 0, \quad \mathbf{x}\in\Delta_d, \tag{3.11}$$

where $d_{\mathbf{v}}(\mathbf{z}) := \|\mathbf{z}-\mathbf{v}\|^2$ is the squared distance to $\mathbf{v}$.

Because $\sum_{i=1}^d x_i^\star = 1$, there is at least one positive entry in $\mathbf{x}^\star$. It remains to show that we cannot have $x_i^\star = 0$ and $x_{i+1}^\star > 0$. Indeed, in this situation, we could decrease $x_{i+1}^\star$ by some small positive ε and simultaneously increase $x_i^\star$ to ε to obtain a vector $\mathbf{x}\in\Delta_d$ such that

$$(\mathbf{x}^\star-\mathbf{v})^\top(\mathbf{x}-\mathbf{x}^\star) = (0-v_i)\varepsilon - (x_{i+1}^\star - v_{i+1})\varepsilon = \varepsilon(\underbrace{v_{i+1}-v_i}_{\leq 0} - \underbrace{x_{i+1}^\star}_{>0}) < 0,$$

contradicting the optimality (3.11). □

But we can say even more about $\mathbf{x}^\star$.

Lemma 3.11. *Under the assumption of Fact 3.9, and with p as in Lemma 3.10,*

$$x_i^\star = v_i - \Theta_p, \quad i\leq p,$$

where

$$\Theta_p = \frac{1}{p}\Big(\sum_{i=1}^p v_i - 1\Big).$$

Proof. Again, we argue by contradiction. If not all $x_i^\star - v_i, i \le p$ have the same value $-\Theta_p$, then we have $x_i^\star - v_i < x_j^\star - v_j$ for some $i, j \le p$. As before, we can then decrease $x_j^\star > 0$ by some small positive ε and simultaneously increase $x_i^\star$ by ε to obtain $\mathbf{x} \in \Delta_d$ such that

$$(\mathbf{x}^\star - \mathbf{v})^\top (\mathbf{x} - \mathbf{x}^\star) = (x_i^\star - v_i)\varepsilon - (x_j^\star - v_j)\varepsilon = \varepsilon \underbrace{((x_i^\star - v_i) - (x_j^\star - v_j))}_{<0} < 0,$$

again contradicting (3.11). The expression for Θ_p is then obtained from

$$1 = \sum_{i=1}^p x_i^\star = \sum_{i=1}^p (v_i - \Theta_p) = \sum_{i=1}^p v_i - p\Theta_p.$$

□

Let us summarize the situation: we now have d candidates for $\mathbf{x}^\star$, namely the vectors

$$\mathbf{x}^\star(p) := (v_1 - \Theta_p, \dots, v_p - \Theta_p, 0, \dots, 0), \quad p \in \{1, \dots, d\}, \tag{3.12}$$

and we just need to find the right one. In order for candidate $\mathbf{x}^\star(p)$ to comply with Lemma 3.10, we must have

$$v_p - \Theta_p > 0, \tag{3.13}$$

and this actually ensures $\mathbf{x}^\star(p)_i > 0$ for all $i \le p$ by the assumption of Fact 3.9 and therefore $\mathbf{x}^\star(p) \in \Delta_d$. But there could still be several values of p satisfying (3.13). Among them, we simply pick the one for which $\mathbf{x}^\star(p)$ minimizes the distance to $\mathbf{v}$. It is not hard to see that this can be done in time $\mathcal{O}(d \log d)$, by first sorting v and then carefully updating the values Θ_p and $\|\mathbf{x}^\star(p) - \mathbf{v}\|^2$ as we vary p to check all candidates.

But actually, there is an even simpler criterion that saves us from comparing distances.

Lemma 3.12. *Under the assumption of Fact 3.9, with $\mathbf{x}^\star(p)$ as in (3.12), and with*

$$p^\star := \max\Big\{p \in \{1, \dots, d\} : v_p - \frac{1}{p}\Big(\sum_{i=1}^p v_i - 1\Big) > 0\Big\},$$

it holds that

$$\operatorname*{argmin}_{\mathbf{x} \in \Delta_d} \|\mathbf{x} - \mathbf{v}\|^2 = \mathbf{x}^\star(p^\star).$$

The proof is Exercise 24. Together with our previous reductions, we obtain the following result.

Theorem 3.13. *Let $\mathbf{v} \in \mathbb{R}^d$, $R \in \mathbb{R}_+$, $X = B_1(R)$ the ℓ_1-ball around $\mathbf{0}$ of radius R. The projection*

$$\Pi_X(\mathbf{v}) = \operatorname*{argmin}_{\mathbf{x} \in X} \|\mathbf{x} - \mathbf{v}\|^2$$

of $\mathbf{v}$ onto $B_1(R)$ can be computed in time $\mathcal{O}(d \log d)$.

This can be improved to time $\mathcal{O}(d)$, based on the observation that a given p can be compared to the value $p^\star$ in Lemma 3.12 in linear time, without the need to presort $\mathbf{v}$ [DSSSC08].

3.6 Proximal gradient descent

Many optimization problems in applications come with additional structure. An important class of objective functions is composed as

$$f(\mathbf{x}) := g(\mathbf{x}) + h(\mathbf{x}) \tag{3.14}$$

where g is a "nice" function, where as h is a "simple" additional term, which however doesn't satisfy the assumptions of niceness which we used in the convergence analysis so far. In particular, an important case is when h is not differentiable.

The classical gradient step for unconstrained minimization of a function g can be equivalently written as

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{y} \in \mathbb{R}^d} \; g(\mathbf{x}_t) + \nabla g(\mathbf{x}_t)^\top (\mathbf{y} - \mathbf{x}_t) + \frac{1}{2\gamma} \|\mathbf{y} - \mathbf{x}_t\|^2 \tag{3.15}$$

$$= \operatorname*{argmin}_{\mathbf{y} \in \mathbb{R}^d} \; \frac{1}{2\gamma} \|\mathbf{y} - (\mathbf{x}_t - \gamma \nabla g(\mathbf{x}_t))\|^2 \,. \tag{3.16}$$

To obtain the last equality, we have just completed the quadratic $\|\mathbf{v}\|^2 + 2\mathbf{v}^\top \mathbf{w} + \|\mathbf{w}\|^2 = \|\mathbf{v} + \mathbf{w}\|^2$ for $\mathbf{v} := \gamma \nabla g(\mathbf{x}_t)$ and $\mathbf{w} := \mathbf{y} - \mathbf{x}_t$. Here it is crucial that $\mathbf{v}$ is independent of the optimization variable $\mathbf{y}$, so therefore the term can be ignored when taking the argmin. The scaling by $\frac{1}{2\gamma}$ is also irrelevant but we keep it for better illustrating the next step.

The interpretation of the above equivalent reformulation of the classic gradient step is important for us, and is what has enabled the previous convergence analysis in Section 2.5 for smooth unconstrained optimization: For the particular choice of stepsize $\gamma := \frac{1}{L}$ which we have used, the above formulation shows that the gradient descent step exactly minimizes the local quadratic model of g at our current iterate $\mathbf{x}_t$, formed by the smoothness property with parameter L as defined in (2.8).

