Continuous Optimization
Lecture Notes

Aurelien Lucchi
Department of Mathematics and Computer Science
University of Basel

May 4, 2025

Contents

Preface

These lecture notes were prepared for a lecture delivered at the Department of Mathematics and Computer Science, University of Basel, Switzerland. They are designed for Master's level students with prior knowledge in machine learning and a solid understanding of mathematics, particularly in algebra and calculus.

Acknowledgment Parts of these lecture notes are based on resources from other researchers and teachers whom I would like to acknowledge:

- Parts of several chapters, including Chapter 2, are based on the Convex Optimization lecture of Prof. Ryan Tibshirani at Berkeley (previously at CMU).
- Chapter 9 on non-convexity is based on the Optimization lecture notes from Prof. Chi Jin (Princeton).

Additionally, many figures are sourced from Wikipedia and occasionally other references. If any uncredited material is discovered, please notify me so that I can properly credit the sources.

Lastly, I want to extend my gratitude to the teaching assistants (Rustem Islamov, Navish Kumar, Enea Monzio Compagnoni) for their ongoing contributions to enhancing the course material over the years.

Chapter 1

Prerequisites

1.1 Notations & Math symbols

1.1.1 Vectors

A column vector is a $d \times 1$ array that is, an array consisting of a single column of d elements, denoted as

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}.$$

Similarly, a row vector is a $1 \times d$ array that is, an array consisting of a single row of d elements, denoted as

$$\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \dots & x_d \end{bmatrix}.$$

Throughout, boldface is used for the row and column vectors. The transpose (indicated by $\top$) of a row vector is a column vector. We also recall that the inner product between two vectors $\mathbf{u} \in \mathbb{R}^d$ and $\mathbf{v} \in \mathbb{R}^d$ is defined as

$$\mathbf{u}^\top \mathbf{v} = \sum_{i=1}^{d} u_i v_i.$$

The **span** of a set of vectors is the set of all possible linear combinations of those vectors. If we have a set of vectors $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$ in a vector space V, the span of these vectors, denoted by $\mathrm{span}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k)$, is defined as:

$$\mathrm{span}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k) = \{c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_k\mathbf{v}_k \mid c_1, c_2, \dots, c_k \in \mathbb{R}\}.$$

Consider the following example illustrated in Figure 1.1 where we show the span of two vectors $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{R}^2$. Taking for instance coefficients $c_1 = 0.5$ and $c_2 = 1$ yields the vector $\begin{pmatrix} 0.5 \\ 4 \end{pmatrix}$ since:

$$\underbrace{\begin{pmatrix} -3 & 2 \\ 2 & 3 \end{pmatrix}}_{\mathbf{V}} \begin{pmatrix} 0.5 \\ 1 \end{pmatrix} = 0.5 \underbrace{\begin{pmatrix} -3 \\ 2 \end{pmatrix}}_{\mathbf{v}_1} + 1 \underbrace{\begin{pmatrix} 2 \\ 3 \end{pmatrix}}_{\mathbf{v}_1} = \begin{pmatrix} 0.5 \\ 4 \end{pmatrix}$$

As another example, consider the span of two dependent vectors $\mathbf{v}_2 = 2\mathbf{v}_1$, then one can check that we can only obtain a multiple of the vector $\mathbf{v}_1$, for instance:

$$\underbrace{\begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix}}_{\mathbf{V}} \begin{pmatrix} 1 \\ 0.5 \end{pmatrix} = 1 \underbrace{\begin{pmatrix} 1 \\ 2 \end{pmatrix}}_{\mathbf{v}_1} + 0.5 \underbrace{\begin{pmatrix} 2 \\ 4 \end{pmatrix}}_{\mathbf{v}_1} = \begin{pmatrix} 2 \\ 4 \end{pmatrix}$$

If we consider the vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ as columns of a matrix $\mathbf{V}$, then the linear combinations of these vectors constitute the column space, as described in the definition below. A similar definition applies to the row space.

Definition 1. *The column space (also called the range or image) of a matrix* $\mathbf{A}$ *is the span (set of all possible linear combinations) of its column vectors.*

1.1.2 Matrices

Matrices will be denoted by a boldface capital letter, for instance

$$\mathbf{A} = \begin{pmatrix} A_{11} & \dots & A_{1d} \\ & \dots & \\ A_{p1} & \dots & A_{pd} \end{pmatrix}, \quad \mathbf{A}^\top = \begin{pmatrix} A_{11} & \dots & A_{p1} \\ & \dots & \\ A_{1d} & \dots & A_{pd} \end{pmatrix}$$

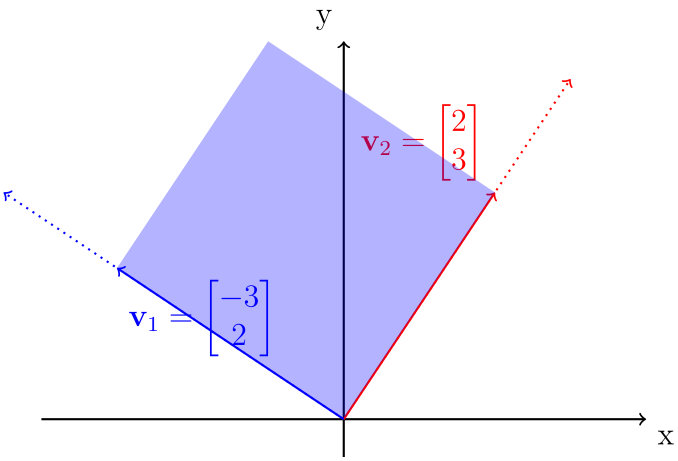

y
x
$\mathbf{v}_2 = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$
$\mathbf{v}_1 = \begin{bmatrix} -3 \\ 2 \end{bmatrix}$

Figure 1.1: Illustration of the span of two vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ in $\mathbb{R}^2$. The blue parallelogram shows the span of these two vectors for coefficients $\mathbf{c} = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$ where $c_i \in [0, 1]$.

Recall that the multiplication of two matrices $\mathbf{A} \in \mathbb{R}^{m\times n}, \mathbf{B} \in \mathbb{R}^{n\times d}$ is

$$\mathbf{C} = \mathbf{A}\mathbf{B}, \quad C_{ij} = \sum_{k=1}^{n} A_{ik}B_{kj}.$$

The matrix multiplication operation has the following properties:

- **Distributive property:** Matrix multiplication is distributive over matrix addition. For matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ of compatible dimensions,

$$\mathbf{A}(\mathbf{B}+\mathbf{C}) = \mathbf{A}\mathbf{B} + \mathbf{A}\mathbf{C} \quad \text{and} \quad (\mathbf{A}+\mathbf{B})\mathbf{C} = \mathbf{A}\mathbf{C} + \mathbf{B}\mathbf{C}.$$

- **Associative property:** Matrix multiplication is associative. For matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ of compatible dimensions,

$$(\mathbf{A}\mathbf{B})\mathbf{C} = \mathbf{A}(\mathbf{B}\mathbf{C}).$$

- **Not commutative:** Matrix multiplication is generally not commutative. For matrices $\mathbf{A}$ and $\mathbf{B}$ of compatible dimensions,

$$\mathbf{A}\mathbf{B} \neq \mathbf{B}\mathbf{A} \quad \text{in general.}$$

The **inverse** of a matrix $\mathbf{A}$ is a matrix, denoted by $\mathbf{A}^{-1}$, such that when it is multiplied by $\mathbf{A}$, it yields the identity matrix. Specifically, for an $n \times n$ matrix $\mathbf{A}$,

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}_n,$$

where $\mathbf{I}_n$ is the $n \times n$ identity matrix.

1.1.3 Notation

- Bold small letter $\mathbf{x}$ is a column vector
- Bold capital letter $\mathbf{A}$ is a matrix
- $\mathbf{x}^\top, \mathbf{A}^\top$: transpose of a vector (i.e. a row vector) and a matrix respectively
- $\mathbb{R}$: reals
- $a := b$: a is defined by b
- $\frac{\partial f}{\partial x}$: partial derivative of a function f with respect to x
- $\frac{df}{dx}$: total derivative of a function f with respect to x
- ∇: gradient
- $\|\cdot\|$: norm (by default $\|\mathbf{x}\| = \|\mathbf{x}\|_2$).

Math symbols

- $C(X, Y)$ Space of continuous functions $f : X \to Y$.
- $C(\mathbb{R})$ space of continuous functions $f : \mathbb{R} \to \mathbb{R}$, usually endowed with the uniform norm topology
- $C_c(\mathbb{R})$ continuous functions with compact support
- $C([a, b])$ space of all continuous functions that are defined on a closed interval $[a, b]$
- $C_b(\mathbb{R})$ space of continuous bounded functions
- $B(\mathbb{R})$ bounded functions
- $C_0(\mathbb{R})$ continuous functions which vanish at infinity
- $C^r(\mathbb{R})$ continuous functions that have continuous first r derivatives.
- $C^\infty(\mathbb{R})$ smooth functions
- $C_c^\infty(\mathbb{R})$ smooth functions compact support
- $\mathbb{F}$ Field of either real $\mathbb{R}$ or complex numbers $\mathcal{C}$
- ℓ^p is used to indicate a p-summable *discrete* set of values. For example, $\ell^p(\mathbb{Z}^+)$ is the set of complex-valued sequences $\{(a_n)\}$ such that $\sum_{n \in \mathbb{Z}^+} |a_n|^p < \infty$. For example:
  - ℓ^1, the space of sequences whose series is absolutely convergent
  - ℓ^2, the space of square-summable sequences, which is a Hilbert space
  - ℓ^∞, the space of bounded sequences
- L^p is typically used to indicate p-summable functions (with respect to some measure) on a *non-discrete* measure space, such as the usual $L^p(\mathbb{R})$, the set of functions $f : \mathbb{R} \to \mathbb{C}$ such that $\int_{\mathbb{R}} |f(x)|^p \, dx < \infty$.
- The expectation of a random variable X is denoted by $\mathbb{E}[X]$ or $\mathbb{E}_D[X]$ to make the distribution of X, denoted by D, explicit
- Given a symmetric matrix $\mathbf{A}$, $\mathbf{A} \succcurlyeq 0$ means that $\mathbf{A}$ is positive semidefinite, i.e. $\mathbf{x}^\top \mathbf{A} \mathbf{x} \geq 0$ for all $\mathbf{x}$.

1.2 Vector spaces

Vector space Recall that a vector space (also called a linear space) is a collection of objects called vectors, which may be added together and multiplied by scalars. Formally, a vector space over a field F (e.g. the field of real numbers) is a set V together with two operations that satisfy several axioms (associativity, commutativity, identity, inverse).

Subspace A subspace of a vector space V is a subset U of V that is itself a vector space under the same operations of vector addition and scalar multiplication as those defined on V. One simple example is a line through the origin in $\mathbb{R}^2$.

Norm A normed vector space is a space equipped with a norm, i.e. a function from V to $\mathbb{R}$ (we give a formal definition of a norm below). Informally this means that we need to be able to add vectors and scale them with a scalar.

Definition 2. *Given a vector space V over a subfield F of the complex numbers, a norm on V is a nonnegative-valued scalar function $p : V \to [0, +\infty)$ with the following properties:*
For all $a \in F$ and all $u, v \in V$,

- *$p(v) \geq 0$ (non-negativity)*
- *If $p(v) = 0$ then $v = 0$ (positive definite)*
- *$p(av) = |a|p(v)$ (absolutely homogeneous)*
- *$p(u + v) \leq p(u) + p(v)$ (triangle inequality).*

Example 1: vector norm A vector norm is a function that assigns a non-negative scalar value to a vector in a vector space, which intuitively represents the length or size of the vector. More formally, if $\mathbf{v}$ is a vector in a vector space V, a norm $\|\mathbf{v}\|$ satisfies the following properties:

i) **Non-negativity (or Positivity)**:

$$\|\mathbf{v}\| \geq 0 \quad \text{and} \quad \|\mathbf{v}\| = 0 \iff \mathbf{v} = \mathbf{0}$$

This means the norm of any vector is always non-negative, and it is zero if and only if the vector itself is the zero vector.

ii) **Scalar Multiplication**:

$$\|\alpha\mathbf{v}\| = |\alpha|\|\mathbf{v}\|$$

for any scalar α. This means that scaling a vector by a scalar α scales the norm of the vector by the absolute value of α.

iii) **Triangle Inequality**:

$$\|\mathbf{u}+\mathbf{v}\| \leq \|\mathbf{u}\| + \|\mathbf{v}\|$$

for any vectors $\mathbf{u}$ and $\mathbf{v}$. This means that the norm of the sum of two vectors is less than or equal to the sum of the norms of the two vectors.

Common examples of vector norms include:

- **Euclidean norm (or 2-norm)**:

$$\|\mathbf{v}\|_2 = \left(\sum_{i=1}^{n} |v_i|^2 \right)^{1/2}$$

where $\mathbf{v} = (v_1, v_2, \ldots, v_n)$.

- **1-norm (or Manhattan norm)**:

$$\|\mathbf{v}\|_1 = \sum_{i=1}^{n} |v_i|$$

- **Infinity norm (or maximum norm)**:

$$\|\mathbf{v}\|_\infty = \max_{1\leq i\leq n} |v_i|$$

Example 2: matrix norm Suppose a vector norm $\|\cdot\|$ on $\mathbb{R}^m$ is given (e.g. the Euclidean norm). Any $m \times n$ matrix $\mathbf{A}$ induces a linear operator from $\mathbb{R}^n$ to $\mathbb{R}^m$ with respect to the standard basis, and one defines the corresponding operator norm on the space $\mathbb{R}^{m\times n}$ of all $m \times n$ matrices as follows:

$$\begin{aligned}\|\mathbf{A}\| &= \sup\{\|\mathbf{A}\mathbf{x}\| : \mathbf{x} \in \mathbb{R}^n \text{ with } \|\mathbf{x}\| = 1\} \\ &= \sup\left\{ \frac{\|\mathbf{A}\mathbf{x}\|}{\|\mathbf{x}\|} : \mathbf{x} \in \mathbb{R}^n \text{ with } \mathbf{x} \neq 0 \right\}. \end{aligned} \tag{1.1}$$

In particular, if the p-norm for vectors is used for both spaces $\mathbb{R}^n$ and $\mathbb{R}^m$, the corresponding induced operator norm is:

$$\|\mathbf{A}\|_p = \sup\left\{ \frac{\|\mathbf{A}\mathbf{x}\|_p}{\|\mathbf{x}\|_p} : \mathbf{x} \in \mathbb{R}^n \text{ with } \mathbf{x} \neq 0 \right\}. \tag{1.2}$$

In practice, one is often interested in the case $p = 2$, for which

$$\begin{aligned}\|\mathbf{A}\|_2 &= \sup\left\{ \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} : \mathbf{x} \in \mathbb{R}^n \text{ with } \mathbf{x} \neq 0 \right\} \\ &= \sigma_1(\mathbf{A}), \end{aligned} \tag{1.3}$$

where $\sigma_1(\mathbf{A})$ is the largest singular value of $\mathbf{A}$.

Norm inequalities We introduce the Cauchy-Schwarz inequality, which is a pillar inequality in the field of mathematics and will be a recurrent inequality in many proofs discussed in this course.

Proposition 1. *The Cauchy-Schwarz inequality states that for all vectors* $\mathbf{x}$ *and* $\mathbf{y}$ *of an inner product space, the following holds:*

$$|\langle \mathbf{x}, \mathbf{y} \rangle| \leq \|\mathbf{x}\| \|\mathbf{y}\|. \tag{1.4}$$

Writing $\mathbf{x} = [x_1 \ldots x_n]$ *and* $\mathbf{y} = [y_1 \ldots y_n]$, *this can also be written as*

$$\left(\sum_{i=1}^{n} x_i y_i \right)^2 \leq \sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i^2. \tag{1.5}$$

Equality occurs if and only if there exists a constant c *such that* $x_i = c y_i \; \forall i = 1, \ldots n$.

Proof. We will give two different proofs (there are many more):

Proof I) A very simple proof can be obtained using the definition of an inner product:

$$\mathbf{x}^\top \mathbf{y} = \|\mathbf{x}\| \|\mathbf{y}\| \cos(\theta), \tag{1.6}$$

with the fact that $\cos(\theta) \leq 1$.

Proof II) Another proof is as follows. Define the following non-negative function:

$$\begin{aligned} f(z) &= \sum_{i=1}^{n} (x_i - z y_i)^2 \geq 0 \\ &= \sum_{i=1}^{n} (x_i^2 - 2 x_i y_i z + z^2 y_i^2) \\ &= \left(\sum_{i=1}^{n} y_i^2 \right) z^2 - 2 \left(\sum_{i=1}^{n} x_i y_i \right) z + \sum_{i=1}^{n} x_i^2. \end{aligned}$$

Note that this is a quadratic function in z of the form:

$$f(z) = A z^2 - B z + C,$$

and whose minimum value occurs when $z = \frac{B}{2A}$. Since the whole expression has to be non-negative, we need

$$\Delta = B^2 - 4AC \leq 0 \implies C \geq \frac{B^2}{4A} \implies B^2 \leq 4AC.$$

One can verify that $B^2 \leq 4AC$ implies the inequality we are looking for:

$$\left(\sum_{i=1}^{n} x_i y_i\right)^2 \leq 4\left(\sum_{i=1}^{n} y_i^2\right)\left(\sum_{i=1}^{n} x_i^2\right). \tag{1.7}$$

□

1.3 Functions

We here review the definition of a gradient and then discuss two fundamental properties of functions.

1.3.1 Gradients

Consider a real-valued (univariate) function $f : \mathbb{R} \to \mathbb{R}$. Its derivative is defined by

$$f'(x) := \lim_{\epsilon \to 0} \frac{f(x+\epsilon) - f(x)}{\epsilon}. \tag{1.8}$$

Let's generalize this definition to the case where the function f is multivariate, i.e. $f : \mathbb{R}^d \to \mathbb{R}$. In this case, we have one (partial) derivative for each dimension, i.e. $\frac{\partial f}{\partial x_i}$ defined as

$$\frac{\partial f}{\partial x_i} := \lim_{\epsilon \to 0} \frac{f(x_1, \dots, x_i + \epsilon, \dots, x_d) - f(x_1, \dots, x_i, \dots, x_d))}{\epsilon}. \tag{1.9}$$

The function f is *differentiable* if the limit in Eq. (1.9) exists for all x_i.

The gradient denoted by $\nabla f(\mathbf{x})$ gives us a way to pack all partial derivatives into one vector. Denoting by $\mathbf{x} = (x_1, x_2, \dots x_d)$, we then define the gradient as

$$\nabla f(\mathbf{x}) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_d}, \end{pmatrix} \tag{1.10}$$

where we can also define

$$\frac{\partial f}{\partial x_i} := \lim_{\epsilon \to 0} \frac{f(\mathbf{x} + \epsilon \mathbf{e}_i) - f(\mathbf{x})}{\epsilon}, \tag{1.11}$$

with $\mathbf{e}_i$ being a standard basis vector (composed of zeros and a single one at the i-th coordinate). Note that $\frac{\partial f}{\partial x_i}$ is also the directional derivative of f along the direction $\mathbf{e}_i$, and can be expressed as $\nabla f(\mathbf{x}) \cdot \mathbf{e}_i$ (where $\cdot$ is the standard inner product in $\mathbb{R}^d$.

1.3.2 Chain rule

The chain rule is a formula to compute the derivative of a composite function $f(g(x))$. We start by describing the case where the functions involved in the composition are single variable functions, i.e. $f : \mathbb{R} \to \mathbb{R}$ and $g : \mathbb{R} \to \mathbb{R}$. In this case, the chain rule is

$$\frac{d}{dx} f(g(x)) = f'(g(x))g'(x)$$

Alternatively, it can also be written as

$$\frac{d}{dx} f(g(x)) = \frac{df}{dg} \cdot \frac{dg}{dx},$$

where dg is an abbreviation for $dg(x)$.

Proof idea. First note that by definition of the derivative as a limit:

$$(f \circ g)'(a) = \lim_{x \to a} \frac{f(g(x)) - f(g(a))}{x - a}.$$

Assuming that $g(x) \neq g(a)$ any x near a, then the previous expression is equal to the product of two factors:

$$\lim_{x \to a} \frac{f(g(x)) - f(g(a))}{g(x) - g(a)} \cdot \frac{g(x) - g(a)}{x - a} = \lim_{x \to a} \frac{f(g(x)) - f(g(a))}{g(x) - g(a)} \cdot \lim_{x \to a} \frac{g(x) - g(a)}{x - a},$$

where the equality uses the fact that the limit of a product is equal to the product of the limits (limit law).

We have therefore reached the desired expression. The case $g(x) = g(a)$ has to be handled with more care, see e.g. Rudin et al. (1976) for a complete proof.

□

Let's consider the general case where $\mathbf{f} : \mathbb{R}^m \to \mathbb{R}^d$ and $\mathbf{g} : \mathbb{R}^k \to \mathbb{R}^m$. Then the composed function is $f(g(\mathbf{x})) : \mathbb{R}^k \to \mathbb{R}^d$.

In this case, the chain rule is written as

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{f}(\mathbf{g}(\mathbf{x})) = \frac{\partial \mathbf{f}}{\partial \mathbf{g}}(g(\mathbf{x})) \cdot \frac{\partial \mathbf{g}}{\partial \mathbf{x}}(\mathbf{x}) \in \mathbb{R}^{d \times k},$$

where $\frac{\partial \mathbf{f}}{\partial \mathbf{g}}(g(\mathbf{x})) \in \mathbb{R}^{d \times m}$ is the Jacobian matrix of $\mathbf{f}$ with respect to $\mathbf{g}$ and $\frac{\partial \mathbf{g}}{\partial \mathbf{x}}(\mathbf{x}) \in \mathbb{R}^{m \times k}$ is the Jacobian matrix of $\mathbf{g}$ with respect to $\mathbf{x}$.

The Jacobian $\frac{\partial \mathbf{f}}{\partial \mathbf{g}}(g(\mathbf{x})) \in \mathbb{R}^{d \times m}$ contains all first-order partial derivatives of the components of $\mathbf{f}$ w.r.t. the components of $\mathbf{g}$. Specifically, if $\mathbf{f} = [f_1, f_2, \ldots, f_d]^\top$ and $\mathbf{g} = [g_1, g_2, \ldots, g_m]^\top$, then the (i, j)-th entry of this Jacobian matrix is $\frac{\partial f_i}{\partial g_j}$.

1.3.3 Taylor's Theorem

Taylor's theorem gives an approximation of a k-times differentiable function f around a given point.

Scalar functions We first state the scalar version where the function f is defined over $\mathbb{R}$.

Theorem 2 (Taylor's theorem)**.** *Let $k \geq 1$ be an integer and let the function $f : \mathbb{R} \to \mathbb{R}$ be k times differentiable at the point $a \in \mathbb{R}$. Then there exists a function $h : \mathbb{R} \to \mathbb{R}$ such that*

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots + \frac{f^{(k)}(a)}{k!}(x-a)^k + h_k(x)(x-a)^k, \tag{1.12}$$

and

$$\lim_{x \to a} h_k(x) = 0. \tag{1.13}$$

Proof. We prove it for $k = 3$ as the generalization follows easily.

Note that we assumed that the function f is k-times differentiable. We can therefore repeatedly apply the fundamental theorem of calculus as follows:

$$\begin{aligned}
f(x) &= f(a) + (f(x) - f(a)) = f(a) + \int_a^x f'(x_1)dx_1 \\
&= f(a) + \int_a^x f'(a) + (f'(x_1) - f'(a))dx_1 \\
&= f(a) + f'(a)(x-a) + \int_a^x (f'(x_1) - f'(a))dx_1 \\
&= f(a) + f'(a)(x-a) + \int_a^x \int_a^{x_1} f^{(2)}(x_2)dx_2dx_1 \\
&= \dots \\
&= f(a) + f'(a)(x-a) + \int_a^x \int_a^{x_1} f^{(2)}(a) + (f^{(2)}(x_2) - f^{(2)}(a))dx_2dx_1 \\
&= f(a) + f'(a)(x-a) + f^{(2)}(a)(x-a)^2 + \int_a^x \int_a^{x_1} \int_a^{x_2} f^{(3)}(x_3)dx_3dx_2dx_1.
\end{aligned}$$

Recall $f^{(2)}$ is continuous, therefore $|f^{(3)}(x)| \leq M \ \forall x$. The remainder can therefore be bounded as follows:

$$
\begin{aligned}
\left| \int_a^x \int_a^{x_1} \int_a^{x_2} f^{(3)}(x_3)\, dx_3 dx_2 dx_1 \right| &\leq \int_a^x \int_a^{x_1} \int_a^{x_2} |f^{(3)}(x_3)|\, dx_3 dx_2 dx_1 \\
&\leq M \int_a^x \int_a^{x_1} \int_a^{x_2} 1\, dx_3 dx_2 dx_1 \\
&= M \frac{(x-a)^3}{3!}
\end{aligned}
$$

□

By choosing $k = 1$ in Theorem 2, we recover the mean value theorem stated below.

Theorem 3 (Mean value theorem (scalar version))**.** *Let $f : [a, b] \to \mathbb{R}$ be a continuous function on the closed interval $[a, b]$, and differentiable on the open interval (a, b), where $a < b$. Then there exists some $c \in (a, b)$ such that*

$$
f'(c) = \frac{f(b) - f(a)}{b - a} \tag{1.14}
$$

Multivariable functions An extension of the previous result to a multivariate function $f : \mathbb{R}^d \to \mathbb{R}$ can be stated in two ways, either as

$$
f(\mathbf{x} + \mathbf{y}) - f(\mathbf{x}) = \nabla f(\mathbf{x} + c\mathbf{y})^\top \mathbf{y}, \text{ for some } c \in (0, 1), \tag{1.15}
$$

or in an integral form, as stated in the theorem below.

Theorem 4 (Mean value theorem (multivariable version))**.** *Given a continuously differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ and any vector $\mathbf{y} \in \mathbb{R}^d$, we have that*

$$
f(\mathbf{x} + \mathbf{y}) - f(\mathbf{x}) = \int_0^1 \nabla f(\mathbf{x} + t\mathbf{y})^\top \mathbf{y}\, dt. \tag{1.16}
$$

Proof. Define $g : [0, 1] \to \mathbb{R}$ as $g(t) = f(\mathbf{x} + t\mathbf{y})$. Then by applying the fundamental theorem of calculus to g, we get

$$
f(\mathbf{x} + \mathbf{y}) - f(\mathbf{x}) = g(1) - g(0) = \int_0^1 g'(t) dt.
$$

Let $\mathbf{u}(t) = \mathbf{x} + t\mathbf{y}$ with entries $u_i(t)$. By the multivariate chain rule,

$$
g'(t) = \sum_{i=1}^d \frac{\partial f(\mathbf{x} + t\mathbf{y})}{\partial u_i} \cdot \frac{\partial u_i}{\partial t} = \sum_{i=1}^d \frac{\partial f(\mathbf{x} + t\mathbf{y})}{\partial u_i} \cdot y_i = \nabla f(\mathbf{x} + t\mathbf{y})^\top \mathbf{y}.
$$

Therefore

$$f(\mathbf{x}+\mathbf{y}) - f(\mathbf{x}) = \int_0^1 \nabla f(\mathbf{x}+t\mathbf{y})^\top \mathbf{y}\; dt.$$

□

Exercise Try to prove Eq. (1.15) by modifying the proof of Theorem 4.

High-order variant A similar expression to Eq. (1.16) can be stated for twice differentiable functions:

$$f(\mathbf{x}+\mathbf{y}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^\top \mathbf{y} + \frac{1}{2}\int_0^1 \mathbf{y}^\top \nabla^2 f(\mathbf{x}+t\mathbf{y})\mathbf{y}\; dt. \tag{1.17}$$

Gradient Note that the mean value theorem also applies to the gradient ∇f for functions f that are twice differentiable. It of course also applies to higher-order derivatives if they exist. For the first derivative, we simply get:

$$\nabla f(\mathbf{x}+\mathbf{y}) = \nabla f(\mathbf{x}) + \int_0^1 \nabla^2 f(\mathbf{x}+t\mathbf{y})^\top \mathbf{y}\; dt. \tag{1.18}$$

1.3.4 Calculating gradients

Example: Gradient of a Multivariable Function

Consider the function $f(\mathbf{x}) = \mathbf{A}\mathbf{x} \in \mathbb{R}^p$, where $\mathbf{x} \in \mathbb{R}^d, \mathbf{A} \in \mathbb{R}^{p\times d}$. Our goal is to compute the gradient $\frac{df}{d\mathbf{x}}$.
First, note that the dimension of the gradient $\frac{df}{d\mathbf{x}}$ is $\mathbb{R}^{p\times d}$. Let's compute the partial derivative of f w.r.t. a single x_j. We have

$$f_i(\mathbf{x}) = \sum_{j=1}^d A_{ij}x_j \implies \frac{\partial f_i}{\partial x_j} = A_{ij}.$$

Collecting all the partial derivatives in the Jacobian, we obtain the following expression for the gradient:

$$\frac{df}{d\mathbf{x}} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_d} \\ & \dots & \\ \frac{\partial f_p}{\partial x_1} & \dots & \frac{\partial f_p}{\partial x_d} \end{pmatrix} = \begin{pmatrix} A_{11} & \dots & A_{1d} \\ & \dots & \\ A_{p1} & \dots & A_{pd} \end{pmatrix} = \mathbf{A} \in \mathbb{R}^{p\times d}.$$

Composition of functions and chain rule

When considering compositions of functions of vectors or matrices, one often requires the use of the chain rule to calculate gradients. For instance, assume we want to calculate a loss $L(\mathbf{g}(\mathbf{x})) := \|\mathbf{g}(\mathbf{x})\|_2^2$, where $\mathbf{g} : \mathbb{R}^d \to \mathbb{R}^n$. By the chain rule, we have

$$\frac{\partial L}{\partial \mathbf{x}} = \frac{\partial L}{\partial \mathbf{g}} \cdot \frac{\partial \mathbf{g}}{\partial \mathbf{x}}. \tag{1.19}$$

As an example, let's consider the minimization of the least-square loss (with a linear model), which is defined as

$$\min_{\mathbf{x} \in \mathbb{R}^d} \left[L(\mathbf{A}\mathbf{x}) := \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 \right], \tag{1.20}$$

where $\mathbf{A} \in \mathbb{R}^{n \times d}$, $\mathbf{x} \in \mathbb{R}^d$ and $\mathbf{y} \in \mathbb{R}^n$ (thus $L : \mathbb{R}^n \to \mathbb{R}$). Sometimes, we will simply write $L(\mathbf{x}) := L(\mathbf{A}\mathbf{x})$.

In this case, we have $\mathbf{g}(\mathbf{x}) = \mathbf{A}\mathbf{x} - \mathbf{y} \in \mathbb{R}^{n \times 1}$ (column vector) and $L(\mathbf{g}) = \|\mathbf{g}\|_2^2$, thus $\frac{\partial L}{\partial \mathbf{g}} = 2\mathbf{g}^\top$. The other derivative is simply $\frac{\partial \mathbf{g}}{\partial \mathbf{x}} = \mathbf{A}$, therefore

$$\frac{\partial L}{\partial \mathbf{x}} = \frac{\partial L}{\partial \mathbf{g}} \cdot \frac{\partial \mathbf{g}}{\partial \mathbf{x}} = 2(\mathbf{A}\mathbf{x} - \mathbf{y})^\top \mathbf{A}. \tag{1.21}$$

Note that with the above notation, $\frac{\partial L}{\partial \mathbf{x}}$ is a row vector since $\frac{\partial L}{\partial \mathbf{x}} \in \mathbb{R}^{1 \times d}$, but by convention, one might want a column vector, in which case we should transpose the result to obtain $2\mathbf{A}^\top(\mathbf{A}\mathbf{x} - \mathbf{y})$.

One alternative to the chain rule for this loss can be obtained by noting that

$$L(\mathbf{x}) := \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 = (\mathbf{A}\mathbf{x} - \mathbf{y})^\top (\mathbf{A}\mathbf{x} - \mathbf{y}), \tag{1.22}$$

and taking the derivative of the inner product (exercise: try to figure this out yourself).

1.3.5 Lipschitz property

Intuitively, a Lipschitz continuous function is limited in how fast it can change. See the formal definition below.

Definition 3. *The function $f : \mathbb{R}^d \to \mathbb{R}$ is L-Lipschitz continuous if:*

$$|f(\mathbf{x}) - f(\mathbf{y})| \le L \|\mathbf{x} - \mathbf{y}\| \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{1.23}$$

The following lemma gives a characterization of Lipschitz continuity using the gradient.

Lemma 5. *A function $f(\mathbf{x})$ is L-Lipschitz continuous if its gradient is bounded by L.*

Proof. By the mean value theorem, we have :

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

Using the Cauchy-Schwarz inequality (see Proposition 1), we conclude that

$$|f(\mathbf{x}) - f(\mathbf{y})| \leq \sup_{\mathbf{z} \in \mathbb{R}^d} \|\nabla f(\mathbf{z})\| \|\mathbf{x} - \mathbf{y}\| \leq L \|\mathbf{x} - \mathbf{y}\|.$$

□

1.3.6 Smoothness

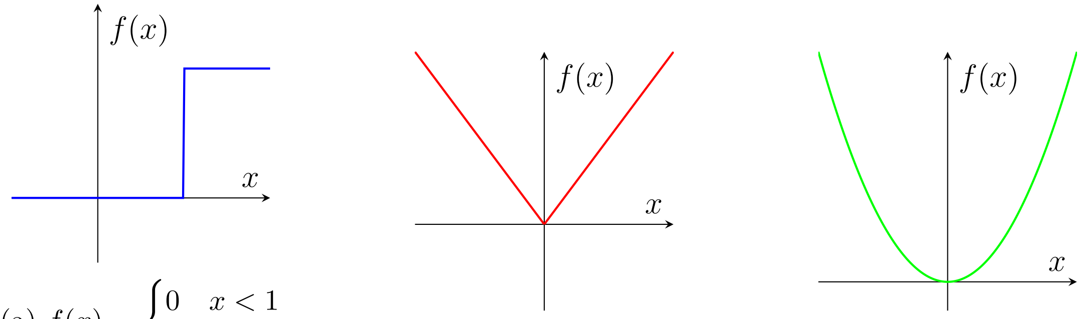
We say that a differentiable function is smooth if its gradient is Lipschitz continuous. We formalize this below.

Definition 4. *The function $f : \mathbb{R}^d \to \mathbb{R}$ is differentiable with L-Lipschitz-continuous gradient, i.e.*

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L \, \|\mathbf{x} - \mathbf{y}\| \, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{1.24}$$

Figure 1.2 shows some examples of continuous and differentiable functions. For instance, the function $f(x) = x^2$ is smooth with constant $L = 2$ since:

$$|f'(x) - f'(y)| = 2|x - y|.$$



(a) $f(x) = \begin{cases} 0 & x < 1 \\ 1 & x \geq 1 \end{cases}$

(b) $f(x) = |x|$

(c) $f(x) = x^2$

Continuous: No
Differentiable: No

Continuous: Yes
Differentiable: No

Continuous: Yes
Differentiable: Yes

Figure 1.2: Examples of functions illustrating their continuity and differentiability properties.

1.4 Linear Algebra

1.4.1 Eigenvalues

We recall the definition of eigenvalues and eigenvectors in a real-vector space (this definition can be generalized to the complex domain but we will mostly deal with reals in this lecture). Consider a square matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$ (recall that eigenvalues are not defined for rectangular matrices, for that we need the concept of singular values). Then $\lambda \in \mathbb{R}$ is an eigenvalue of $\mathbf{A}$ and $\mathbf{v} \in \mathbb{R}^d \backslash \{\mathbf{0}\}$ is the corresponding eigenvector of $\mathbf{A}$ if the following holds:

$$\mathbf{A}\mathbf{v} = \lambda \mathbf{v}. \tag{1.25}$$

Note that this relation is also equivalent to $(\mathbf{A} - \lambda \mathbf{I})\mathbf{v} = 0$ or $\det(\mathbf{A} - \lambda \mathbf{I}) = 0$. The polynomial $\det(\mathbf{A} - \lambda \mathbf{I})$ plays an important role in linear algebra. For instance, the number of distinct eigenvalues of $\mathbf{A}$ is equal to the multiplicity of λ as a root of this polynomial.