Our goal in this section is to minimize $f = g + h$, instead of only the smooth part g alone. The idea of the proximal gradient method is to modify the simple quadratic model (3.15) above, so as to make it a valid model for f, that is a model which upper bounds f at all points. The simplest way to do this is to just treat the h function separately by adding it unmodified. We obtain the update equation for *proximal gradient descent*

$$\mathbf{x}_{t+1} := \operatorname*{argmin}_{\mathbf{y}\in\mathbb{R}^d} \; g(\mathbf{x}_t) + \nabla g(\mathbf{x}_t)^\top(\mathbf{y}-\mathbf{x}_t) + \frac{1}{2\gamma}\|\mathbf{y}-\mathbf{x}_t\|^2 + h(\mathbf{y}) \tag{3.17}$$

$$= \operatorname*{argmin}_{\mathbf{y}} \; \frac{1}{2\gamma}\|\mathbf{y}-(\mathbf{x}_t - \gamma\nabla g(\mathbf{x}_t))\|^2 + h(\mathbf{y}) \; . \tag{3.18}$$

The last formulation makes clear that the resulting update tries to combine the two goals, staying close to the classic gradient update, as well as also to minimize h.

3.6.1 The proximal gradient algorithm

We define the *proximal mapping* for a given function h, and parameter $\gamma > 0$:

$$\operatorname{prox}_{h,\gamma}(\mathbf{z}) := \operatorname*{argmin}_{\mathbf{y}} \left\{ \frac{1}{2\gamma}\|\mathbf{y}-\mathbf{z}\|^2 + h(\mathbf{y}) \right\}$$

An iteration of *proximal gradient descent* is defined as

$$\mathbf{x}_{t+1} := \operatorname{prox}_{h,\gamma}(\mathbf{x}_t - \gamma\nabla g(\mathbf{x}_t)) \; . \tag{3.19}$$

This same update step can also be written in different form as

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \gamma G_\gamma(\mathbf{x}_t) \tag{3.20}$$

for $G_{h,\gamma}(\mathbf{x}) := \frac{1}{\gamma}\Big(\mathbf{x} - \operatorname{prox}_{h,\gamma}(\mathbf{x} - \gamma\nabla g(\mathbf{x}))\Big)$ being the so called generalized gradient of f.

A generalization of gradient descent. The proximal gradient descent method (3.19) is also known as generalized gradient descent. In the special case $h \equiv 0$, we of course recover classic gradient descent.

More interestingly, it is also a generalization of projected gradient descent as we have discussed in the previous sections. Given a closed convex set X, the *indicator function* of the set X is given as the convex function

$$\begin{aligned} \boldsymbol{\iota}_X : \mathbb{R}^d &\to \mathbb{R} \cup +\infty \\ \mathbf{x} &\mapsto \boldsymbol{\iota}_X(\mathbf{x}) := \begin{cases} 0 & \text{if } \mathbf{x} \in X, \\ +\infty & \text{otherwise.} \end{cases} \end{aligned} \tag{3.21}$$

When using the indicator function of our constraint set X as $h \equiv \boldsymbol{\iota}_X$, it is easy to see that the proximal mapping simply becomes

$$\begin{aligned} \operatorname{prox}_{h,\gamma}(\mathbf{z}) &:= \operatorname*{argmin}_{\mathbf{y}} \left\{ \frac{1}{2\gamma} \|\mathbf{y} - \mathbf{z}\|^2 + \boldsymbol{\iota}_X(\mathbf{y}) \right\} \\ &= \operatorname*{argmin}_{\mathbf{y} \in X} \ \|\mathbf{y} - \mathbf{z}\|^2 = \Pi_X(\mathbf{z}) \,, \end{aligned}$$

which is the projection of $\mathbf{z}$ onto X.

As we will see, the convergence of proximal gradient will be as fast as classic gradient descent. However, this still comes not entirely for free. In every iteration, we now have to additionally compute the proximal mapping. This can be very expensive if h is complex. Nevertheless, for some important examples of h the proximal mapping is efficient to compute, such as for the ℓ_1-norm.

3.6.2 Convergence in $\mathcal{O}(1/\varepsilon)$ steps

Interestingly, the vanilla convergence analysis for smooth functions as in Theorem 2.7 directly applies for the more general case of proximal gradient descent. Intuitively, this means that proximal method only "sees" the nice smooth part g of the objective, and is not impacted by the additional h which it treats separately in each step.

Theorem 3.14. *Let $g : \mathbb{R}^d \to \mathbb{R}$ be convex and smooth with parameter L, and also h convex and $prox_{h,\gamma}(\mathbf{x}) := \operatorname{argmin}_{\mathbf{z}}\{\|\mathbf{x} - \mathbf{z}\|^2/(2\gamma) + h(\mathbf{z})\}$ can be computed. Choosing the fixed stepsize*

$$\gamma := \frac{1}{L},$$

proximal gradient descent (3.19) with arbitrary $\mathbf{x}_0$ satisfies

$$f(\mathbf{x}_T) - f(\mathbf{x}^\star) \leq \frac{L}{2T}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2, \quad T > 0.$$

Proof. The proof follows the vanilla analysis for the smooth case, applying it only to g, while always keeping h separate, as in (3.17). We leave the details as Exercise 25 for the reader. □

3.7 Exercises

Exercise 21. *Consider the projected gradient descent algorithm as in (3.1) and (3.2), with a convex differentiable function f. Suppose that for some iteration t, $\mathbf{x}_{t+1} = \mathbf{x}_t$. Prove that in this case, $\mathbf{x}_t$ is a minimizer of f over the closed and convex set X!*

Exercise 22. *Prove that in Theorem 3.4 (i),*

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t).$$

Exercise 23. *Let $X \subseteq \mathbb{R}^d$ be a nonempty closed and convex set, and let f be strongly convex over X. Prove that f has a unique minimizer $\mathbf{x}^\star$ over X! In particular, for $X = \mathbb{R}^d$, we obtain the existence of a unique global minimum.*

Exercise 24. *Prove Lemma 3.12!*

Hint: *It is useful to prove that with $\mathbf{x}^\star(p)$ as in (3.12) and satisfying (3.13),*

$$\mathbf{x}^\star(p) = \operatorname{argmin}\{\|\mathbf{x} - \mathbf{v}\| : \sum_{i=1}^{d} x_i = 1, x_{p+1} = \cdots = x_d = 0\}.$$

Exercise 25. *Prove Theorem 3.14!*

Chapter 4

Subgradient Descent

Contents

4.1 Subgradients

Definition 4.1. *Let* $f : \mathbf{dom}(f) \to \mathbb{R}$. *Then* $\mathbf{g} \in \mathbb{R}^d$ *is a* subgradient *of* f *at* $\mathbf{x} \in \mathbf{dom}(f)$ *if*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top(\mathbf{y} - \mathbf{x}) \quad \forall \mathbf{y} \in \mathbf{dom}(f). \tag{4.1}$$

The set of subgradients of f *at* $\mathbf{x}$ *is called the* subdifferential *at* $\mathbf{x}$ *and is denoted by* $\partial f(\mathbf{x})$.

The notion of a subgradient can be seen as a generalization of the gradient, for functions which are not necessarily differentiable. A prominent example is the ℓ_1-norm, which we have discussed in Exercise 8. Figure 4.1 shows that this function has several subgradients at $x = 0$ (one-dimensional case).

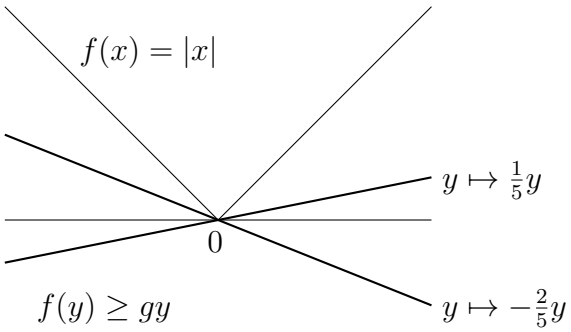


Figure 4.1: The function $f(x) = |x|$ has subgradients $g \in [-1, 1]$ at 0, since $f(y) \geq gy$ for exactly $g \in [-1, 1]$.

Lemma 4.2 (Exercise 26)**.** *If* $f : \mathbf{dom}(f) \to \mathbb{R}$ *is differentiable at* $\mathbf{x} \in \mathbf{dom}(f)$, *then* $\partial f(\mathbf{x}) \subseteq \{\nabla f(\mathbf{x})\}$.

This means that in the differentiable case, there is either exactly one subgradient $\nabla f(\mathbf{x})$, or no subgradient at all (if f is *not* above its tangent hyperplane at $\mathbf{x}$; see Figure 1.1).

Definition 4.1 above looks suspiciously similar to the first-order characterization of convexity (1.3) that we discussed earlier. Indeed, the only difference is that here we have replaced $\nabla f(\mathbf{x})$ by $\mathbf{g}$. It turns out that convexity is equivalent to the existence of subgradients everywhere. So we

get a "first order characterization" of convexity that also covers the non-differentiable case.

Lemma 4.3 (Exercise 27)**.** *A function $f : \mathbf{dom}(f) \to \mathbb{R}$ is convex if and only if $\mathbf{dom}(f)$ is convex and $\partial f(\mathbf{x}) \neq \emptyset$ for all $\mathbf{x} \in \mathbf{dom}(f)$.*

It turns out that Lipschitz continuity can be characterized by bounded subgradients. For real-valued convex functions, this is a generalization of Lemma 1.9 to the non-differentiable case.

Lemma 4.4 (Exercise 28)**.** *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be convex, $\mathbf{dom}(f)$ open, $B \in \mathbb{R}_+$. Then the following two statements are equivalent.*

(i) $\|\mathbf{g}\| \leq B$ *for all* $\mathbf{x} \in \mathbf{dom}(f)$ *and all* $\mathbf{g} \in \partial f(\mathbf{x})$.

(ii) $|f(\mathbf{x}) - f(\mathbf{y})| \leq B\|\mathbf{x} - \mathbf{y}\|$ *for all* $\mathbf{x}, \mathbf{y} \in \mathbf{dom}(f)$.

Subgradient optimality condition. Subgradients also allow us to describe cases of optimality for functions which are not necessarily differentiable (and not necessarily convex), in the spirit of Lemma 1.21:

Lemma 4.5. *Suppose that $f : \mathbf{dom}(f) \to \mathbb{R}$ and $\mathbf{x} \in \mathbf{dom}(f)$. If $\mathbf{0} \in \partial f(\mathbf{x})$, then $\mathbf{x}$ is a global minimum.*

Proof. By (4.1), $\mathbf{g} = \mathbf{0} \in \partial f(\mathbf{x})$ gives

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top(\mathbf{y} - \mathbf{x}) = f(\mathbf{x})$$

for all $\mathbf{y} \in \mathbf{dom}(f)$, so $\mathbf{x}$ is a global minimum. □

Here we see (again) that subgradients are "stronger" than gradients for differentiable functions. Indeed, if $\nabla f(\mathbf{x}) = \mathbf{0}$ for a differentiable function f and $\mathbf{x} \in \mathbf{dom}(f)$, we can only say that $\mathbf{x}$ is a *critical point*, but not necessarily a global minimum. Unlike the gradient, a subgradient yields by definition a linear lower bound to the function.

4.2 Differentiability of convex functions

Before we move on to subgradient descent, we want to get a feeling for how "wild" non-differentiable convex functions can be. The answer is: they are surprisingly tame. While there are continuous functions that are *nowhere* differentiable (the classical example is the *Weierstrass function*), convex function cannot be as pathological. In fact, a convex function f is differentiable *almost everywhere*. Formally, this means that wherever you are in $\mathbf{dom}(f)$, you find points arbitrarily close to you at which f is differentiable. In still other words, the set of points where f is not differentiable has measure 0 [Roc97, Theorem 25.5]. Again, all of this requires $\mathbf{dom}(f) \subseteq \mathbb{R}^d$, so let us remind ourselves that we are always in finite dimension throughout this text.

This does not mean that we can ignore non-differentiability in optimization. For example, as Figure 4.1 demonstrates, the global minimum $\mathbf{x}^\star$ can easily be a "kink", a point where f is not differentiable. Also, while running an iterative optimization scheme, we may always stumble upon an intermediate kink.

An important fact is the following characterization of subdifferentials;

Theorem 4.6 ([Roc97, Theorem 25.6])**.** *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be convex, $\mathbf{dom}(f)$ open, $\mathbf{x} \in \mathbf{dom}(f)$. Then $\partial f(\mathbf{x})$ is the convex hull of the set*

$$S(\mathbf{x}) = \{ \lim_{n\to\infty} \nabla f(\mathbf{x}_n) \mid \lim_{n\to\infty} \mathbf{x}_n = \mathbf{x} \}.$$

In words, we consider sequences $(\mathbf{x}_n)_{n\in\mathbb{N}}$ that converge to $\mathbf{x}$ and for which the sequence of gradients $(\nabla f(\mathbf{x}_n))_{n\in\mathbb{N}}$ exists and also converges; the theorem says that the limit is a subgradient at $\mathbf{x}$, and that *any* subgradient can be obtained as a convex combination of such limit subgradients.

In the example of Figure 4.1, there are two types of sequences converging to 0 such the gradients converge as well. These are sequences that have almost all elements negative (gradients converge to -1), and sequences that have almost all elements positive (gradients converge to 1). Consequently, the subgradients at 0 are formed by the set $[-1, 1]$, the convex hull of -1 and 1.

4.3 The algorithm

An iteration of *subgradient descent* is defined as

$$\begin{aligned} &\text{Let } \mathbf{g}_t \in \partial f(\mathbf{x}_t) \\ &\mathbf{x}_{t+1} := \mathbf{x}_t - \gamma_t \mathbf{g}_t. \end{aligned} \tag{4.2}$$

In contrast to our previous descent algorithms, we allow a time-varying stepsize here. This can of course be done for any descent algorithm but so far, we just did not need it. Later in this chapter, we will make use of a time-varying step size.