In the following, we will denote the eigenvalues of $\mathbf{A}$ by λ_i and assume there are sorted as follows:

$$\lambda_1(\mathbf{A}) \geq \ldots \geq \lambda_n(\mathbf{A}).$$

Trace of a matrix The trace of a square matrix $\mathbf{A}$, denoted as $\mathrm{tr}(\mathbf{A})$, is the sum of its diagonal elements. If $\mathbf{A}$ is an $n \times n$ matrix given by:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

then the trace of $\mathbf{A}$ is defined as:

$$\mathrm{tr}(\mathbf{A}) = \sum_{i=1}^{n} a_{ii}$$

The trace of a matrix is also equal to the sum of its eigenvalues. If $\lambda_1, \lambda_2, \ldots, \lambda_n$ are the eigenvalues of $\mathbf{A}$, then:

$$\mathrm{tr}(\mathbf{A}) = \sum_{i=1}^{n} \lambda_i.$$

Below are some important properties of the trace we will be using later on:

- **Linearity**: For any two $n \times n$ matrices $\mathbf{A}$ and $\mathbf{B}$, and any scalars α and β,

$$\mathrm{tr}(\alpha \mathbf{A} + \beta \mathbf{B}) = \alpha \, \mathrm{tr}(\mathbf{A}) + \beta \, \mathrm{tr}(\mathbf{B})$$

- **Cyclic property**: For any $n \times n$ matrices $\mathbf{A}$ and $\mathbf{B}$,

$$\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$$

- **Trace of a transpose**: For any $n \times n$ matrix $\mathbf{A}$,
$$\operatorname{tr}(\mathbf{A}^\top) = \operatorname{tr}(\mathbf{A})$$

Loewner order Recall that a matrix $\mathbf{A}$ is positive semi-definite (PSD) if $\mathbf{u}^\top \mathbf{A}\mathbf{u} \geq 0$ for all $\mathbf{u} \in \mathbb{R}^d$, or equivalently all the eigenvalues λ_i of $\mathbf{A}$ are non-negative, i.e. $\lambda_i(\mathbf{A}) \geq 0 \;\; \forall\, i = 1, \ldots d$. We will use the notation $\mathbf{A} \succcurlyeq 0$ to denote that $\mathbf{A}$ is PSD.

Definition 5 (Loewner order)**.** *Let $\mathbf{A}$ and $\mathbf{B}$ be two symmetric matrices. We say that $\mathbf{A} \succcurlyeq \mathbf{B}$ if $\mathbf{A} - \mathbf{B}$ is positive semi-definite.*

Note that $\succcurlyeq$ is only a partial order (instead of a total order).

1.4.2 Singular values

The main object of interest will now be a rectangular $m \times n$ matrix $\mathbf{A}$ with real entries. We denote by $\sigma_i(\mathbf{A})$ the i-th singular value of $\mathbf{A}$ which is equal to

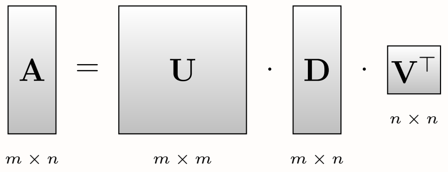
$$\sigma_i(\mathbf{A}) = \sqrt{\lambda_i(\mathbf{A}\mathbf{A}^\top)} = \sqrt{\lambda_i(\mathbf{A}^\top \mathbf{A})}. \tag{1.26}$$

The number of nonzero singular values of $\mathbf{A}$ equals to $\operatorname{rank}(\mathbf{A}) \leq \min(m, n)$, where we recall that the rank of a matrix is the maximum number of linearly independent rows or columns in the matrix.

We also note that given a matrix $\mathbf{A}$, the largest singular value $\sigma_1(\mathbf{A})$ is equal to the operator norm of $\mathbf{A}$.

Singular Value Decomposition (SVD) The Singular Value Decomposition (SVD) is a widely-used technique to decompose a matrix $\mathbf{A}$, and to expose some of its properties.

Theorem 6 (SVD)**.** *Every real (or complex) matrix $\mathbf{A}$ can be decomposed into*



where $\mathbf{U}$, $\mathbf{V}$ orthogonal (or unitary), $\mathbf{D}$ diagonal. More precisely:

- $\mathbf{U}$ *is an m by m orthogonal matrix,* $\mathbf{U}^\top\mathbf{U} = \mathbf{U}\mathbf{U}^\top = \mathbf{I}$*,*
- $\mathbf{D}$ *is an m by n diagonal matrix, padded with* $\max\{m, n\} - \min\{m, n\}$ *zero rows or columns,*
- $\mathbf{V}$ *is an n by n orthogonal matrix,* $\mathbf{V}^\top\mathbf{V} = \mathbf{V}\mathbf{V}^\top = \mathbf{I}$.

Proof. Omitted. See standard linear algebra textbook, e.g. (Golub and Van Loan, 2013). □

We here recall the definition of orthogonal vectors and matrices for the convenience of the reader.

Definition 6 (Orthogonal vectors)**.** *A set of vectors* $\{\mathbf{v}_1, \ldots, \mathbf{v}_K\}$ *in an inner product space are orthogonal if all pairwise inner products are zero, i.e.*

$$\langle \mathbf{v}_i, \mathbf{v}_j \rangle = 0, \quad (\forall i \neq j).$$

Definition 7 (Orthogonal matrix)**.** *An orthogonal matrix* $\mathbf{A}$ *is a square matrix with real entries whose columns and rows are orthogonal unit vectors (i.e. orthonormal vectors):*

$$\langle \mathbf{a}_i, \mathbf{a}_j \rangle = \delta_{ij} \quad (\forall i, j).$$

Equivalently, $\mathbf{A}\mathbf{A}^\top = \mathbf{A}^\top\mathbf{A} = \mathbf{I}$.

Theorem 7. *The inverse of an orthogonal matrix is its transpose.*

Proof. It follows directly from the above equations as

$$\mathbf{A}^\top\mathbf{A} = \mathbf{I} \quad \Longrightarrow \quad \mathbf{A}^\top = \mathbf{A}^{-1}.$$

□

SVD: Singular Values The elements on the diagonal of $\mathbf{D}$ are the singular values and will be denoted by $\sigma_i := d_{ii}$ $(i = 1, \ldots, \min\{m, n\})$, i.e.

$$\mathbf{D} = \operatorname{diag}\left(\sigma_1, \ldots, \sigma_{\min\{m,n\}}\right).$$

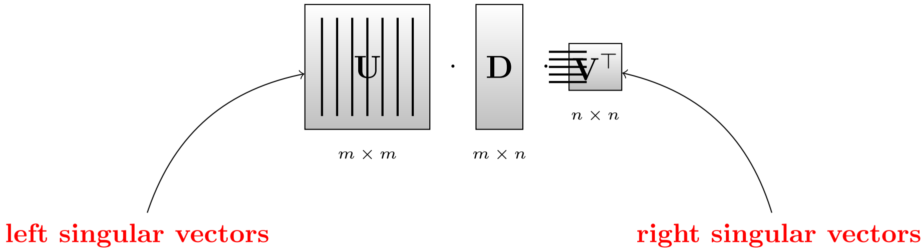
By convention, we typically order the singular values in decreasing order:

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{\min\{m,n\}} \geq 0$$

Also note the non-negativity of singular values. Finally, an important notion related to singular values is the rank of the matrix $\mathbf{A}$ defined as

$$\sigma_i = d_{ii} = 0, \quad (\forall i > \operatorname{rank}(\mathbf{A})).$$

SVD: Singular Vectors The columns of $\mathbf{U}$ (denoted by $\mathbf{u}_i \in \mathbb{R}^m$) are called the left singular vectors and they form an orthonormal basis for columns space of $\mathbf{A}$. Similarly, the rows of $\mathbf{V}^\top$ = columns of $\mathbf{V}$ (denoted by $\mathbf{v}_i \in \mathbb{R}^n$) are called the right singular vectors and form an orthonormal basis for the row space of $\mathbf{A}$.



The left and right singular vectors $\mathbf{u}_i \in \mathbb{R}^m$ and $\mathbf{v}_i \in \mathbb{R}^n$ are also the eigenvectors of $\mathbf{A}\mathbf{A}^\top$ and $\mathbf{A}^\top\mathbf{A}$ respectively.

We note that the SVD decomposition is only unique up to rotation and degeneracies (i.e. σ_i with two or more linearly independent left (or right) singular vectors).

SVD as sum of rank-1 matrices Let the columns of $\mathbf{U}$ be denoted by $\mathbf{u}_i$ and the columns of $\mathbf{V}$ be denoted by $\mathbf{v}_i$. Then by multiplying the SVD equation by $\mathbf{V}$ one gets

$$\mathbf{A}\mathbf{V} = \mathbf{U}\mathbf{D} \quad \iff \quad \mathbf{A}\mathbf{v}_i = \sigma_i \mathbf{u}_i \quad (\forall i \leq \min\{m, n\})$$

Similarly

$$\mathbf{A}^\top\mathbf{U} = \mathbf{V}\mathbf{D}^\top \iff \mathbf{A}^\top\mathbf{u}_i = \sigma_i \mathbf{v}_i \quad (\forall i \leq \min\{m, n\})$$

Note that for the special case of $m = n$ and $\mathbf{U} = \mathbf{V}$ (i.e. $\mathbf{A}$ symmetric) we retrieve the eigendecomposition. In this case, the vectors $\mathbf{u}_i = \mathbf{v}_i$ are just the eigenvectors.

Based on the above equations, we observe that we can write the SVD decomposition of a matrix as a sum of rank-1 matrices:

$$\mathbf{A} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^\top, \text{where } r = \text{rank}(\mathbf{A}), \tag{1.27}$$

where $\mathbf{u}_i\mathbf{v}_i^\top$ is an outer product between two vectors and therefore has rank 1 (exercise: try to prove this claim yourself).

Interpretation As shown in Figure 1.3, the SVD gives us a way to decompose a linear map as three subsequent operations: rotation, axis scaling, and another rotation

Relation between singular values and eigenvalues For symmetric matrices, the eigenvalues and singular values are closely related. A nonnegative eigenvalue, $\lambda \geq 0$, is also a singular value, $\sigma = \lambda$. The corresponding vectors are equal to each other, $\mathbf{u} = \mathbf{v}$. A negative eigenvalue, $\lambda < 0$, must reverse its sign to become a singular value, $\sigma = |\lambda|$. One of the corresponding singular vectors is the negative of the other, $\mathbf{u} = -\mathbf{v}$. So in general, if $\mathbf{A}$ is a symmetric matrix then the singular values of $\mathbf{A}$ are the absolute values of the eigenvalues λ_i of $\mathbf{A}$: $\sigma_i(\mathbf{A}) = |\lambda_i(\mathbf{A})|$.

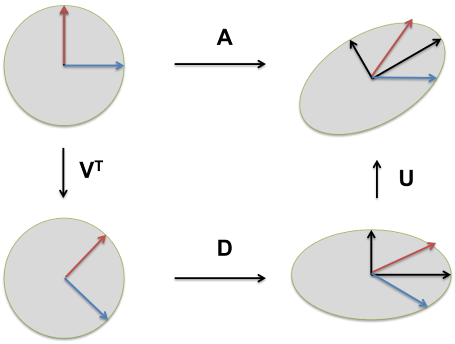
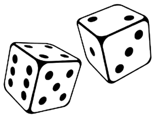


Figure 1.3: Illustration of the SVD transformation where: **Top:** The effect of $\mathbf{A}$ on the unit disc D and the canonical unit vectors $\mathbf{e}_1$ and $\mathbf{e}_2$. **Left:** The effect of $\mathbf{V}^\top$ is a rotation on D, $\mathbf{e}_1$, and $\mathbf{e}_2$. **Bottom:** The effect of $\mathbf{D}$, scaling horizontally by the singular value σ_1 and vertically by σ_2. **Right:** The effect of $\mathbf{U}$, another rotation. Source: Wikipedia.

1.5 Probability Theory

We first recall the definition of a probability space.

Definition 8 (Probability space)**.** *A probability space W is a unique triple $W = \{\Omega, \mathcal{F}, \mathbb{P}\}$, where Ω is its sample space , $\mathcal{F}$ its σ-algebra of events, and $\mathbb{P}$ its probability measure.*



Sample space The sample space Ω is the set of all possible samples or elementary events ω. Take for example the case where we throw a die once and define the random Variable X (we will later give a formal definition of this concept) as ”the score shown on the top face”. This random variable X can take the values 1, 2, 3, 4, 5 or 6. Therefore, the sample space is $\Omega := \{1, 2, 3, 4, 5, 6\}$. Let us list a few more examples of sample spaces:

- Toss of a coin (with head H and tail T): $\Omega = \{H, T\}$
- Two tosses of a coin: $\Omega = \{HH, HT, TH, TT\}$
- The positive integers: $\Omega = 1, 2, 3, \ldots$.

σ-algebra A σ-algebra is a collection of subsets of the sample space that satisfies certain properties. The σ-algebra represents the collection of events for which we can assign probabilities. It provides a structure to define and manipulate sets of outcomes or events.

Formally, the σ-algebra $\mathcal{F}$ is the set of all of the *considered* events A, i.e., subsets of Ω: $\mathcal{F} = \{A | A \subseteq \Omega, A \in \mathcal{F}\}$. It also has to satisfy the following properties:

1. $\Omega \in \mathcal{F}$ and $\emptyset \in \mathcal{F}$

2. $A \in \mathcal{F} \Rightarrow \Omega \backslash A \in \mathcal{F}$ (closed under complementation)

3. $A_1, A_2, \cdots \in \mathcal{F} \Rightarrow \cup_n A_n \in \mathcal{F}$ (closed under countable unions)

Below are several examples of σ-algebra:

- **The trivial σ-algebra:**
  The smallest σ-algebra on any set X is the trivial σ-algebra, which contains only the empty set and the set itself.
  $$\mathcal{A} = \{\emptyset, X\}$$

- **The Power set σ-algebra:**
  The largest σ-algebra on a set X is the power set of X, which contains all subsets of X.
  $$\mathcal{A} = \mathcal{P}(X)$$

- **The σ-Algebra generated by a partition:**
  If $X = \{1, 2, 3, 4\}$ and we consider the partition $\{\{1, 2\}, \{3, 4\}\}$, the σ-algebra generated by this partition is:
  $$\mathcal{A} = \{\emptyset, \{1, 2\}, \{3, 4\}, \{1, 2, 3, 4\}\}$$

- **The Borel σ-algebra on $\mathbb{R}$:**
  The Borel σ-algebra on the real numbers $\mathbb{R}$, denoted by $\mathcal{B}(\mathbb{R})$, is generated by all open intervals $(a, b) \subset \mathbb{R}$. This σ-algebra contains all Borel sets, which are the sets that can be formed from open intervals using countable unions, countable intersections, and relative complements.

- **σ-Algebra on a finite set:**
  If $X = \{a, b, c\}$, one possible σ-algebra on X could be:
  $$\mathcal{A} = \{\emptyset, \{a\}, \{b, c\}, \{a, b, c\}\}$$

- **The σ-Algebra generated by a single set:**
  Let X be a set and $A \subseteq X$. The σ-algebra generated by A is:
  $$\mathcal{A} = \{\emptyset, A, A^c, X\}$$

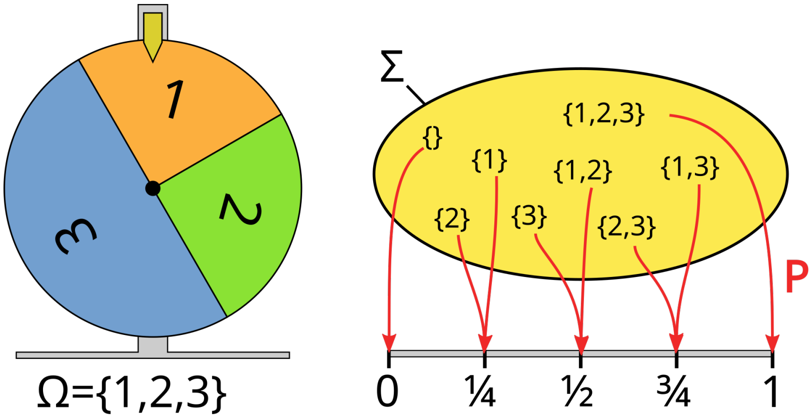


Figure 1.4: A probability measure mapping the σ-algebra for 2^3 events to the unit interval. For example, given three elements 1, 2, and 3 with probabilities 1/4, 1/4, and 1/2, the value assigned to the set $\{1, 3\}$ is $1/4 + 1/2 = 3/4$. Source: Wikipedia.

Probability measure A function $\mathbb{P} : \mathcal{F} \to [0, 1]$ is called a probability measure if it satisfies the following three properties:

1. Non-negativity: For every $A \in \mathcal{F}$, $\mathbb{P}(A) \geq 0$.
2. Normalization: $\mathbb{P}(\Omega) = 1$, where Ω is the entire sample space.
3. σ-additivity: For any countable collection $\{A_n\}$ of pairwise disjoint sets in $\mathcal{F}$ (i.e., $A_i \cap A_j = \varnothing$ for $i \neq j$), the probability of the union is equal to the sum of the probabilities of the individual sets:

$$\mathbb{P}\left(\bigcup_{n=1}^{\infty} A_n\right) = \sum_{n=1}^{\infty} \mathbb{P}(A_n).$$

An illustration of a probability measure is shown in Figure 1.4.

Random variable A random variable is similar to a variable in mathematics, but instead of taking on just one value, it can take on a range of possible values, each with a certain probability. Below, we state a more formal definition of a random variable.

Definition 9 (Random variable)**.** *Given a probability space* $(\Omega, \mathcal{F}, \mathbb{P})$*, a random variable is a function from* Ω *to a measurable space* E [a]*. This function must satisfy the constraint* $\{\omega : X(\omega) \in B\} \in \mathcal{F}$ *for any Borel set* $B \subset E$[b]*, i.e. it is a valid event* $\forall B \in E$*. Each random variable has a distribution* $F_X(x) = \mathbb{P}(X \leq x)$*.*

[a] A measurable space is a set equipped with a collection of subsets that are designated as measurable.

[b] In simple terms, a Borel set is any set that you can create starting from open sets and applying these operations a finite or countable number of times

More formal definition of a Borel set A Borel set is a special type of set that is defined using the concept of open sets in a topological space. The collection of Borel sets includes all open sets and is closed under operations like countable union, countable intersection, and relative complement. For example, consider the real number line. An open interval like $(0, 1)$ is a Borel set because it is an open set. If you take the union of open intervals $(0, 1)$ and $(2, 3)$, you get another Borel set, $(0, 1) \cup (2, 3)$. Even if you take more complex combinations like countable unions or intersections of such intervals, the resulting sets are still Borel sets. So, any set that can be built from open intervals on the real line using operations like union, intersection, and complement will be a Borel set.

Given a set $S \subseteq E$, the probability of a random variable is defined as

$$\mathbb{P}(X \in S) = \mathbb{P}(\{\omega \in \Omega | X(\omega) \in S\}). \tag{1.28}$$

If X maps onto a finite or countable set, it is *discrete* and has a probability mass function (PMF) where $p_X(x) = \mathbb{P}(X = x)$.
If $dF_X(x)/dx$ exists and is finite for all x, then X is continuous and has a density $f_X(x) = dF_X(x)/dx$.

Example of random variables:
a) Throw two dices and take $\Omega = \{1, 2, 3, 4, 5, 6\}^2$ and $\mathcal{F} = P(\Omega)$ where P is the power set, i.e. the set of all subsets of Ω (thus $\mathcal{F} = \{(1, 1), (1, 2), \ldots\}$). Then $X : \Omega \to \mathbb{R}$ defined as $(\omega_1, \omega_2) \to \omega_1 + \omega_2$ (i.e. sum the numbers on each die) is indeed a random variable.
b) Throw two dices and take $\Omega = \{1, 2, 3, 4, 5, 6\}^2$ and $\mathcal{F} = \{\emptyset, \Omega\}$. Then X as defined in a) is not a random variable. For instance $X^{-1}(\{2\}) = \{(1, 1)\} \notin \mathcal{F}$.

Continuous vs discrete probability spaces The following table compares the case where the probability space is continuous and discrete (note that one can also mix them but we won't be discussing this case in these notes).

Probability density function (PDF) and probability mass function (PMF) A probability density function is most commonly associated with absolutely continuous univariate distributions. A random variable X has density f_X, where f_X is a non-negative Lebesgue-integrable function, if:

$$\mathbb{P}[a \leq X \leq b] = \int_a^b f_X(x)\, dx. \tag{1.29}$$

Hence, if F_X is the cumulative distribution function of X, then:

$$F_X(x) = \int_{-\infty}^{x} f_X(u)\, du, \tag{1.30}$$

Characteristic	**Discrete Probability Space**	**Continuous Probability Space**
Definition	Consists of a sample space Ω, a σ-algebra $\mathcal{F}$, and a probability mass function (PMF) $\mathbb{P}(X = x)$	Consists of a sample space Ω, a σ-algebra $\mathcal{F}$, and a probability density function (PDF) $f(x)$
Sample Space	Countable and finite or countably infinite, e.g. $\Omega = \{1, 2, 3, 4, 5, 6\}$ (for a six-sided die)	Uncountably infinite, e.g. $\Omega = [0, 1]$
σ-algebra	Typically the power set $\mathcal{F} = P(\Omega)$	Typically the Borel set formed by Ω
Probability Function	PMF $\mathbb{P}(X = x)$ for each x in Ω	PDF $f(x)$ such that $\mathbb{P}(a \leq X \leq b) = \int_a^b f(x)\, dx$
Probability Assignment	$\mathbb{P}(X = x) \geq 0$ for all x and $\sum_{x\in\Omega} \mathbb{P}(X = x) = 1$	$f(x) \geq 0$ for all x and $\int_{-\infty}^{\infty} f(x)\, dx = 1$
Probability at a Point	$\mathbb{P}(X = x) > 0$ for some x	$\mathbb{P}(X = x) = 0$ for all x
Sum/Integral of Probabilities	$\sum_{x\in\Omega} \mathbb{P}(X = x) = 1$	$\int_{-\infty}^{\infty} f(x)\, dx = 1$
Random Variable	X takes values in Ω with probabilities assigned by the PMF	X takes values in Ω with probabilities assigned by the PDF
Example	Throwing a six-sided die: $\mathbb{P}(X = 1) = \frac{1}{6}, \mathbb{P}(X = 2) = \frac{1}{6}, \ldots, \mathbb{P}(X = 6) = \frac{1}{6}$	Choosing a point in $[0, 1]$: $f(x) = 1$ for $0 \leq x \leq 1$, and $f(x) = 0$ elsewhere

Table 1.1: Differences Between Discrete and Continuous Probability Spaces

and (if f_X is continuous at x)

$$f_X(x) = \frac{d}{dx} F_X(x). \tag{1.31}$$

Intuitively, one can think of $f_X(x)\, dx$ as being the probability of X falling within the infinitesimal interval$[x, x + dx]$.

Similar properties hold for discrete probability spaces where the density function $f_X(x)$ is replaced by a discrete function $p : \mathcal{F} \to [0, 1]$ defined by $p_X(x) = \mathbb{P}(X = x)$. Since the function p is defined over a discrete set, the integral is replaced by a sum, therefore for a given set A, we have

$$\mathbb{P}(A) = \sum_{\omega\in A} p_X(\omega).$$

Expectation If a random variable X has a continuous density $f_X(x)$, then its expectation is given by

$$\mathbb{E}[X] = \int_{-\infty}^{\infty} x f_X(x)\,dx.$$

For a discrete random variable with a finite list $x_1, \ldots, x_k$ of possible outcomes each of which (respectively) has probability $p_1, \ldots, p_k$ of occurring, the expectation can be written as

$$\mathbb{E}[X] = \sum_{i=1}^{k} x_i\, p_i.$$

Finally, the expected value of a non-negative random variable can be written as the integral of its tail distribution. This is because if X is a nonnegative real number, then

$$X = \int_0^{+\infty} [X \geq t] \mathrm{d}t,$$

where $[\cdot]$ is the indicator function.

Then one integrates both sides of the relevant identity with respect to the distribution $\mathbb{P}_X$ of X and one uses Fubini's theorem to change the order of the summation/integral and of the expectation:

$$\mathbb{E}[X] = \int_\Omega X \ \mathrm{d}\mathbb{P} = \int_\Omega \int_0^{+\infty} [X > t] \mathrm{d}t \ \mathrm{d}\mathbb{P} = \int_0^{+\infty} \int_\Omega [X > t] \mathrm{d}\mathbb{P} \ \mathrm{d}t$$

that is,

$$\mathbb{E}[X] = \int_0^{+\infty} \mathbb{P}(X \geq t) \ \mathrm{d}t.$$

1.6 Basic Topological Concepts

In this section, we explained some important concepts in topology that are needed for the course. We will only need a basic understanding of these concepts but we refer the interested reader to (Munkres, 2000) for more detailed explanations.

Let S be a subset of a topological space X, i.e. a set whose collection of open subsets satisfies certain conditions. We recall that topological spaces are abstractions of other spaces such as metric spaces, see illustration in Figure 1.5.

Open set A set is open if it does *not* include any of its boundary points. Formally, the concept of open sets can be formalized with various degrees of generality. For example, in the case of metric spaces, we have the following definition.

Definition 10 (Open set)**.** *A subset U of a metric space (M, d) is called open if, given any point $x \in U$, there exists a real number $\epsilon > 0$ such that, given any point $y \in M$ with $d(x, y) < \epsilon$, y also belongs to U. Equivalently, U is open if every point in U has a neighborhood contained in U.*

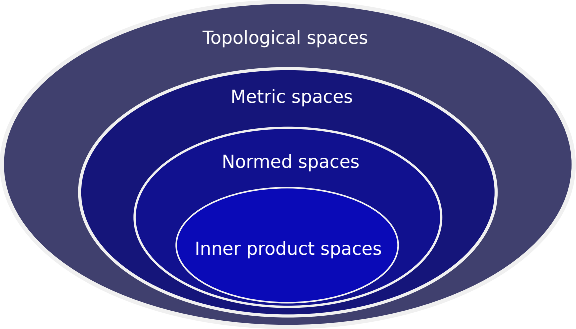


Figure 1.5: Source: Wikimedia Commons

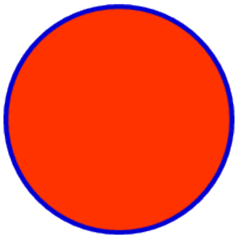


Figure 1.6: Example: The blue circle represents the set of points (x, y) satisfying $x^2 + y^2 = r^2$. The red disk represents the set of points (x, y) satisfying $x^2 + y^2 < r^2$. The red set is an open set, the blue set is its boundary set, and the union of the red and blue sets is a closed set.

Closed sets, Limit points and closure We start with the definition of a closed set.

Definition 11 (Closed set)**.** *A set is closed if its complement is open.*

Note that closed balls are closed sets (proof: show by contradiction that the complement of a closed ball is an open set, i.e. show it violates the triangle inequality).

Definition 12 (Limit point)**.** *We say that $p \in X$ is a limit point of S if every open neighborhood of p contains one point in S (other than p).*

One can give a characterization of a closed set using limit points, see the next proposition.

Proposition 8. *A closed set is a set that contains all of its limit points.*

Proof. Proof by contradiction: Assume a closed set S does not contain all its limit points, i.e. $\exists y \in S^{\complement}$ such that y is a limit point of S. Since $S^{\complement}$ is open and y is an interior point, then there exists a neighborhood of y, denoted by $\mathcal{B}(y, \epsilon)$ s.t. $\mathcal{B}(y, \epsilon) \subseteq$

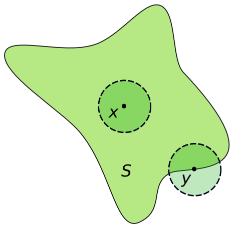


Figure 1.7: Example of limit points: Both x (interior point) and y (boundary point) are limit points of the set S since any neighborhood of these points contain a point in S.

$S^{\complement}$ but this contradicts the fact that y is a limit point of S.
For the converse, show that any $y \in S^{\complement}$ is not a limit point of S therefore proving $S^{\complement}$ is open. □

The last proposition can also be stated as follows.

Proposition 9. *A set $A \subseteq X$ is closed if and only if for every convergent sequence $(a_n)_{n \in N} \subseteq A$, we have $\lim_{n\to\infty} a_n \in A$.*

Definition 13 (Closure)**.** *The closure of a set A is the union of all its limit points. It is usually denoted by $\bar{A}$ or $cl(A) = A \cup A'$, where A' is the set of all limit points.*

Definition 14 (Dense set)**.** *A subset A of a topological space X is called dense (in X) if every point $x \in X$ either belongs to A or is a limit point of A. Alternatively, A is dense if it has* ***non-empty intersection*** *with an arbitrary non-empty open subset $B \subset X$.*

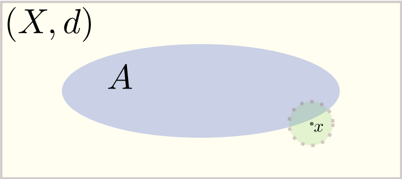


Figure 1.8: The set A shown in blue is dense in X if every $x \in X$ is a limit point of A.

A well-known example is the fact that the rationals are dense in the set of reals, which we formalize in the next theorem.

Theorem 10 (Density theorem, $\mathbb{Q}$ is dense on $\mathbb{R}$)**.**

$$\forall a < b \in \mathbb{R}, \exists x \in \mathbb{Q} \; s.t. \; x \in (a, b), \; i.e. \; a < x < b$$

Compact sets. There are typically two characterizations of compact spaces, one in terms of open sets and another one in terms of convergent sequences. We start with the definition in terms of open sets.

Definition 15 (Compact set, definition 1)**.** *A topological space X is called compact if each of its open covers* [a] *has a finite subcover.*

[a] A cover of a set X is a collection of sets whose union includes X as a subset.

Explicitly, this means that for every arbitrary collection $\{U_\alpha\}_{\alpha \in A}$ of open subsets of X such that $X = \bigcup_{\alpha \in A} U_\alpha$, there is a **finite** subset J of A such that $X = \bigcup_{i \in J} U_i$. See illustration in Figure 1.9.

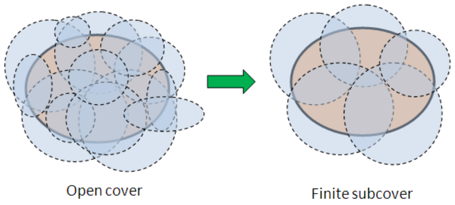


Figure 1.9: Source: `https://mathstrek.blog/`

Definition 16 (Compact set, definition 2)**.** *We call X a compact set if all sequences $(f_n)_{n \geq 1} \subset X$ have a convergent subsequence $(f_{n(k)})$ with limit point in X.*

Finally, we note that the following characterization of compact sets is often used in the literature: a subset of $\mathbb{R}^d$ is compact if it is closed and bounded (Heine-Borel theorem).

Chapter 2

Fundamentals

Notation We begin by outlining the key notational conventions that will be used throughout these lecture notes.

- Bold small letter $\mathbf{x}$ is a column vector
- Bold capital letter $\mathbf{A}$ is a matrix
- $\mathbf{x}^\top, \mathbf{A}^\top$: transpose of a vector (i.e. a row vector) and a matrix respectively
- $\mathbb{R}$: reals, $\mathbb{R}^+$: positive reals, $\mathbb{R}_0^+$: positive reals including zero
- $a := b$: a is defined by b
- $\frac{\partial f}{\partial x}$: partial derivative of a function f with respect to x
- $\frac{df}{dx}$: total derivative of a function f with respect to x
- ∇: gradient
- $\|\cdot\|$: norm (by default $\|\mathbf{x}\| = \|\mathbf{x}\|_2$).

Why should I care about optimization? Optimization plays a crucial role in many fields of science, including for instance in engineering where one often has to adjust the parameters of a system to obtain the best possible performance. Often, this can be framed by defining an objective function $f(\mathbf{x})$ and optimizing its parameters $\mathbf{x} \in \mathbb{R}^d$. Mathematically, we formalize this problem as:

$$\min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x}). \tag{2.1}$$

We will denote the optimum solution (assuming for now it is unique) as $\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x})$.

In some cases, the problem requires the parameters to be constrained. This is typically achieved by incorporating constraint functions $c_i(\mathbf{x})$ that define certain equations that the solution $\mathbf{x}$ must satisfy. Formally, this optimization problem can be defined as

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) \\ &\text{s.t. } c_i(\mathbf{x}) = 0, i \in \mathcal{E} \\ &\qquad c_i(\mathbf{x}) \geq 0, i \in \mathcal{I}, \end{aligned} \tag{2.2}$$

where the sets $\mathcal{E}, \mathcal{I}$ define equality and inequality constraints. W.l.o.g. we will often assume we want to minimize the function f (take $f \to -f$ for maximization problems).

Continuous vs Discrete Optimization In some problems of interest, the variables take on integer values. Consider for instance the following problem. You have a collection of n objects $\mathcal{C} = \{i\}_{i=1}^n$, each with a given value $v_i \in \mathbb{R}$. One possible goal is to select the maximum number of objects so that their combined value is less than a given threshold.

In discrete problems, the feasible set of solutions is a countable set. In contrast, *continuous problems*, are defined by a feasible set that is usually uncountable, which is for instance the case when $\mathbf{x} \in \mathbb{R}^d$.

Global vs Local Solution One often differentiates between two types of solution:

- Global solution: the corresponding function value is lower than all other values,
- Local solution: the corresponding function value is lower than all other values in a given neighborhood.

Finding the global solution to an optimization problem is very difficult in general. This is only feasible for specific problems where the class of functions f has some special properties (such as convexity, which we will discuss in detail in this course).

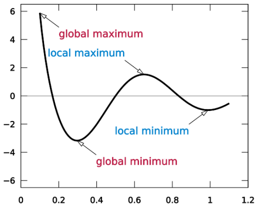


Figure 2.1: Local vs global solutions. Source: Wikipedia.

Optimization algorithms In this class, we will almost exclusively consider iterative methods which begin from an initial solution $\mathbf{x}_0 \in \mathbb{R}^d$, and iteratively improve this solution. A typical example is gradient descent, which minimizes the function f (assuming it is differentiable) by taking the following steps:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \nabla f(\mathbf{x}_k), \tag{2.3}$$

where $\nabla f(\mathbf{x})$ is the gradient of the function f, and $\eta > 0$ is a step-size (aka learning rate) parameter.

Some examples of questions we will study in this class are:

- What's the speed of convergence of various optimization algorithms to a solution $\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x})$?
- How optimal are various optimization algorithms?
- What's the difference between convex and non-convex optimization?
- etc.

The remaining of this chapter Next, we will introduce some important properties that will be required to establish formal guarantees of convergence for optimization methods. This will, for instance, include Lipschitz functions that provide a way to measure the rate at which a function changes over its domain, or convex functions that have many desirable properties, such as being easy to optimize and having a unique global minimum. Finally, we will formally introduce the concept of optimality, as well as discuss necessary and sufficient conditions to find an optimal solution.

2.1 Lipschitz property

Definition 17. *The function $f : \mathbb{R}^d \to \mathbb{R}$ is L-Lipschitz continuous if:*

$$|f(\mathbf{x}) - f(\mathbf{y})| \leq L \|\mathbf{x} - \mathbf{y}\| \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{2.4}$$

Intuitively, a Lipschitz continuous function is limited in how fast it can change.

Lemma 11. *A function $f(\mathbf{x})$ is L-Lipschitz continuous if its gradient norm is bounded by L.*

Proof. By the mean value theorem, we have :

$$|f(\mathbf{x}) - f(\mathbf{y})| \leq \sup_{\mathbf{c}\in\mathbb{R}^d} \|\nabla f(\mathbf{c})\| \|\mathbf{x} - \mathbf{y}\| \leq L\|\mathbf{x} - \mathbf{y}\|.$$

□

2.2 Smoothness

We say that a differentiable function is smooth if its gradient is Lipschitz continuous. In Figure 2.2, we provide a visual example of such a function (as well as a counter-example), followed by a formal definition.

Definition 18. *The function $f : \mathbb{R}^d \to \mathbb{R}$ is smooth if it is differentiable with L-Lipschitz-continuous gradient, i.e.*

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\| \, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{2.5}$$

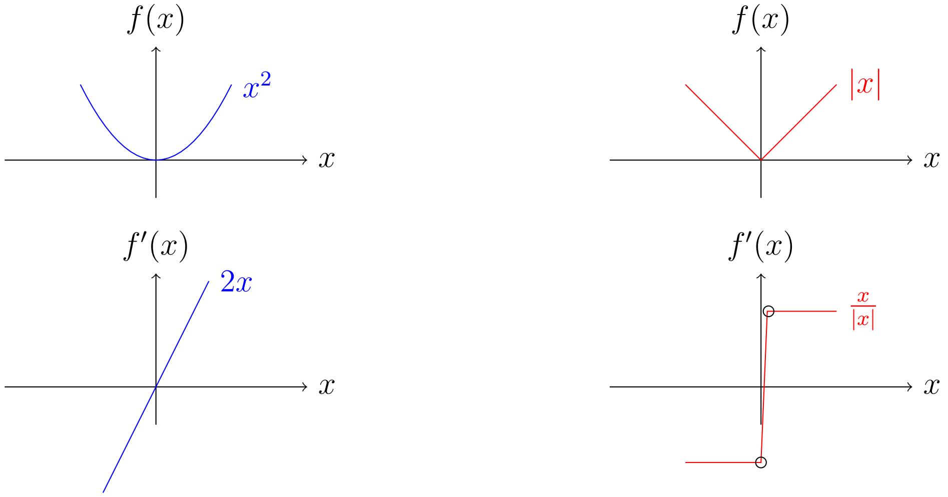


Figure 2.2: Illustration of smooth and non-smooth functions. The top figure shows the functions $f(x) = x^2$ (left) and $f(x) = |x|$ (right) with their respective derivatives below. The function $f(x) = x^2$ is smooth as its derivative is continuous while the function $f(x) = |x|$ is not smooth as its derivative has a discontinuity at $x = 0$.