4.4 Lipschitz convex functions: $\mathcal{O}(1/\varepsilon^2)$ steps

The following result gives the convergence for Subgradient Descent. It is identical to Theorem 2.1, up to relaxing the requirement of differentiability.

Theorem 4.7. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and B-Lipschitz continuous with a global minimum $\mathbf{x}^\star$; furthermore, suppose that $\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$. Choosing the constant stepsize*

$$\gamma_t = \gamma := \frac{R}{B\sqrt{T}},$$

subgradient descent (4.2) yields

$$\frac{1}{T}\sum_{t=0}^{T-1} f(\mathbf{x}_t) - f(\mathbf{x}^\star) \leq \frac{RB}{\sqrt{T}}.$$

Proof. The proof is identical to the one of Theorem 2.1 presented in Section 2.4. The only change is that $\mathbf{g}_t$ is a subgradient now and not a gradient, so that the inequality (2.5) now follows from the subgradient property (4.1) instead of the first-order characterization of convexity. The required bound $\|\mathbf{g}_t\|^2 \leq B^2$ follows from Lemma 4.4 ("convex and Lipschitz = bounded subgradients"). □

Projected subgradient descent. Theorem 3.2 for constrained optimization in $\mathcal{O}(1/\varepsilon^2)$ steps directly extends to the case of subgradient descent as well.

4.5 Tame strong convexity: $\mathcal{O}(1/\varepsilon)$ steps

(Projected) gradient descent converges in $\mathcal{O}(\log(1/\varepsilon))$ steps for functions that are both smooth and strongly convex. But if a function is non-differentiable, then it cannot be smooth under the natural definition of smoothness (Exercise 29). It can still be strongly convex, however, so it is natural to ask whether strong convexity alone allows us to obtain a convergence result. The answer is no in general, but before we discuss this, let us define strong convexity for not necessarily differentiable functions. This is straightforward; for differentiable functions, we recover Definition 2.9. Here, we restrict to the unconstrained case for simplicity.

Definition 4.8. *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be convex, $\mu \in \mathbb{R}_+, \mu > 0$. Function f is called* strongly convex *(with parameter μ) if*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top(\mathbf{y}-\mathbf{x}) + \frac{\mu}{2}\|\mathbf{x}-\mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbf{dom}(f),\ \forall \mathbf{g} \in \partial f(\mathbf{x}). \tag{4.3}$$

Actually, requiring (4.3) only for *some* $\mathbf{g} \in \partial f(\mathbf{x})$ would be another straightforward generalization of Definition 2.9, so which one is the "right" one? The answer is that it does not matter if $\mathbf{dom}(f)$ is open. We could even afford to not require *anything* for points $\mathbf{x}$ where f is not differentiable. This is a consequence of Theorem 4.6 (Exercise 30).

Strong convexity has the following useful characterization.

Lemma 4.9 (Exercise 31)**.** *Let $f : \mathbf{dom}(f) \to \mathbb{R}$ be convex, $\mathbf{dom}(f)$ open, $\mu \in \mathbb{R}_+, \mu > 0$. f is strongly convex with parameter μ if and only if $f_\mu : \mathbf{dom}(f) \to \mathbb{R}$ defined by*

$$f_\mu(\mathbf{x}) = f(\mathbf{x}) - \frac{\mu}{2}\|\mathbf{x}\|^2, \quad \mathbf{x} \in \mathbf{dom}(f)$$

is convex.

Let's look at the problem with (sub)gradient descent on strongly convex functions.

Lemma 4.10 (Exercise 32)**.** *The function $f(x) = e^{|x|}$ is strongly convex with parameter $\mu = 1$.*

This function is of course far from being smooth; it grows exponentially, so there can't be any quadratic upper bounds. In fact, as strong

convexity ony requires quadratic *lower* bounds, strongly convex functions can be extremely fast-growing. In such a situation, (sub)gradient descent will overshoot already for tiny step sizes and diverge.

In case of $f(x) = e^{|x|}$, the function is differentiable at $x \neq 0$ with $f'(x) = \operatorname{sgn}(x)e^{|x|}$, so the (sub)gradient step is

$$x_{t+1} = x_t - \gamma_t \operatorname{sgn}(x_t) e^{|x_t|}.$$

For $|x|$ only mildly larger than 0, the step will overshoot the optimum $x^* = 0$ and take us (much) further away. To compensate for this, we would need extremely small stepsizes. These in turn would lead to extremely poor convergence for functions such as $f(x) = x^2/2$ (which is also strongly convex with $\mu = 1$) . Hence, there are no stepsizes that fit all strongly convex functions with a fixed strong convexity parameter μ.

To succeed with (sub)gradient descent in this situation, we therefore need to make some additional assumptions. Smoothness (quadratic upper bounds) is such an assumption, but in the non-differentiable case, this is precisely not an option. What people have done instead is to assume that the subgradients g_t that we encounter during the algorithm are bounded in norm.

To ensure bounded subgradients, we could simply assume that f is Lipschitz, but then we will only make a statement about an empty function class. The reason is that a function cannot be globally strongly convex and Lipschitz at the same time (Exercise 33). It can be strongly convex *and* have bounded gradients over a closed and bounded set X, so analyzing projected subgradient descent is an alternative.

But even when we optimize over $\mathbb{R}^d$, we may be lucky and only hit iterates with small subgradients. This will typically happen if we start sufficiently close to optimality. In this case, there are step sizes γ_t (not depending on the observed gradients) that give us useful error bounds.

Below, we prove such a bound for subgradient descent, and this result then clearly extends to gradient descent on differentiable and strongly convex (but not necessarily smooth) functions. The bound on the number of steps will be $\mathcal{O}(1/\varepsilon)$ which is of course much worse than $\mathcal{O}(\log(1/\varepsilon))$, but still better than $\mathcal{O}(1/\varepsilon^2)$ that we get in the Lipschitz case. So assuming strong convexity results in a convergence behavior as in the smooth case—*if* the gradients stay bounded, and this is what we mean by "tame".

In order to analyze subgradient descent on strongly convex functions,

we will for the first time depart from algorithm variants with a constant stepsize γ, but instead use a time-varying stepsize γ_t decreasing over time.

Theorem 4.11. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be strongly convex with parameter $\mu > 0$ and let $\mathbf{x}^\star$ be the unique global minimum of f. With decreasing step size*

$$\gamma_t := \frac{2}{\mu(t+1)}, \quad t > 0,$$

subgradient descent (4.2) yields

$$f\left(\frac{2}{T(T+1)}\sum_{t=1}^{T} t \cdot \mathbf{x}_t\right) - f(\mathbf{x}^\star) \leq \frac{2B^2}{\mu(T+1)},$$

where $B = \max_{t=1}^{T} \|\mathbf{g}_t\|$.

Unlike in previous convergence results, small error is not achieved by some iterate that we have gone through, but by a convex combination of iterates.

Proof. We start from the vanilla analysis (2.3) (with $\gamma = \gamma_t$):

$$\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star) = \frac{\gamma_t}{2}\|\mathbf{g}_t\|^2 + \frac{1}{2\gamma_t}\left(\|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2\right).$$

Now we plug in the lower bound $\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star) \geq f(\mathbf{x}_t) - f(\mathbf{x}^\star) + \frac{\mu}{2}\|\mathbf{x}_t - \mathbf{x}^\star\|^2$ resulting from strong convexity to obtain (with $\|\mathbf{g}_t\|^2 \leq B^2$) that

$$f(\mathbf{x}_t) - f(\mathbf{x}^\star) \leq \frac{B^2\gamma_t}{2} + \frac{(\gamma_t^{-1} - \mu)}{2}\|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \frac{\gamma_t^{-1}}{2}\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2. \quad (4.4)$$

Unlike in the vanilla analysis (where we had $\gamma_t = \gamma, \mu = 0$), the right-hand side does not telescope anymore when we sum over all $t \leq T$; to fix this, we precisely need the time-varying stepsize.

Let's make a small computation: to get telescoping behavior, we would need that $\gamma_t^{-1} = \gamma_{t+1}^{-1} - \mu$. For example, $\gamma_t^{-1} = \mu(1+t)$ satisfies this, but our choice $\gamma_t^{-1} = \mu(1+t)/2$ does not. Exercise 34 asks you to compute what happens when we actually choose $\gamma_t^{-1} = \mu(1+t)$; this will let you

appreciate the seemingly "wrong" choice of $\gamma_t = \frac{2}{\mu(t+1)}$ here. Plugging in this stepsize and multiplying with t on both the sides, we get

$$\begin{aligned} t \cdot \big(f(\mathbf{x}_t) - f(\mathbf{x}^\star)\big) &\leq \frac{B^2 t}{\mu(t+1)} + \frac{\mu}{4}\Big(t(t-1)\left\|\mathbf{x}_t - \mathbf{x}^\star\right\|^2 - (t+1)t\left\|\mathbf{x}_{t+1} - \mathbf{x}^\star\right\|^2\Big) \\ &\leq \frac{B^2}{\mu} + \frac{\mu}{4}\Big(t(t-1)\left\|\mathbf{x}_t - \mathbf{x}^\star\right\|^2 - (t+1)t\left\|\mathbf{x}_{t+1} - \mathbf{x}^\star\right\|^2\Big). \end{aligned}$$

Summing from $t = 1, \ldots, T$, we obtain a telescoping sum:

$$\sum_{t=1}^{T} t \cdot \big(f(\mathbf{x}_t) - f(\mathbf{x}^\star)\big) \leq \frac{TB^2}{\mu} + \frac{\mu}{4}\Big(0 - T(T+1)\left\|\mathbf{x}_{T+1} - \mathbf{x}^\star\right\|^2\Big) \leq \frac{TB^2}{\mu}.$$

Since

$$\frac{2}{T(T+1)}\sum_{t=1}^{T} t = 1,$$

Jensen's inequality (Lemma 1.12) yields

$$f\left(\frac{2}{T(T+1)}\sum_{t=1}^{T} t \cdot \mathbf{x}_t\right) - f(\mathbf{x}^\star) \leq \frac{2}{T(T+1)}\sum_{t=1}^{T} t \cdot \big(f(\mathbf{x}_t) - f(\mathbf{x}^\star)\big).$$

This in turn implies

$$f\left(\frac{2}{T(T+1)}\sum_{t=1}^{T} t \cdot \mathbf{x}_t\right) - f(\mathbf{x}^\star) \leq \frac{2B^2}{\mu(T+1)}\,.$$

□

Unlike all previous bounds, this bound seems to be independent from the initial distance $\|\mathbf{x}_o - \mathbf{x}^\star\|$ to the optimum. However, there is no free lunch here. The initial distance will typically affect the bound B (think of a quadratic function where B is proportional to $\|\mathbf{x}_o - \mathbf{x}^\star\|$).

4.6 Optimality of first-order methods

With all the convergence rates we have seen so far, a very natural question to ask is if these rates are best possible or not. Surprisingly, the rate can indeed not be improved in general.

Theorem 4.12 (Nesterov). *For any $T \leq d-1$ and starting point $\mathbf{x}_0$, there is a function f in the problem class of B-Lipschitz functions over $\mathbb{R}^d$, such that any (sub)gradient method has an objective error at least*

$$f(\mathbf{x}_T) - f(\mathbf{x}^\star) \geq \frac{RB}{2(1+\sqrt{T+1})} \,.$$

The above theorem applies to all first-order methods which form iterates by linearly combining past iterates and (sub)gradients, and requires the dimension d to be sufficiently large.

4.7 Exercises

Exercise 26. *Prove Lemma 4.2, meaning that a function that is differentiable at $\mathbf{x}$ has at most one subgradient there, namely $\nabla f(\mathbf{x})$.*

Exercise 27. *Prove the easy direction of Lemma 4.3, meaning that the existence of subgradients everywhere implies convexity!*

Exercise 28. *Prove Lemma 4.4 (Lipschitz continuity and bounded subgradients).*

Exercise 29. *Generalizing Definition 2.2, let us call a (not necessarily differentiable) function $f : \mathbb{R}^d \to \mathbb{R}$ smooth with parameter $L \in \mathbb{R}_+$ if for all $\mathbf{x} \in \mathbb{R}^d$, there exists a subgradient $\mathbf{g}_\mathbf{x} \in \mathbb{R}^d$ such that*

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \mathbf{g}_\mathbf{x}^\top(\mathbf{y}-\mathbf{x}) + \frac{L}{2}\|\mathbf{x}-\mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d.$$

This means that for every point $\mathbf{x}$, the graph of f is below the graph of the quadratic function $f(\mathbf{x}) + \mathbf{g}_\mathbf{x}^\top(\mathbf{y}-\mathbf{x}) + \frac{L}{2}\|\mathbf{x}-\mathbf{y}\|^2$.

Prove that if f is smooth according to this definition, then f is differentiable, with $\mathbf{g}_\mathbf{x} = \nabla f(\mathbf{x})$ for all $\mathbf{x}$. In particular, for differentiable functions, the notion of smoothness introduced above coincides with the one of Definition 2.2; moreover, non-differentiable functions cannot be smooth.

Does the above hold if $\mathbf{g}_\mathbf{x}$ is not a subgradient?