Proposition 12. *If f is twice differentiable, then there is an equivalent definition of Lipschitz continuity of the gradient that gives us a bound on the eigenvalues of $\nabla^2 f(\mathbf{x})$:*

$$-L\mathbf{I} \preccurlyeq \nabla^2 f(\mathbf{x}) \preccurlyeq L\mathbf{I}. \tag{2.6}$$

Proof. The proof is based on the mean value theorem and is left as an exercise for the reader. □

Example of smooth function Take a quadratic function $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{H}\mathbf{x} \in \mathbb{R}^d$ where $\mathbf{H} \in \mathbb{R}^{d \times d}$. Then $\nabla f(\mathbf{x}) = \mathbf{H}\mathbf{x}$ and therefore

$$\begin{aligned} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| &= \|\mathbf{H}(\mathbf{x} - \mathbf{y})\| \\ &\leq \|\mathbf{H}\| \cdot \|\mathbf{x} - \mathbf{y}\|. \end{aligned} \tag{2.7}$$

We now present some lemmas for functions with Lipschitz-continuous gradients (see Nesterov (2003)) and Lipschitz-continuous Hessian.

Lemma 13. *If f has an L-Lipschitz-continuous gradient, then*

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

Proof. By the mean value theorem,

$$
\begin{aligned}
f(\mathbf{y}) = f(\mathbf{x}) &+ \int_0^1 \langle \nabla f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x})), \mathbf{y} - \mathbf{x} \rangle d\tau \\
&= f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \int_0^1 \langle \nabla f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle d\tau \qquad (2.9)
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
|f(\mathbf{y}) - f(\mathbf{x}) - \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle| &= \left| \int_0^1 \langle \nabla f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle d\tau \right| \\
&\leq \int_0^1 |\langle \nabla f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle| d\tau \\
&\overset{(i)}{\leq} \int_0^1 \|\nabla f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x})\| \|\mathbf{y} - \mathbf{x}\| d\tau \\
&\leq \int_0^1 \tau L \|\mathbf{y} - \mathbf{x}\|^2 d\tau = \frac{L}{2} \|\mathbf{y} - \mathbf{x}\|^2, \qquad (2.10)
\end{aligned}
$$

where we used the Cauchy–Schwarz inequality in (i).

□

2.3 Convexity

In this section, we will introduce the concept of convex sets and convex functions.

Definition 19 (Convex set)**.** *A set Q is* convex *if the line segment between any two points of Q lies in Q, i.e., if for any $\mathbf{x}, \mathbf{y} \in Q$ and any θ with $0 \leq \theta \leq 1$, we have*

$$\theta \mathbf{x} + (1 - \theta) \mathbf{y} \in Q.$$

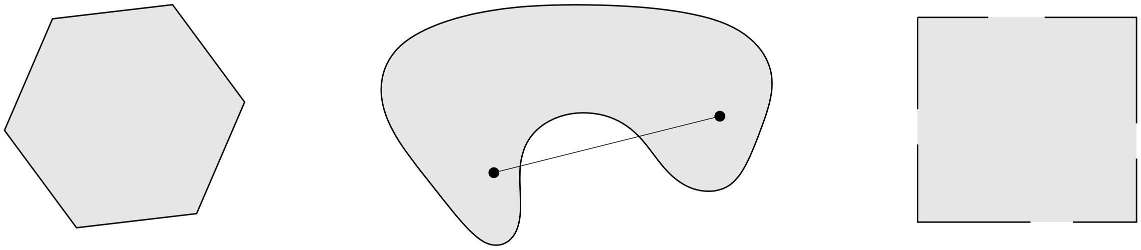


Figure 2.3: [Left] Convex, [Middle] Not convex, since line segment not in set, [Right] Not convex, since some, but not all boundary points are contained in the set. Source: Figure 2.2 from S. Boyd, L. Vandenberghe

Properties of convex sets

- Intersections of convex sets are convex
- Projections onto convex sets are *unique* (and often efficient to compute)

  recall $P_Q(\mathbf{x}') := \mathrm{argmin}_{\mathbf{y}\in Q} \|\mathbf{y}-\mathbf{x}'\|$.

Now that we have defined convex sets, we are ready to define convex functions.

Convex functions We start with the definition of a convex function.

Definition 20. *A function* $f : \mathbb{R}^d \to \mathbb{R}$ *is* convex *if* $\operatorname{dom} f$ *is a convex set and if for all* $\mathbf{x}, \mathbf{y} \in \operatorname{dom} f$*, and* θ *with* $0 \le \theta \le 1$*, we have*

$$f(\theta\mathbf{x} + (1-\theta)\mathbf{y}) \le \theta f(\mathbf{x}) + (1-\theta) f(\mathbf{y}). \tag{2.11}$$

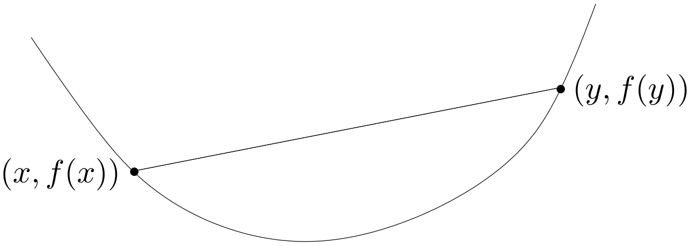


Figure 2.4: *Geometrically*: The line segment between $(\mathbf{x}, f(\mathbf{x}))$ and $(\mathbf{y}, f(\mathbf{y}))$ lies above the graph of f. Source: Figure 3.1 from S. Boyd, L. Vandenberghe

The *graph* of a function $f : \mathbb{R}^d \to \mathbb{R}$ is defined as

$$\{(\mathbf{x}, f(\mathbf{x})) \,|\, \mathbf{x} \in \operatorname{dom} f\},$$

The *epigraph* of a function $f : \mathbb{R}^d \to \mathbb{R}$ is defined as

$$\{(\mathbf{x}, t) \,|\, \mathbf{x} \in \operatorname{dom} f, f(\mathbf{x}) \le t\}.$$

It is illustrated in Figure 2.5.

A function is convex *iff* its epigraph is a convex set.

An equivalent definition that is often used in convergence proofs is given in the following theorem:

Theorem 14. *If the domain* S *is open and* $f : S \to \mathbb{R}$ *is differentiable on* $S \subseteq \mathbb{R}^d$*, then* f *is convex if and only if:*

$$f(\mathbf{y}) \ge f(\mathbf{x}) + (\mathbf{y}-\mathbf{x})^\top \nabla f(\mathbf{x}) \;\; \forall \mathbf{x}, \mathbf{y} \in S. \tag{2.12}$$

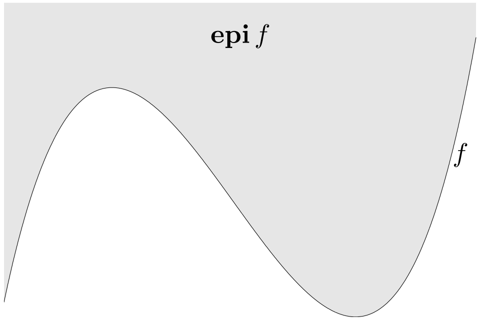


Figure 2.5: Source: Figure 3.5 from S. Boyd, L. Vandenberghe

Proof. Using the definition of the gradient (as a directional derivative),

$$\begin{aligned}(\mathbf{y}-\mathbf{x})^\top \nabla f(\mathbf{x}) &= \lim_{h\to 0^+} \frac{1}{h}[f(\mathbf{x}+h(\mathbf{y}-\mathbf{x}))-f(\mathbf{x})] \\ &= \lim_{h\to 0^+} \frac{1}{h}[f((1-h)\mathbf{x}+h\mathbf{y})-f(\mathbf{x})] \\ &\leq \lim_{h\to 0^+} \frac{1}{h}[(1-h)f(\mathbf{x})+hf(\mathbf{y})-f(\mathbf{x})] \\ &\leq f(\mathbf{y})-f(\mathbf{x}) \qquad (2.13)\end{aligned}$$

□

By the last theorem, if f is convex, then

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

and

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

So by adding the last two inequalities, we obtain the gradient monotonicity condition:

$$(\mathbf{y}-\mathbf{x})^\top(\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})) \leq 0. \qquad (2.14)$$

Properties of convex functions Some properties of convex functions include:

- Locally Lipschitz in the interior of its domain (but it can have discontinuities at boundaries)
- Differentiable almost everywhere

The definition of convexity (Eq. (2.11)) can be extended to show that if $f : \mathbb{R}^d \to \mathbb{R}$ is a convex function, and $\lambda_i > 0$ with $\sum_{i=1}^d \lambda_i = 1$, then

$$f\left(\sum_{i=1}^d \lambda_i \mathbf{x}_i\right) \leq \sum_{i=1}^d \lambda_i f(\mathbf{x}_i). \qquad (2.15)$$

This inequality is known as Jensen's inequality.

Examples of convex functions

1. The simplest example of a convex function is an affine function (i.e. a sum of a linear form plus a constant $b \in \mathbb{R}$):

   $$f(\mathbf{x}) = \mathbf{a}^\top \mathbf{x} + b, \quad \mathbf{a}, \mathbf{x} \in \mathbb{R}^d.$$

   Note that this function satisfies Eq. 2.12 with equality. It is also concave (negative of a convex function) and one can easily show that functions that are both convex and concave on the entire space are affine functions.

2. Any norm. Indeed, by the triangle inequality $f(\mathbf{x} + \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y})$ and homogeneity of a norm $f(a\mathbf{x}) = |a| f(\mathbf{x})$, a scalar:

   $$f(\theta \mathbf{x} + (1-\theta)\mathbf{y}) \leq f(\theta \mathbf{x}) + f((1-\theta)\mathbf{y}) = \theta f(\mathbf{x}) + (1-\theta) f(\mathbf{y}), \quad 0 \leq \theta \leq 1.$$

   We used the triangle inequality for the inequality and homogeneity for the equality.

3. Exponential function $\exp(x)$.

4. A sum of convex functions is convex.

Convex and twice differentiable Next, we will introduce an equivalent definition of convexity that applies to twice differentiable functions and that is often useful to check whether a function is convex. In the following, we will use the notation $\mathbf{A} \succcurlyeq 0$ to denote that $\mathbf{A}$ is positive semi-definite (PSD). Recall that a matrix $\mathbf{A}$ is PSD if $\mathbf{u}^\top \mathbf{A} \mathbf{u} \geq 0$ for all $\mathbf{u} \in \mathbb{R}^d$, or equivalently all the eigenvalues λ_i of $\mathbf{A}$ are non-negative, i.e. $\lambda_i(\mathbf{A}) \geq 0 \ \forall \, i = 1, \ldots d$. We will also denote by $\nabla^2 f(\mathbf{x})$ the Hessian matrix of the function $f : \mathbb{R}^d \to \mathbb{R}$, which is a square matrix that contains all partial derivatives, i.e.

$$\nabla^2 f(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_d} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_d} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_d \partial x_1} & \frac{\partial^2 f}{\partial x_d \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_d^2}. \end{bmatrix} \tag{2.16}$$

Theorem 15. *The function $f \in C^2(\mathbb{R}^d)$ is convex if and only if the Hessian $\nabla^2 f(\mathbf{x}) \succcurlyeq 0$ for all $\mathbf{x} \in \mathbb{R}^d$.*

The proof of this theorem relies on the characterization of an optimal solution that will be presented in Section 2.4. We therefore invite the reader to skip this proof for now and come back to it after reading Section 2.4.

Proof. $\Longrightarrow$ Let's first assume that f is convex and fix $\mathbf{x} \in \mathbb{R}^d$. Let $g(\mathbf{y}) := f(\mathbf{y}) - \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x})$, which is convex. We have

$$\nabla g(\mathbf{y}) = \nabla f(\mathbf{y}) - \nabla f(\mathbf{x}),$$

and

$$\nabla^2 g(\mathbf{y}) = \nabla^2 f(\mathbf{y}).$$

Note that $\nabla g(\mathbf{x}) = 0$, therefore $\mathbf{x}$ is a global minimizer of g, which in turns implies that $\nabla^2 g(\mathbf{x}) \succcurlyeq 0$. Since $\nabla^2 g(\mathbf{x}) = \nabla^2 f(\mathbf{x})$ and the choice of $\mathbf{x}$ was arbitrary, the desired result holds.

$\Longleftarrow$ We next assume that $\nabla^2 f(\mathbf{x}) \succcurlyeq 0$ for all $\mathbf{x} \in \mathbb{R}^d$. For all $\mathbf{y} \in \mathbb{R}^d$, by the mean-value theorem,

$$f(\mathbf{y}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x}) + \frac{1}{2}(\mathbf{y} - \mathbf{x})^\top \nabla^2 f(\mathbf{x} + \alpha(\mathbf{y} - \mathbf{x}))(\mathbf{y} - \mathbf{x}),$$

for some $0 \leq \alpha \leq 1$. Since $\nabla^2 f$ is positive semi-definite, we have $\frac{1}{2}(\mathbf{y} - \mathbf{x})^\top \nabla^2 f(\mathbf{x} + \alpha(\mathbf{y} - \mathbf{x}))(\mathbf{y} - \mathbf{x}) \geq 0$, therefore

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

which proves that f is convex.

□

Summary: how do you prove a function is convex?

- If f is twice differentiable, then check whether $\nabla^2 f(\mathbf{x}) \succcurlyeq 0$
- You can verify that the equation in the definition of convexity holds
- You can use the fact that certain operations preserve convexity:
  - If $f(\mathbf{x})$ is convex, then $g(\mathbf{x}) = cf(\mathbf{x})$ is convex if $c > 0$
  - If $f(\mathbf{x})$ is convex, then $g(\mathbf{x}) = f(a\mathbf{x}+\mathbf{b})$ is convex given arbitrary constants $a, b \in \mathbb{R}$
  - Other operations such as taking the sum of the maximum of convex functions.

2.3.1 Strong convexity

We start with a definition.

Definition 21. *A (differentiable) function* $f : \mathbb{R}^d \to \mathbb{R}$ *is* μ-strongly-convex *if* $\operatorname{dom} f$ *is a convex set and if for all* $\mathbf{x}, \mathbf{y} \in \operatorname{dom} f$*, and* θ *with* $0 \leq \theta \leq 1$*, we have*

$$f(\theta\mathbf{x} + (1-\theta)\mathbf{y}) \leq \theta f(\mathbf{x}) + (1-\theta)f(\mathbf{y}) - \frac{\mu}{2}\theta(1-\theta)\|\mathbf{x}-\mathbf{y}\|_2^2.$$

Note that strong convexity does not necessarily require the function to be differentiable, as the gradient can be replaced by a sub-gradient when the function is non-smooth (we will discuss sub-gradients later in the class, you can think of them as a generalization of gradients for non-differentiable functions).

Equivalent definitions The following are equivalent definitions for strong convexity:

a)

$$f(\mathbf{y}) \geq f(\mathbf{x}) + (\mathbf{y}-\mathbf{x})^\top \nabla f(\mathbf{x}) + \frac{\mu}{2}\|\mathbf{y}-\mathbf{x}\|^2 \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{2.17}$$

We illustrate Condition (2.17) in Figure 2.6.

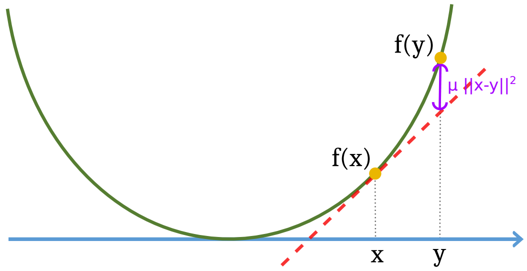


Figure 2.6: Strong convexity. The dotted red line is a linear function that lower bounds the function f (which is what's needed for convexity). Adding the term $\|\mathbf{x}-\mathbf{y}\|^2$ makes the function strongly convex.

b) $g(\mathbf{x}) = f(\mathbf{x}) - \mu\|\mathbf{x}\|_2^2$ is convex, $\forall \mathbf{x} \in \mathbb{R}^d$.

c) Strong monotonicity: $\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle \geq \mu\|\mathbf{x}-\mathbf{y}\|^2$.

Proof. *a)* $\equiv$ *b)*: this follows from the fact that, by Theorem 14, $g(\mathbf{x})$ is convex if and only if $g(\mathbf{y}) \geq g(\mathbf{x}) + \nabla g(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}), \ \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.

b) $\equiv$ *c)*: this follows from the monotone gradient condition (Eq. (2.14)), i.e. $g(\mathbf{x})$ is convex if and only if $(\nabla g(\mathbf{x}) - \nabla g(\mathbf{y}))^\top(\mathbf{x}-\mathbf{y}) \geq 0 \ \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.

□

Another less-known implication of strong convexity is (try to prove it):

$$f(\mathbf{y}) \leq f(\mathbf{x}) + (\mathbf{y}-\mathbf{x})^\top \nabla f(\mathbf{x}) + \frac{1}{2\mu}\|\nabla f(\mathbf{y}) - \nabla f(\mathbf{x})\|^2 \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{2.18}$$

Properties at $\mathbf{x}^*$ Further properties can be shown to hold for $\mathbf{y} = \mathbf{x}^*$.

For instance, Eq (2.17) implies that $\forall \mathbf{x} \in \mathbb{R}^d$:

$$\|\mathbf{x} - \mathbf{x}^*\|^2 \leq \frac{2}{\mu} \left(f(\mathbf{x}) - f(\mathbf{x}^*) \right), \tag{2.19}$$

or,

$$\begin{aligned} -(\mathbf{x}^* - \mathbf{x})^\top \nabla f(\mathbf{x}) &\geq f(\mathbf{x}) - f(\mathbf{x}^*) + \frac{\mu}{2} \|\mathbf{x}^* - \mathbf{x}\|^2 \\ &\geq \mu \|\mathbf{x}^* - \mathbf{x}\|^2 . \end{aligned} \tag{2.20}$$

We can also derive a lower bound on the norm of $\nabla f(\mathbf{x})$ by minimizing both sides of Eq. 2.17 with respect to $\mathbf{y}$. The minimizer of the LHS is $\mathbf{y} = \mathbf{x}^*$ while the one for the RHS is $\mathbf{y} = \mathbf{x} - \frac{1}{\mu} \nabla f(\mathbf{x})$. Thus

$$f(\mathbf{x}^*) \geq f(\mathbf{x}) - \frac{1}{2\mu} \|\nabla f(\mathbf{x})\|^2 \quad \forall \mathbf{x} \in \mathbb{R}^d, \tag{2.21}$$

which implies

$$\|\nabla f(\mathbf{x})\|^2 \geq 2\mu (f(\mathbf{x}) - f(\mathbf{x}^*)) \;\; \forall \mathbf{x} \in \mathbb{R}^d. \tag{2.22}$$

Hessian The following bound on the Hessian is implied by strong convexity:

$$\mu \mathbf{I} \preceq \nabla^2 f(\mathbf{x}). \tag{2.23}$$

Lipschitz continuity and strong convexity In general, strongly-convex functions can not be Lipschitz continuous over their whole domain. To see why this is the case, note that

$$\begin{aligned} f(\mathbf{x}) - f(\mathbf{y}) &\geq \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) + \frac{\mu}{2} \|\mathbf{x} - \mathbf{y}\|_2^2 \\ &\geq \left(\frac{\mu}{2} \|\mathbf{x} - \mathbf{y}\|_2 - \|\nabla f(\mathbf{y})\|_2 \right) \|\mathbf{x} - \mathbf{y}\|_2, \end{aligned} \tag{2.24}$$

which follows from the Cauchy-Schwarz inequality

$$-\|\mathbf{u}\|_2 \|\mathbf{v}\|_2 \leq \mathbf{u}^\top \mathbf{v} \leq \|\mathbf{u}\|_2 \|\mathbf{v}\|_2.$$

Therefore, for $\mathbf{x} \neq \mathbf{y}$, we have

$$\frac{f(\mathbf{x}) - f(\mathbf{y})}{\|\mathbf{x} - \mathbf{y}\|_2} \geq -\|\nabla f(\mathbf{y})\|_2 + \frac{\mu}{2} \|\mathbf{x} - \mathbf{y}\|_2.$$

The right-hand side is unbounded as $\|\mathbf{x} - \mathbf{y}\|_2 \to +\infty$, so no fixed value of the Lipschitz constant L can be found. One therefore needs to restrict the domain of the function f to ensure that L is finite.

2.3.2 Polyak-Lojasiewicz inequality

In 1963, Boris Polyak proposed a simple condition that is sufficient to show a global linear convergence rate for gradient descent (this will be the subject of another chapter). This condition is a special case of the Lojasiewicz inequality proposed in the same year, and it does not require strong convexity (or even convexity).

Definition 22. *We will say that a function satisfies the PL inequality if the following holds for some $\mu > 0$ and all $\mathbf{x} \in \mathbb{R}^d$,*

$$\frac{1}{2}\|\nabla f(\mathbf{x})\|^2 \geq \mu|f(\mathbf{x}) - f(\mathbf{x}^*)|. \tag{2.25}$$

Relation to other inequalities Karimi et al. (2016) showed that the Polyak-Lojasiewicz (PL) inequality is actually weaker than the main conditions that have been explored to show linear convergence rates without strong convexity over the last 25 years. We restate the different conditions below. All of these definitions involve some constant $\mu > 0$ (which may not be the same across conditions), and we will use the convention that $\mathbf{x}_p$ is the projection of $\mathbf{x}$ onto the solution set $\mathcal{X}^*$.

1. **Strong Convexity** (SC): For all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ we have
$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 .$$

2. **Essential Strong Convexity** (ESC): For all $\mathbf{x}$ and $\mathbf{y}$ such that $\mathbf{x}_p = \mathbf{y}_p$ we have
$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 .$$

3. **Weak Strong Convexity** (WSC): For all $\mathbf{x}$ we have
$$f^* \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{x}_p - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{x}_p - \mathbf{x}\|^2 .$$

4. **Restricted Secant Inequality (aka one-point strong convexity)** (RSI): For all $\mathbf{x}$ we have
$$\langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{x}_p \rangle \geq \mu \|\mathbf{x}_p - \mathbf{x}\|^2 .$$
If the function f is also convex it is called **restricted strong convexity** (RSC).

5. **Error Bound** (EB): For all $\mathbf{x}$ we have
$$\|\nabla f(\mathbf{x})\| \geq \mu \|\mathbf{x}_p - \mathbf{x}\| .$$

6. **Polyak-Łojasiewicz** (PL): For all $\mathbf{x}$ we have
$$\frac{1}{2} \|\nabla f(\mathbf{x})\|^2 \geq \mu(f(\mathbf{x}) - f^*).$$

7. **Quadratic Growth** (QG): For all $\mathbf{x}$ we have

$$f(\mathbf{x}) - f^* \geq \frac{\mu}{2} \left\| \mathbf{x}_p - \mathbf{x} \right\|^2 .$$

If the function f is also convex it is called **optimal strong convexity** (OSC) or **semi-strong convexity** or sometimes WSC (but we'll reserve the expression WSC for the definition above).

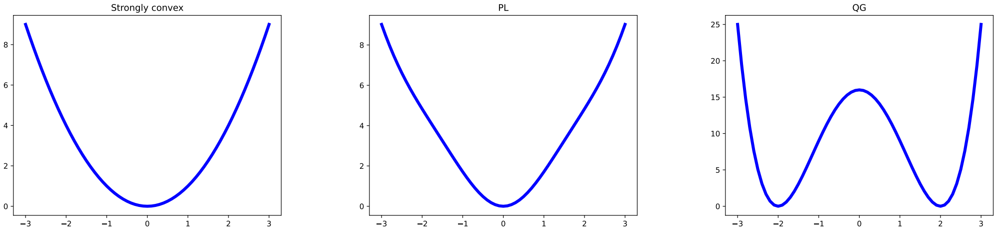


Figure 2.7: Example of strongly-convex (left), PL (middle) and QG functions (right).

The result proved in Karimi et al. (2016) is the theorem restated below. The notation $x \to y$ means that if a function f belongs to class x, then it also belongs to class y.

Theorem 16. *For a function f with a Lipschitz-continuous gradient, the following implications hold:*

$$(SC) \to (ESC) \to (WSC) \to (RSI) \to (EB) \equiv (PL) \to (QG).$$

If we further assume that f is convex then we have

$$(RSI) \equiv (EB) \equiv (PL) \equiv (QG).$$

Proof. We here prove that strong convexity implies PL and leave the other relations as exercises. Taking $\mathbf{y} = \mathbf{x}^*$ in Eq. (2.17), we get (for all $\mathbf{x} \in \mathbb{R}^d$),

$$f(\mathbf{x}^*) - f(\mathbf{x}) \geq (\mathbf{x}^* - \mathbf{x})^\top \nabla f(\mathbf{x}) + \frac{\mu}{2} \left\| \mathbf{x}^* - \mathbf{x} \right\|^2 ,$$

which implies

$$\begin{aligned} f(\mathbf{x}) - f(\mathbf{x}^*) &\leq (\mathbf{x} - \mathbf{x}^*)^\top \nabla f(\mathbf{x}) - \frac{\mu}{2} \left\| \mathbf{x}^* - \mathbf{x} \right\|^2 \\ &= -\frac{1}{2} \left\| \sqrt{\mu}(\mathbf{x} - \mathbf{x}^*) - \frac{1}{\sqrt{\mu}} \nabla f(\mathbf{x}) \right\|^2 + \frac{1}{2\mu} \|\nabla f(\mathbf{x})\|^2 \\ &\leq \frac{1}{2\mu} \|\nabla f(\mathbf{x})\|^2. \end{aligned}$$

□

2.4 Optimality

In the following, we will characterize the optimality of a solution using the concept of a necessary condition (a condition that must be present for a particular outcome) and of a sufficient condition (a condition that guarantees a particular outcome). But first, we start with the different definitions of minimizers.

2.4.1 Definition of a minimum

One typically distinguishes between global and local minimum as follows.

Definition 23 (Global minimum)**.** *A point* $\mathbf{x} \in \mathcal{X}$ *is a global minimum of a function* $f : \mathcal{X} \to \mathbb{R}$ *if* $\forall \mathbf{y} \in \mathcal{X}, f(\mathbf{x}) \leq f(\mathbf{y})$.

Definition 24 (Local minimum)**.** *A point* $\mathbf{x} \in \mathcal{X}$ *is a local minimum of a function* $f : \mathcal{X} \to \mathbb{R}$ *if* $\exists \delta > 0, \forall \mathbf{y} \in \mathcal{X}$ *such that* $\|\mathbf{y} - \mathbf{x}\| \leq \delta$, *then* $f(\mathbf{y}) \geq f(\mathbf{x})$.

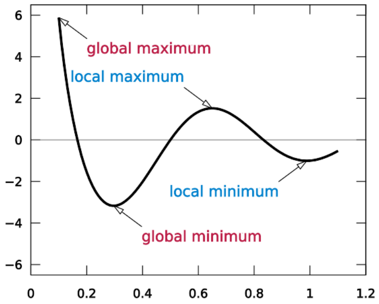


Figure 2.8: Local vs global solutions. Source: Wikipedia.

Of course, a global minimum is also a local minimum. For convex functions, the two notions are exactly the same.

Theorem 17. *If* $f : \mathbb{R}^d \to \mathbb{R}$ *is convex, then* $\mathbf{x}^*$ *is a local minimum of* $f(\mathbf{x})$ *if and only if* $\mathbf{x}^*$ *is a global minimum of* $f(\mathbf{x})$.

Proof. Suppose $\mathbf{x}^*$ is not a global minimum and show it then can not be a local minimum either. Indeed, we have $\exists\ \hat{\mathbf{x}} \in \mathbb{R}^d : f(\hat{\mathbf{x}}) < f(\mathbf{x}^*)$. For each $\lambda \in (0, 1)$, let $\mathbf{x}(\lambda) = (1 - \lambda)\mathbf{x}^* + \lambda\hat{\mathbf{x}}$, then using convexity:

$$\begin{aligned} f(\mathbf{x}(\lambda)) &\leq (1 - \lambda)f(\mathbf{x}^*) + \lambda f(\hat{\mathbf{x}}) \qquad (2.26) \\ &= f(\mathbf{x}^*) + \lambda(f(\hat{\mathbf{x}}) - f(\mathbf{x}^*)) \\ &< f(\mathbf{x}^*). \end{aligned}$$

Therefore every neighborhood of $\mathbf{x}^*$ contains points $\mathbf{x}(\lambda)$ such that $f(\mathbf{x}(\lambda)) < f(\mathbf{x}^*)$.

□

Descent direction Assume $f \in C^1(\mathbb{R}^d), \mathbf{x} \in \mathbb{R}^d$ and $\mathbf{s} \in \mathbb{R}^d$ such that $\mathbf{s} \neq \mathbf{0}$. Then $\mathbf{s}$ is a descent direction if

$$\nabla f(\mathbf{x})^\top \mathbf{s} < 0 \implies f(\mathbf{x} + \alpha \mathbf{s}) < f(\mathbf{x}) \quad \forall \alpha \text{ sufficiently small.} \tag{2.27}$$

The condition $\nabla f(\mathbf{x})^\top \mathbf{s} < 0$ also implies:

$$\cos(-\nabla f(\mathbf{x}), \mathbf{s}) = \frac{(-\nabla f(\mathbf{x}))^\top \mathbf{s}}{\|\nabla f(\mathbf{x})\| \cdot \|\mathbf{s}\|} = \frac{|\nabla f(\mathbf{x}))^\top \mathbf{s}|}{\|\nabla f(\mathbf{x})\| \cdot \|\mathbf{s}\|} > 0. \tag{2.28}$$

One can easily verify that $\mathbf{s} = -\nabla f(\mathbf{x})$ is a descent direction and so is any $\mathbf{s}$ that does not deviate too much from $-\nabla f(\mathbf{x})$, i.e. $\langle -\nabla f(\mathbf{x}), \mathbf{s} \rangle \in [0, \pi/2)$.

Existence of a global minimum One sufficient condition for a minimum to exist is that the function f is continuous on a closed and bounded domain, as stated by the extreme-value theorem.

Theorem 18 (Extreme Value Theorem)**.** *If $f : [a, b] \to \mathbb{R}$ is continuous, then f has a maximum and a minimum.*

Proof. We skip the proof and refer the reader to a standard analysis textbook. □

Next, we will see alternative conditions to guarantee the existence of a local minimum.

2.4.2 Necessary and sufficient conditions

First-order necessary condition (FONC) We start with the concept of a necessary condition that is used to determine the points where a function may achieve its local maximum or minimum, or more generally, where a function might have a critical point. The FONC is called ”necessary” because, for a point to be an extremum (maximum or minimum), this condition must hold. However, it is not sufficient by itself to guarantee an extremum.

Theorem 19. *Assume $f \in C^1(\mathbb{R}^d)$. If $\mathbf{x}^* \in \mathbb{R}^d$ is a local minimizer of f, then* $\nabla f(\mathbf{x}^*) = 0$.

Proof. We proceed by contradiction, i.e. we suppose that $\nabla f(\mathbf{x}^*) \neq 0$. Define $\mathbf{p} = -\nabla f(\mathbf{x}^*)$, then $\mathbf{p}^\top \nabla f(\mathbf{x}^*) = -\|\nabla f(\mathbf{x}^*)\|^2 < 0$. Since $\nabla f(\cdot)$ is continuous, there exists $T > 0$ such that for $t \in [0, T]$

$$\mathbf{p}^\top \nabla f(\mathbf{x}^* + t\mathbf{p}) < 0.$$

By the mean value theorem, for any $\bar{t} \in (0, T]$, $\exists t \in (0, \bar{t})$ such that

$$f(\mathbf{x}^* + \bar{t}\mathbf{p}) = f(\mathbf{x}^*) + \bar{t}\mathbf{p}^\top \nabla f(\mathbf{x}^* + t\mathbf{p}) < f(\mathbf{x}^*),$$

which contradicts the fact that $\mathbf{x}^*$ is a local minimizer.

□

For non-differentiable convex functions, $\mathbf{x}^*$ is an optimal solution for $\min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x})$ if $\mathbf{0} \in \partial f(\mathbf{x}^*)$ where $\partial f(\mathbf{x}^*)$ is the sub-differential (a generalization of the concept of derivatives for non-differentiable function, we will discuss this later).

Recall that $\mathbf{g} \in \partial f(\mathbf{x})$ if and only if

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \mathbf{g}, \mathbf{y} - \mathbf{x} \rangle, \quad \forall \mathbf{y}.$$

If $\mathbf{0} \in \partial f(\mathbf{x}^*)$, the above definition translates to

$$f(\mathbf{y}) \geq f(\mathbf{x}^*), \quad \forall \mathbf{y},$$

i.e. $\mathbf{x}^*$ is optimal.

Second-order necessary condition Second-order condition are often needed to determine the nature of the critical point, i.e. whether it's a maximum, minimum, or saddle point.

Theorem 20. *Assume $f \in C^2(\mathbb{R}^d)$. Then $\mathbf{x}^*$ is a local minimizer of f implies $\nabla^2 f(\mathbf{x}^*)$ is positive semidefinite, i.e. $\mathbf{s}^\top \nabla^2 f(\mathbf{x}^*) \mathbf{s} \geq 0$ for all $\mathbf{s} \in \mathbb{R}^d$.*

Proof. We have already shown in Theorem 19 that $\nabla f(\mathbf{x}^*) = 0$. We proceed similarly for the condition $\nabla^2 f(\mathbf{x}^*)$ and assume it is not positive semidefinite. Then, we can choose $\mathbf{p}$ such that $\mathbf{p}^\top \nabla^2 f(\mathbf{x}^*) \mathbf{p} < 0$ and since $\nabla^2 f$ is continuous, there exists $T > 0$ such that $\mathbf{p}^\top \nabla^2 f(\mathbf{x}^* + t\mathbf{p}) \mathbf{p} < 0$ for all $t \in [0, T]$.

By the mean value theorem, for any $\bar{t} \in (0, T]$, $\exists t \in (0, \bar{t})$ such that

$$f(\mathbf{x}^* + \bar{t}\mathbf{p}) = f(\mathbf{x}^*) + \bar{t}\mathbf{p}^\top \nabla f(\mathbf{x}^*) + \frac{1}{2}\bar{t}^2 \mathbf{p}^\top \nabla^2 f(\mathbf{x}^* + t\mathbf{p}) \mathbf{p} < f(\mathbf{x}^*),$$

which contradicts the fact that $\mathbf{x}^*$ is a local minimizer.

□

Example: $f(x) = x^3$: $x^* = 0$ is not a local minimizer but $f'(0) = f''(0) = 0$.

Second-order sufficient conditions While necessary conditions are required for a point to be an extremum, sufficient conditions go further - they guarantee that the point is indeed a local maximum or minimum.

Theorem 21. *Assume $f \in C^2(\mathbb{R}^d)$. Then $\nabla f(\mathbf{x}^*) = 0$ and $\nabla^2 f(\mathbf{x}^*)$ is positive definite (i.e. $\mathbf{s}^\top \nabla^2 f(\mathbf{x}^*)\mathbf{s} \geq 0$ for all $\mathbf{s} \in \mathbb{R}^d$) implies that $\mathbf{x}^*$ is a local minimizer of f.*

Proof. Since $\nabla^2 f$ is continuous, we can choose $r > 0$ such that $\nabla^2 f(\mathbf{x})$ is positive definite for all $\mathbf{x} \in \mathcal{D} = \{\mathbf{z} \mid \|\mathbf{z} - \mathbf{x}^*\| < r\}$. Consider $\mathbf{p} \neq 0$ such that $\|\mathbf{p}\| < r$, then we have $\mathbf{x}^* + \mathbf{p} \in \mathcal{D}$, therefore

$$\begin{aligned} f(\mathbf{x}^* + \mathbf{p}) &= f(\mathbf{x}^*) + \mathbf{p}^\top \nabla f(\mathbf{x}^*) + \frac{1}{2}\mathbf{p}^\top \nabla^2 f(\mathbf{z})\mathbf{p} \\ &= f(\mathbf{x}^*) + \frac{1}{2}\mathbf{p}^\top \nabla^2 f(\mathbf{z})\mathbf{p}, \end{aligned}$$

where $\mathbf{z} = \mathbf{x}^* + t\mathbf{p}$ for some $t \in (0,1)$. Since $\mathbf{z} \in \mathcal{D}$, $\mathbf{p}^\top \nabla^2 f(\mathbf{z})\mathbf{p} > 0$, and we therefore conclude that

$$f(\mathbf{x}^* + \mathbf{p}) > f(\mathbf{x}^*).$$

□

Note that the sufficient condition requires $\nabla^2 f(\mathbf{x}^*)$ to be positive definite, while the necessary condition implies positive semi-definiteness.

Simple example: Consider the function $f(x) = x^4$. The point $x^* = 0$ is a local minimizer but $f''(0) = 0$.

Example: least-squares Consider a set of n pairs of points, denoted by $\mathcal{D} = (\xi_i, y_i)_{i=1}^n$ where $\xi_i \in \mathbb{R}^d$ (called features or independent variables in statistics or machine learning) and $\mathbf{y}_i \in \{-1, +1\}$ are the corresponding labels. The least-squares objective is then defined as:

$$f(\mathbf{x}) = \frac{1}{2n}\sum_{i=1}^{n}(y_i - \mathbf{x}^\top \xi_i)^2 + \frac{\lambda}{2}\|\mathbf{x}\|^2, \tag{2.29}$$

where $\mathbf{x} \in \mathbb{R}^d$. We can also rewrite the equation above in a vector form as

$$f(\mathbf{x}) = \frac{1}{2n}\|\Xi\mathbf{x} - \mathbf{y}\|^2 + \frac{\lambda}{2}\|\mathbf{x}\|^2, \tag{2.30}$$

where $\mathbf{y} = [y_1 \dots y_n] \in \mathbb{R}^d$ and Ξ is matrix whose rows are the $\xi_i's$, i.e. $\Xi = [\xi_1 \dots \xi_n]^\top \in \mathbb{R}^{n \times d}$.

We can then derive a closed-form solution for $\mathbf{x}$ using the necessary and sufficient conditions as follows. First, by setting the gradient to zero, we obtain:

$$\nabla f(\mathbf{x}^*) = \Xi^\top(\Xi\mathbf{x}^* - \mathbf{y}) + \lambda\mathbf{x}^* \overset{!}{=} 0 \tag{2.31}$$

$$\implies \left(\Xi^\top\Xi + \lambda\mathbf{I}\right)\mathbf{x}^* = \Xi^\top\mathbf{y} \tag{2.32}$$

$$\implies \mathbf{x}^* = \left(\Xi^\top\Xi + \lambda\mathbf{I}\right)^{-1}\Xi^\top\mathbf{y}. \tag{2.33}$$

Then, one can check that the Hessian is $\nabla^2 f(\mathbf{x}) = \Xi^\top\Xi$ which is positive definite (since it is a covariance matrix). We conclude that $\mathbf{x}^*$ is a minimizer.

2.5 Cheat sheet

The following is taken from `https://fa.bianp.net/blog/2017/optimization-inequalities-cheatsheet` and summarizes some of the key concepts seen in this lecture.

f is L-smooth. This is the class of functions that are differentiable and its gradient is Lipschitz continuous.

- $\|\nabla f(\mathbf{y}) - \nabla f(\mathbf{x})\| \leq L\|\mathbf{x} - \mathbf{y}\|$
- $|f(\mathbf{y}) - f(\mathbf{x}) - \langle\nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle| \leq \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|^2$
- $\nabla^2 f(\mathbf{x}) \preceq L$ (assuming f is twice differentiable)

f is convex.

- $f(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1-\lambda)f(\mathbf{y})$ for all $\lambda \in [0, 1]$.
- $f(\mathbf{x}) \leq f(\mathbf{y}) + \langle\nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y}\rangle$
- $0 \leq \langle\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y}\rangle$
- $f(\mathbb{E}\mathbf{x}) \leq \mathbb{E}[f(\mathbf{x})]$ where $\mathbf{x}$ is a random variable (Jensen's inequality).

f is both L-smooth and convex.

- $\frac{1}{L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \leq \langle\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y}\rangle$
- $0 \leq f(\mathbf{y}) - f(\mathbf{x}) - \langle\nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle \leq \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2$
- $f(\mathbf{x}) \leq f(\mathbf{y}) + \langle\nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y}\rangle - \frac{1}{2L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2$

f is μ-strongly convex. Set of functions f such that $f - \frac{\mu}{2}|\cdot|^2$ is convex. It includes the set of convex functions with $\mu = 0$. Here $\mathbf{x}^*$ denotes the minimizer of f.

- $f(\mathbf{x}) \leq f(\mathbf{y}) + \langle\nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y}\rangle - \frac{\mu}{2}\|\mathbf{x} - \mathbf{y}\|^2$
- $f(\mathbf{x}) \leq f(\mathbf{y}) + \langle\nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y}\rangle + \frac{1}{2\mu}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2$
- $\mu\|\mathbf{x} - \mathbf{y}\|^2 \leq \langle\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y}\rangle$
- $\frac{\mu}{2}\|\mathbf{x} - \mathbf{x}^*\|^2 \leq f(\mathbf{x}) - f(\mathbf{x}^*)$

f is both L-smooth and μ-strongly convex.

- $\frac{\mu L}{\mu+L}\|\mathbf{x}-\mathbf{y}\|^2 + \frac{1}{\mu+L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \leq \langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y}\rangle$
- $\mu \preceq \nabla^2 f(\mathbf{x}) \preceq L$ (assuming f is twice differentiable)

2.6 Exercise: Fundamentals

Problem 1 (Smooth functions):

If f is both convex ($\nabla^2 f(\mathbf{x}) \succeq 0$) and L-smooth ($\nabla^2 f(\mathbf{x}) \preceq LI$), then for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$,

$$f(\mathbf{y}) \;\geq\; f(\mathbf{x}) \;+\; \nabla f(\mathbf{x})^\top \big(\mathbf{y} - \mathbf{x}\big) \;+\; \frac{1}{2L} \left\|\nabla f(\mathbf{y}) - \nabla f(\mathbf{x})\right\|^2 .$$

Problem 2 (Functions with Lipschitz-continuous Hessian):

If f has an L-Lipschitz-continuous Hessian, show that

1.
$$\|\nabla f(\mathbf{y}) - \nabla f(\mathbf{x}) - \nabla^2 f(\mathbf{x})(\mathbf{y} - \mathbf{x})\| \leq \frac{L}{2} \|\mathbf{y} - \mathbf{x}\|^2 \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d, \tag{2.34}$$

2.
$$|f(\mathbf{y}) - f(\mathbf{x}) - (\mathbf{y} - \mathbf{x})^\top \nabla f(\mathbf{x}) - (\mathbf{y} - \mathbf{x})^\top \nabla^2 f(\mathbf{x})(\mathbf{y} - \mathbf{x})| \leq \frac{L}{6}\|\mathbf{y} - \mathbf{x}\|^3 \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{2.35}$$

Problem 3 (Convex sets):

1. Show that every ball $B(\mathbf{a}, r) := \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x} - \mathbf{a}\| \leq r$ for $\mathbf{a} \in \mathbb{R}^d$ and $r \geq 0$ is convex.
2. Let $\mathbf{A} \in \mathbb{R}^{m \times d}$ and $C = \{\mathbf{x} \in \mathbb{R}^d : \mathbf{A}\mathbf{x} \leq 0\}$. Prove that C is a convex cone. Recall that a cone C is a convex cone if $\alpha\mathbf{x} + \beta\mathbf{y} \in C$ for any $\mathbf{x}, \mathbf{y} \in C$ and $\alpha, \beta > 0$.
3. Is the union of two convex sets convex?

Problem 4 (Convex functions):

1. Show that the exponential function $\exp(x)$ is convex.
2. Show that a sum of convex functions is convex.
3. Show that if f is μ-strongly-convex and L-smooth, then for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, we have:

$$\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \frac{\mu L}{\mu + L} \|\mathbf{x} - \mathbf{y}\|^2 + \frac{1}{\mu + L} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \tag{2.36}$$

Chapter 3

Gradient Descent

3.1 Steepest Descent

Our goal is to minimize a differentiable function $f : \mathbb{R}^d \to \mathbb{R}$. To do so, we will consider iterative algorithms that, given an iterate $\mathbf{x}_k$, decide how to move to $\mathbf{x}_{k+1}$ by considering the gradient information at $\mathbf{x}_k$ (and potentially at previous iterates $\mathbf{x}_{k-1}, \mathbf{x}_{k-2}, \ldots$). The first question we should ask is what should be our criterion to move from $\mathbf{x}_k$? A reasonable guess seems to ask for a monotonic function decrease, i.e. $f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k)$. It turns out that there are different strategies to achieve such a decrease, and we introduce some of the most important ones below.

Line search Given a direction $\mathbf{p}_k$, line search finds the amount to move along $\mathbf{p}_k$ by solving the following one-dimensional problem:

$$\min_{\alpha > 0} f(\mathbf{x}_k + \alpha \mathbf{p}_k). \tag{3.1}$$

Solving Eq. (3.1) exactly is computationally expensive in general. Instead, one often resorts to inexact line-search techniques that compute a step length α to satisfy certain conditions (see e.g. Wolfe and Goldstein conditions, which we will not be discussing in detail here).

Trust region Trust region methods construct and optimize a local model m_k of the objective function at $\mathbf{x}_k$ within a region whose radius r depends on how well the model approximates the real objective. The candidate step direction $\mathbf{p}_k$ is found by solving the following subproblem:

$$\begin{aligned} &\min_{\mathbf{p} \in \mathbb{R}^d} m_k(\mathbf{x}_k + \mathbf{p}) \\ &\text{s.t. } \|\mathbf{p}\| \leq r. \end{aligned} \tag{3.2}$$

If the candidate solution $\mathbf{p}_k$ does not produce a sufficient decrease in terms of function value, then we conclude that the trust-region radius r is too large, and we, therefore, shrink it.

One of the keys to the efficiency of trust-region methods is to pick a model m_k that is easy to optimize, such as a quadratic function:

$$m_k(\mathbf{x}_k + \mathbf{p}) = f(\mathbf{x}_k) + \mathbf{p}^\top \nabla f(\mathbf{x}_k) + \frac{1}{2}\mathbf{p}^\top \mathbf{B}_k \mathbf{p}, \tag{3.3}$$

where the matrix $\mathbf{B}_k \in \mathbb{R}^{d\times d}$ is either the Hessian $\nabla^2 f(\mathbf{x}_k)$ or an approximation to it.

Steepest descent direction The most obvious direction to choose to move between $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ is the direction that gives the fastest decreases, which can be estimated by solving the problem

$$\min_{\mathbf{p}\in\mathbb{R}^d} \mathbf{p}^\top \nabla f(\mathbf{x}_k) \text{ s.t. } \|\mathbf{p}\| = 1. \tag{3.4}$$

The solution to the above problem is $\mathbf{p}_k = \frac{\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|}$. One can show that this direction is orthogonal to the contour of the function (exercise).

Gradient Descent A well-known variant of steepest descent is gradient descent that starts from some initial $\mathbf{x}_0$ and then iteratively updates the iterate $\mathbf{x}_k \in \mathbb{R}^d$ as follows,

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \nabla f(\mathbf{x}_k),$$

where $\eta > 0$ is a chosen fixed step size. Larger step sizes are usually preferred from the standpoint of computational complexity and performance, but there is a limit to how large the step size can be. We will discuss this later on.

3.2 Quadratic Model

Let's consider a local approximation of f formed by taking the second-order Taylor expansion of f as follows,

$$f(\mathbf{x}+\mathbf{p}) \approx f(\mathbf{x}) + \nabla f(\mathbf{x})^\top \mathbf{p} + \frac{1}{2}\mathbf{p}^\top \nabla^2 f(\mathbf{x})\mathbf{p}.$$

The minimizer of the right-hand side over $\mathbf{p}$ is:

$$\mathbf{p} = -[\nabla^2 f(\mathbf{x})]^{-1}\nabla f(\mathbf{x}).$$

We recover the update of Newton's method, which will be discussed in detail later in the class.

Recovering Gradient Descent: Note that if $\nabla^2 f(\mathbf{x}) = \mathbf{I}$, then

$$f(\mathbf{x}+\mathbf{p}) \approx f(\mathbf{x}) + \nabla f(\mathbf{x})^\top \mathbf{p} + \frac{1}{2\eta}\|\mathbf{p}\|^2.$$

The minimizer is just $\mathbf{p} = -\eta\nabla f(\mathbf{x})$, which recovers the update of gradient descent. This gives us a different interpretation of gradient descent as optimizing a surrogate quadratic model. This is illustrated in Figure 3.1.

3.3 Convergence for Quadratic Functions

We will start by analysing the gradient dynamics and its convergence properties on a simple convex quadratic objective $f : \mathbb{R}^d \to \mathbb{R}$ defined as

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{Q}\, \mathbf{x} - \mathbf{q}^\top \mathbf{x}, \quad \mathbf{Q} \in \mathbb{R}^{d\times d} \text{ is positive definite.}$$

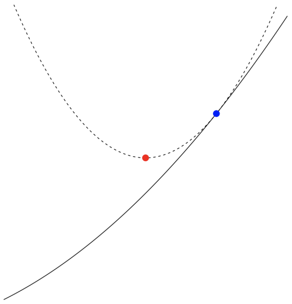


Figure 3.1: A function f is drawn as a plain line, along with its quadratic approximation as a dotted line computed at the blue point. The red point corresponds to the minimizer of the quadratic approximation.

To simplify the problem, we diagonalize $\mathbf{Q} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$ where $\mathbf{U}$ is an orthogonal matrix. We then perform a change of basis $\mathbf{x} \leftarrow \mathbf{U}^\top\mathbf{x}$, $\mathbf{q} \leftarrow \mathbf{U}^\top\mathbf{q}$, which yields an objective function that is separable over the coordinates of $\mathbf{x} = [x^{(1)}, \dots, x^{(d)}]$:

$$f(\mathbf{x}) = \sum_{i=1}^{n} g_i(x^{(i)}), \quad g_i(z) = \frac{\lambda_i}{2} z^2 - p_i z, \quad \lambda_i > 0\,.$$

Since the problem is separable, we can consider a generic function $g := g_i : \mathbb{R} \to \mathbb{R}$, $g(z) = \frac{\lambda}{2}z^2 - pz$. The derivative is

$$g'(z) = \lambda z - p.$$

From first-order optimality, we obtain the following minimizer:

$$z^* = p/\lambda, \quad g(z^*) = \frac{p^2}{2\lambda} - \frac{p^2}{\lambda} = -\frac{p^2}{2\lambda}.$$

To simplify the analysis, we shift g as follows:

$$g(z) \leftarrow g(z) - g(z^*) = \frac{\lambda}{2}\left(z^2 - \frac{2p}{\lambda}z + \frac{p^2}{\lambda^2}\right) = \frac{1}{2\lambda}\left(\lambda z - p\right)^2.$$

A gradient step results in

$$\begin{aligned} g(z - \eta(\lambda z - p)) &= \frac{1}{2\lambda}\left((1-\lambda\eta)(\lambda z - p)\right)^2 \\ &= (1-\lambda\eta)^2 g(z). \end{aligned}$$

We see that we need $\eta < \frac{1}{\lambda}$ for the objective decrease by a constant factor < 1 in every step. This yields exponentially fast convergence to the minimum. I.e. for K steps of gradient descent, we have

$$g(z^K) = (1-\lambda\eta)^{2k} g(z^0).$$

Let's now come back to the multi-dimensional quadratic. The step size condition becomes

$$\eta \leq \frac{1}{\lambda_{max}(\mathbf{Q})}.$$

The optimal step size:

$$\begin{aligned}\eta^* &= \min_\eta \max_i (1-\eta\lambda_i)^2 \\ &= \min_\eta \max\{\eta\lambda_{\max} - 1, 1-\eta\lambda_{min}\}\end{aligned}$$

is attained at

$$\eta\lambda_{max} - 1 \overset{!}{=} 1 - \eta\lambda_{min}$$
$$\iff \eta^* = \frac{2}{\lambda_{max}+\lambda_{min}}.$$

Optimal Rates The slowest rate is in direction of the eigenvector with smallest eigenvalue, for which

$$\rho = (1-\lambda_{min}\eta^*)^2 = \left(\frac{\lambda_{max}-\lambda_{min}}{\lambda_{max}+\lambda_{min}}\right)^2$$
$$\leq (1-\kappa)^2, \quad \kappa = \frac{\lambda_{max}}{\lambda_{min}}.$$

The constant κ is known as the condition number of $\mathbf{Q}$. It shows that gradient descent can be very slow for functions where $\lambda_{max} \gg \lambda_{min}$. We show an example in Figure 3.2, as well as the effect of a preconditioner in Figure 3.3.

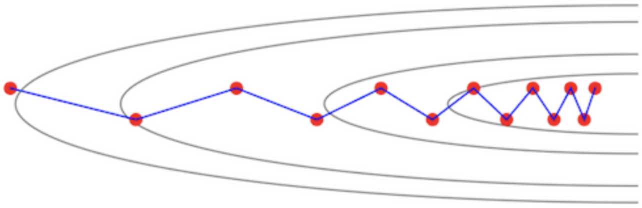


Figure 3.2: Example of a two-dimensional ill-conditioned function where $\lambda_{max} \gg \lambda_{min}$.

3.4 Convergence for Smooth and Strongly-convex Functions

We recall a few definitions introduced earlier in the class.

Definition 25 (L-smooth function)**.** *A function f is L-smooth (alternatively it is said to have L-Lipschitz-continuous gradients) if:*

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}+\mathbf{p})\| \leq L\|\mathbf{p}\|.$$

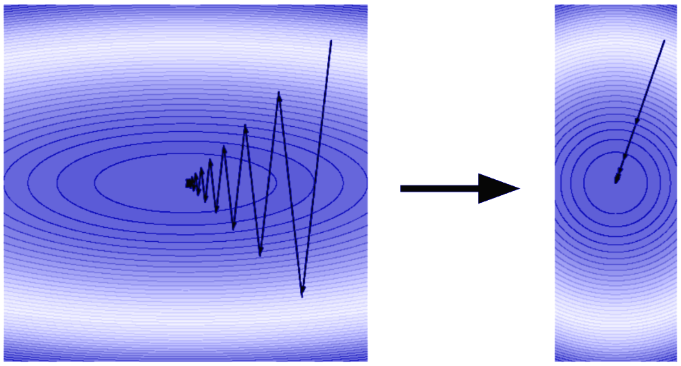


Figure 3.3: Desired effect of a preconditioner. It rescales the different dimensions to make the level sets isotropic. This implies that the condition number $\kappa = 1$.

Definition 26 (μ-strongly convex function)**.** *A function f is μ-strongly convex if*

$$f(\mathbf{x}+\mathbf{p}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top \mathbf{p} + \frac{\mu}{2}\|\mathbf{p}\|^2. \tag{3.5}$$

The function is convex if $\mu = 0$.

If f is twice differentiable then smoothness can be restated as

$$f(\mathbf{x}+\mathbf{p}) \leq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top \mathbf{p} + \frac{L}{2}\|\mathbf{p}\|^2. \tag{3.6}$$

The above simple inequality will serve as the **basis of many convergence proofs** to follow.

We also recall that the Hessian of a (twice differentiable) L-smooth and μ-strongly-convex function is sandwiched as

$$\mu\mathbf{I} \preceq \nabla^2 f \preceq L\mathbf{I}.$$

This connects back to the quadratic case, where $\lambda_{\text{min}} = \mu$ and $\lambda_{max} = L$.

In the following, we will denote by $\mathbf{x}^*$ the unique minimizer of f. We now state our first main convergence result.

Theorem 22. *For a μ-strongly convex, L-smooth function f, the gradient descent iterates $\mathbf{x}_k$ with step size $0 < \eta \leq 1/L$ converge to the unique minimizer $\mathbf{x}^*$ at rate*

$$\|\mathbf{x}_k - \mathbf{x}^*\|^2 \leq (1-\eta\mu)^k \|\mathbf{x}_0 - \mathbf{x}^*\|^2.$$

Corollary 23. *Using L-smoothness, we also have*

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{L}{2}\|\mathbf{x}_k - \mathbf{x}^*\|^2 \leq \frac{L}{2}(1-\eta\mu)^k\|\mathbf{x}_0 - \mathbf{x}^*\|^2.$$

The proof of Theorem 22 makes use of the following so-called descent lemma:

Lemma 24 (Descent lemma)**.** *If f is differentiable and L-smooth, then for all $\mathbf{x} \in \mathbb{R}^d$, and for $\eta \leq \frac{1}{L}$,*

$$f(\mathbf{x}) - f(\mathbf{x}^*) \geq \frac{1}{2L}\|\nabla f(\mathbf{x})\|^2.$$

Proof. From the definition of $\mathbf{x}^*$, we have, for all $\mathbf{x} \in \mathbb{R}^d$,

$$\begin{aligned}
f(\mathbf{x}^*) - f(\mathbf{x}) &\leq f\left(\mathbf{x} - \eta\nabla f(\mathbf{x})\right) - f(\mathbf{x}) \\
&\overset{(3.6)}{\leq} f(\mathbf{x}) - \eta\|\nabla f(\mathbf{x})\|^2 + \frac{L}{2}\|\eta\nabla f(\mathbf{x})\|^2 - f(\mathbf{x}) \\
&= \left(-\eta + \frac{L}{2}\eta^2\right)\|\nabla f(\mathbf{x})\|^2 \\
&\leq -\frac{\eta}{2}\|\nabla f(\mathbf{x})\|^2 \\
&\leq -\frac{1}{2L}\|\nabla f(\mathbf{x})\|^2.
\end{aligned}$$

Note that $\left(-\eta + \frac{L}{2}\eta^2\right)$ is a quadratic function whose minimum is achieved for $\eta = \frac{1}{L}$. □

Proof Theorem 22. From strong convexity (Eq. (3.5)), we have

$$f(\mathbf{x}^*) - f(\mathbf{x}) \geq \nabla f(\mathbf{x})^\top(\mathbf{x}^* - \mathbf{x}) + \frac{\mu}{2}\|\mathbf{x}^* - \mathbf{x}\|^2. \tag{3.7}$$

Using the gradient descent update, we obtain

$$\begin{aligned}
\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2 &= \|\mathbf{x}_k - \eta\nabla f(\mathbf{x}_k) - \mathbf{x}^*\|^2 \\
&= \|\mathbf{x}_k - \mathbf{x}^*\|^2 - 2\eta\nabla f(\mathbf{x}_k)^\top(\mathbf{x}_k - \mathbf{x}^*) + \eta^2\|\nabla f(\mathbf{x}_k)\|^2 \\
&\overset{(\mu)}{\leq} (1-\eta\mu)\|\mathbf{x}_k - \mathbf{x}^*\|^2 - 2\eta(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \eta^2\|\nabla f(\mathbf{x}_k)\|^2 \\
&\overset{(L)}{\leq} (1-\eta\mu)\|\mathbf{x}_k - \mathbf{x}^*\|^2 - 2\eta(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + 2L\eta^2(f(\mathbf{x}_k) - f(\mathbf{x}^*)) \\
&= (1-\eta\mu)\|\mathbf{x}_k - \mathbf{x}^*\|^2 - 2\eta(1-\eta L)(f(\mathbf{x}_k) - f(\mathbf{x}^*)) \\
&\leq (1-\eta\mu)\|\mathbf{x}_k - \mathbf{x}^*\|^2,
\end{aligned}$$

where (μ) uses strong convexity (Eq. (8.11)), and (L) uses the descent lemma.

We conclude the proof by induction over k. □

The convexity condition alone (with smoothness) can only guarantee a slower convergence, as stated below.

Theorem 25. *Let f be convex, differentiable and L-smooth. Then with step size $\eta \leq \frac{1}{L}$, the suboptimality along the gradient descent iterates decays as*

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{1}{2\eta k}\|\mathbf{x}_0 - \mathbf{x}^*\|^2,$$

where $\mathbf{x}^*$ is a minimizer of f.

Proof. Since f is L-smooth, we have

$$\begin{aligned} f(\mathbf{x}_{k+1}) &\leq f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k \rangle + \frac{L}{2}\left\|\mathbf{x}_{k+1} - \mathbf{x}_k\right\|^2 \\ &= f(\mathbf{x}_k) - \omega \left\|\nabla f(\mathbf{x}_k)\right\|^2, \end{aligned} \tag{3.8}$$

where $\omega = \eta\left(1 - \frac{L}{2}\eta\right)$. We therefore get a bound on the gradient norm as

$$\begin{aligned} \omega \left\|\nabla f(\mathbf{x}_k)\right\|^2 &\leq f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) \\ &= (f(\mathbf{x}_k) - f(\mathbf{x}^*)) - (f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)). \end{aligned} \tag{3.9}$$

Using convexity of f, we also have the following relation:

$$\begin{aligned} f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \langle \nabla f(\mathbf{x}_k), \mathbf{x}_k - \mathbf{x}^* \rangle &\leq \|\mathbf{x}_k - \mathbf{x}^*\| \left\|\nabla f(\mathbf{x}_k)\right\| \\ &= C_k \left\|\nabla f(\mathbf{x}_k)\right\|, \end{aligned} \tag{3.10}$$

where $C_k := \|\mathbf{x}_k - \mathbf{x}^*\|$.

Let $\Delta_k := f(\mathbf{x}_k) - f(\mathbf{x}^*)$. Then combining Eq. (3.9) and (3.10) yields:

$$\begin{aligned} \Delta_{k+1} &\leq \Delta_k - \frac{\omega}{C_k^2}\Delta_k^2 \\ \overset{\times\frac{1}{\Delta_k \Delta_{k+1}}}{\Longrightarrow} \frac{1}{\Delta_k} &\leq \frac{1}{\Delta_{k+1}} - \frac{\omega}{C_k^2}\frac{\Delta_k}{\Delta_{k+1}} \\ \Longrightarrow \frac{\omega}{C_k^2}\frac{\Delta_k}{\Delta_{k+1}} &\leq \frac{1}{\Delta_{k+1}} - \frac{1}{\Delta_k} \\ \overset{\Delta_{k+1} \leq \Delta_k}{\Longrightarrow} \frac{\omega}{C_k^2} &\leq \frac{1}{\Delta_{k+1}} - \frac{1}{\Delta_k}. \end{aligned} \tag{3.11}$$

Summing up over $i = 0, \ldots, k$, we obtain

$$k\frac{\omega}{\sup_i C_i^2} \leq \sum_{i=0}^{k} \frac{\omega}{C_i^2} \leq \frac{1}{\Delta_k} - \frac{1}{\Delta_0} \leq \frac{1}{\Delta_k}. \tag{3.12}$$

We conclude that

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{\sup_i C_i^2}{\omega k}. \tag{3.13}$$

Finally, we simply need to select the optimal step size η that maximizes ω which is shown to be $\eta^* = \frac{1}{L}$ in Nesterov (2003) (see Section 2.1.5).

□

The rate $\mathcal{O}\left(\frac{1}{k}\right)$ obtained for convex functions is called "sublinear", while the rate $\mathcal{O}(1-\eta\mu)^k$ obtained for strongly-convex functions is a linear rate, which is faster than sublinear. The name "linear" can be understood by taking the log of $f(\mathbf{x}_k) - f(\mathbf{x}^*)$ and seeing that we obtain a straight line, see illustration in Figure 3.4.

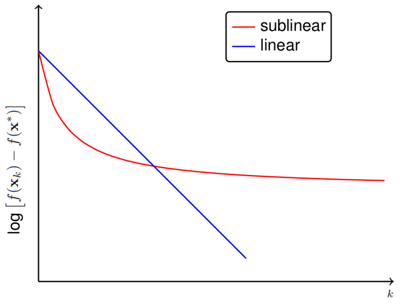


Figure 3.4: Sublinear vs linear rate of convergence.

3.5 Convergence without Convexity

In the non-convex case, local convergence is typically measured in terms of a vanishing gradient norm (i.e. first-order criticality).

$$\boxed{\|\nabla f(\mathbf{x})\|^2 \leq \epsilon} \tag{ϵ-stationarity}$$

Theorem 26. *Assume f is differentiable, L-smooth, but not necessarily convex and with minimum f^*. The gradient descent iterates with step size $\eta \leq 1/L$ satisfy*

$$\min_{i=0}^{k} \|\nabla f(\mathbf{x}_i)\|^2 \leq \frac{2(f(\mathbf{x}_0) - f^*)}{\eta(k+1)}.$$

Note that in the convex case, the decay of the iterate suboptimality also implies that the squared gradient norm vanishes at the same rate. Here the gradient norm vanishes at rate $1/k$ without making any statement about the suboptimality of the iterates.

1. For any $\eta \leq 1/L$ by the descent lemma above

$$\frac{\eta}{2}\|\nabla f(\mathbf{x})\|^2 \leq f(\mathbf{x}) - f(\mathbf{x} - \eta\nabla f(\mathbf{x}))$$

2. Summing over iterates we get a telescoping sum

$$\frac{\eta}{2}\sum_{i=0}^{k}\|\nabla f(\mathbf{x}_i)\|^2 \leq f(\mathbf{x}_0) - f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_0) - f(\mathbf{x}^*)$$

3. We can lower bound the sum by the smallest summand

$$\sum_{i=0}^{k} \|\nabla f(\mathbf{x}_i)\|^2 \geq (k+1) \min_{i=0}^{k} \|\nabla f(\mathbf{x}_i)\|^2.$$

Polyak-Łojasiewicz (PL) Condition The PL condition is a weaker assumption than convexity that still allows gradient descent to obtain a fast (i.e. linear) rate of convergence. It is defined as follows.

Definition 27. *For $\mu > 0$, a function f is μ-PL if*

$$\frac{1}{2}\|\nabla f(\mathbf{x})\|^2 \geq \mu(f(\mathbf{x}) - f^*), \ \forall \mathbf{x}, \ f^* = \min_{\mathbf{x}} f(\mathbf{x}).$$

Two examples of PL functions are shown in Figure 3.5.

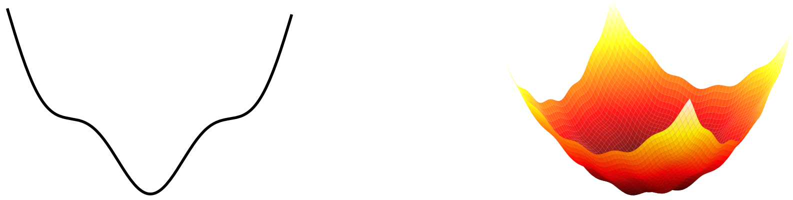


Figure 3.5: Examples of PL functions: 2-dimensional case on the left and 3-dimensional case on the right.

The PL condition ensures fast convergence, see the next theorem.

Theorem 27 (Convergence Theorem with PL Condition)**.** *Let f be differentiable and L-smooth, not necessarily convex with minimum f^* and fulfilling the PL condition with $\mu > 0$. The gradient descent iterates with step size $\eta \leq 1/L$ satisfy*

$$f(\mathbf{x}_k) - f^* \leq \left(1 - \frac{\mu}{L}\right)^k (f(\mathbf{x}_0) - f^*).$$

Proof sketch.

1. By the descent lemma for L-smooth functions, with $\eta \leq \frac{1}{L}$,:

$$f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k) \leq -\frac{1}{2L}\|\nabla f(\mathbf{x}_k)\|^2.$$

2. By the PL condition, we have

$$-\frac{1}{2L}\|\nabla f(\mathbf{x}_k)\|^2 \leq -\frac{\mu}{L}(f(\mathbf{x}_k) - f^*).$$

3. Subtracting f^* on both sides

$$f(\mathbf{x}_{k+1}) - f^* \leq \left(1 - \frac{\mu}{L}\right)(f(\mathbf{x}_k) - f^*).$$

The claim follows by induction.

□

3.6 Summary

We summarize the rates of convergence of gradient descent for various types of smooth functions in Table 9.1.

Function	Quantity	Rate	Theorem
Smooth μ-strongly-convex	$f(\mathbf{x}_K) - f(\mathbf{x}^*)$	$\mathcal{O}((1-\frac{\mu}{L})^K)$	Thm. 22
Smooth Convex	$f(\mathbf{x}_K) - f(\mathbf{x}^*)$	$\mathcal{O}\left(\frac{1}{K}\right)$	Thm. 25
Smooth μ-PL	$f(\mathbf{x}_K) - f(\mathbf{x}^*)$	$\mathcal{O}\left(\left(1-\frac{\mu}{L}\right)^K\right)$	Thm. 27
Smooth (non-convex)	$\inf_{s\in[0,K]} \left\lVert\nabla f(\mathbf{x}_s)\right\rVert^2$	$\mathcal{O}\left(\frac{1}{K}\right)$	Thm. 26

Table 3.1: Convergence rates of gradient descent for different function classes.

One can also ask for the number of iterations required to achieve $f(\mathbf{x}_K) - f(\mathbf{x}^*) \leq \epsilon$ (ϵ optimality). Let's for instance consider the convex and strongly-convex case.

Convex functions Choosing $\eta = \frac{1}{2L}$, we obtain

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{LR_0^2}{k},$$

where $R_0 = \|\mathbf{x}_0 - \mathbf{x}^*\|$. Setting $\frac{LR_0^2}{k} \leq \epsilon$ is equivalent to $k \geq \frac{LR_0^2}{\epsilon}$.

Strongly-convex functions Choosing $\eta = \frac{1}{2L}$, we obtain

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \frac{L}{2}\left(1 - \frac{\mu}{2L}\right)^k R_0^2,$$

where $R_0 = \|\mathbf{x}_0 - \mathbf{x}^*\|$.

We can obtain the number of iterations k to achieve $f(\mathbf{x}_k) - f(\mathbf{x}^*) \leq \epsilon$ as follows:

$$\begin{aligned}
&\frac{L}{2}\left(1 - \frac{\mu}{2L}\right)^k R_0^2 \leq \epsilon \\
\implies &\log\left(\frac{L}{2}\left(1 - \frac{\mu}{2L}\right)^k R_0^2\right) \leq \log\epsilon \\
\implies &\log\left(\frac{LR_0^2}{2}\right) + \log\left(\left(1 - \frac{\mu}{2L}\right)^k\right) \leq \log\epsilon \\
\implies &k\log\left(1 - \frac{\mu}{2L}\right) \leq \log\epsilon - \log\left(\frac{LR_0^2}{2}\right) = \log\left(\frac{2\epsilon}{LR_0^2}\right) \\
\implies &k \geq \left(\log\left(1 - \frac{\mu}{2L}\right)\right)^{-1} \log\left(\frac{2\epsilon}{LR_0^2}\right),
\end{aligned}$$

where the inequality in last step switches direction since $\log\left(1 - \frac{\mu}{2L}\right) < 0$.

3.7 Exercise: Gradient Descent

Problem 1 (Quadratic function):

Consider a quadratic function $f : \mathbb{R}^d \to \mathbb{R}$ of the form $f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}\mathbf{x} + \mathbf{b}^\top \mathbf{x} + c$, where $\mathbf{A} \in \mathbb{R}^{d\times d}$ is symmetric invertible and $\mathbf{b} \in \mathbb{R}^d, c \in \mathbb{R}$.

1. Prove that f is smooth with constant $2\|\mathbf{A}\|$, where we recall that $\|\mathbf{A}\| := \sup_{\mathbf{x}\neq 0} \frac{\|\mathbf{A}\mathbf{x}\|}{\|\mathbf{x}\|}$.
2. What's the minimum value of f?

Problem 2 (Biased gradients):

Consider the gradient descent update with a bias:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta\nabla f(\mathbf{x}_k) + \epsilon_k, \tag{3.14}$$

where $\eta > 0$ is the step size and $\epsilon_k > 0$ is a bias. We assume that $\eta \leq \frac{1}{L}$.

1. Show that
$$f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k) + \frac{\eta}{2}\left(-\|\nabla f(\mathbf{x}_k)\|^2 + \|\epsilon_k\|^2\right).$$
2. Conclude that
$$\min_{k=1\ldots K} \|\nabla f(\mathbf{x}_k)\|^2 \leq \frac{\eta}{2K}(f(\mathbf{x}_1) - f(\mathbf{x}^*)) + \frac{1}{K}\sum_{k=1}^{K} \|\epsilon_k\|^2,$$

Problem 3 (Normalized Gradient Descent):

In this exercise, we consider a variant of gradient descent known as normalized gradient descent. At each iteration, it normalizes the gradient by its norm, which yields the following update step:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta\frac{\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|}, \tag{3.15}$$

where $\eta > 0$ is a chosen step size.

We assume that f is convex and L-smooth. Prove that

1. $$\|\nabla f(\mathbf{x}_k)\| \leq \frac{f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})}{\eta} + \frac{L\eta}{2}.$$
2. If we choose $\eta = \frac{2\epsilon}{L}$, how many iterations do we need to obtain $\frac{1}{k}\sum_{i=0}^{k-1} \|\nabla f(\mathbf{x}_k)\| \leq \epsilon$?

Problem 4 (Programming):

Complete TODOs in the Jupyter Notebook provided by implementing the Gradient Descent optimizer for a Linear Regression task. Then, study the behavior of the optimizer for different step sizes, initialization, and maximum number of iterations.