Exercise 30. *Suppose that $f : \mathbb{R}^d \to \mathbb{R}$ is convex and satisfies*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) + \frac{\mu}{2}\|\mathbf{x}-\mathbf{y}\|^2$$

for all $\mathbf{x}$ *such that* $\nabla f(\mathbf{x})$ *exists, and for all* $\mathbf{y}$*. Prove that this implies*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}_{\mathbf{x}}^{\top}(\mathbf{y} - \mathbf{x}) + \frac{\mu}{2}\|\mathbf{x} - \mathbf{y}\|^2$$

for all $\mathbf{x}$*, all* $\mathbf{g}_{\mathbf{x}} \in \partial f(\mathbf{x})$ *and all* $\mathbf{y}$*.*

Exercise 31. *Prove Lemma 4.9: f is strongly convex with parameter μ over an open domain if and only if $f_\mu : \mathbf{x} \mapsto f(\mathbf{x}) - \frac{\mu}{2}\|\mathbf{x}\|^2$ is convex over the same domain.*

Exercise 32. *Prove Lemma 4.10: $f(x) = e^{|x|}$ is strongly convex with parameter $\mu = 1$.*

Exercise 33. *Prove that a function $f : \mathbb{R}^d \to \mathbb{R}$ cannot simultaneously be Lipschitz and strongly convex!*

Exercise 34. *Which result can you prove when you use the "telescoping stepsize"*

$$\gamma_t = \frac{1}{\mu(t+1)}$$

in Theorem 4.11 instead of $\gamma_t = \frac{2}{\mu(t+1)}$?

Chapter 5

Stochastic Gradient Descent

Contents

5.1 The algorithm

Many objective functions occurring in machine learning are formulated as *sum structured objective functions*

$$f(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} f_i(\mathbf{x}). \tag{5.1}$$

Here f_i is typically the cost function of the i-th datapoint, taken from a training set of n elements in total.

We have already seen an example for this: the loss function (1.13) in the handwritten digit recognition (Section 1.6.1) has one term for each of the n training images $\mathbf{x} \in P$:

$$\ell(W) = -\sum_{\mathbf{x} \in P} \ln z_{d(\mathbf{x})}(W\mathbf{x}).$$

The normalizing factor $1/n$ that we assume in the general setting (5.1) will just simplify the following a bit.

An iteration of *stochastic gradient descent* (SGD) in its basic form is defined as

$$\begin{aligned} &\text{sample } i \in [n] \text{ uniformly at random} \\ &\mathbf{x}_{t+1} := \mathbf{x}_t - \gamma_t \nabla f_i(\mathbf{x}_t). \end{aligned} \tag{5.2}$$

This update looks almost identical to the classical gradient method, the only difference being that we have computed the gradient not of the entire f but only of one particular (randomly chosen) function f_i. As we will need varying stepsizes a bit later, we allow for the stepsize to depend on t now.

In the above setting, the update vector $\mathbf{g}_t := \nabla f_i(\mathbf{x}_t)$ is called a *stochastic gradient*. Formally, $\mathbf{g}_t$ is a vector of d random variables, but we will also simply call this a random variable.

The crucial advantage of SGD versus its classical gradient descent counterpart is the efficiency per iteration: While computing the full gradient for a sum structured problem (5.1) would require us to compute n individual gradients of the f_i functions, an iteration of SGD requires only a single one of those, and therefore is n times cheaper. SGD has therefore become the main workhorse for training machine learning models. Whether such cheaper iterations also give similar progress is another question, which we analyze next.

5.2 Unbiasedness

We would like to start with the vanilla analysis again, but now we cannot bound the random variable $\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)$ from below using (2.5), as the inequality

$$f(\mathbf{x}_t) - f(\mathbf{x}^\star) \leq \mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)$$

may hold or not hold, depending on how $\mathbf{g}_t$ turns out. But it still holds *in expectation*, as we show now.

The vector $\mathbf{g}_t$ may be far from the true gradient, and of high variance, but in expectation over the random choice of i, it does coincide with the full gradient of f. We formalize this as

$$\mathbb{E}\big[\mathbf{g}_t \big| \mathbf{x}_t = \mathbf{x}\big] = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\mathbf{x}) = \nabla f(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^d. \tag{5.3}$$

Here, $\mathbb{E}\big[\mathbf{g}_t \big| \mathbf{x}_t = \mathbf{x}\big]$ is the *conditional expectation* of $\mathbf{g}_t$, given the event $\{\mathbf{x}_t = \mathbf{x}\}$. If this event is nonempty, linearity of conditional expectations yields that

$$\mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x} - \mathbf{x}^\star) \big| \mathbf{x}_t = \mathbf{x}\big] = \mathbb{E}\big[\mathbf{g}_t \big| \mathbf{x}_t = \mathbf{x}\big]^\top(\mathbf{x} - \mathbf{x}^\star) = \nabla f(\mathbf{x})^\top(\mathbf{x} - \mathbf{x}^\star).$$

Using the fact that $\{\mathbf{x}_t = \mathbf{x}\}$ can occur only for $\mathbf{x}$ in some finite set X (one element for every choice of indices throughout all iterations), the partition theorem further gives us

$$\begin{aligned}
\mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] &= \sum_{\mathbf{x}\in X} \mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x} - \mathbf{x}^\star) \big| \mathbf{x}_t = \mathbf{x}\big]\ \mathrm{prob}(\mathbf{x}_t = \mathbf{x}) \\
&= \sum_{\mathbf{x}\in X} \nabla f(\mathbf{x})^\top(\mathbf{x} - \mathbf{x}^\star)\ \mathrm{prob}(\mathbf{x}_t = \mathbf{x}) \\
&= \mathbb{E}\big[\nabla f(\mathbf{x}_t)^\top(\mathbf{x}_t - \mathbf{x}^\star)\big].
\end{aligned}$$

Hence, we have

$$\mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] = \mathbb{E}\big[\nabla f(\mathbf{x}_t)^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] \geq \mathbb{E}\big[f(\mathbf{x}_t) - f(\mathbf{x}^\star)\big]. \tag{5.4}$$

The last inequality is by convexity, and this is means that the lower bound (2.5) holds in expectation.

Exercise 35 lets you recall some basics around conditional expectations. Under (5.3) we say that the stochastic gradient $\mathbf{g}_t$ is an *unbiased* estimator of the gradient, for any time-step t.

5.3 Bounded stochastic gradients: $\mathcal{O}(1/\varepsilon^2)$ steps

To get a first result out of the vanilla analysis, we assumed in Section 2.4 that $\|\nabla f(\mathbf{x})\|^2 \leq B^2$ for all $\mathbf{x} \in \mathbb{R}^d$, where B was a constant. Here, we are assuming the same for the *expected* squared norms of our stochastic gradients. And we are getting the same result, except that it now holds for the *expected* function values.