Chapter 4

Subgradient Method

Disclaimer 2: *Part of these lecture notes are based on the Convex Optimization lecture of Prof. Ryan Tibshirani (Berkeley).*

4.1 Subgradient

Subgradients generalize the notion of derivative to convex functions that are not necessarily differentiable. As an example, consider the function $f : I \to \mathbb{R}$ (where I is an open interval) shown in blue in Figure 4.1. Although the function f is not differentiable at the point x_0, one can draw functions (shown in red) that go through the point $(x_0, f(x_0))$ and which are everywhere either touching f or below the graph of f. The slope of such a line is called a subderivative (subgradient in general for multi-dimensional functions).

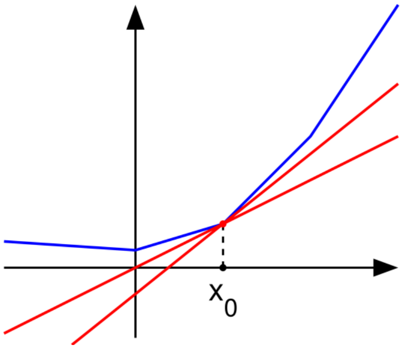


Figure 4.1: Illustration subgradient (source: Wikipedia): a convex function (blue) and its "subtangent lines" at x_0 (red).

Definition 28. *A subgradient of the convex function $f : \mathbb{R}^d \to \mathbb{R}$ at $\mathbf{x}$ is defined as a vector $\mathbf{g}$, such that*

$$\mathbf{g}^\top(\mathbf{y} - \mathbf{x}) \leq f(\mathbf{y}) - f(\mathbf{x}), \quad \forall \mathbf{y} \in \mathbb{R}^d. \tag{4.1}$$

For convex functions f, subgradients are guaranteed to exist. This is however not the case for non-convex functions. If f is differentiable, then $\mathbf{g} = \nabla f(\mathbf{x})$ uniquely.

Examples

1. $f(x) = |x|$ (see Figure 4.2). Then $g = \begin{cases} \text{sign}(x) \text{ if } x \neq 0 \\ [-1, 1] \text{ if } x = 0 \end{cases}$ where $\text{sign}(x) = \begin{cases} -1 \text{ if } x < 0 \\ +1 \text{ if } x > 0 \end{cases}$.

2. $f(\mathbf{x}) = \|\mathbf{x}\|_2$. Then $\mathbf{g} = \begin{cases} \frac{\mathbf{x}}{\|\mathbf{x}\|_2} \text{ if } \mathbf{x} \neq 0 \\ \{\mathbf{z} \mid \|\mathbf{z}\|_2 \leq 1\} \text{ if } \mathbf{x} = 0 \end{cases}$.

3. $f(\mathbf{x}) = \|\mathbf{x}\|_1 = \sum_{i=1}^{d} |x_i|$ (see Figure 4.3). Then the i-th coordinate of g is $g_i = \begin{cases} \text{sign}(x_i) \text{ if } x_i \neq 0 \\ [-1, 1] \text{ if } x_i = 0. \end{cases}$

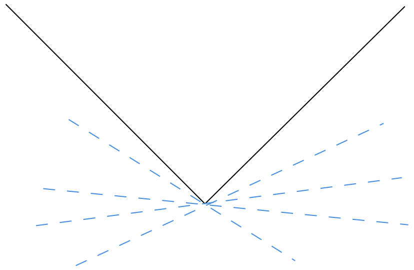


Figure 4.2: Absolute function $f(x) = |x|$ show in black and some examples of subgradient in blue (dotter lines) at $x = 0$.

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

We list some important properties of the subdifferential:

- The subdifferential is always a closed convex set.
- The subdifferential is always non-empty for convex functions (it can be empty for non-convex functions)
- If f is convex and differentiable at $\mathbf{x}$, then $\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}$.
- Conversely, if $\partial f(\mathbf{x}) = \{\mathbf{g}\}$, then f is differentiable at $\mathbf{x}$ and $\nabla f(\mathbf{x}) = \mathbf{g}$.

Subgradient calculus Assuming f is convex, the following properties hold:

- Scaling: $\partial(af) = a \cdot \partial f$, for $a > 0$
- Addition: $\partial(f_1 + f_2) = \partial f_1 + \partial f_2$
- Affine composition: if $g(\mathbf{x}) = f(\mathbf{A}\mathbf{x} + \mathbf{b})$, then $\partial g(\mathbf{x}) = \mathbf{A}^\top \partial f(\mathbf{A}\mathbf{x} + \mathbf{b})$

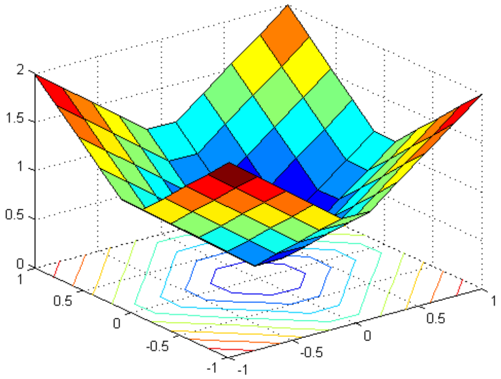


Figure 4.3: Illustration of the function $f(\mathbf{x}) = \|\mathbf{x}\|_1$ in dimension 3.

- Finite pointwise maximum: if $f(\mathbf{x}) = \max_{i=1,\dots,m} f_i(\mathbf{x})$, then $\partial f(\mathbf{x}) = \operatorname{conv}\left(\cup_{i:f_i(\mathbf{x})=f(\mathbf{x})} \partial f_i(\mathbf{x})\right)$, i.e. we first form the union of the functions $f_i(\mathbf{x})$ that achieve the maximum $f(\mathbf{x})$, and we then take the convex hull (the union alone is not necessarily convex).

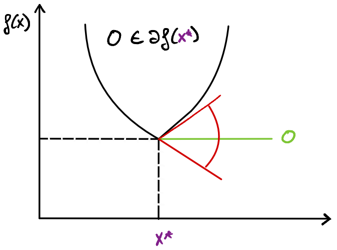
Let's look at some examples:

1. Given two differentiable functions $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$, define $f(\mathbf{x}) = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$. Then
$$\partial f(\mathbf{x}) = \operatorname{conv}(\{\nabla f_1(\mathbf{x}), \nabla f_2(\mathbf{x})\}) = \{\alpha \nabla f_1(\mathbf{x}) + (1-\alpha)\nabla f_2(\mathbf{x}) : \alpha \in [0,1]\}$$

2. Let $f(\mathbf{x}) = \|\mathbf{x}\|_p$ and q be such that $\frac{1}{q} + \frac{1}{p} = 1$. Then $f(\mathbf{x}) = \|\mathbf{x}\|_p = \max_{\mathbf{z} | \|\mathbf{z}\|_q \le 1} \mathbf{z}^\top \mathbf{x}$, therefore
$$\partial f(\mathbf{x}) = \operatorname*{argmax}_{\mathbf{z} | \|\mathbf{z}\|_q \le 1} \mathbf{z}^\top \mathbf{x}.$$

Optimality condition The following gives a first-order optimality condition based on the subgradient of any function f.

Proposition 28. *Given a function $f : \mathbb{R}^d \to \mathbb{R}$ (not necessarily convex), then*
$$f(\mathbf{x}^*) = \min_{\mathbf{x}} f(\mathbf{x}) \iff \mathbf{0} \in \partial f(\mathbf{x}^*).$$



Proof. It simply follows from the following observation: $\mathbf{g} = \mathbf{0} \in \partial f(\mathbf{x}^*)$ means that for all $\mathbf{y} \in \mathbb{R}^d$,
$$f(\mathbf{y}) \ge f(\mathbf{x}^*) + \mathbf{0}^\top(\mathbf{y} - \mathbf{x}^*) = f(\mathbf{x}^*).$$
□

Example 1: lasso optimality conditions Consider the following lasso optimization problem. Given $\mathbf{y} \in \mathbb{R}^n$, $\mathbf{X} \in \mathbb{R}^{n \times d}$, we seek

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^d} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1, \lambda \geq 0. \tag{4.2}$$

The subgradient optimality condition is then

$$\begin{aligned} \mathbf{0} &\in -\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \lambda \partial \|\boldsymbol{\beta}\|_1 \\ &\implies \mathbf{X}^\top(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = \lambda \mathbf{v}, \end{aligned}$$

where $\mathbf{v} = (v_i)_i \in \mathbb{R}^d$ is such that, $\forall i = 1, \ldots d$,

$$v_i \in \begin{cases} \{1\} & \text{if } \beta_i > 0 \\ \{-1\} & \text{if } \beta_i < 0 \\ [-1, 1] & \text{if } \beta_i = 0. \end{cases}$$

4.2 Subgradient Method

We consider the minimization of a non-smooth function $f(\mathbf{x})$, for which we can compute a subgradient $\mathbf{g}$.

The subgradient method minimizes $f(\mathbf{x})$ using the following simple iterative update:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta_k \mathbf{g}_k, \tag{4.3}$$

where $\eta_k > 0$ is the step size and $\mathbf{g}_k$ is a subgradient at $\mathbf{x}_k$. The initial vector $\mathbf{x}_0 \in \mathbb{R}^d$ is typically chosen at random (or hand-picked).

Not a descent method Let's consider again the subgradient inequality stated in Eq. (4.1):

$$\forall \mathbf{y} \in \mathbb{R}^d, \mathbf{g}^\top(\mathbf{y} - \mathbf{x}) \leq f(\mathbf{y}) - f(\mathbf{x}). \tag{4.4}$$

Take $\mathbf{g} := \mathbf{g}_k$ to be a subgradient of f at $\mathbf{x}_k$ and $\mathbf{y} = \mathbf{x}_{k+1}$, $\mathbf{x} = \mathbf{x}_k$ then the sub-gradient inequality states that:

$$\begin{aligned} f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k - \eta_k \mathbf{g}_k) &\geq f(\mathbf{x}_k) + \mathbf{g}_k^\top(\mathbf{x}_{k+1} - \mathbf{x}_k) \\ &= f(\mathbf{x}_k) + \mathbf{g}_k^\top(-\eta_k \mathbf{g}_k) \\ &= f(\mathbf{x}_k) - \eta_k \|\mathbf{g}_k\|_2^2. \end{aligned}$$

This inequality implies that $f(\mathbf{x}^{k+1})$ can be any value *larger* than $f(\mathbf{x}_k) - \eta_k \|\mathbf{g}_k\|_2^2$, and in particular, any value above $f(\mathbf{x}_k)$. Therefore, the subgradient inequality does not ensure that the subgradient method is a descent method (this is why it is not named "subgradient descent"). In contrast, when the function f is smooth and differentiable, it's worth recalling that we have demonstrated the validity of the descent lemma, which guarantees we minimize the function at every step of gradient descent.

4.2.1 Convergence analysis for convex functions

Since the subgradient method is not necessarily a descent method, we will need to keep track of the best iterate $\mathbf{x}_{\min}$ in the sequence $\mathbf{x}_1, \mathbf{x}_2, \ldots$ generated by the algorithm.

Lemma 29. *Given $R, L > 0$, let $\|\mathbf{x}_1 - \mathbf{x}^*\|_2 \leq R$ and $\|\mathbf{g}_i\| \leq L$ $\forall i = 1, \ldots d$, then the iterates produced by the subgradient method satisfy*

$$\min_{i=1\ldots k} f(\mathbf{x}_i) - f^* \leq \frac{R^2 + L^2 \sum_{i=1}^k \eta_i^2}{2(\sum_{i=1}^k \eta_i)}. \tag{4.5}$$

Proof. Since $\mathbf{g}_k$ is a subgradient of the objective function f, we have

$$f(\mathbf{x}^*) \geq f(\mathbf{x}_k) + \mathbf{g}_k^\top (\mathbf{x}^* - \mathbf{x}_k). \tag{4.6}$$

Then

$$\begin{aligned} \|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 &= \|\mathbf{x}_k - \eta_k \mathbf{g}_k - \mathbf{x}^*\|_2^2 \\ &= \|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - 2\eta_k \langle \mathbf{x}_k - \mathbf{x}^*, \mathbf{g}_k \rangle + \eta_k^2 \|\mathbf{g}_k\|_2^2 \\ &\overset{(4.6)}{\leq} \|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - 2\eta_k (f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \eta_k^2 \|\mathbf{g}_k\|_2^2. \end{aligned} \tag{4.7}$$

Applying the inequality above recursively, we have

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 \leq \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2 - 2\sum_{i=1}^k \eta_i (f(\mathbf{x}_i) - f(\mathbf{x}^*)) + \sum_{i=1}^k \eta_i^2 \|\mathbf{g}_i\|_2^2, \tag{4.8}$$

which implies

$$\begin{aligned} 2\sum_{i=1}^k \eta_i (f(\mathbf{x}_i) - f(\mathbf{x}^*)) &\leq \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2 - \|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 + \sum_{i=1}^k \eta_i^2 \|\mathbf{g}_i\|_2^2 \\ &\leq \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2 + \sum_{i=1}^k \eta_i^2 \|\mathbf{g}_i\|_2^2. \end{aligned} \tag{4.9}$$

Combining this inequality with

$$\sum_{i=1}^k \eta_i (f(\mathbf{x}_i) - f(\mathbf{x}^*) \geq \left(\sum_{i=1}^k \eta_i\right) \min_{i=1\ldots k} (f(\mathbf{x}_i) - f(\mathbf{x}^*), \tag{4.10}$$

we get

$$\min_{i=1\ldots k} f(\mathbf{x}_i) - f^* \leq \frac{\|\mathbf{x}_1 - \mathbf{x}^*\|_2^2 + \sum_{i=1}^k \eta_i^2 \|\mathbf{g}_i\|_2^2}{2(\sum_{i=1}^k \eta_i)}. \tag{4.11}$$

□

Constant step-size

We consider the case where the step size $\eta_i = \eta > 0$ is constant over all iterations i.

Theorem 30. *Given $R, L > 0$, let $\|\mathbf{x}_1 - \mathbf{x}^*\|_2 \leq R$ and $\|\mathbf{g}_i\| \leq L\ \forall i$, then the iterates produced by the subgradient method with a constant step-size $\eta^* = \frac{R/L}{\sqrt{k}}$ satisfy*

$$\min_{i=1 \dots k} f(\mathbf{x}_i) - f^* \leq \frac{RL}{\sqrt{k}}. \tag{4.12}$$

Proof. We simply need to optimize the bound obtained in Lemma 29. To do so, we define the following function

$$h(\eta_1, \dots \eta_k) = \frac{R^2 + L^2 \sum_{i=1}^k \eta_i^2}{2(\sum_{i=1}^k \eta_i)}. \tag{4.13}$$

The function h is convex and symmetric with respect to the η_i's (i.e. exchanging any two η_i's yields the same function value). Therefore, the optimal value occurs when all η_i's are equal (let η be this value). We therefore need to minimize the function

$$h(\eta) = \frac{R^2 + L^2 k \eta^2}{2k\eta}. \tag{4.14}$$

The minimum is attained at $\eta^* = \frac{R/L}{\sqrt{k}}$, for which we obtain

$$\min_{i=1 \dots k} f(\mathbf{x}_i) - f^* \leq \frac{RL}{\sqrt{k}}. \tag{4.15}$$

□

Decreasing step-size See Boyd et al. (2003) for a discussion of the rate of convergence with a decreasing step size η_i.

4.2.2 Convergence analysis for strongly-convex functions

Recall that a differentiable function f that is μ-strongly-convex functions f satisfies the inequality:

$$f(\mathbf{y}) \geq f(\mathbf{x}) + (\mathbf{y} - \mathbf{x})^\top \nabla f(\mathbf{x}) + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{4.16}$$

One can also require a similar inequality to hold for non-differentiable functions where $\nabla f(\mathbf{x})$ is replaced by a subgradient. Then, one can first prove that (exercise):

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 \leq (1 - \mu\eta_k)\|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - 2\eta_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \eta_k^2 \|\mathbf{g}_k\|_2^2. \tag{4.17}$$

For a decreasing step size, we obtain the following rate of convergence.

Theorem 31. *Let f be μ-strongly convex and L-Lipschitz continuous ($\|\mathbf{g}_i\| \leq L\ \forall i$), then the iterates produced by the subgradient method with a step-size $\eta_i = \frac{1}{\mu i}$ satisfy*

$$\min_{i=1\ldots k} f(\mathbf{x}_i) - f^* \leq \frac{L^2(\log k + 1)}{2\mu k}. \tag{4.18}$$

Proof. See exercise sheet. □

4.2.3 Summary: worst-case iteration complexity

Table 9.1 compares the rate of convergence of gradient descent and subgradient method to reach $\min_{i=1,\ldots,k} f(\mathbf{x}_i) - f(\mathbf{x}^*) \leq \epsilon$. Keep in mind that for $\epsilon < 1$, we have that $k = \mathcal{O}(\epsilon^{-1})$ is (quadratically) better than $k = \mathcal{O}(\epsilon^{-2})$ iterations.

	L-Lipschitz	L_2-smooth
Convex	$\left(\frac{RL}{\epsilon}\right)^2$	$\frac{R^2 L_2}{\epsilon}$
μ-strongly-convex	$\frac{L^2}{\mu\epsilon}$	$\kappa \log\left(\frac{R^2 L_2}{\epsilon}\right)$

Table 4.1: Iteration complexity gradient descent and subgradient method to reach $\min_{i=1,\ldots,k} f(\mathbf{x}_i) - f(\mathbf{x}^*) \leq \epsilon$. κ is the condition number of the function f (which is always greater than 1).

4.3 Exercise: Subgradient Method

Problem 1 (Properties subdifferential):

Prove that:

1. The subdifferential is always a closed convex set.
2. The subdifferential is not always non-empty for non-convex functions.

Problem 2 (Convergence rate for strongly-convex functions):

Prove that if $\|\mathbf{g}_i\| \leq L \ \forall i$,

$$f\left(\frac{2}{K(K+1)}\sum_{k=0}^{K-1}(k+1)\mathbf{x}_k\right) - f^* \leq \frac{2L^2}{\mu(K+1)}.$$

1. Prove that $\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 \leq (1-\mu\eta_k)\|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - 2\eta_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \eta_k^2 L^2$
2. Rewrite the above expression using $\eta_k = \frac{2}{\mu(k+1)}$ and calculate $k(f(\mathbf{x}_k) - f(\mathbf{x}^*))$
3. Observe that $k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) \leq \frac{\mu}{4}\left(k(k-1)\|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - k(k+1)\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2\right) + \frac{1}{\mu}L^2$
4. Sum the above expression for $k \in \{1, \cdots, K\}$. Hint: The series is telescopic.
5. Conclude using convexity.

Problem 3 (Polyak's step size):

In the subgradient lecture, we have seen that

$$\begin{aligned}\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 &= \|\mathbf{x}_k - \eta_k\mathbf{g}_k - \mathbf{x}^*\|_2^2 \\ &= \|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - 2\eta_k\langle \mathbf{x}_k - \mathbf{x}^*, \mathbf{g}_k\rangle + \eta_k^2\|\mathbf{g}_k\|_2^2 \\ &\leq \|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - 2\eta_k(f(\mathbf{x}_k) - f(\mathbf{x}^*)) + \eta_k^2\|\mathbf{g}_k\|_2^2. \end{aligned} \tag{4.19}$$

1. Prove that the optimal step size is equal to $\eta_k = \frac{(f(\mathbf{x}_k) - f(\mathbf{x}^*))}{\|\mathbf{g}_k\|_2^2}$ (this step size is known as Polyak's step size).
2. Does this step size guarantee convergence?
3. What's the drawback of this approach?

Problem 4 (Programming: regularized logistic regression):

Complete TODOs in the Jupyter Notebook provided to implement the (sub)-gradient method for the ℓ^1 and ℓ^2-regularized Logistic Regression.

Chapter 5

Constrained Optimization

5.1 Introduction

Assume we want to solve the constrained problem

$$\mathbf{y} = \operatorname*{argmin}_{\mathbf{x} \in C} f(\mathbf{x}), \tag{5.1}$$

where $f : C \to \mathbb{R}$ is a convex function and $C \subset \mathbb{R}^d$ is a convex set.

There are several ways to encode the set of constraints C, including:

1. Equality constraints are constraints of the form $g(\mathbf{x}) = 0$, where $g(\mathbf{x})$ is a function of the decision variables $\mathbf{x}$.
2. Inequality constraints are of the form $g(\mathbf{x}) \leq 0$ or $g(\mathbf{x}) \geq 0$.
3. Linear constraints are constraints of the form $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ or $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A}$ is a matrix of coefficients and $\mathbf{b}$ is a vector of constants.
4. Nonlinear constraints are constraints of the form $h(\mathbf{x}) \leq 0$ or $h(\mathbf{x}) = 0$, where $h(\mathbf{x})$ is a nonlinear function of the decision variables $\mathbf{x}$.

Terminology

- If the constraint is **loose**, it implies that no force is required to prevent the optimization variable from violating the constraint. Consequently, at any optimum $\mathbf{x}^*$, we have $\nabla f(\mathbf{x}^*) = 0$, resembling an unconstrained optimization problem.
- Alternatively, a constraint could be **tight**, signifying that the constraint is exactly satisfied (e.g. $g(\mathbf{x}^*) = 0$ for an inequality constraint $g(\mathbf{x}) \leq 0$). In this case, we may have $\nabla f(\mathbf{x}^*) \neq 0, h(\mathbf{x}^*) \neq 0, g(\mathbf{x}^*) \neq 0$. We will come back to this later.

Example Consider the following two-dimensional constrained problem:

$$\begin{aligned} \mathbf{y} = \operatorname*{argmin}_{\mathbf{x} \in \mathbb{R}^2} f(\mathbf{x}) \\ \text{s.t. } x_1^2 + x_2^2 = 1, \end{aligned}$$

which is also illustrated in Figure 5.1. One can clearly see how the unconstrained optimum $\operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^2} f(\mathbf{x})$ shown as a purple star is different from the constrained optimum shown as an orange start.

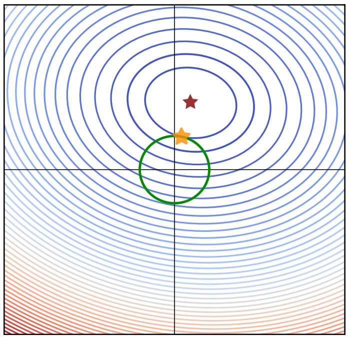


Figure 5.1: Illustration of a constrained problem in 2-dimension. The level sets of the function $f(\mathbf{x})$ are shown with colored ellipses. The constraint $x_1^2 + x_2^2 = 1$ is shown as a green circle. The optimum of the constrained problem is shown with an orange star. Note that the constrained optimum lies at the intersection of the green circle with one of the contour lines of the function $f(\mathbf{x})$. We will later see that this point is such that the gradients of $f(\mathbf{x})$ and $g(\mathbf{x}) = x_1^2 + x_2^2$ are aligned with each other.

5.2 Geometric optimality conditions

First-order optimality conditions We first introduce the concept of a cone, which will be needed to define optimality conditions for a constrained problem.

Definition 29. *Let C be a non-empty subset of $\mathbb{R}^d$. We say that C is a cone if*

$$\lambda \mathbf{x} \in C,$$

whenever $\mathbf{x} \in C$ and $\lambda \geq 0$. Additionally, if C is convex, then it is a convex cone.

A first-order optimality condition for Eq. (5.1) can be defined by using the concept of normal cone defined below [1].

Definition 30 (Normal cone)**.** *The normal cone of a convex set C at a point $\mathbf{x} \in C$ is given by*

$$N_C(\mathbf{x}) = \{\mathbf{w} : \langle \mathbf{w}, \mathbf{y} - \mathbf{x} \rangle \leq 0, \mathbf{y} \in C\}. \tag{5.2}$$

The normal cone defines all directions that are negatively correlated with $\mathbf{y}-\mathbf{x}$, including directions that point straight out (90-degree angle) from the set C.

Examples We compute normal cones for various examples of sets (see Figure 5.2):

[1] A related concept is one of the tangent cone (which coincides with the concept of normal cone for convex sets, see proof in Rockafellar and Wets, 2004, Theorem 6.9). Formally, the tangent cone of a convex set C at a point $\mathbf{x} \in C$ is given by

$$T_C(\mathbf{x}) = \mathrm{cl}\{s(\mathbf{y} - \mathbf{x}) : \mathbf{y} \in C, s \geq 0\},$$

where cl denotes the closure of the set.

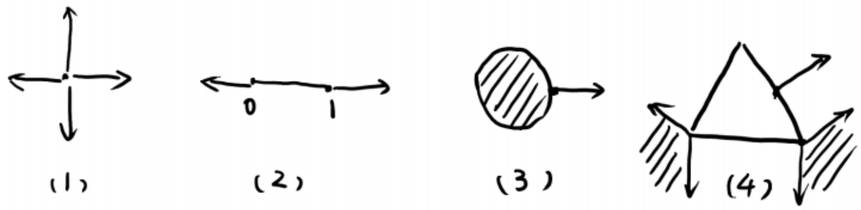


Figure 5.2: Illustration taken from `https://people.orie.cornell.edu/dsd95/teaching/orie6300/lec05.pdf`. Normal cones of several convex sets.

1. Let $S = \{\mathbf{z}\}$ (a singleton), then

$$N_S(\mathbf{x}) = \begin{cases} \mathbb{R}^d & \text{if } \mathbf{x} = \mathbf{z} \\ \emptyset & \text{otherwise} \end{cases}$$

(indeed if $\mathbf{x} = \mathbf{z}$, the required condition is trivially satisfied for all vectors, while if $\mathbf{x} \neq \mathbf{z}$, no vector can satisfy the required condition since it would need to hold for all vectors in $\mathbb{R}^d$)

2. Let $S = [0, 1]$, then

$$N_S(x) = \begin{cases} \mathbb{R}_{\leq 0} & \text{if } x = 0 \\ \mathbb{R}_{\geq 0} & \text{if } x = 1 \\ \{0\} & \text{if } x \in (0, 1) \\ \emptyset & \text{otherwise} \end{cases}$$

3. Let $S = \{\mathbf{x} | \|\mathbf{x}\| \leq 1, \mathbf{x} \in \mathbb{R}^d\}$, then

$$N_S(\mathbf{x}) = \begin{cases} \mathbb{R}_{\geq 0}\mathbf{x} & \text{if } \|\mathbf{x}\| = 1 \\ \{0\} & \text{if } \|\mathbf{x}\| < 1 \\ \emptyset & \text{otherwise} \end{cases}$$

4. The normal cone of a triangle, computed at some but not all points, is depicted in Figure 5.2.

Theorem 32. *Given a closed convex set C, the normal cone $N_C(\mathbf{x})$ is a closed convex cone.*

Proof. 1) $N_C(\mathbf{x})$ is a cone: For any $\mathbf{w} \in N_C(\mathbf{x})$, we have

$$\begin{aligned} &\langle \mathbf{w}, \mathbf{y} - \mathbf{x} \rangle \leq 0 \quad \forall \mathbf{y} \in C \\ \implies &\langle \lambda \mathbf{w}, \mathbf{y} - \mathbf{x} \rangle = \lambda \langle \mathbf{w}, \mathbf{y} - \mathbf{x} \rangle \leq 0 \quad \forall \mathbf{y} \in C. \end{aligned}$$

Therefore $\lambda \mathbf{w} \in N_C(\mathbf{x})$.

2) $N_C(\mathbf{x})$ is convex:
Simply apply the definition of a convex set. Let $\mathbf{v}_1, \mathbf{v}_2 \in N_C(\mathbf{x})$. We have

$$\langle \mathbf{v}_1, \mathbf{y} - \mathbf{x} \rangle \leq 0 \quad \forall \mathbf{y} \in C$$
$$\langle \mathbf{v}_2, \mathbf{y} - \mathbf{x} \rangle \leq 0 \quad \forall \mathbf{y} \in C.$$

It follows that

$$\langle \lambda \mathbf{v}_1 + (1-\lambda)\mathbf{v}_2, \mathbf{y} - \mathbf{x} \rangle = \lambda \langle \mathbf{v}_1, \mathbf{y} - \mathbf{x} \rangle + (1-\lambda)\langle \mathbf{v}_2, \mathbf{y} - \mathbf{x} \rangle \leq 0, \tag{5.3}$$

i.e. $\lambda \mathbf{v}_1 + (1-\lambda)\mathbf{v}_2 \in N_C(\mathbf{x})$.

3) Closeness: The closeness property also follows from the continuity of the norm (since the preimage of any closed set is closed under a continuous map). Specifically, any sequence $\mathbf{w}_i \in N_C(\mathbf{x})$ satisfies $\langle \mathbf{w}_i, \mathbf{y} - \mathbf{x} \rangle \leq 0$ for all $\mathbf{y} \in C$ and therefore if $\mathbf{w}_i \to \mathbf{w}$, continuity implies that $\langle \mathbf{w}, \mathbf{y} - \mathbf{x} \rangle \leq 0$ for all $\mathbf{y} \in C$, i.e. $\mathbf{w} \in N_C(\mathbf{x})$ and therefore $N_C(\mathbf{x})$ is indeed closed.

□

Theorem 33. *Let C be a convex subset of $\mathbb{R}^d$. If $\mathbf{x} \in int(C)$, then*

$$N_C(\mathbf{x}) = \{\mathbf{0}\}.$$

Proof. Since $\mathbf{x} \in \text{int}(C), \exists \delta > 0$ s.t. $B_\delta(\mathbf{x}) \subset C$. Fix $\mathbf{w} \in N_C(\mathbf{x})$. By the definition,

$$\langle \mathbf{w}, \mathbf{y} - \mathbf{x} \rangle \leq 0 \quad \forall \mathbf{y} \in C.$$

For sufficiently small $t > 0$, we have

$$\mathbf{x} + t\mathbf{w} \in B_\delta(\mathbf{x}) \subset C,$$

thus

$$\langle \mathbf{w}, \mathbf{x} + t\mathbf{w} - \mathbf{x} \rangle = t\langle \mathbf{w}, \mathbf{w} \rangle \leq 0 \quad \forall \mathbf{y} \in C$$
$$\implies \|\mathbf{w}\|^2 = 0 \implies \mathbf{w} = \mathbf{0}. \tag{5.4}$$

□

Finally, we state one more theorem that is sometimes helpful. We will not prove it here and instead refer the reader to standard convex analysis textbooks, e.g. Borwein and Lewis (2006).

Theorem 34. *Let f be a proper convex function and $\bar{\mathbf{x}}$ be an interior point of* dom$f$. *Denote the sublevel set $\{\mathbf{x} : f(\mathbf{x}) \leq f(\bar{\mathbf{x}})\}$ by C and the normal cone to C at $\bar{\mathbf{x}}$ by $N_C(\bar{\mathbf{x}})$. Moreover, assume that $f(\bar{\mathbf{x}}) > \inf f(\mathbf{x})$. Then, we have $N_C(\bar{\mathbf{x}}) = \text{cone}\,\partial f(\bar{\mathbf{x}})$.*

Here is another example demonstrating the usefulness of Theorem 34. Consider the set

$$\Omega_2 := \{\mathbf{x} \in \mathbb{R}^2 : \|\mathbf{x}\|_2 \leq 1\}.$$

Equivalently, Ω_2 is generated by the inequality constraint

$$0 \geq c(\mathbf{x}) = 1 - x_1^2 - x_2^2.$$

Thus, by Theorem 34, the normal cone is generated by $\nabla c(\mathbf{x}) = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$ or equivalently by $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$, therefore

$$N_{\Omega_2} = \left\{ \begin{pmatrix} \lambda \\ 0 \end{pmatrix} : \lambda \geq 0 \right\}.$$

First-order optimality Given a convex set C and a convex and differentiable function $f : \mathbb{R}^d \to \mathbb{R}$, the optimization problem

$$\begin{aligned} &\min_{\mathbf{x}} f(\mathbf{x}) \\ &\text{s.t. } \mathbf{x} \in C, \end{aligned}$$

is solved at $\mathbf{x}^*$ if and only if

$$\nabla f(\mathbf{x}^*)^\top (\mathbf{y} - \mathbf{x}^*) \geq 0 \quad \forall \mathbf{y} \in C.$$

Intuitively, this condition states that the gradient increases as we move away (inside the set C) from $\mathbf{x}^*$. We make this argument more formal in the next proposition.

Proposition 35. *The first-order optimality condition of Eq.* (5.1) *is*

$$-\nabla f(\mathbf{x}) \in N_C(\mathbf{x}) := \{\mathbf{w} \in \mathbb{R}^d \mid \mathbf{w}^\top (\mathbf{y} - \mathbf{x}) \leq 0 \; \forall \mathbf{y} \in C\}, \tag{5.5}$$

where $N_C(\mathbf{x})$ is usually called the normal cone. It consists of all directions negatively correlated with $\mathbf{y} - \mathbf{x}$.

Proof. Intuition: Assume that $-\nabla f(\mathbf{x}) \notin N_C(\mathbf{x})$. Then, there exists a direction inside C positively correlated with $-\nabla f(\mathbf{x})$, from which we conclude that $\mathbf{x}$ is not a local (therefore global) minimum.

We make this argument more formal below. First, we restate the objective as follows:

$$\min_{\mathbf{x}} f(\mathbf{x}) + I_C(\mathbf{x}),$$

where $I_C(\mathbf{x}) = \begin{cases} 0 \text{ if } \mathbf{x} \in C \\ \infty \text{ if } \mathbf{x} \notin C. \end{cases}$ By the first-order optimality condition, we have

$$\mathbf{0} \in \partial(f(\mathbf{x}) + I_C(\mathbf{x})).$$

We will use the known result that the subdifferential of the indicator function at $\mathbf{x}$ is normal cone $N_C(\mathbf{x})$.

This implies that

$$\begin{aligned} \mathbf{0} \in \partial(f(\mathbf{x}) + I_C(\mathbf{x})) &\iff \mathbf{0} \in \{\nabla f(\mathbf{x})\} + N_C(\mathbf{x}) \\ &\iff -\nabla f(\mathbf{x}) \in N_C(\mathbf{x}). \end{aligned}$$

□

Computability The geometric optimality conditions are an elegant characterization of local minima. However, in practice, it is not always possible to compute $N_C(\mathbf{x})$. We will therefore introduce constraint qualifications that assure that one can compute the optimality condition. This will give rise to the KKT condition, which we discuss next.

5.3 Karush-Kuhn-Tucker conditions

Given a minimization problem of the form

$$\begin{aligned} \min_{\mathbf{x}} \ & f(\mathbf{x}) \\ \text{s.t. } & h_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m \\ & l_j(\mathbf{x}) = 0, \quad j = 1, \dots, r. \end{aligned} \tag{P}$$

Primal and Dual forms The functions h_i encode inequality constraints, while the functions l_j encode equality constraints. This is what we will call the primal form of the problem, referring to the fact that optimization problems may be viewed from either of two perspectives: primal and dual perspectives. If the primal is a minimization problem then the dual is a maximization problem.

Next, we define the vectors $\mathbf{u} = (u_i) \in \mathbb{R}^m$ and $\mathbf{v} = (v_j) \in \mathbb{R}^r$. We will form what is known as the Lagrangian:

$$L(\mathbf{x}, \mathbf{u}, \mathbf{v}) = f(\mathbf{x}) + \sum_{i=1}^{m} u_i h_i(\mathbf{x}) + \sum_{j=1}^{r} v_j l_j(\mathbf{x}), \tag{5.6}$$

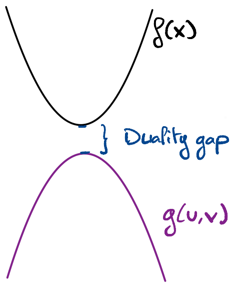
and the Lagrangian dual function:

$$g(\mathbf{u}, \mathbf{v}) = \min_{\mathbf{x}} L(\mathbf{x}, \mathbf{u}, \mathbf{v}). \tag{5.7}$$

The variables $(u_i)_i$ and $(v_j)_j$ are called dual variables. By introducing these dual variables, we have reformulated our constrained optimization problem into an unconstrained system, wherein both the objective function and the constraints are handled equivalently.

We denote by f^* the primal optimal value. The Lagrangian dual g gives a lower bound on f^*:

$$f^* \geq g(\mathbf{u}, \mathbf{v}). \tag{5.8}$$



The subsequent dual problem is given by

$$\begin{aligned} &\max_{\mathbf{u},\mathbf{v}} g(\mathbf{u},\mathbf{v}) \\ &\text{s.t. } \mathbf{u} \geq 0. \end{aligned} \tag{D}$$

We denote by g^* the dual optimal value. Notably, the dual problem is always concave (negative of a convex function), even if the primal problem is not convex. The difference between the primal and dual solutions is called the duality gap. Also note that:

- In the context of an equality constraint, the dual variable has no restrictions placed upon it.
- In the case of an inequality constraint, the dual variable is constrained to be either positive or negative, determined by the sign of the inequality.

Important properties: weak and strong duality

- Weak duality states that the duality gap is always greater than or equal to 0, i.e. $f^* \geq g^*$.
- Strong duality states some conditions in which the primal optimal objective and the dual optimal objective are equal (i.e. zero duality gap). These conditions are known as Slater's conditions and require there exists $\mathbf{x}$ such that $h_i(\mathbf{x}) < 0 \; \forall i$ and $l_j(\mathbf{x}) = 0 \; \forall j$.

We will not be discussing duality in more detail in this course, but we refer the interested reader to Boyd et al. (2004).

KKT conditions The following four groups of conditions have to hold for the solution $\mathbf{x}^*$ to be optimal (first-order necessary conditions):

- Stationarity:
$$\mathbf{0} \in \partial f(\mathbf{x}^*) + \sum_{i=1}^{m} u_i \partial h_i(\mathbf{x}^*) + \sum_{j=1}^{r} v_j \partial l_j(\mathbf{x}^*).$$
- Primal feasibility:
$$\begin{aligned} h_i(\mathbf{x}^*) &\leq 0, \text{ for } i = 1, \ldots, m \\ l_j(\mathbf{x}^*) &= 0, \text{ for } j = 1, \ldots, r. \end{aligned}$$
- Dual feasibility
$$u_i \geq 0, \text{ for } i = 1, \ldots, m.$$
- Complementary slackness
$$u_i h_i(\mathbf{x}^*) = 0, \text{ for } i = 1, \ldots, m.$$

The stationary condition is a first-order optimality for the Lagrangian function. Note that it requires that at the optimum, the gradients of the objective function f and of the constraints h and l balance out. Primal and dual feasibility requires that the solution satisfies the desired constraint. Finally, complementary slackness requires that either the i-th inequality constraint is tight, or its dual variable u_i is zero. Informally, the latter is like a balancing act between the primal and dual problems. If one is "slack" (not fully utilized), then the other must be "tight" (fully utilized) in order for the optimal solution to be reached.

Example Consider the following constrained optimization problem:

$$\begin{aligned}&\min x_1 + x_2\\ &\text{s.t. } x_1^2 + x_2^2 = 1.\end{aligned} \tag{5.9}$$

The Lagrangian is $L(x_1, x_2, v) = x_1 + x_2 + v(x_1^2 + x_2^2 - 1)$. By the stationarity condition, we have

$$\begin{pmatrix} 1 + 2vx_1 \\ 1 + 2vx_2 \end{pmatrix} = \mathbf{0},$$

i.e. $x_1 = x_2 = -\frac{1}{2v}$.

By primal feasibility:

$$x_1^2 + x_2^2 = 1 \implies \frac{1}{4v^2} + \frac{1}{4v^2} = 1 \implies v = \pm\frac{1}{\sqrt{2}}.$$

We conclude that there are two KKT points: $\mathbf{x}^a = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$ and $\mathbf{x}^b = \left(-\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right)$. Among them, $\mathbf{x}^b$ is the minimizer.

Theorem 36. *Given an optimization problem, the following statements are equivalent:*

- $\mathbf{x}^*$ *and* $(\mathbf{u}^*, \mathbf{v}^*)$ *are primal and dual solutions with zero duality gap,*
- $\mathbf{x}^*$ *and* $(\mathbf{u}^*, \mathbf{v}^*)$ *satisfy the KKT conditions.*

Proof. Necessity (the KKT conditions are always sufficient):

Let $\mathbf{x}^*$ be the primal solution and $(\mathbf{u}^*, \mathbf{v}^*)$ be the dual solution with zero duality gap. Then we have

$$\begin{aligned} f(\mathbf{x}^*) &= g(\mathbf{u}^*, \mathbf{v}^*)\\ &= \min_{\mathbf{x}} f(\mathbf{x}) + \sum_{i=1}^{m} u_i^* h_i(\mathbf{x}) + \sum_{j=1}^{r} v_j^* l_j(\mathbf{x})\\ &\le f(\mathbf{x}^*) + \sum_{i=1}^{m} u_i^* h_i(\mathbf{x}^*) + \sum_{j=1}^{r} v_j^* l_j(\mathbf{x}^*)\\ &\le f(\mathbf{x}^*), \end{aligned}$$

where the last inequality follows from the fact that $\mathbf{x}^*$ satisfies the required constraints.

Therefore, the above inequalities are equalities, which implies that:

1. $\mathbf{x}^*$ minimizes $L(\mathbf{x}, \mathbf{u}^*, \mathbf{v}^*)$ and so we must have that $0 \in \partial L(\mathbf{x}, \mathbf{u}^*, \mathbf{v}^*)$. This implies that the stationary condition is satisfied.
2. The second inequality implies that $\sum_{i=1}^m u_i^* h_i(\mathbf{x}^*) + \sum_{j=1}^r v_j^* l_j(\mathbf{x}^*) = 0$, which in turns implies that $\sum_{i=1}^m u_i^* h_i(\mathbf{x}^*) = 0$ since $\sum_{j=1}^r v_j^* l_j(\mathbf{x}^*) = 0$ due to the feasibility of $\mathbf{x}^*$. Each summand is ≤ 0, which implies that $u_i^* h_i(\mathbf{x}^*) = 0$, which is nothing but the complementary slackness condition.
3. Primal and dual feasibility also hold due to the optimality of the solution.

We conclude that the KKT conditions are indeed satisfied.

Sufficiency (the KKT conditions are necessary under strong duality):

We now assume we have a triplet $(\mathbf{x}^*, \mathbf{u}^*, \mathbf{v}^*)$ that satisfies the KKT conditions and we want to show that it is optimal. By primal feasibility and complementary slackness, we have

$$\begin{aligned} g(\mathbf{u}^*, \mathbf{v}^*) &= f(\mathbf{x}^*) + \sum_{i=1}^m u_i^* h_i(\mathbf{x}^*) + \sum_{j=1}^r v_j^* l_j(\mathbf{x}^*) \\ &= f(\mathbf{x}^*). \end{aligned}$$

Therefore, the primal solution and the dual solutions are equal, which means that the duality gap is zero. This guarantees that $\mathbf{x}^*$ is optimal for the primal, and $(\mathbf{u}^*, \mathbf{v}^*)$ is optimal for the dual.

□

5.4 Projections onto convex sets

Example Consider the following problem:

$$\begin{aligned} \underset{\mathcal{X}}{\operatorname{proj}}(\mathbf{y}) = &\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|_2^2 \\ &\text{s.t. } \mathbf{x} \in \mathcal{X}. \end{aligned}$$

$\mathbf{x}^*$ is an optimal solution if and only if

$$0 \in \nabla f(\mathbf{x}^*) + N_{\mathcal{X}}(\mathbf{x}^*) \implies \mathbf{y} - \mathbf{x}^* \in N_{\mathcal{X}}(\mathbf{x}^*), \tag{5.10}$$

i.e. for any $\mathbf{x} \in \mathcal{X}$,

$$\langle \mathbf{y} - \mathbf{x}^*, \mathbf{x} - \mathbf{x}^* \rangle \leq 0. \tag{5.11}$$

As shown in Figure 5.3, the angle between $\mathbf{y} - \mathbf{x}^*$ and $\mathbf{x} - \mathbf{x}^*$ is greater or equal to 90 degrees. This is a special property of convex sets.

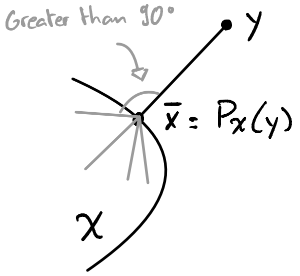


Figure 5.3: Projection onto a convex set $\mathcal{X}$.

Properties of projections Next, we delve into a fundamental characteristic of projections, specifically, that projecting onto convex sets does not lead to an increase in distances.

Proposition 37 (Projection onto convex set is a contraction)**.** *Consider a convex set $C \subset \mathbb{R}^d$ and let $\mathbf{x}_1 = \mathrm{proj}_C(\mathbf{y}_1)$ and $\mathbf{x}_2 = \mathrm{proj}_C(\mathbf{y}_2)$. Then*

$$\|\mathbf{y}_1 - \mathbf{y}_2\|_2 \geq \|\mathbf{x}_1 - \mathbf{x}_2\|_2.$$

Proof. By the optimality condition in Eq. (5.11), we have

$$\langle \mathbf{y}_1 - \mathbf{x}_1, \mathbf{x}_2 - \mathbf{x}_1 \rangle \leq 0$$
$$\langle \mathbf{y}_2 - \mathbf{x}_2, \mathbf{x}_1 - \mathbf{x}_2 \rangle \leq 0.$$

Adding up the last two inequalities, we obtain

$$\begin{aligned}
&\langle \mathbf{y}_1 - \mathbf{x}_1, \mathbf{x}_2 - \mathbf{x}_1 \rangle - \langle \mathbf{y}_2 - \mathbf{x}_2, \mathbf{x}_2 - \mathbf{x}_1 \rangle \leq 0 \\
\Longrightarrow\ &\langle \mathbf{y}_1 - \mathbf{y}_2 - (\mathbf{x}_1 - \mathbf{x}_2), \mathbf{x}_1 - \mathbf{x}_2 \rangle \geq 0 \\
\Longrightarrow\ &\langle \mathbf{y}_1 - \mathbf{y}_2, \mathbf{x}_1 - \mathbf{x}_2 \rangle - \langle \mathbf{x}_1 - \mathbf{x}_2, \mathbf{x}_1 - \mathbf{x}_2 \rangle \geq 0 \\
\Longrightarrow\ &\langle \mathbf{y}_1 - \mathbf{y}_2, \mathbf{x}_1 - \mathbf{x}_2 \rangle \geq \|\mathbf{x}_1 - \mathbf{x}_2\|_2^2.
\end{aligned}$$

By Cauchy-Schwartz, we also have

$$\|\mathbf{y}_1 - \mathbf{y}_2\|_2 \cdot \|\mathbf{x}_1 - \mathbf{x}_2\|_2 \geq \langle \mathbf{y}_1 - \mathbf{y}_2, \mathbf{x}_1 - \mathbf{x}_2 \rangle.$$

By combining the last two inequalities, we conclude that

$$\|\mathbf{y}_1 - \mathbf{y}_2\|_2 \geq \|\mathbf{x}_1 - \mathbf{x}_2\|_2.$$

□

The result of Proposition 37 is illustrated in Figure 5.4.

In the next lecture, we will study an algorithm named Projected Gradient Descent that solves the constrained problem introduced in Eq. (5.1). As its name suggests, it takes a gradient descent step and then applies a projection onto the constrained set.

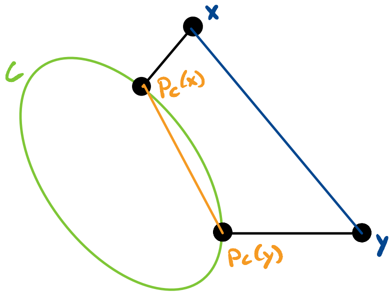


Figure 5.4: Projecting two points $\mathbf{x}$ and $\mathbf{y}$ onto a convex set C does not increase the distance between the points.

5.5 Exercise: Constrained Optimization

Problem 1 (Constrained problem):

Consider the following 2-dimensional problem

$$\begin{aligned} \min f(x,y) &:= x(1-y^2) \\ \text{s.t. } x^2+y^2 &= 1. \end{aligned}$$

1. Write the stationary and primal feasibility conditions.
2. Derive the optimal solution (x^*, y^*).

Problem 2 (KKT problem with two constraints):

Consider the following 3-dimensional problem

$$\begin{aligned} &\min f(x,y,z) := x+y+z \\ &\text{s.t. } x^2-y^2 = 1 \text{ and } 2x+z-1=0. \end{aligned}$$

1. Write the stationary and primal feasibility conditions.
2. Derive all the optimal solutions.
3. Can you comment on the results?

Problem 3 (Projection onto hyperplane):

Consider the projection of a vector onto a hyperplane identified by the equation $\mathbf{Ax} = \mathbf{b}$, i.e.

$$\mathbf{x} = \text{Proj}_{\mathbf{Ax}=\mathbf{b}}(\mathbf{y}) = \underset{\mathbf{x}:\mathbf{Ax}=\mathbf{b}}{\text{argmin}} \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|^2. \tag{5.12}$$

1. Write down the Lagrangian corresponding to the constrained problem defined in Eq. (5.12).

2. Calculate the optimal value of $\mathbf{x}$ (using the KKT conditions). Show that

$$\mathbf{x} = \mathbf{P}\mathbf{y} + \mathbf{A}^\top(\mathbf{A}\mathbf{A}^\top)^{-1}\mathbf{b},$$

   where $\mathbf{P} := (\mathbf{I} - \mathbf{A}^\top(\mathbf{A}\mathbf{A}^\top)^{-1}\mathbf{A})$ is a projection matrix.

Problem 4 (Normal cones):

Consider the following two sets:

$$\Omega_\infty := \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\|_\infty \leq 1\},$$

and

$$\Omega_2 := \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\|_2 \leq 1\}.$$

1. Show that Ω_∞ and Ω_2 are non-empty, convex and closed.
2. Determine the normal cones of Ω_∞ and Ω_2 for $d = 2$ at the point $\mathbf{x} = (1, 0)$.

Chapter 6

Proximal Gradient Descent

6.1 Proximal and Projected Gradient Descent

The main topic for this chapter is proximal gradient descent, which is related to projected gradient descent. In short, proximal gradient descent is a more general method, from which we get projected gradient descent (when choosing the prox operator to be an indicator function, this will become clear later).

6.1.1 Projected subgradient method

Consider the optimization problem

$$\begin{aligned} &\min_{\mathbf{x}} f(\mathbf{x}) \\ &\text{s.t. } \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{6.1}$$

where $\mathcal{X}$ is a convex set and f is a convex function. We assume that $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$ is a bounded subgradient, i.e. $\|\mathbf{g}_k\| \leq G$ for $G > 0$.

Starting from $\mathbf{x}_0 \in \mathcal{X}$, the projected subgradient method updates the iterates as follows:

$$\begin{aligned} \mathbf{y}_{k+1} &= \mathbf{x}_k - \eta \mathbf{g}_k, \quad \mathbf{g}_k \in \partial f(\mathbf{x}_k) \\ \mathbf{x}_{k+1} &= \mathrm{Proj}_{\mathcal{X}}(\mathbf{y}_{k+1}), \end{aligned} \tag{6.2}$$

where $\mathrm{Proj}_{\mathcal{X}}$ is the projection operator onto the set $\mathcal{X}$.

Let's first recall the standard analysis of subgradient method (assuming that f is convex).

Recall: standard analysis subgradient method

$$\begin{aligned}\|\mathbf{x}_{k+1}-\mathbf{x}^*\|^2 &= \|\mathbf{x}_k-\eta\mathbf{g}_k-\mathbf{x}^*\|^2\\ &= \|\mathbf{x}_k-\mathbf{x}^*\|^2-2\eta\mathbf{g}_k^\top(\mathbf{x}_k-\mathbf{x}^*)+\eta^2\|\mathbf{g}_k\|^2\\ &\overset{(i)}{\leq} \|\mathbf{x}_k-\mathbf{x}^*\|^2-2\eta(f(\mathbf{x}_k)-f(\mathbf{x}^*))+\eta^2G^2,\end{aligned}$$

where (i) holds due to the definition of subgradients, and the bounded assumption $\|\mathbf{g}_k\|\leq G$.
By rearranging, we get

$$f(\mathbf{x}_k)-f(\mathbf{x}^*)\leq\frac{1}{2\eta}(\|\mathbf{x}_k-\mathbf{x}^*\|^2-\|\mathbf{x}_{k+1}-\mathbf{x}^*\|^2)+\frac{\eta}{2}G^2. \tag{6.3}$$

We can perform the same analysis as above by replacing $\mathbf{x}_{k+1}$ with $\mathbf{y}_{k+1}$:

$$\begin{aligned}\|\mathbf{y}_{k+1}-\mathbf{x}^*\|^2 &= \|\mathbf{x}_k-\eta\mathbf{g}_k-\mathbf{x}^*\|^2\\ &= \|\mathbf{x}_k-\mathbf{x}^*\|^2-2\eta\mathbf{g}_k^\top(\mathbf{x}_k-\mathbf{x}^*)+\eta^2\|\mathbf{g}_k\|^2\\ &\overset{(i)}{\leq} \|\mathbf{x}_k-\mathbf{x}^*\|^2-2\eta(f(\mathbf{x}_k)-f(\mathbf{x}^*))+\eta^2G^2.\end{aligned}$$

By rearranging, we get

$$f(\mathbf{x}_k)-f(\mathbf{x}^*)\leq\frac{1}{2\eta}(\|\mathbf{x}_k-\mathbf{x}^*\|^2-\|\mathbf{y}_{k+1}-\mathbf{x}^*\|^2)+\frac{\eta}{2}G^2. \tag{6.4}$$

We then use the property that the projection operator does not increase distances, therefore for $\mathbf{x}_{k+1}=\mathrm{Proj}(\mathbf{y}_{k+1})$ and $\mathbf{x}^*=\mathrm{Proj}(\mathbf{x}^*)$, we have

$$\|\mathbf{y}_{k+1}-\mathbf{x}^*\|^2\geq\|\mathbf{x}_{k+1}-\mathbf{x}^*\|^2. \tag{6.5}$$

We can therefore replace $\|\mathbf{y}_{k+1}-\mathbf{x}^*\|^2$ by $\|\mathbf{x}_{k+1}-\mathbf{x}^*\|^2$ in Eq. (6.4) and continue with the analysis as usual. We have

$$f(\mathbf{x}_k)-f(\mathbf{x}^*)\leq\frac{1}{2\eta}(\|\mathbf{x}_k-\mathbf{x}^*\|^2-\|\mathbf{x}_{k+1}-\mathbf{x}^*\|^2)+\frac{\eta}{2}G^2. \tag{6.6}$$

Recall that by Jensen's inequality (and convexity), we have

$$f\left(\sum_{i=1}^k\tfrac{1}{k}\mathbf{x}_i\right)\leq\sum_{i=1}^k\tfrac{1}{k}f(\mathbf{x}_i). \tag{6.7}$$

By combining the last two inequalities (as well as averaging Eq. (6.6) over k iterations, and simplifying the telescoping sum):

$$f\left(\tfrac{1}{k}\sum_{i=1}^k\mathbf{x}_i\right)-f(\mathbf{x}^*)\leq\frac{1}{2\eta k}\|\mathbf{x}_0-\mathbf{x}^*\|^2+\frac{\eta}{2}G^2, \tag{6.8}$$

Choosing $\eta = \frac{1}{\sqrt{k}}$, we obtain

$$f\left(\frac{1}{k}\sum_{i=1}^{k}\mathbf{x}_i\right) - f(\mathbf{x}^*) \leq \frac{1}{2\sqrt{k}}\|\mathbf{x}_0 - \mathbf{x}^*\|^2 + \frac{G^2}{2\sqrt{k}}, \tag{6.9}$$

i.e. we have a convergence of order $\mathcal{O}\left(\frac{1}{\sqrt{k}}\right)$.

6.1.2 Proximap map

Relation projection and prox operators We start from the fact that Eq. (6.1) can be written as

$$\min_{\mathbf{x}} f(\mathbf{x}) + I_{\mathcal{X}}(\mathbf{x}), \tag{6.10}$$

where $I_{\mathcal{X}}(\mathbf{x})$ is the indicator function defined as

$$I_{\mathcal{X}}(\mathbf{x}) = \begin{cases} +\infty, & \mathbf{x} \notin \mathcal{X} \\ 0, & \mathbf{x} \in \mathcal{X}. \end{cases} \tag{6.11}$$

Equivalently, we can define the projection operator as

$$\text{Proj}_{\mathcal{X}}(\mathbf{x}) = \operatorname*{argmin}_{\mathbf{u}} I_{\mathcal{X}}(\mathbf{u}) + \frac{1}{2}\|\mathbf{u} - \mathbf{x}\|_2^2. \tag{6.12}$$

The second term penalizes the distance to $\mathbf{x}$, trying to ensure that the projection is close to it.

Next, we will see a generalization of the above definition where $I_{\mathcal{X}}(\mathbf{u})$ is replaced by another convex function. This is the so-called proximal operator or mapping (abbreviated by prox).

Definition 31 (Proximal mapping (Moreau, 1965))**.** *Given a convex function h, we define the proximal operator as*

$$\mathit{prox}_h(\mathbf{x}) := \operatorname*{argmin}_{\mathbf{u}} h(\mathbf{u}) + \frac{1}{2}\|\mathbf{u} - \mathbf{x}\|_2^2. \tag{6.13}$$

Examples If we choose $h(\mathbf{x}) = \delta(\mathbf{x})$ (dirac function) we get back the projected operator, and if we choose $h(\mathbf{x}) = \eta\|\mathbf{x}\|_1$, we get the soft-thresholding operator $T_\eta(\mathbf{x}) := \text{sign}(\mathbf{x}) \odot [|\mathbf{x}| - \eta\mathbf{1}_d]_+$ for all $\mathbf{x} \in \mathbb{R}^d$, where $\odot$ denotes the componentwise product.

Indeed, for $h(\mathbf{x}) = \eta\|\mathbf{x}\|_1$, we know that by the first-order optimality condition:

$$0 \in \partial\|\mathbf{u}\|_1 + \frac{1}{\eta}(\mathbf{u} - \mathbf{x}) \implies \mathbf{x} - \mathbf{u} \in \eta\partial\|\mathbf{u}\|_1.$$

Recall that $\mathbf{z} \in \partial\|\mathbf{y}\|_1$ if $z_i = \begin{cases} \text{sign}(y_i) & \text{if } y_i \neq 0 \\ \in [-1, 1] & \text{if } y_i = 0. \end{cases}$

Therefore, $\mathbf{x} - \mathbf{u} \in \eta \partial \|\mathbf{u}\|_1$ implies that

$$u_i = \begin{cases} x_i - \eta & \text{if } x_i \geq \eta \\ 0 & \text{if } |x_i| \leq \eta \\ x_i + \eta & \text{if } x_i \leq -\eta. \end{cases}$$

We can see that this operator shrinks the coordinates to zero.

Proximal map: fixed point theory and smoothing Next, we will introduce a tool used in convex analysis named the Moreau envelope that allows for smoothing out a non-smooth convex function f.

We start with a formal definition, which we will later use to show that the proximal operator is closely tied with smoothing and regularization.

Definition 32 (Moreau envelope/Moreau-Yosida regularization)**.** *The Moreau envelope or Moreau-Yosida regularization is given by*

$$M_f(\mathbf{x}) = \inf_{\mathbf{u}} \left\{ f(\mathbf{u}) + \frac{1}{2\lambda} \|\mathbf{u} - \mathbf{x}\|_2^2 \right\}, \tag{6.14}$$

where $\lambda \in \mathbb{R}$ is a chosen parameter.

The smoothing effect of the Moreau envelope is illustrated in Figure 6.1.

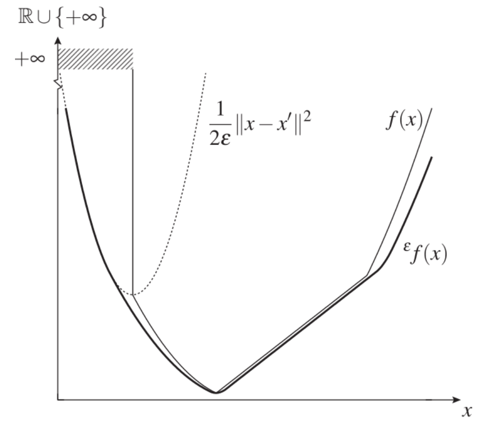


Figure 6.1: Smoothing effect of the Moreau envelope for a simple convex non-smooth function $f : \mathbb{R} \to \mathbb{R}$. $\epsilon_{f(x)} := M_f(x)$ is the Moreau envelope. Source: Kvaal et al. (2014).

Notably, the definition of the Moreau envelope can be rewritten using the proximal operator as

$$M_f(\mathbf{x}) = f(\text{prox}_f(\mathbf{x})) + \frac{1}{2\lambda} \|\text{prox}_f(\mathbf{x}) - \mathbf{x}\|_2^2. \tag{6.15}$$

The smoothing effect of the Moreau regularization can be characterized by the next proposition (see Lemaréchal and Sagastizábal, 1997, for elementary proofs).

Proposition 38 (Regularization properties of the Moreau Envelope (Lin et al., 2018)). *Given a convex continuous function f and a regularization parameter $\kappa = \frac{1}{\lambda} > 0$, consider the Moreau envelope M_f. Then,*

1. *M_f is convex and minimizing f and M_f are equivalent in the sense that*

$$\min_{\mathbf{x}\in\mathbb{R}^d} M_f(\mathbf{x}) = \min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) \; .$$

   Moreover, the solution set of the two above problems coincides with each other.

2. *M_f is continuously differentiable even when f is not and*

$$\nabla M_f(\mathbf{x}) = \kappa(\mathbf{x} - \mathit{prox}_f(\mathbf{x})) \; . \tag{6.16}$$

   Moreover the gradient ∇M_f is Lipschitz continuous with constant $L_{M_f} = \kappa$.

3. *If f is μ-strongly convex, then M_f is μ_{M_f}-strongly convex with $\mu_{M_f} = \frac{\mu\kappa}{\mu+\kappa}$.*

The previous proposition states two important properties that make the Moreau envelope a useful tool in optimization: 1) f and M_f have the same minimizers, and 2) M_f is smooth (therefore it is ”easier” to optimize).

Example Consider $f(x) = \lambda x$ for $x \in \mathbb{R}$, then

$$M_\lambda(x) = \begin{cases} \frac{x^2}{2\lambda}, & |x| \le \lambda \\ |x| - \frac{\lambda}{2}, & |x| > \lambda. \end{cases} \tag{6.17}$$

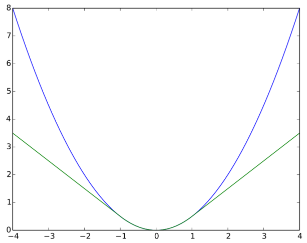


Figure 6.2: Huber loss (green, $\lambda = 1$) and squared error loss (blue). Source: wikipedia

We see that λ acts as a smoothing parameter. Note that this function is also known as the Huber loss function, illustrated in the figure on the right for $\lambda = 1$.

6.2 Proximal Gradient Descent: Convergence analysis

Consider the unconstrained problem with an objective function that is split into two components:

$$\min_{\mathbf{x}} f(\mathbf{x}) := g(\mathbf{x}) + h(\mathbf{x}), \tag{6.18}$$

where

- g is convex and differentiable,

- h is closed [1], convex, possibly non-differentiable, but prox_h is inexpensive to compute.

This sort of decomposition is common in many applications, where g is typically a nice function, while h is not as nice but still simple (such that the proximal can be computed).

The proximal gradient algorithm optimizes the function f by performing the following update:

$$\mathbf{x}_{k+1} = \text{prox}_{\eta h}(\mathbf{x}_k - \eta \nabla g(\mathbf{x}_k)). \tag{6.19}$$

The proximal gradient algorithm can be viewed as gradient descent on the Moreau envelope.

We will analyze the convergence properties of this algorithm. We start with a fundamental contraction property of the proximal operator.

Lemma 39. *Let h be convex, then for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$,*

$$\|prox_h(\mathbf{x}) - prox_h(\mathbf{y})\|_2 \leq \|\mathbf{x} - \mathbf{y}\|_2.$$

Proof. Let $\mathbf{u} = \text{prox}_h(\mathbf{x})$ and $\mathbf{v} = \text{prox}_h(\mathbf{y})$. By first-order optimality,

$$\begin{aligned}
\mathbf{u} = \text{prox}_h(\mathbf{x}) = \underset{\mathbf{u}}{\text{argmin}}\, h(\mathbf{u}) + \frac{1}{2}\|\mathbf{x} - \mathbf{u}\|_2^2 && (6.20)\\
\implies 0 \in \partial h(\mathbf{u}) + (\mathbf{u} - \mathbf{x}) &&\\
\implies (\mathbf{x} - \mathbf{u}) \in \partial h(\mathbf{u}). && (6.21)
\end{aligned}$$

Similarly, we also have $(\mathbf{y} - \mathbf{v}) \in \partial h(\mathbf{v})$.

Since h is convex, ∂h is monotone, i.e. $\langle \partial h(\mathbf{u}) - \partial h(\mathbf{v}), \mathbf{u} - \mathbf{v}\rangle \geq 0$, which implies that

$$\begin{aligned}
&\langle (\mathbf{x} - \mathbf{u}) - (\mathbf{y} - \mathbf{v}), \mathbf{u} - \mathbf{v}\rangle \geq 0\\
\implies &\langle \mathbf{x} - \mathbf{y}, \mathbf{u} - \mathbf{v}\rangle \geq \|\mathbf{u} - \mathbf{v}\|_2^2\\
\implies &\langle \mathbf{x} - \mathbf{y}, \text{prox}_h(\mathbf{x}) - \text{prox}_h(\mathbf{y})\rangle \geq \|\text{prox}_h(\mathbf{x}) - \text{prox}_h(\mathbf{y})\|^2.
\end{aligned}$$

Using the Cauchy–Schwarz inequality, we conclude that

$$\begin{aligned}
&\|\text{prox}_h(\mathbf{x}) - \text{prox}_h(\mathbf{y})\|^2 \leq \langle \mathbf{x} - \mathbf{y}, \text{prox}_h(\mathbf{x}) - \text{prox}_h(\mathbf{y})\rangle \leq \|\mathbf{x} - \mathbf{y}\|\|\text{prox}_h(\mathbf{x}) - \text{prox}_h(\mathbf{y})\|\\
\implies &\|\text{prox}_h(\mathbf{x}) - \text{prox}_h(\mathbf{y})\| \leq \|\mathbf{x} - \mathbf{y}\|.
\end{aligned}$$

□

[1] A function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be closed if for each $\alpha \in \mathbb{R}$, the sublevel set $\{x \in \text{dom} f | f(\mathbf{x}) \leq \alpha\}$ is a closed set.

Proximal mapping From the definition of the proximal operator:

$$\begin{aligned}\mathbf{x}_{k+1} &= \text{prox}_{\eta h}(\mathbf{x}_k - \eta \nabla g(\mathbf{x}_k)) && (6.22)\\ &= \arg\min_{\mathbf{u}} \left(h(\mathbf{u}) + \frac{1}{2\eta} \|\mathbf{u} - \mathbf{x}_k + \eta \nabla g(\mathbf{x}_k))\|^2 \right) \\ &= \arg\min_{\mathbf{u}} \left(h(\mathbf{u}) + g(\mathbf{x}_k) + \nabla g(\mathbf{x}_k)^\top (\mathbf{u} - \mathbf{x}_k) + \frac{1}{2\eta} \|\mathbf{u} - \mathbf{x}_k\|^2 \right). && (6.23)\end{aligned}$$

Therefore $\mathbf{x}_{k+1}$ minimizes $h(\mathbf{u})$ plus a simple quadratic local model of $g(\mathbf{u})$ around $\mathbf{x}_k$. By setting the derivative of Eq. (6.23) to 0, we get:

$$\partial h(\mathbf{u}) + \nabla g(\mathbf{x}_k) + \frac{\mathbf{u} - \mathbf{x}_k}{\eta} := 0 \implies \mathbf{u} = \mathbf{x}_k - \eta(\partial h(\mathbf{u}) + \nabla g(\mathbf{x}_k)). \tag{6.24}$$

We conclude that we can also interpret the prox operator as a gradient step:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta(\partial h(\mathbf{u}) + \nabla g(\mathbf{x}_k)) = \mathbf{x}_k - \eta G_\eta(\mathbf{x}_k), \tag{6.25}$$

where

$$G_\eta(\mathbf{x}_k) = \frac{\mathbf{x}_k - \text{prox}_{\eta h}(\mathbf{x}_k - \eta \nabla g(\mathbf{x}_k))}{\eta}. \tag{6.26}$$

Note that typically $G_\eta(\mathbf{x}_k) \notin \partial f(\mathbf{x}_k)$. However, we know that an optimal solution is a fixed point of the proximal gradient update, i.e. $G_\eta(\mathbf{x}^*) = 0$ if and only if $\mathbf{x}^*$ minimizes $f(\mathbf{x})$. This condition is equivalent to

$$\begin{aligned}&\frac{\mathbf{x}^* - \text{prox}_{\eta h}(\mathbf{x}^* - \eta \nabla g(\mathbf{x}^*))}{\eta} = 0\\ \implies &\mathbf{x}^* = \text{prox}_{\eta h}(\mathbf{x}^* - \eta \nabla g(\mathbf{x}^*)).\end{aligned}$$

By the first-order optimality condition in Eq. (6.21) (where $\mathbf{u} = \mathbf{x}^*$ and $\mathbf{x} = \mathbf{x}^* - \eta \nabla g(\mathbf{x}^*)$), the last equality also implies

$$\begin{aligned}&\mathbf{x}_k - \eta \nabla g(\mathbf{x}_k) - \mathbf{x}_k \in \eta \partial h(\mathbf{x}_k)\\ \implies &- \eta \nabla g(\mathbf{x}_k) \in \eta \partial h(\mathbf{x}_k)\\ \implies &0 \in \nabla g(\mathbf{x}_k) + \partial h(\mathbf{x}_k),\end{aligned}$$

i.e. $\mathbf{x}_k$ is a first-order optimal solution of $g(\mathbf{x}) + h(\mathbf{x})$.

Lemma 40 (Proximal Descent Lemma)**.** *Let $f(\mathbf{x}) = g(\mathbf{h}) + h(\mathbf{x})$ and assume that g is β-smooth and μ strongly-convex. Then for all $\mathbf{x}, \mathbf{z} \in \mathbb{R}^d$, $\eta \leq \frac{1}{\beta}$,*

$$f(\mathbf{x} - \eta G_\eta(\mathbf{x})) \leq f(\mathbf{z}) + \langle G_\eta(\mathbf{x}), \mathbf{x} - \mathbf{z} \rangle - \frac{\eta}{2} \|G_\eta(\mathbf{x})\|_2^2 - \frac{\mu}{2} \|\mathbf{x} - \mathbf{z}\|_2^2.$$

Proof. By definition,

$$f(\mathbf{x}-\eta G_\eta(\mathbf{x})) = g(\mathbf{x}-\eta G_\eta(\mathbf{x})) + h(\mathbf{x}-\eta G_\eta(\mathbf{x})).$$

We first upper bound g. Since g is smooth, we have

$$\begin{aligned} g(\mathbf{x}-\eta G_\eta(\mathbf{x})) &\le g(\mathbf{x}) - \eta\langle \nabla g(\mathbf{x}), G_\eta(\mathbf{x}))\rangle + \frac{\eta}{2}\|G_\eta(\mathbf{x})\|^2 \\ &\le g(\mathbf{z}) + \langle \nabla g(\mathbf{z}), \mathbf{x}-\mathbf{z}\rangle - \frac{\mu}{2}\|\mathbf{z}-\mathbf{x}\|^2 - \eta\langle \nabla g(\mathbf{x}), G_\eta(\mathbf{x}))\rangle + \frac{\eta}{2}\|G_\eta(\mathbf{x})\|^2, \end{aligned}$$

where the second inequality is due to the μ strong-convexity of g.

Next, we upper bound h. By the definition of the gradient mapping G_η (Eq. (6.26)),

$$\mathbf{u} := \mathbf{x}-\eta G_\eta(\mathbf{x}) = \mathbf{x} - (\mathbf{x} - \mathrm{prox}_{\eta h}(\mathbf{x}-\eta\nabla g(\mathbf{x}))) = \mathrm{prox}_{\eta h}(\mathbf{x}-\eta\nabla g(\mathbf{x})).$$

Recall that by first-order optimality, we have

$$\mathbf{u} = \mathrm{prox}_h(\mathbf{x}-\eta\nabla g(\mathbf{x})) \implies (\mathbf{x}-\eta\nabla g(\mathbf{x}) - \mathbf{u}) \in \partial h(\mathbf{u}).$$

Therefore,

$$G_\eta(\mathbf{x}) - \nabla g(\mathbf{x}) \in \partial h(\mathbf{x}-\eta G_\eta(\mathbf{x})).$$

By convexity of h (and the definition of a subgradient), we have

$$h(\mathbf{x}-\eta G_\eta(\mathbf{x})) \le h(\mathbf{z}) - \langle G_\eta(\mathbf{x}) - \nabla g(\mathbf{x}), \mathbf{z} - (\mathbf{x}-\eta G_\eta(\mathbf{x}))\rangle. \tag{6.27}$$

By combining the two upper bounds, we have

$$\begin{aligned} f(\mathbf{x}-\eta G_\eta(\mathbf{x})) &\le g(\mathbf{z}) + \langle \nabla g(\mathbf{z}), \mathbf{x}-\mathbf{z}\rangle - \frac{\mu}{2}\|\mathbf{z}-\mathbf{x}\|^2 - \eta\langle \nabla g(\mathbf{x}), G_\eta(\mathbf{x}))\rangle + \frac{\eta}{2}\|G_\eta(\mathbf{x})\|^2 \\ &\quad + h(\mathbf{z}) - \langle G_\eta(\mathbf{x}) - \nabla g(\mathbf{x}), \mathbf{z} - (\mathbf{x}-\eta G_\eta(\mathbf{x}))\rangle \\ &= f(\mathbf{z}) + \langle G_\eta(\mathbf{x}), \mathbf{x}-\mathbf{z}\rangle - \frac{\eta}{2}\|G_\eta(\mathbf{x})\|^2 - \frac{\mu}{2}\|\mathbf{x}-\mathbf{z}\|^2 \end{aligned}$$

□

Taking $\mathbf{z} = \mathbf{x}_k, \mathbf{x} = \mathbf{x}_k$, then the above lemma says that

$$f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k - \eta G(\mathbf{x}_k)) \le f(\mathbf{x}_k) - \frac{\eta}{2}\|G_\eta(\mathbf{x}_k)\|_2^2, \tag{6.28}$$

i.e. we get a descent inequality (similar to the descent lemma we have seen earlier for gradient descent).