Theorem 5.1. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex and differentiable function, and let $\mathbf{x}^\star$ be a global minimum of f; furthermore, suppose that $\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$, and that $\mathbb{E}\big[\|\mathbf{g}_t\|^2\big] \leq B^2$ for all t. Choosing the constant stepsize*

$$\gamma := \frac{R}{B\sqrt{T}}$$

stochastic gradient descent (5.2) yields

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\big[f(\mathbf{x}_t)\big] - f(\mathbf{x}^\star) \leq \frac{RB}{\sqrt{T}}.$$

Proof. Taking expectations on both sides of the vanilla analyis (2.4) and using linearity of expectations, we get

$$\sum_{t=0}^{T-1} \mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] \leq \frac{\gamma}{2}\sum_{t=0}^{T-1} \mathbb{E}\big[\|\mathbf{g}_t\|^2\big] + \frac{1}{2\gamma}\|\mathbf{x}_0 - \mathbf{x}^\star\|^2. \tag{5.5}$$

By (5.4),

$$\mathbb{E}\big[f(\mathbf{x}_t) - f(\mathbf{x}^\star)\big] \leq \mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)\big].$$

Plugging this into (5.5), using $\mathbb{E}\big[\|\mathbf{g}_t\|^2\big] \leq B^2$ and $\|\mathbf{x}_0 - \mathbf{x}^\star\| \leq R$, we get

$$\sum_{t=0}^{T-1} \mathbb{E}\big[f(\mathbf{x}_t) - f(\mathbf{x}^\star)\big] \leq \frac{\gamma}{2}B^2T + \frac{1}{2\gamma}R^2,$$

from which the statement follows from the choice of γ as in Theorem 2.1. □

Constrained optimization. For constrained optimization, Theorem 5.1 for the convergence in $\mathcal{O}(1/\varepsilon^2)$ steps directly extends to constrained problems as well. After every step of SGD, projection back to X is applied as usual. The resulting algorithm is called *projected SGD*.

5.4 Tame strong convexity: $\mathcal{O}(1/\varepsilon)$ steps

It is possible to strengthen our above SGD analysis. One way to do so is under the additional assumption of strong convexity of the objective function f (as in Definition 2.9). Again, the proof works by "taking expectations" over a previous analysis, in this case the one for subgradient descent in the tame strongly convex case (Theorem 4.11).

Theorem 5.2. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable and strongly convex with parameter $\mu > 0$; let $\mathbf{x}^\star$ be the unique global minimum of f. With decreasing step size*

$$\gamma_t := \frac{2}{\mu(t+1)}$$

stochastic gradient descent (5.2) yields

$$\mathbb{E}\Big[f\Big(\frac{2}{T(T+1)}\sum_{t=1}^{T} t \cdot \mathbf{x}_t\Big) - f(\mathbf{x}^\star)\Big] \leq \frac{2B^2}{\mu(T+1)},$$

where $B = \max_{t=1}^{T} \mathbb{E}\big[\,\|\mathbf{g}_t\|\,\big]$.

Proof. We start from the vanilla analysis (2.3) (with $\gamma = \gamma_t$) and take expectations on both sides:

$$\mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] = \frac{\gamma_t}{2}\mathbb{E}\big[\|\mathbf{g}_t\|^2\big] + \frac{1}{2\gamma_t}\left(\mathbb{E}\big[\|\mathbf{x}_t - \mathbf{x}^\star\|^2\big] - \mathbb{E}\big[\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2\big]\right).$$

Now we use (5.4) along with strong convexity to get a lower bound

$$\begin{aligned}\mathbb{E}\big[\mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] &= \mathbb{E}\big[\nabla f(\mathbf{x}_t)^\top(\mathbf{x}_t - \mathbf{x}^\star)\big] \\ &\geq \mathbb{E}\big[f(\mathbf{x}_t) - f(\mathbf{x}^\star)\big] + \frac{\mu}{2}\mathbb{E}\big[\|\mathbf{x}_t - \mathbf{x}^\star\|^2\big]\end{aligned}$$

for the left-hand side. Combining the previous two equations and using $\mathbb{E}\big[\|\mathbf{g}_t\|^2\big] \leq B^2$, we get the "expected version" of (4.4):

$$\mathbb{E}[f(\mathbf{x}_t) - f(\mathbf{x}^\star)] \leq \frac{B^2\gamma_t}{2} + \frac{(\gamma_t^{-1} - \mu)}{2}\mathbb{E}\big[\,\|\mathbf{x}_t - \mathbf{x}^\star\|^2\,\big] - \frac{\gamma_t^{-1}}{2}\mathbb{E}\big[\,\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2\,\big].$$

The proof continues as in Theorem 4.11, with every step being the "expected version" of the corresponding step in the earlier proof. □

5.5 Stochastic Subgradient Descent

For problems which are not necessarily differentiable, we modify SGD to use a subgradient of f_i in each iteration. The update of stochastic subgradient descent is given by

$$\begin{aligned} &\text{sample } i \in [n] \text{ uniformly at random} \\ &\text{let } \mathbf{g}_t \in \partial f_i(\mathbf{x}_t) \\ &\mathbf{x}_{t+1} := \mathbf{x}_t - \gamma_t \mathbf{g}_t. \end{aligned} \tag{5.6}$$

Let $\mathbf{g}^i : \mathbb{R}^d \to \mathbb{R}^d$ denote the function that selects the subgradient of f_i at the current point. Then we have $\mathbf{g}_t = \mathbf{g}^i(\mathbf{x}_t)$ for random i. Unbiasedness now becomes

$$\mathbb{E}\big[\mathbf{g}_t \big| \mathbf{x}_t = \mathbf{x}\big] = \frac{1}{n}\sum_{i=1}^{n} \mathbf{g}^i(\mathbf{x}) =: \mathbf{g}(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^d.$$

It is immediate from the subgradient property that $\mathbf{g}(\mathbf{x}) \in \partial f(\mathbf{x})$ if $\mathbf{g}^i(\mathbf{x}) \in \partial f_i(\mathbf{x})$ for all i. As in Section 5.2 for SGD, we then get

$$\mathbb{E}\big[\mathbf{g}_t^\top (\mathbf{x}_t - \mathbf{x}^\star)\big] = \mathbb{E}\big[\mathbf{g}(\mathbf{x}_t)^\top (\mathbf{x}_t - \mathbf{x}^\star)\big].$$

This in turn can be lower bounded by

$$\mathbb{E}\big[f(\mathbf{x}_t) - f(\mathbf{x}^\star)\big] + \frac{\mu}{2}\mathbb{E}\big[\|\mathbf{x}_t - \mathbf{x}^\star\|^2\big],$$

with $\mu = 0$ in the convex case and $\mu > 0$ in the strongly convex case, now using $\mathbf{g}(\mathbf{x}_t)$'s subgradient property (4.1) in the convex and (4.3) in the strongly convex case instead of the first-order condition for $\nabla f(\mathbf{x}_t)$. As this lower bound is the crucial ingredient in the previous two analyses of convergence in $\mathcal{O}(1/\varepsilon^2)$ and $\mathcal{O}(1/\varepsilon)$ steps, the results directly extend to the case of subgradient descent as well.

5.6 Mini-batch variants

Instead of using a single element f_i of our sum objective (5.1) to form a stochastic gradient $\mathbf{g}_t = \nabla f_i(\mathbf{x}_t)$, another variant is to use an average of several of them:

$$\tilde{\mathbf{g}}_t := \frac{1}{m}\sum_{j=1}^{m} \mathbf{g}_t^j. \tag{5.7}$$

where $\mathbf{g}_t^j = \nabla f_{i_j}(\mathbf{x}_t)$ for an index i_j. The set of the (distinct) i_j indices is called a mini-batch, and m is the mini batch size.

Using the step direction $\tilde{\mathbf{g}}_t$ defines mini-batch SGD. For $m = 1$, we recover SGD as originally defined, while for $m = n$ we recover full gradient descent.