Convergence for smooth, convex functions We first derive a proof of convergence for the smooth and convex case.

Theorem 41 (Smooth, convex). *Assume that f is β-smooth and convex, then*

$$f(\mathbf{x}_K) - f(\mathbf{x}^*) \leq \frac{1}{2\eta K}\left(\|\mathbf{x}_1 - \mathbf{x}^*\|^2\right).$$

Proof. Using Lemma 40 with $\mathbf{z} = \mathbf{x}^*, \mathbf{x} = \mathbf{x}_k$, we have

$$f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k - \eta G_\eta(\mathbf{x}_k)) \leq f(\mathbf{x}^*) + \langle G_\eta(\mathbf{x}_k), \mathbf{x}_k - \mathbf{x}^*\rangle - \frac{\eta}{2}\|G_\eta(\mathbf{x}_k)\|_2^2, \tag{6.29}$$

which implies

$$\begin{aligned} f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*) &\leq \langle G_\eta(\mathbf{x}_k), \mathbf{x}_k - \mathbf{x}^*\rangle - \frac{\eta}{2}\|G_\eta(\mathbf{x}_k)\|_2^2 \\ &= \frac{1}{2\eta}\left(\|\mathbf{x}_k - \mathbf{x}^*\|^2 - \|\mathbf{x}_k - \mathbf{x}^* - \eta G_\eta(\mathbf{x}_k)\|^2\right) \\ &= \frac{1}{2\eta}\left(\|\mathbf{x}_k - \mathbf{x}^*\|^2 - \|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2\right). \end{aligned}$$

Note that we have a telescoping sum, therefore

$$\begin{aligned} f(\mathbf{x}_K) - f(\mathbf{x}^*) &\leq \frac{1}{K}\left(\sum_{i=1}^{K} f(\mathbf{x}_i) - f(\mathbf{x}^*)\right) \\ &\leq \frac{1}{2\eta K}\left(\|\mathbf{x}_1 - \mathbf{x}^*\|^2 - \|\mathbf{x}_K - \mathbf{x}^*\|^2\right) \\ &\leq \frac{1}{2\eta K}\left(\|\mathbf{x}_1 - \mathbf{x}^*\|^2\right). \end{aligned}$$

□

Note that we obtained a $\mathcal{O}\left(\frac{1}{K}\right)$ rate of convergence. Next, we will see that we obtain a linear rate of convergence in the strongly-convex case.

Theorem 42 (Smooth, strongly-convex). *Assume that f is β-smooth and μ strongly-convex, then*

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2 \leq (1 - \eta\mu)^k \|\mathbf{x}_1 - \mathbf{x}^*\|^2.$$

Proof. See the exercise sheet.

□

ISTA For $f(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$ and $g(\mathbf{x}) = \lambda\|\mathbf{x}\|_1$ (basis pursuit denoising problem), we get an algorithm known as ISTA (Iterative Shrinkage-Thresholding Algorithm) which is commonly used for solving linear inverse problems such as compressed sensing, image denoising, and deblurring.

Chapter 7

Newton's method

7.1 Newton's method for optimization

Newton's method was originally invented to find the roots of a differentiable univariate function $f : \mathbb{R} \to \mathbb{R}$. Starting from an initial point $x_0 \in \mathbb{R}$, it simply computes the following update iteratively:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}. \tag{7.1}$$

By re-arranging, one can simply show that Eq. (7.1) finds a point where the derivative is zero, i.e. $f'(x_k) = \frac{f(x_k)-0}{x_k - x_{k+1}}$.

This method can also be adapted to optimize twice differentiable function $f : \mathbb{R} \to \mathbb{R}$ by searching for a zero of the derivative of f. The update in Eq. (7.1) then becomes

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}. \tag{7.2}$$

The same principle can be applied to multi-dimensional functions $f : \mathbb{R}^d \to \mathbb{R}$, in which case Newton's update becomes

$$\mathbf{x}_{k+1} = \mathbf{x}_k - [\nabla^2 f(\mathbf{x}_k)]^{-1} \nabla f(\mathbf{x}_k). \tag{7.3}$$

Note that $\mathbf{H}_k := \nabla^2 f(\mathbf{x}_k)$ is a $d \times d$ matrix, which we need to invert so Newton's method is only applicable if the Hessian is invertible.

In the following, we will first analyze Newton's method and show it exhibits a local quadratic convergence rate (i.e. a fast local rate of convergence). There exist two types of analysis: (i) the classical analysis assumes the function has a global Lipschitz Hessian and (2) a self-concordant analysis.

In the classical analysis of Newton's method, the number of steps needed to reach the superlinear convergence zone depends on the condition number $\kappa = \frac{L}{\mu} \geq 1$ (where L is the smoothness constant and μ is the strong-convexity constant of the objective function f), as is the case also for gradient descent; more specifically, it scales quadratically in κ. However, one can show that the convergence of Newton's method is independent of the condition number but to do so, we will have to resort to the machinery of self-concordant functions (a special class of functions with specific curvature properties).

7.2 Classical analysis

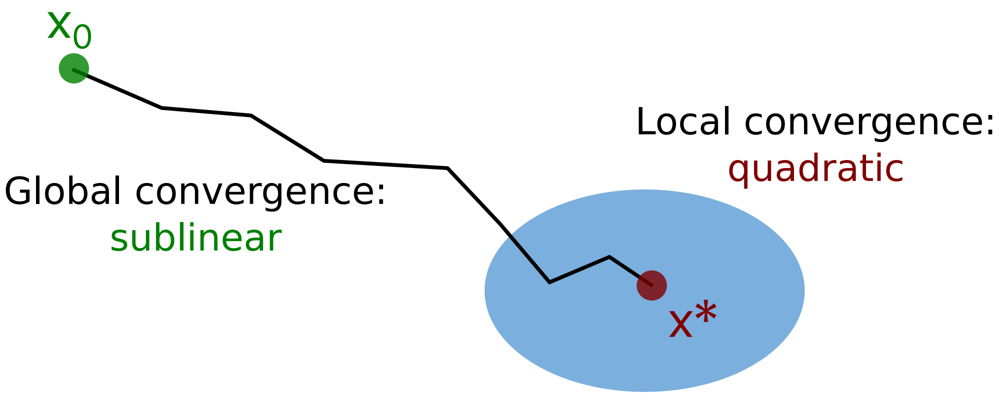


Figure 7.1: Global and local convergence of Newton's method.

We will conduct our analysis under the following assumptions:

Assumption 1. *We assume that:*

- *f is twice differentiable,*
- *The Hessian of f is L-Lipschitz continuous on $\mathbb{R}^d$, i.e.*

$$\left\|\nabla^2 f(\mathbf{x}) - \nabla^2 f(\mathbf{y})\right\| \leq L \left\|\mathbf{x} - \mathbf{y}\right\|, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d, \tag{7.4}$$

- *f is μ strongly-convex.*

We will differentiate between two phases illustrated in Figure 7.1:

- Phase 1 (damped phase):

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta(\nabla^2 f(\mathbf{x}_k))^{-1} \nabla f(\mathbf{x}_k), \tag{7.5}$$

  where the step size η is chosen according to a line-search procedure.

- Phase 2 (undamped phase):

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (\nabla^2 f(\mathbf{x}_k))^{-1} \nabla f(\mathbf{x}_k). \tag{7.6}$$

  Phase 2 will yield a local quadratic rate of convergence (i.e. a very fast rate but it will require being close enough to the optimum). Note that this is achieved using a unit step size $\eta = 1$.

Damped phase In this first phase, the step size is typically chosen using a backtracking line search procedure detailed in Algorithm 1 and illustrated in Figure 7.2. This backtracking approach ensures either that the selected step size is some fixed value (the initial

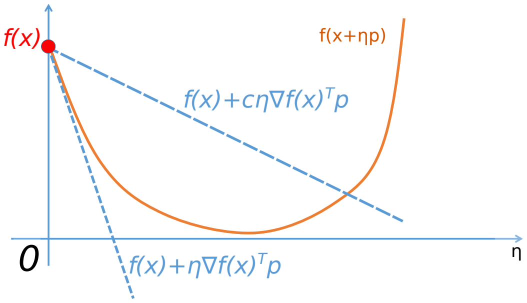


Figure 7.2: The backtracking condition requires that f lies below the upper dashed line. Note that since $\mathbf{p}$ is a descent direction, then $\nabla f(\mathbf{x})^\top \mathbf{p} < 0$.

Algorithm 1 Backtracking line search for a given descent direction $\mathbf{p}_k$.

```
Choose $\bar{\eta}, \rho \in (0,1), c \in (0,1)$
Set $\eta \leftarrow \bar{\eta}$
while $f(\mathbf{x}_k + \eta \mathbf{p}_k) > f(\mathbf{x}_k) + c\eta \nabla f(\mathbf{x}_k)^\top \mathbf{p}_k$ do
    $\eta \leftarrow \rho\eta$    # reduce step size
end while
return $\eta_k = \eta$
```

choice), or else that it is short enough to satisfy a sufficient decrease condition. It starts with a large step size and reduces it until a sufficient decrease condition is satisfied.

In the damped phase, one can use the condition required by the backtracking search to show that $f(\mathbf{x}_k) - f(\mathbf{x}_{k-1}) \leq \gamma$ where $\gamma = \mathcal{O}\left(\frac{\eta^2 \mu}{L^2}\right)$. We will leave this as an exercise for the reader.

Undamped phase We start with a lemma that we will use to prove the main theorem.

Lemma 43. *Assume that $f(\cdot)$ satisfies Assumption 1 and let $\Delta := \|\mathbf{y} - \mathbf{x}\|$, then*

$$\nabla^2 f(\mathbf{x}) - L\Delta\mathbf{I} \preccurlyeq \nabla^2 f(\mathbf{y}) \preccurlyeq \nabla^2 f(\mathbf{x}) + L\Delta\mathbf{I}. \tag{7.7}$$

Proof. The proof follows from Assumption 1 and the notation $\mathbf{A} \succcurlyeq 0$ (which denotes that $\mathbf{A}$ is PSD). □

Next, we state a local convergence result for the undampled phase.

Theorem 44 (Undamped)**.** *Assume that $f(\cdot)$ satisfies Assumption 1 and that $\|\mathbf{x}_k - \mathbf{x}^*\| \leq \frac{2\mu}{3L}$, then*

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \frac{3L}{2\mu} \|\mathbf{x}_k - \mathbf{x}^*\|^2 . \tag{7.8}$$

Proof. We will use the shorthand notation $\nabla^2 f_k := \nabla^2 f(\mathbf{x}_k)$. We consider the Newton step $\mathbf{p}_k = -\nabla^2 f_k^{-1} \nabla f_k$. Then we have

$$\begin{aligned} \mathbf{x}_{k+1} - \mathbf{x}^* = \mathbf{x}_k + \mathbf{p}_k - \mathbf{x}^* &= \mathbf{x}_k - \mathbf{x}^* - \nabla^2 f_k^{-1} \nabla f(\mathbf{x}_k) \\ &= \nabla^2 f_k^{-1} [\nabla^2 f_k(\mathbf{x}_k - \mathbf{x}^*) - (\nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}^*))], \end{aligned} \tag{7.9}$$

where we recall that $\nabla f(\mathbf{x}^*) = 0$.

By the mean-value theorem,

$$\nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}^*) = \int_0^1 \nabla^2 f(\mathbf{x}_k + t(\mathbf{x}^* - \mathbf{x}_k))(\mathbf{x}_k - \mathbf{x}^*) \, \mathrm{d}t. \tag{7.10}$$

We then get

$$\begin{aligned} &\left\|\nabla^2 f_k(\mathbf{x}_k - \mathbf{x}^*) - (\nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}^*))\right\| \\ &= \left\| \int_0^1 [\nabla^2 f(\mathbf{x}_k) - \nabla^2 f(\mathbf{x}_k + t(\mathbf{x}^* - \mathbf{x}_k))](\mathbf{x}_k - \mathbf{x}^*) \, \mathrm{d}t \right\| \\ &\leq \int_0^1 \left\|\nabla^2 f(\mathbf{x}_k) - \nabla^2 f(\mathbf{x}_k + t(\mathbf{x}^* - \mathbf{x}_k))\right\| \|\mathbf{x}_k - \mathbf{x}^*\| \, \mathrm{d}t \\ &\overset{(i)}{\leq} \|\mathbf{x}_k - \mathbf{x}^*\|^2 \int_0^1 Lt \, \mathrm{d}t = \frac{1}{2} L \|\mathbf{x}_k - \mathbf{x}^*\|^2 , \end{aligned} \tag{7.11}$$

where (i) follows from Assumption 1.

Let $\Delta_k := \|\mathbf{x}_k - \mathbf{x}^*\|$. Using Lemma 43, we have

$$\nabla^2 f(\mathbf{x}_k) \succcurlyeq \nabla^2 f(\mathbf{x}^*) - L\Delta_k \mathbf{I} \succcurlyeq \mu \mathbf{I} - L\Delta_k \mathbf{I}. \tag{7.12}$$

Note that if $\Delta_k < \frac{\mu}{L}$ then $\nabla^2 f(\mathbf{x}_k)$ is positive definite.

Combined with the previous equations, we get

$$\begin{aligned} \|\mathbf{x}_{k+1} - \mathbf{x}^*\| &\leq \|\nabla^2 f_k^{-1}\| \cdot \left\|\nabla^2 f_k(\mathbf{x}_k - \mathbf{x}^*) - (\nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}^*))\right\| \\ &\leq \frac{L}{2(\mu - L\Delta_k)} \Delta_k^2. \end{aligned} \tag{7.13}$$

If Δ_k is small enough ($\Delta_k < \frac{2\mu}{3L}$), we have

$$\Delta_{k+1} \leq \frac{3L}{2\mu} \Delta_k^2 \quad (< \Delta_k). \tag{7.14}$$

We conclude that the quadratic rate holds. The inequality in brackets can easily be verified using the fact that $\Delta_k < \frac{2\mu}{3L}$. □

The last theorem shows a local quadratic rate of convergence. However, this rate of convergence still depends on the constants L and μ. Therefore, the theoretic rate of convergence could still be slow for ill-conditioned optimization problems. Next, we will see a different (and more recent) analysis that will prove that Newton's method achieves a faster rate of convergence than gradient descent for ill-conditioned optimization problems. This rate will not only be faster but it will also be independent of the condition number.

7.3 Self-concordance analysis of Newton's method

First, we introduce some notations:

$$\begin{aligned} \|\mathbf{u}\|_{\mathbf{x}} &:= \langle \nabla^2 f(\mathbf{x})\mathbf{u}, \mathbf{u}\rangle^{\frac{1}{2}} \\ \|\mathbf{u}\|_{\mathbf{x}}^* &:= \langle [\nabla^2 f(\mathbf{x})]^{-1}\mathbf{u}, \mathbf{u}\rangle^{\frac{1}{2}}, \end{aligned} \tag{7.15}$$

and

$$\nabla^3 f(\mathbf{x})[\mathbf{u}, \mathbf{u}, \mathbf{u}] = \langle \nabla^3 f(\mathbf{x})[\mathbf{u}]\mathbf{u}, \mathbf{u}\rangle. \tag{7.16}$$

(this notation is analogous to $\nabla f(\mathbf{x})[\mathbf{u}] = \langle \nabla f(\mathbf{x}), \mathbf{u}\rangle$ or $\nabla^2 f(\mathbf{x})[\mathbf{u}, \mathbf{u}] = \langle \nabla^2 f(\mathbf{x})\mathbf{u}, \mathbf{u}\rangle$).

We also define the Dikin ellipsoid of function f at $\mathbf{x}$ as

$$W^0(\mathbf{x}, r) = \{\mathbf{y} \in \mathbb{R}^d \mid \|\mathbf{y} - \mathbf{x}\|_{\mathbf{x}} < r\}. \tag{7.17}$$

We will perform the analysis of Newton's method for a type of function called self-concordant, which is defined below.

Definition 33. *A self-concordant function $f : \mathbb{R}^d \to \mathbb{R}$ is a function that satisfies the following property for any $\mathbf{x} \in \mathbb{R}^d$ and for all $\mathbf{u} \in \mathbb{R}^d$,*

$$|\nabla^3 f(\mathbf{x})[\mathbf{u}, \mathbf{u}, \mathbf{u}]| \leq M_f \|\mathbf{u}\|_{\mathbf{x}}^{3/2}.$$

where $M_f > 0$ is a constant.

The self-concordant assumption will replace the convexity and Lipschitz Hessian assumptions used in the classical analysis discussed in Section 7.2.

For $d = 1$, this definition simply reduces to $|f'''(x)| \leq M_f f''(x)^{3/2}$. Intuitively, self-concordant functions are functions whose rate of change in curvature is bounded by the curvature.

Examples of self-concordant functions

1. Linear functions $f(x) = a \cdot x + b$. Then $f'(x) = a, f''(x) = 0, f'''(x) = 0$, therefore f is self-concordant with $M_f = 0$.

2. The quadratic form $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{A}\mathbf{x} - \mathbf{b}^\top \mathbf{x} + c$, where $\mathbf{A} \in \mathbb{R}^{d\times d}$ is a positive definite matrix, $\mathbf{b} \in \mathbb{R}^d$, and $c \in \mathbb{R}$, is self-concordant with $M_f = 0$.

3. So-called logarithmic barrier functions $f(x) = -\ln x$. Then $f'(x) = \frac{1}{x}, f''(x) = \frac{1}{x^2}, f'''(x) = -\frac{2}{x^3},$, therefore f is self-concordant with $M_f = 2$.

We will need the following lemma that allows us to relate the Hessian matrix at two different points (similarly to the result of Lemma 43).

Lemma 45. *Let* $\mathbf{x} \in \mathbb{R}^d$*. Then for any* $\mathbf{y} \in W^0(\mathbf{x}; 1)$*, we have*

$$(1 - \|\mathbf{y} - \mathbf{x}\|_{\mathbf{x}})^2 \nabla^2 f(\mathbf{x}) \preccurlyeq \nabla^2 f(\mathbf{y}) \preccurlyeq \frac{1}{(1 - \|\mathbf{y} - \mathbf{x}\|_{\mathbf{x}})^2} \nabla^2 f(\mathbf{x}). \tag{7.18}$$

Proof. See Theorem 4.1.6 in Nesterov (2003). □

Corollary 46. *Let* $\mathbf{x} \in \mathbb{R}^d$ *and* $r = \|\mathbf{y} - \mathbf{x}\|_{\mathbf{x}} < 1$*. Define* $\mathbf{D} = \int_0^1 \nabla^2 f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x}))\,\mathrm{d}\tau$*. Then,*

$$\left(1 - r + \frac{r^2}{3}\right) \nabla^2 f(\mathbf{x}) \preccurlyeq \mathbf{D} \preccurlyeq \frac{1}{1 - r} \nabla^2 f(\mathbf{x}).$$

Proof. By Lemma 45:

$$\mathbf{D} = \int_0^1 \nabla^2 f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x}))\,\mathrm{d}\tau \succcurlyeq \nabla^2 f(\mathbf{x}) \int_0^1 (1 - \tau r)^2\,\mathrm{d}\tau = \left(1 - r + \frac{r^2}{3}\right) \nabla^2 f(\mathbf{x}),$$

and

$$\mathbf{D} \preccurlyeq \nabla^2 f(\mathbf{x}) \int_0^1 \frac{\mathrm{d}\tau}{(1 - \tau r)^2} = \frac{1}{1 - r} \nabla^2 f(\mathbf{x}).$$

□

The analysis for self-concordant functions relies on the local norm of the gradient $\lambda_f(\mathbf{x}) := \|\nabla f(\mathbf{x})\|_{\mathbf{x}}^*$ (also called the "Newton decrement" of the function $f(\mathbf{x})$). For small enough $\lambda_f(\mathbf{x})$, we will see that we obtain a quadratic rate of convergence.

Theorem 47. *Assume that* $f(\cdot)$ *is self-concordant. Let* $\mathbf{x} \in \mathbb{R}^d$ *and* $\lambda_f(\mathbf{x}) < 1$*, and consider* $\mathbf{x}^+ = \mathbf{x} - [\nabla^2 f(\mathbf{x})]^{-1} \nabla f(\mathbf{x})$*, then*

$$\lambda_f(\mathbf{x}^+) \leq \left(\frac{\lambda_f(\mathbf{x})}{1 - \lambda_f(\mathbf{x})}\right)^2. \tag{7.19}$$

Proof. The full proof below is from Nesterov (2003), Theorem 4.1.12. Note that it relies on the local bound of the Hessian given in Lemma 45.

Let $\mathbf{p} = \mathbf{x}^+ - \mathbf{x}$ and $\lambda := \lambda_f(\mathbf{x})$, then $\|\mathbf{p}\|_{\mathbf{x}} = \lambda < 1$, and

$$\begin{aligned}\lambda_f(\mathbf{x}^+) &= \langle [\nabla^2 f(\mathbf{x}^+)]^{-1} \nabla f(\mathbf{x}^+), \nabla f(\mathbf{x}^+)\rangle^{1/2} \\ &\overset{(i)}{\leq} \frac{1}{1 - \|\mathbf{p}\|_{\mathbf{x}}} \|\nabla f(\mathbf{x}^+)\|_{\mathbf{x}}^* = \frac{1}{1-\lambda}\|\nabla f(\mathbf{x}^+)\|_{\mathbf{x}}^*, \end{aligned} \tag{7.20}$$

where (i) is due to Lemma 45 applied to the inverse of $\nabla f^2(\mathbf{x}^+)$ (using the fact that if $\mathbf{A} \succcurlyeq \mathbf{B}$, then $\mathbf{B}^{-1} \succcurlyeq \mathbf{A}^{-1}$).

We now derive a bound on $\|\nabla f(\mathbf{x}^+)\|_{\mathbf{x}}^*$ as follows,

$$\begin{aligned}&\nabla f(\mathbf{x}^+) - \nabla f(\mathbf{x}) = \left(\int_0^1 \nabla^2 f(\mathbf{x} + \tau\mathbf{p}) d\tau\right) \mathbf{p} \\ &\Rightarrow \nabla f(\mathbf{x}^+) = \underbrace{\left(\int_0^1 \nabla^2 f(\mathbf{x} + \tau\mathbf{p}) - \nabla^2 f(\mathbf{x}) d\tau\right)}_{\mathbf{G}} \mathbf{p}, \\ &\Rightarrow (\|\nabla f(\mathbf{x}^+)\|_{\mathbf{x}}^*)^2 = \langle [\nabla^2 f(\mathbf{x})]^{-1} \mathbf{G}\mathbf{p}, \mathbf{G}\mathbf{p}\rangle \leq \|\mathbf{Q}\|^2 \|\mathbf{p}\|_{\mathbf{x}}^2, \end{aligned} \tag{7.21}$$

where $\mathbf{Q} := [\nabla^2 f(\mathbf{x})]^{-1/2} \mathbf{G} [\nabla^2 f(\mathbf{x})]^{-1/2}$.

Note that the second term $\|\mathbf{p}\|_{\mathbf{x}}^2$ is a norm w.r.t. $\nabla^2 f(\mathbf{x})$ which is why we have $\nabla^2 f(\mathbf{x})^{-1/2}$ on both sides in $\mathbf{Q}$ (crucial for the next equality so that terms nicely cancel out).

From Corollary 46 applied to $\mathbf{G}$,

$$\left(-\lambda + \frac{1}{3}\lambda^2\right) \nabla^2 f(\mathbf{x}) \preccurlyeq \mathbf{G} \preccurlyeq \frac{\lambda}{1-\lambda} \nabla^2 f(\mathbf{x}), \tag{7.22}$$

therefore

$$\|\mathbf{Q}\| \leq \max\left\{\frac{\lambda}{1-\lambda}, \lambda - \frac{1}{3}\lambda^2\right\} \overset{\lambda \leq 1}{=} \frac{\lambda}{1-\lambda}. \tag{7.23}$$

Combining Eq. (7.20), (7.21) and (7.23), we get

$$\begin{aligned}\lambda_f(\mathbf{x}^+)^2 &\leq \frac{1}{(1-\lambda)^2} (\|\nabla f(\mathbf{x}^+)\|_{\mathbf{x}}^*)^2 \\ &\leq \frac{1}{(1-\lambda)^2} \|\mathbf{Q}\|^2 \|\mathbf{p}\|_{\mathbf{x}}^2 \\ &\leq \frac{\lambda^4}{(1-\lambda)^4}. \end{aligned}$$

□

Based on the above theorem, one can once again check that for λ small enough, we obtain a quadratic rate of convergence.

7.4 Computational Complexity

An obvious drawback of Newton's method is that its update requires computing and inverting the Hessian, which is expensive in high dimensions. Alternatively, we would like to choose a different matrix $\mathbf{B}$ that is close to the Hessian (i.e. $\|\nabla^2 f(\mathbf{x}) - \mathbf{B}\|_2 \leq \epsilon$) but with lower computational complexity. What could it be? In some cases a diagonal approximation $\mathbf{B} = \mathrm{Diag}(\nabla^2 f(\mathbf{x}))$ might work and would be cheaper to compute. Yet another alternative would be a block-diagonal approximation.

In terms of convergence speed, if the approximate matrices $\{\mathbf{B}_k\}_k$ are positive definite with eigenvalues that are uniformly upper and lower bounded, then the convergence rate of the damped phase is preserved (up to constants that depend on the eigenvalues). For the undamped phase, superlinear convergence is achieved if and only if the following Dennis-Moré condition holds:

$$\lim_{k\to\infty} \frac{\|(\mathbf{B}_k - \nabla^2 f(\mathbf{x}_k))\mathbf{B}_k^{-1}\nabla f(\mathbf{x}_k)\|}{\|\mathbf{B}_k^{-1}\nabla f(\mathbf{x}_k)\|} = 0. \tag{7.24}$$

We refer the reader to Nocedal and Wright (1999) (Chapter 3) for a detailed analysis.

7.5 Exercise: Newton's method

Problem 1 (Affine invariance property of Newton's method):

Consider a function $f : \mathbb{R}^d \to \mathbb{R}$ and a non-singular matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$. Let $\mathbf{x} = \mathbf{A}\mathbf{y}$ and $g(\mathbf{y}) = f(\mathbf{A}\mathbf{y})$.

1. Show that the per-step update of Newton's method that minimizes $g(\mathbf{y})$ is equal to
$$\mathbf{y}^+ = \mathbf{y} - \mathbf{A}^{-1}(\nabla^2 f(\mathbf{A}\mathbf{y}))^{-1}\nabla f(\mathbf{A}\mathbf{y}).$$
2. Show that
$$\mathbf{x}^+ = \mathbf{x} - (\nabla^2 f(\mathbf{x}))^{-1}\nabla f(\mathbf{x}).$$
3. What conclusion can you draw from the previous equation?

Hint: Given a composition of functions $g = f(h(\mathbf{y}))$, where $h : \mathbb{R}^d \to \mathbb{R}^p$ and $f : \mathbb{R}^p \to \mathbb{R}$, and such that $\frac{\partial^2 h}{\partial y_i \partial y_j} = 0$, then
$$\nabla^2 g = \mathbf{J}_h^\top \mathbf{H}_f \mathbf{J}_h, \tag{7.25}$$
where $\mathbf{J}_h$ is the Jacobian matrix of h, and $\mathbf{H}_g$ is the Hessian matrix of g.

Problem 2 (Quadratic convergence of Newton's method):

1. Recall the following theorem derived in class.

> **Theorem 48** (Undamped)**.** *Assume that $f(\cdot)$ satisfies the Assumptions seen in class and that $\|\mathbf{x}_k - \mathbf{x}^*\| \le \frac{2\mu}{3L}$, then*
> $$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \le \frac{3L}{2\mu}\|\mathbf{x}_k - \mathbf{x}^*\|^2. \tag{7.26}$$

   Using the above theorem, provide a bound on $\|\mathbf{x}_{k+s} - \mathbf{x}^*\|$.

2. Assuming $\|\mathbf{x}_0 - \mathbf{x}^*\| \le \frac{\mu}{3L}$, prove that $\|\mathbf{x}_k - \mathbf{x}^*\| \le \left(\frac{1}{2}\right)^{2^k-1} \cdot \frac{\mu}{3L}$.
3. Assuming $\|\mathbf{x}_0 - \mathbf{x}^*\| \le \frac{\mu}{3L}$, prove that the Hessian satisfies the following relative error bound:
$$\frac{\|\nabla f^2(\mathbf{x}_k) - \nabla f^2(\mathbf{x}^*)\|}{\|\nabla f^2(\mathbf{x}^*)\|} \le \frac{1}{3}\left(\frac{1}{2}\right)^{2^k-1}.$$

Problem 3 (Convergence in terms of gradient norm):

We optimize a function $f : \mathbb{R}^d \to \mathbb{R}$ using Newton's method that produces the iterates:
$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{p}_k, \tag{7.27}$$
where $\mathbf{p}_k := -[\nabla^2 f(\mathbf{x}_k)]^{-1}\nabla f(\mathbf{x}_k)$.

1. Using the relation $\nabla f(\mathbf{x}_k) + \nabla^2 f(\mathbf{x}_k)\mathbf{p}_k = 0$, prove that

$$\|\nabla f(\mathbf{x}_{k+1})\| \leq \int_0^1 \|\nabla^2 f(\mathbf{x}_k + t\mathbf{p}_k) - \nabla^2 f(\mathbf{x}_k)\| \|\mathbf{p}_k\| \, \mathrm{d}t.$$

2. Since $\nabla^2 f$ is non-singular and Lipschitz continuous, there is a radius $r > 0$ such that $\|\nabla^2 f(\mathbf{x}_k)^{-1}\| \leq 2\|\nabla^2 f(\mathbf{x}^*)^{-1}\|$ for all $\mathbf{x}_k$ such that $\|\mathbf{x}_k - \mathbf{x}^*\| \leq r$. Use this result to prove that

$$\|\nabla f(\mathbf{x}_{k+1})\| \leq 2L\|[\nabla^2 f(\mathbf{x}^*)]^{-1}\|^2 \|\nabla f(\mathbf{x}_k)\|^2 \quad \text{if } \|\mathbf{x}_k - \mathbf{x}^*\| \leq r,$$

i.e. the gradient norm converges to zero quadratically.

Problem 4 (Programming):

Implementation of Undamped Newton's method taught in the class. Fill in the `TODOs` in the given jupyter notebook.

Chapter 8

Stochastic Optimization

In this chapter, we will consider optimization problems where the objective function is given in the form of an expectation, i.e.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \left[f(\mathbf{x}) := \mathbb{E}_\xi f(\mathbf{x}, \xi)\right], \tag{8.1}$$

where ξ is a random vector whose probability distribution $\mathbb{P}$ is supported on a set $\mathcal{P} \subseteq \mathbb{R}^d$, and $f : \mathbb{R}^d \times \mathcal{P}$ is a measurable function [1].

One important difficulty for solving Eq. (8.1) is that the expectation $\mathbb{E}_\xi f(\mathbf{x}, \xi) = \int_{\mathcal{P}} f(\mathbf{x}, \xi) d\mathbb{P}(\xi)$ is a multi-dimensional integral that is typically expensive to compute. In order to circumvent this computational issue, stochastic methods generate independent i.i.d. samples $\xi_1, \xi_2, \dots$ from the distribution $\mathbb{P}$ and use them to compute an estimate of the gradient. Assuming that f is continuous, we have $\nabla_{\mathbf{x}} f(\mathbf{x}) = \mathbb{E}_\xi \partial_{\mathbf{x}} f(\mathbf{x}, \xi)$ (note that we can interchange the expectation and the gradient using Leibniz integral rule). Therefore, one can estimate the expectation $\mathbb{E}_\xi \partial_{\mathbf{x}} f(\mathbf{x}, \xi)$ by an empirical average, and use standard concentration inequalities (e.g. Hoeffding's inequality) to guarantee that the estimated gradient is indeed valid.

In order to simplify the notation, we will use the shortcut $f_\xi(\mathbf{x}) := f(\mathbf{x}, \xi)$ in the remainder of these notes.

Example One concrete example of such an objective is the general empirical risk function that typically appears in machine learning:

$$f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^n f_i(\mathbf{x}), \text{ s.t. } \nabla f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(\mathbf{x}), \tag{8.2}$$

where f_i is a function that depends on a specific datapoint/feature (also called covariate in statistics).

When dealing with Eq. (8.2), the computational complexity of gradient descent is *linear* in the number of samples n, which becomes prohibitive for large data sets (where n can be in the billions). How can we reduce the complexity of this approach? As discussed above, a

[1] Formally, we recall that a measurable function is a function between two measurable spaces, such that the preimage of any measurable set in the codomain is a measurable set in the domain.

simple idea would be to approximate the empirical average over all training instances by an empirical average over a smaller set. In order to retain statistical efficiency, this will require resampling at every update. This is the main principle behind *stochastic* gradient descent.

Stochastic Gradient Descent Based on what we discussed so far, we are ready to introduce the Stochastic Gradient Descent (SGD) algorithm. Starting from an iterate $\mathbf{x}_1$, SGD performs the following iterative update:

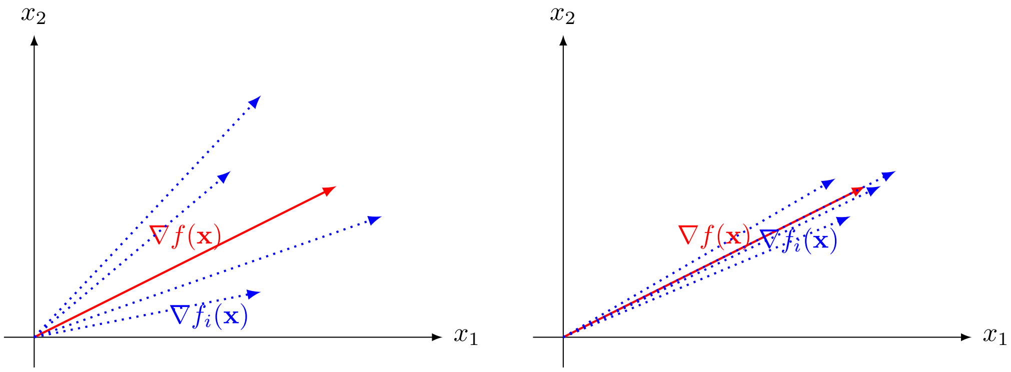
$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta_k \nabla f_{I(k)}(\mathbf{x}_k), \quad I(k) \sim \text{Uniform}\{1, \ldots, n\},$$

where $\eta_k > 0$ is a step size schedule. Note that, unlike gradient descent, the step size η_k typically depends on the iteration number k, although using a constant step size is also possible (we will discuss this shortly).

Note on stochastic processes Since the dynamics of SGD is subject to random fluctuations (or uncertainty in its behavior due to the use of sampling), we say that SGD is a stochastic process. This means that the iterates of SGD $\mathbf{x}_k$ are random variables. In order to analyze the convergence of SGD, we will consider its expected behavior (see chapter on prerequisites).

In our context, we will use the notation $\mathbb{E}[g(\mathbf{x}_k)|\mathbf{x}_{k-1}]$ to mean the conditional expectation of a measurable function g of the random variable $\mathbf{x}_k$ with respect to the σ-algebra of the random variable $\mathbf{x}_{k-1}$, i.e. formally $\mathbb{E}[g(\mathbf{x}_k)|\sigma(\mathbf{x}_{k-1})]$.

In order to simplify the notation and help readers who have less familiarity with measure theory, we will use the notation $\mathbb{E}[\cdot]$ to mean that we take the expectation over all sources of randomness.



(a) Large variance

(b) Small variance

Figure 8.1: Illustration of stochastic gradients in $\mathbb{R}^2$ shown as dotted blue vectors, with their expectation $\nabla f(\mathbf{x})$ shown as a plain red vector.