Mini-batch SGD can be advantageous in several applications. For example, parallelization over up to m processors will easily give a speed-up for the gradient computation, which is typically the main cost of running SGD. Here, parallelization exploits the fact that all $\mathbf{g}_t^j$ are defined at the same iterate $\mathbf{x}_t$ and can therefore be computed independently.

Taking an average of many independent random variables reduces the variance. In the context of mini-batch SGD, we obtain that for larger size of the mini-batch m our estimate $\tilde{\mathbf{g}}_t$ will be closer to the true gradient, in expectation:

$$\begin{aligned}\mathbb{E}\Big[\Big\|\tilde{\mathbf{g}}_t - \nabla f(\mathbf{x}_t)\Big\|^2\Big] =& \mathbb{E}\Big[\Big\|\frac{1}{m}\sum_{j=1}^{m}\mathbf{g}_t^j - \nabla f(\mathbf{x}_t)\Big\|^2\Big] \\ =& \frac{1}{m}\mathbb{E}\big[\|\mathbf{g}_t^1 - \nabla f(\mathbf{x}_t)\|^2\big] \\ =& \frac{1}{m}\mathbb{E}\big[\|\mathbf{g}_t^1\|^2\big] - \frac{1}{m}\|\nabla f(\mathbf{x}_t)\|^2 \leq \frac{B^2}{m}\,.\end{aligned}$$

Using a modification of the above analysis, it is possible to use this property to relate the above convergence rate of SGD to the rate of full gradient descent.

5.7 Exercises

Exercise 35. *Let Y be a random variable over a finite probability space (Ω, prob) where $\mathrm{prob} : 2^\Omega \to [0,1]$; this avoids subtleties in defining conditional probabilities and expectations; and it covers the random variables occurring in SGD, since in each step, we are randomly choosing among a finite set of n indices. Furthermore, let $B \subseteq \Omega$ be an event.*

For nonemepty B, the conditional expectation of Y given B *is the number*

$$\mathbb{E}\big[Y\big|B\big] := \sum_{y \in Y(\Omega)} y \cdot \mathrm{prob}\big(Y = y\big|B\big).$$

where $Y = y$ is shorthand for the event $\{\omega \in \Omega : Y(\omega) = y\}$.

Finally, for two events A and $B \neq \emptyset$, the conditional probability $\mathrm{prob}\big[A\big|B\big]$ is defined as

$$\mathrm{prob}\big(A\big|B\big) := \frac{\mathrm{prob}\big(A \cap B\big)}{\mathrm{prob}\big(B\big)}.$$

If $B = \emptyset$, $\mathbb{E}\big[Y\big|B\big]$ can be defined arbitrarily.

Prove the following statements.

(i) Alternative definition of conditional expectation:

$$\mathrm{prob}\big(B\big) \cdot \mathbb{E}\big[Y\big|B\big] = \sum_{\omega \in B} Y(\omega)\mathrm{prob}(\omega).$$

(ii) Partition Theorem: Let $B_1, \ldots, B_m$ be a partition of Ω. Then

$$\mathbb{E}\big[Y\big] = \sum_{i=1}^{m} \mathbb{E}\big[Y\big|B_i\big]\mathrm{prob}\big(B_i\big).$$

(iii) Linearity of conditional expectation: For random variables $Y_1, \ldots, Y_m$ over (Ω, prob) and real numbers $\lambda_1, \ldots, \lambda_m$, and if $B \neq \emptyset$,

$$\sum_{i=1}^{m} \lambda_i \mathbb{E}\big[Y_i\big|B\big] = \mathbb{E}\big[\sum_{i=1}^{m} \lambda_i Y_i\big|B\big].$$

Bibliography

[ACGH18] Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A convergence analysis of gradient descent for deep linear neural networks. *CoRR*, abs/1810.02281, 2018.

[AE08] Herbert Amann and Joachim Escher. *Analysis II*. Birkhäuser, 2008.

[BG17] Nikhil Bansal and Anupam Gupta. Potential-function proofs for first-order methods. *CoRR*, abs/1712.04581, 2017.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. https://web.stanford.edu/~boyd/cvxbook/.

[Dav59] William C. Davidon. Variable metric method for minimization. Technical Report ANL-5990, AEC Research and Development, 1959.

[Dav91] William C. Davidon. Variable metric method for minimization. *SIAM J. Optimization*, 1(1):1–17, 1991.

[Die69] J. Dieudonneé. *Foundations of Modern Analysis*. Academic Press, 1969.

[DSSSC08] John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Tushar Chandra. Efficient projections onto the ℓ_1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, pages 272–279, 07 2008.

[FM91] M. Furi and M. Martelli. On the mean value theorem, inequality, and inclusion. *The American Mathematical Monthly*, 98(9):840–846, 1991.

[Gol70] D. Goldfarb. A family of variable-metric methods derived by variational means. *Mathematics of Computation*, 24(109):23–26, 1970.

[Gre70] J. Greenstadt. Variations on variable-metric methods. *Mathematics of Computation*, 24(109):1–22, 1970.

[KNS16] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Łojasiewicz Condition. In *ECML PKDD 2016: Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer International Publishing, Cham, September 2016.

[KSJ18] Sai Praneeth Karimireddy, Sebastian U Stich, and Martin Jaggi. Global linear convergence of Newton's method without strong-convexity or Lipschitz gradients. *arXiv*, 2018.

[LW19] Ching-Pei Lee and Stephen Wright. First-order algorithms converge faster than $o(1/k)$ on convex problems. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3754–3762, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

[Nes83] Yurii Nesterov. A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Soviet Math. Dokl.*, 27(2), 1983.

[Nes12] Yurii Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.

[Nes18] Yurii Nesterov. *Lectures on Convex Optimization*, volume 137 of *Springer Optimization and Its Applications*. Springer, second edition, 2018.

[Noc80] J. Nocedal. Updating quasi-newton matrices with limited storage. *Mathematics of Computation*, 35(151):773–782, 1980.

[NP06] Yurii Nesterov and B.T. Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, Aug 2006.

[NSL+15] Julie Nutini, Mark W Schmidt, Issam H Laradji, Michael P Friedlander, and Hoyt A Koepke. Coordinate Descent Converges Faster with the Gauss-Southwell Rule Than Random Selection. In *ICML*, pages 1632–1641, 2015.

[NY83] Arkady. S. Nemirovsky and D. B. Yudin. *Problem complexity and method efficiency in optimization*. Wiley, 1983.

[Roc97] R. Tyrrell Rockafellar. *Convex Analysis*. Princeton Landmarks in Mathematics. Princeton University Press, 1997.

[Tib96] Robert Tibshirani. Regression shrinkage and selection via the LASSO. *J. R. Statist. Soc. B*, 58(1):267–288, 1996.

[Vis15] Nisheeth Vishnoi. A mini-course on convex optimization (with a view toward designing fast algorithms), 2015. https://theory.epfl.ch/vishnoi/Nisheeth-VishnoiFall2014-ConvexOptimization.pdf.

[Zim16] Judith Zimmermann. *Information Processing for Effective and Stable Admission*. PhD thesis, ETH Zurich, 2016. .