Bias and Variance The update direction of SGD is *unbiased*, i.e.

$$\mathbb{E}[\nabla f_I(\mathbf{x}_k)|\mathbf{x}_{k-1}] = \sum_{i=1}^{n} \frac{1}{n} \nabla f_i(\mathbf{x}) = \nabla f(\mathbf{x}).$$

Note that the expectation $\mathbb{E}$ in these notes is always taken w.r.t. the data. The stochastic effect of the sampling process can be quantified by the variance function [2]

$$\operatorname{var}(\mathbf{x}) = \mathbb{E}_I \left\| \frac{1}{|I|} \sum_{i \in I} (\nabla f(\mathbf{x}) - \nabla f_i(\mathbf{x})) \right\|^2 .$$

Figure 8.1 illustrates two scenarios with small and large variance. When the variance is small, the stochastic gradients remain tightly concentrated around their expectation, causing SGD to behave similarly to gradient descent. Conversely, when the variance is large, the stochastic gradients deviate more substantially from their expectation. As we will see, this increased variance has significant implications for the convergence of SGD.

Minibatch SGD The term minibatching refers to sampling more than one function f_i (i.e. $|I(k)| = r > 1$). Minibatch SGD gives an unbiased update and reduces the variance. When optimizing complex objective functions (e.g. the ones arising from training the parameters of deep neural networks), selecting the batch size is not trivial as the theory only gives some rough guidelines. We here give a few recommendations that can be followed in practice:

- size r or the minibatch can range from $r = 1$ (single data point) to r in the 100s or 1000s,
- smaller r introduce more noise, yet usually gives better results in the non-convex case,
- choice of r: it depends on the desired performance and the available hardware (e.g. GPU memory).

General idea to prove convergence The general idea can be seen from the following decomposition:

$$\begin{aligned}\mathbb{E}[\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2 \mid \mathbf{x}_k] &= \mathbb{E}[\|\mathbf{x}_k - \eta_k \nabla f_i(\mathbf{x}_k) - \mathbf{x}^*\|^2 \mid \mathbf{x}_k] \\ &= \mathbb{E}[\|\mathbf{x}_k - \mathbf{x}^*\|^2 \mid \mathbf{x}_k] - \underbrace{2\eta_k \mathbb{E}[(\mathbf{x}_k - \mathbf{x}^*)^\top \nabla f_i(\mathbf{x}_k) \mid \mathbf{x}_k]}_{\text{Progress term}} + \underbrace{\eta_k^2 \mathbb{E}[\|\nabla f_i(\mathbf{x}_k)\|^2 \mid \mathbf{x}_k]}_{\text{Variance term}} .\end{aligned}$$

Note that the progress term can be further reduced to

$$\mathbb{E}[(\mathbf{x}_k - \mathbf{x}^*)^\top \nabla f_i(\mathbf{x}_k) \mid \mathbf{x}_k] = (\mathbf{x}_k - \mathbf{x}^*)^\top \mathbb{E}[\nabla f_i(\mathbf{x}_k) \mid \mathbf{x}_k] = (\mathbf{x}_k - \mathbf{x}^*)^\top \nabla f(\mathbf{x}_k). \tag{8.3}$$

We will see that the progress term contributes to decreasing the distance $\mathbb{E}\,\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2$. If f is strongly convex, this can be seen by using the inequality $f(\mathbf{x}^*) - f(\mathbf{x}) \geq \nabla f(\mathbf{x})^\top (\mathbf{x}^* - \mathbf{x}) + \frac{\mu}{2}\|\mathbf{x}^* - \mathbf{x}\|^2$.

On the contrary, the variance term has the opposite effect. One therefore needs to balance these two terms carefully. How? There are several options such as:

[2] Recall the variance of a random vector $\mathbf{x}$ is defined as the trace of the covariance matrix $\operatorname{cov}(\mathbf{x}, \mathbf{x}) = \mathbb{E}\left[(\mathbf{x} - \mathbb{E}[\mathbf{x}])(\mathbf{x} - \mathbb{E}[\mathbf{x}])^\top\right]$.

1. Averaging iterates: although this is common for analysis purposes it is less often used in practice. One type of averaging is called Polyak-Rupert averages:

$$\bar{\mathbf{x}}_{k+1} = \frac{k}{k+1}\bar{\mathbf{x}}_k + \frac{1}{k+1}\mathbf{x}_{k+1},$$

2. Use an appropriate decreasing step size η_k,

3. Use an explicit mechanism to reduce the variance.

8.1 Convergence for strongly-convex functions

Recall the definition of strong convexity:

$$f(\mathbf{x}+\Delta\mathbf{x}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top \Delta\mathbf{x} + \frac{\mu}{2}\|\Delta\mathbf{x}\|^2, \quad \forall \mathbf{x}, \Delta\mathbf{x} \in \mathbb{R}^d.$$

One can check that it implies:

$$\|\mathbf{x}-\mathbf{x}^*\|^2 \leq \frac{2}{\mu}\left(f(\mathbf{x}) - f(\mathbf{x}^*)\right), \quad \forall \mathbf{x} \in \mathbb{R}^d. \tag{8.4}$$

or

$$\begin{aligned} -(\mathbf{x}^*-\mathbf{x})^\top \nabla f(\mathbf{x}) &\geq f(\mathbf{x}) - f(\mathbf{x}^*) + \frac{\mu}{2}\|\mathbf{x}^*-\mathbf{x}\|^2 \\ &\geq \mu\|\mathbf{x}^*-\mathbf{x}\|^2, \quad \forall \mathbf{x} \in \mathbb{R}^d. \end{aligned} \tag{8.5}$$

Assumption We will conduct the analysis for smooth functions that have bounded stochastic gradients, i.e. $\mathbb{E}_i\|\nabla f_i(\mathbf{x})\|^2 \leq \sigma^2$ for some constant $\sigma^2 > 0$ and for all $\mathbf{x} \in \mathbb{R}^d$. Note it is possible to relax this condition to $\mathbb{E}_i\|\nabla f_i(\mathbf{x}) - \nabla f(\mathbf{x})\|^2 \leq \sigma^2$ by dealing with a few extra terms in the analysis.

Constant step-size We first prove convergence for constant step sizes $\eta_k := \eta > 0$.

Theorem 49 (Constant step-size)**.** *Let f be a L-smooth and μ-strongly-convex function. Then SGD with a constant step size η and with bounded gradients ($\mathbb{E}_i\|\nabla f_i(\mathbf{x}_k)\|^2 \leq \sigma^2$) satisfies*

$$\mathbb{E}\|\mathbf{x}_{k+1}-\mathbf{x}^*\|^2 \leq (1-2\eta_k\mu)^k \mathbb{E}\|\mathbf{x}_1-\mathbf{x}^*\|^2 + \frac{\eta\sigma^2}{2\mu}. \tag{8.6}$$

Proof. Using the update equation of SGD, we get

$$\begin{aligned} \mathbb{E}\|\mathbf{x}_{k+1}-\mathbf{x}^*\|^2 &= \mathbb{E}\|\mathbf{x}_k - \eta\nabla f_i(\mathbf{x}_k) - \mathbf{x}^*\|^2 \\ &= \mathbb{E}\|\mathbf{x}_k-\mathbf{x}^*\|^2 - 2\eta\mathbb{E}(\mathbf{x}_k-\mathbf{x}^*)^\top\nabla f_i(\mathbf{x}_k) + \eta^2\mathbb{E}\|\nabla f_i(\mathbf{x}_k)\|^2 \\ &= \mathbb{E}\|\mathbf{x}_k-\mathbf{x}^*\|^2 - \underbrace{2\eta(\mathbf{x}_k-\mathbf{x}^*)^\top\nabla f(\mathbf{x}_k)}_{\text{Progress term}} + \underbrace{\eta^2\mathbb{E}\|\nabla f_i(\mathbf{x}_k)\|^2}_{\text{Variance term}}, \end{aligned}$$

where we used the unbiasedness of the stochastic gradients, i.e. $\mathbb{E}\nabla f_i(\mathbf{x}_k) = \nabla f(\mathbf{x}_k)$.

We can then use Eq. (8.5) to bound the progress term. The variance term will have to be controlled by setting the step size appropriately.

$$\begin{aligned} \mathbb{E}\left\|\mathbf{x}_{k+1} - \mathbf{x}^*\right\|^2 &\overset{(8.5)}{\leq} \mathbb{E}\left\|\mathbf{x}_k - \mathbf{x}^*\right\|^2 - 2\eta\mathbb{E}\mu\left\|\mathbf{x}_k - \mathbf{x}^*\right\|^2 + \eta^2\mathbb{E}\left\|\nabla f_i(\mathbf{x}_k)\right\|^2 \\ &= (1-2\eta\mu)\mathbb{E}\left\|\mathbf{x}_k - \mathbf{x}^*\right\|^2 + \eta^2\mathbb{E}\left\|\nabla f_i(\mathbf{x}_k)\right\|^2 \end{aligned} \tag{8.7}$$

Since the stochastic gradients are bounded, i.e. $\mathbb{E}_i\left\|\nabla f_i(\mathbf{x}_k)\right\|^2 \leq \sigma^2$, we get

$$\begin{aligned} \mathbb{E}\left\|\mathbf{x}_{k+1} - \mathbf{x}^*\right\|^2 &\leq (1-2\eta\mu)\mathbb{E}\left\|\mathbf{x}_k - \mathbf{x}^*\right\|^2 + \eta^2\sigma^2 \\ &\leq (1-2\eta\mu)^k\mathbb{E}\left\|\mathbf{x}_1 - \mathbf{x}^*\right\|^2 + \sum_{i=1}^{k}(1-2\eta\mu)^i\eta^2\sigma^2. \end{aligned} \tag{8.8}$$

Note that the second term on the RHS is a geometric series of the form $\sum_{i=1}^k ar^{i-1} = \frac{a(1-r^k)}{1-r}$. Therefore,

$$\sum_{i=1}^{k}(1-2\eta\mu)^i\eta^2\sigma^2 = \eta^2\sigma^2\frac{1-(1-2\eta\mu)^k}{2\eta\mu} \leq \frac{\eta^2\sigma^2}{2\eta\mu} = \frac{\eta\sigma^2}{2\mu}, \tag{8.9}$$

which in turn implies

$$\mathbb{E}\left\|\mathbf{x}_{k+1} - \mathbf{x}^*\right\|^2 \leq (1-2\eta\mu)^k\mathbb{E}\left\|\mathbf{x}_1 - \mathbf{x}^*\right\|^2 + \frac{\eta\sigma^2}{2\mu}. \tag{8.10}$$

□

Remark 1. *We make the following observations from the result of Theorem 49:*

1. *Convergence is guaranteed up to a non-vanishing constant term that is proportional to the variance σ^2.*
2. *$\sigma^2 = 0$ recovers the fast exponential convergence of gradient descent.*
3. *An appropriate choice of step size (small enough) can get arbitrarily close to the optimum, further details follow next.*

Decreasing step size From the previous theorem, we see that we obtain linear convergence to a ball whose radius is proportional to the step size and the variance. Next, we discuss how one can converge to the optimum at a slower speed by decreasing the step size. The schedule $\eta_k \propto 1/k$ is often theoretically ideal and this is what we will analyze next. We however note that convergence can be proven under the following general conditions:

$$\sum_{k=1}^{\infty} 1/k = \infty, \quad \sum_{k=1}^{\infty} 1/k^2 = \frac{\pi^2}{6} < \infty.$$

Next, we will use the shortcut notation $\mathbf{g}_k := \nabla f_i(\mathbf{x}_k)$ (we omit the dependence on the specific random variable i as it will not play a role in the analysis). Recall that by the μ-strong-convexity of $f(\mathbf{x})$, we have:

$$\langle \mathbf{g}_k, \mathbf{x}_k - \mathbf{x}^* \rangle \geq \mu \|\mathbf{x}_k - \mathbf{x}^*\|_2^2. \tag{8.11}$$

A variant of the following analysis was originally derived by Rakhlin et al. (2011) (which is itself a variation of Nemirovski et al. (2009)). We will proceed by induction: we first start by bounding $\mathbb{E}\|\mathbf{x}_1 - \mathbf{x}^*\|^2$ and we then derive a bound for $\mathbb{E}\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2$.

Lemma 50. *Assume that f is smooth and μ-strongly convex. If $\mathbb{E}\|\mathbf{g}_1\|^2 \leq \sigma^2$, then the first iterate $\mathbf{x}_1$ of SGD satisfies*

$$\mathbb{E}\|\mathbf{x}_1 - \mathbf{x}^*\|_2^2 \leq \frac{\sigma^2}{\mu^2}. \tag{8.12}$$

Proof. From Eq. (8.11), we have:

$$\langle \mathbf{g}_1, \mathbf{x}_1 - \mathbf{x}^* \rangle \geq \mu \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2.$$

Using the Cauchy-Schwarz inequality ($|\langle \mathbf{x}, \mathbf{y} \rangle| \leq \|\mathbf{x}\|\|\mathbf{y}\|$), we get:

$$\|\mathbf{g}_1\|_2^2 \geq \frac{\left(\mu \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2\right)^2}{\|\mathbf{x}_1 - \mathbf{x}^*\|_2^2} = \mu^2 \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2. \tag{8.13}$$

Taking expectation and from the assumption that $\mathbb{E}\|\mathbf{g}_1\|^2 \leq \sigma^2$, we have that:

$$\mathbb{E}\|\mathbf{x}_1 - \mathbf{x}^*\|_2^2 \leq \frac{\sigma^2}{\mu^2}. \tag{8.14}$$

□

Theorem 51 (Decreasing step-size)**.** *Assume that f is smooth and μ-strongly convex. If $\mathbb{E}\|\mathbf{g}_k\|^2 \leq \sigma^2$ for all k, then the iterates of SGD with a decreasing step size $\eta_k = \frac{1}{\mu k}$ satisfy*

$$\mathbb{E}\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 \leq \frac{2\sigma^2}{\mu^2 k}. \tag{8.15}$$

Proof. By the update rule of SGD, we have

$$\begin{aligned}
\mathbb{E}\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 &= \mathbb{E}\|\mathbf{x}_k - \eta_k \mathbf{g}_k - \mathbf{x}^*\|_2^2 \\
&= \mathbb{E}\|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - 2\eta_k \mathbb{E}\langle \mathbf{g}_k, \mathbf{x}_k - \mathbf{x}^* \rangle + (\eta_k)^2 (\mathbb{E}\|\mathbf{g}_k\|_2^2) \\
&\leq \mathbb{E}\|\mathbf{x}_k - \mathbf{x}^*\|_2^2 - 2\eta_k \mu \mathbb{E}\|\mathbf{x}_k - \mathbf{x}^*\|_2^2 + (\eta_k)^2 \sigma^2 \\
&= (1 - 2\eta_k \mu) \mathbb{E}\|\mathbf{x}_k - \mathbf{x}^*\|_2^2 + (\eta_k)^2 \sigma^2.
\end{aligned} \tag{8.16}$$

Using the above inequality, we see that for $k = 1$,

$$\begin{aligned}\mathbb{E}\|\mathbf{x}_2 - \mathbf{x}^*\|_2^2 &\leq (1 - 2\eta_1\mu)\mathbb{E}\|\mathbf{x}_1 - \mathbf{x}^*\|_2^2 + (\eta_1)^2\sigma^2 \\ &= -\mathbb{E}\|\mathbf{x}_1 - \mathbf{x}^*\|_2^2 + \frac{\sigma^2}{\mu^2} \leq \frac{\sigma^2}{\mu^2}.\end{aligned}$$

By applying inequality (8.16) recursively:

$$\begin{aligned}\mathbb{E}\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 &\leq (1 - 2\eta_k\mu)\mathbb{E}\|\mathbf{x}_k - \mathbf{x}^*\|_2^2 + (\eta_k)^2\sigma^2 \\ &\leq (1 - 2\eta_k\mu)((1 - 2\eta_{k-1}\mu)\mathbb{E}\|\mathbf{x}_{k-1} - \mathbf{x}^*\|_2^2 + (\eta_{k-1})^2\sigma^2) + (\eta_k)^2\sigma^2 \\ &\leq \left(\prod_{i=2}^{k}(1 - 2\eta_i\mu)\right)\mathbb{E}\|\mathbf{x}_2 - \mathbf{x}^*\|_2^2 + \sum_{i=2}^{k}\prod_{j=i+1}^{k}(1 - 2\eta_j\mu)(\eta_i)^2\sigma^2.\end{aligned}$$

Plugging in $\eta_i = \frac{1}{\mu i}$, we get:

$$\begin{aligned}\mathbb{E}\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 &\leq \prod_{i=2}^{k}\left(1 - \frac{2}{i}\right)\mathbb{E}\|\mathbf{x}_2 - \mathbf{x}^*\|_2^2 + \sum_{i=2}^{k}\prod_{j=i+1}^{k}\left(1 - \frac{2}{j}\right)\left(\frac{1}{i}\right)^2\frac{\sigma^2}{\mu^2} \\ &= 0 + \frac{\sigma^2}{\mu^2}\sum_{i=2}^{k}\prod_{j=i+1}^{k}\left(1 - \frac{2}{j}\right)\left(\frac{1}{i}\right)^2.\end{aligned}$$

Then, note that

$$\prod_{j=i+1}^{k}\left(1 - \frac{2}{j}\right) = \prod_{j=i+1}^{k}\left(\frac{j-2}{j}\right) = \frac{(i-1)i}{(k-1)k}, \tag{8.17}$$

and therefore

$$\sum_{i=2}^{k}\frac{1}{i^2}\prod_{j=i+1}^{k}\left(1 - \frac{2}{j}\right) = \sum_{i=2}^{k}\frac{(i-1)}{i(k-1)k} \leq \frac{1}{k}, \tag{8.18}$$

By combining Eq. (8.17) with Eq. (8.18), we then get:

$$\mathbb{E}\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2 \leq \frac{\sigma^2}{\mu^2 k}. \tag{8.19}$$

□

Remark 2 (Step size schedule)**.** *We note that the theoretical schedule for the step size is not always what is used in practice. Instead, one typically relies on empirical observations and adjusts the step size accordingly. If slow convergence is observed, then the schedule would be adjusted to have a slower decay. One popular alternative is to use a constant step size at the beginning and then reduce the step size afterward. In recent years, more complex cyclic step-size schedules have also been explored both theoretically and empirically.*

8.2 Convergence for smooth functions with bounded gradients

In the non-convex case, local convergence is typically measured in terms of a vanishing gradient norm (i.e. first-order criticality). In the stochastic case, we naturally extend this notion by taking an expectation over the randomness of the datapoint sampling, i.e.

$$\mathbb{E}\|\nabla f(\mathbf{x})\| \leq \epsilon \tag{ϵ-stationarity}$$

The following result derived in Ghadimi and Lan (2013); Reddi et al. (2016) demonstrates convergence in terms of the expected norm of the gradient, i.e. we reach a first-order critical point in expectation.

Theorem 52. *Assume that f is L-smooth and the stochastic gradients are bounded by a constant σ. Let $\eta_k = \frac{c}{\sqrt{K}}$ with $c = \sqrt{\frac{2(f(\mathbf{x}_1)-f^*)}{L\sigma^2}}$ and K being the total number of iterations, then the iterates of SGD satisfy*

$$\min_{s\in[1,K]} \mathbb{E}\left\|\nabla f(\mathbf{x}_s)\right\|^2 \leq \sqrt{\frac{2(f(\mathbf{x}_1)-f^*)L}{K}}\sigma. \tag{8.20}$$

Proof. Since the function f is L-smooth, we get

$$\begin{aligned}
\mathbb{E}f(\mathbf{x}_{k+1}) &\leq \mathbb{E}\left[f(\mathbf{x}_k) + \langle\nabla f(\mathbf{x}_k), \mathbf{x}_{k+1}-\mathbf{x}_k\rangle + \frac{L}{2}\left\|\mathbf{x}_{k+1}-\mathbf{x}_k\right\|^2\right] \\
&\leq \mathbb{E}f(\mathbf{x}_k) - \eta\mathbb{E}\|\nabla f(\mathbf{x}_k)\|^2 + \frac{L\eta^2}{2}\mathbb{E}\left\|\nabla f_i(\mathbf{x}_k)\right\|^2 \\
&\leq \mathbb{E}f(\mathbf{x}_k) - \eta\mathbb{E}\left\|\nabla f(\mathbf{x}_k)\right\|^2 + \frac{L\eta^2}{2}\sigma^2,
\end{aligned} \tag{8.21}$$

where the second inequality follows from the unbiasness of the stochastic gradients, i.e. $\mathbb{E}_i\nabla f_i(\mathbf{x}) = \nabla f(\mathbf{x})$.

By rearranging the terms in the equation above, we get

$$\mathbb{E}\left\|\nabla f(\mathbf{x}_k)\right\|^2 \leq \frac{1}{\eta}\mathbb{E}[f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})] + \frac{L\eta}{2}\sigma^2. \tag{8.22}$$

By summing from $k = 1$ to K,

$$\begin{aligned}
\min_{s\in[1,K]} \mathbb{E}\left\|\nabla f(\mathbf{x}_s)\right\|^2 &\leq \frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left\|\nabla f(\mathbf{x}_k)\right\|^2 \\
&\leq \frac{1}{\eta K}\mathbb{E}[f(\mathbf{x}_1) - f(\mathbf{x}_{K-1})] + \frac{L\eta^2}{2}\sigma^2 \\
&\leq \frac{1}{\eta K}[f(\mathbf{x}_1) - f(\mathbf{x}^*)] + \frac{L\eta^2}{2}\sigma^2 \\
&\leq \frac{1}{\sqrt{K}}\left(\frac{1}{c}[f(\mathbf{x}_1) - f(\mathbf{x}^*)] + \frac{Lc}{2}\sigma^2\right),
\end{aligned} \tag{8.23}$$

where we used the specific choice of step size $\eta_k = \frac{c}{\sqrt{K}}$ in the last inequality.

□

Why do we have to take the minimum in Eq. (8.23)**?** From the first inequality in Eq. (8.23), we see that we proved the average gradient norm goes to zero at a rate of $\mathcal{O}\left(\frac{1}{\sqrt{K}}\right)$. We have seen that it implies that there exists at least one iterate $\mathbf{x} \in \{\mathbf{x}_1, \dots \mathbf{x}_K\}$ that has a small expected gradient. This however does not imply that the gradient norm of the last iterate $\mathbf{x}_K$ is small. This is because the process is stochastic and the norm can in fact increase due to the variance (captured by σ^2). The process oscillates around the critical point. The only way to exactly converge would be to ensure that the variance decreases to zero (by sampling the full gradient, or by other means).

Alternatively to the average or the minimum used in the above theorem, the result can also be stated for an iterate $\mathbf{x}_i$ chosen uniformly from the sequence $\mathbf{x}_s$ since $\mathbb{E}\|\nabla f(\mathbf{x}_i)\|^2 = \frac{1}{|S|}\sum_{s\in S}\mathbb{E}\|\nabla f(\mathbf{x}_s)\|^2$ (Ghadimi and Lan, 2013).

Remark 3 (Bounded gradient assumption)**.** *Theorem 52 assumed that the variance of the stochastic gradients is bounded, i.e.*

$$\mathbb{E}[\|\nabla f_i(\mathbf{x}) - \nabla f(\mathbf{x})\|^2] \leq \sigma^2.$$

Since the stochastic gradients are unbiased, this condition is equivalent to

$$\mathbb{E}[\|\nabla f_i(\mathbf{x})\|^2] \leq \|\nabla f(\mathbf{x})\|^2 + \sigma^2.$$

(Khaled and Richtárik, 2020) showed that one can obtain the same rate of convergence under a more general condition that requires a global lower bound $f^{\inf}$ on the function f (rather than a global minimizer). This condition is as follows:

$$\mathbb{E}[\|\nabla f_i(\mathbf{x})\|^2] \leq 2A(f(\mathbf{x}) - f^{\inf}) + B\|\nabla f(\mathbf{x})\|^2 + C, \quad \forall\, \mathbf{x} \in \mathbb{R}^d, \tag{8.24}$$

where $A, B, C \geq 0$ are some constants.

8.3 Summary

We summarize the rates of convergence of stochastic gradient descent for various types of smooth functions in Table 9.1. For the non-convex case, note that Arjevani et al. (2019) showed that the rate $\mathcal{O}(1/\sqrt{K})$ is optimal in the worst-case. However, Fang et al. (2019) showed that for functions that are second-order smooth, a variant of SGD can achieve a convergence rate of $\mathcal{O}(\text{polylog}(d)/K^{4/7})$.

8.4 Additional material: how does the variance scale with the minibatch size?

Consider a minibatch B of size b from which we sample without replacement. To simplify the notation, we introduce the shorthand $\delta_i := \nabla f_i(\mathbf{x}) - \nabla f(x)$ and $\delta_B = \nabla f_B(\mathbf{x}) - \nabla f(\mathbf{x})$. It's important to note that this construction ensures $\mathbb{E}_i[\delta_i] = \mathbb{E}_B[\delta_B(\mathbf{x})] = \delta$, implying

Function	Quantity	Rate GD	Rate SGD	SGD Theorem
μ-strongly-convex	$\mathbb{E}\lVert\mathbf{x}_K - \mathbf{x}^*\rVert^2$	$\mathcal{O}((1-\frac{\mu}{L})^K)$	$\mathcal{O}\left(\frac{1}{K}\right)$	Thm. 51
Convex	$\mathbb{E}f(\mathbf{x}_K) - f(\mathbf{x}^*)$	$\mathcal{O}\left(\frac{1}{K}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{K}}\right)$	Moulines and Bach (2011)
μ-PL	$\mathbb{E}f(\mathbf{x}_K) - f(\mathbf{x}^*)$	$\mathcal{O}\left(\left(1-\frac{\mu}{L}\right)^K\right)$	$\mathcal{O}\left(\frac{L}{\mu k^2}\right)$	Karimi et al. (2016)
Non-convex	$\min_{s\in[1,K]} \mathbb{E}\left\lVert\nabla f(\mathbf{x}_s)\right\rVert^2$	$\mathcal{O}\left(\frac{1}{K}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{K}}\right)$	Thm. 52

Table 8.1: Convergence rates of stochastic gradient descent for different function classes (assuming all functions are smooth).

$C^2 = \mathbb{E}_i[\lVert\delta_i\rVert^2]$. This in turn implies that the variance can be bounded as follows,

$$\begin{aligned}
\mathbb{E}_B[\lVert\delta_B\rVert_2^2] &= \mathbb{E}_B\left[\left\lVert\frac{1}{b}\sum_{i\in B}\delta_i\right\rVert_2^2\right] \\
&= \frac{1}{b^2}\mathbb{E}_B\left[\sum_{i,j\in B}\delta_i^\top\delta_j\right] \\
&= \frac{1}{b^2}\mathbb{E}_B\left[\sum_{i\neq j\in B}\delta_i^\top\delta_j\right] + C^2\frac{b}{b} \\
&= \frac{b-1}{bn(n-1)}\sum_{i,j}\delta_i^\top\delta_j + \frac{C^2}{b} \\
&= \frac{b-1}{bn(n-1)}\sum_{i\neq j}\delta_i^\top\delta_j + \frac{C^2}{b} - \frac{C^2}{b}\frac{b-1}{n-1} \\
&= 0 + \frac{C^2}{b}\frac{n-b}{n-1} = \mathcal{O}\left(\frac{1}{b}\right).
\end{aligned}$$

8.5 Exercise: Stochastic Optimization

Problem 1 (Stochastic Gradient Descent):

Consider an objective function with the following finite-sum structure:

$$f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}, \xi_i) \tag{8.25}$$

where ξ_i is the $i-$th random variable (e.g. a datapoint in a given dataset in a machine learning setting). In this case, the computational cost of one GD step scales as $\mathcal{O}(d)$. One, obviously cheaper alternative is to only compute the update based on the gradient of one specific datapoint. This is the updated of *stochastic* gradient descent (SGD), which is arguably the most widely used optimizer in machine learning:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta\nabla f_i(\mathbf{x}_k), \quad i \in \{1,\dots,n\};\ \eta > 0. \tag{8.26}$$

In each iteration, the datapoint i is chosen uniformly at random such that $\mathbb{E}[\nabla f_i(\mathbf{x}_k)] = \nabla f(\mathbf{x}_k)$. We assume that the loss function f is smooth and μ-strongly convex.

1. In this regime, how many samples are left unseen in expectation after one epoch (n iterations)?

2. Show that given $\mathbf{x}_k$ and a constant step size $\eta = \frac{1}{2L}$, SGD does not converge to a critical point $\mathbf{x}^*$, i.e.

$$\mathbb{E}\left[\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_2^2\right] \geq \frac{1}{(2L)^2}\mathbb{E}\left[\|\nabla f_i(\mathbf{x}_k)\|_2^2\right] \tag{8.27}$$

3. Name two possibilities to retain convergence.

Problem 2 (Convergence of the gradient norm):

Under the same finite-sum setting discussed in Problem 1, assume that f is μ-strongly-convex with L-Lipschitz continuous gradients, H-Lipschitz continuous Hessians, and bounded gradients ($\mathbb{E}_i \|\nabla f_i(\mathbf{x}_k)\|^2 \leq B^2$).

1. Show that $g(\mathbf{x}) := \|\nabla f(\mathbf{x})\|^2$ is $\widetilde{L}$-smooth with $\widetilde{L} := 2HB + 2L^2$.

2. Find the expression for the gradient of $g(\mathbf{x})$

3. Show that

$$g(\mathbf{x}_{k+1}) \leq g(\mathbf{x}_k) - 2\eta\langle\nabla^2 f(\mathbf{x}_k)\nabla f(\mathbf{x}_k), \nabla f_i(\mathbf{x}_k)\rangle + \frac{\widetilde{L}}{2}\eta^2 \|\nabla f_i(\mathbf{x}_k)\|^2 . \tag{8.28}$$

4. Using $\mathbb{E}_i \|\nabla f_i(\mathbf{x}_k)\|^2 \leq B^2$, show that

$$\mathbb{E}[g(\mathbf{x}_{k+1})] \leq (1 - 2\eta\mu)^{k+1} g(\mathbf{x}_0) + \underbrace{\sum_{j=0}^{k} (1 - 2\eta\mu)^j \frac{\widetilde{L}}{2}\eta^2 B^2}_{\text{Noise}} . \tag{8.29}$$

5. Bound the noise term and conclude that

$$\mathbb{E}[g(\mathbf{x}_k)] \leq (1-2\mu\eta)^k \, g(\mathbf{x}_0) + \frac{\widetilde{L}\eta}{4\mu} B^2. \tag{8.30}$$

Problem 3 (Convergence for PL functions):

Under the same finite-sum setting discussed in Problem 1, assume that f is μ-PL with L-Lipschitz continuous gradients and bounded gradients ($\|\nabla f_i(\mathbf{x}_k)\|^2 \leq B^2$).

1. Prove that

$$\mathbb{E}[f(\mathbf{x}_{k+1}) - f^*] \leq (1-2\eta_k\mu)[f(\mathbf{x}_k) - f^*] + \frac{LB^2\eta_k^2}{2}.$$

2. Let $\delta_f(k) \equiv k^2\mathbb{E}[f(\mathbf{x}_k) - f^*]$. Using $\eta_k = \frac{2k+1}{2\mu(k+1)^2}$, show that

$$\delta_f(k+1) \leq \delta_f(k) + \frac{LB^2}{2\mu^2},$$

3. Conclude that

$$\mathbb{E}[f(\mathbf{x}_{k+1}) - f^*] \leq \frac{LB^2}{2\mu^2(k+1)}.$$

Problem 4 (Programming exercise):

Write simple SGD code on least-square problem. You should compute the derivatives on paper, then implement them and run the algorithm, finally check the results by plotting the convergence curves. Also use a constant step size so as to see that SGD does not convergence to the minimum. Then compare with decreasing step size.

Chapter 9

Accelerated Gradient Descent

In previous chapters, we have seen the upper bounds in Table 9.1 hold for objective functions that are Lipschitz or smooth. For functions that are Lipschitz, these bounds are optimal (they can not be improved in terms of ϵ). However, we will now see that there is an algorithm that can achieve faster convergence for smooth functions.

	L-Lipschitz	L_2-smooth
Convex	$\left(\frac{RL}{\epsilon}\right)^2$	$\frac{R^2 L_2}{\epsilon}$
μ-strongly-convex	$\frac{L^2}{\mu\epsilon}$	$\kappa \log\left(\frac{R^2 L_2}{\epsilon}\right)$

Table 9.1: Iteration complexity to reach $f(\mathbf{x}_K) - f(\mathbf{x}^*) \leq \epsilon$. $\kappa = \frac{L_2}{\mu}$ is the condition number (always greater than 1).

9.1 Polyak's momentum (Heavy Ball method)

In vanilla gradient descent, each update is based solely on the negative gradient direction at the current point:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \nabla f(\mathbf{x}_k), \tag{9.1}$$

where η is the step size (learning rate).

In the heavy-ball method, we add a "momentum" term to the update:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \nabla f(\mathbf{x}_k) + \beta(\mathbf{x}_k - \mathbf{x}_{k-1}), \tag{9.2}$$

where $\beta \in ([0, 1)$ controls how much of the previous velocity (the difference $\mathbf{x}_k - \mathbf{x}_{k-1}$) is carried forward.

Physical Intuition: A Rolling Ball Imagine you place a heavy ball on a hilly surface. Gravity (analogous to $-\nabla f$) pulls the ball downhill. However, the ball also has inertia or momentum, so once it starts moving, it doesn't instantly stop if the slope flattens out.

We note that Eq. (9.2) is sometimes also written as

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{y}_k - \eta \nabla f(\mathbf{x}_k) \\ \mathbf{y}_k &= (1 + \beta)\mathbf{x}_k - \beta \mathbf{x}_{k-1}. \end{aligned} \tag{9.3}$$

The resulting algorithm is also called the Heavy ball algorithm. It has a provable accelerated rate of convergence on quadratic functions but it is still unknown whether the rate theoretically holds for more general functions.

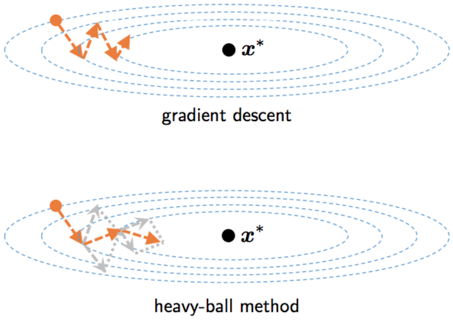


Figure 9.1: (Taken from http://www.princeton.edu/~yc5/ele522_optimization/lectures/accelerated_gradient.pdf)

Intuition (informal): What if the gradient does not change (much)? Let's assume that the gradient is almost constant, i.e. $\nabla f(\mathbf{x}_i) \approx \mathbf{g}$ for all i, then by Eq. (9.2), we have

$$\begin{aligned}
\mathbf{x}_1 - \mathbf{x}_0 &\approx -\eta\mathbf{g} \\
\mathbf{x}_2 - \mathbf{x}_1 &\approx -\eta(1+\beta)\mathbf{g} \\
\mathbf{x}_3 - \mathbf{x}_2 &\approx -\eta(1+\beta(1+\beta))\mathbf{g} = -\eta(1+\beta+\beta^2)\mathbf{g} \\
&\dots
\end{aligned}$$

So that in the limit

$$\lim_{k\to\infty} (\mathbf{x}_k - \mathbf{x}_{k-1}) \approx -\eta\mathbf{g}\sum_{i=0}^{\infty}\beta^i = -\eta\left[\frac{1}{1-\beta}\right]\mathbf{g},$$

such that $\beta = 0.9$ results (potentially) in a 10-fold acceleration.

Figure 9.1 illustrates the effect of the momentum term, where one observes that each step of the algorithm is more consistently moving towards the optimum.

Python code Below is the Python code to implement Heavy-ball momentum on a simple quadratic function. The result is shown in Figure 9.2.

```python
import numpy as np
import matplotlib.pyplot as plt

# ---- Problem setup --------------------------------------------------------
A = np.array([[3.0, 1.0],
              [1.0, 2.0]])

def f(x):
    return 0.5 * x.T @ A @ x

```

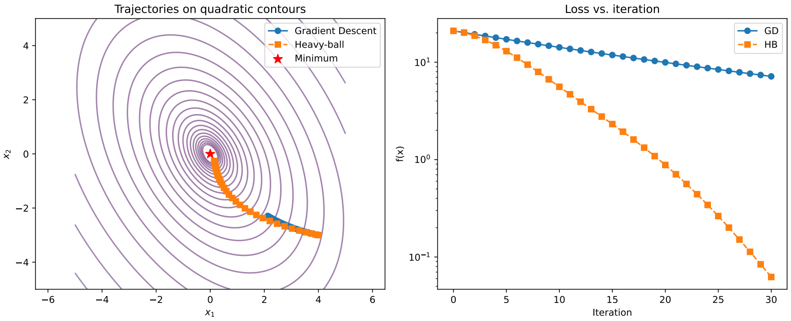


Figure 9.2: Heavy-ball trajectory and loss.

```python
def grad(x):
    return A @ x

alpha, beta, n_iter = 0.01, 0.8, 30
x0 = np.array([4.0, -3.0])

# ---- Gradient Descent ----------------------------------------------------
gd_path = [x0.copy()]
x = x0.copy()
for _ in range(n_iter):
    x = x - alpha * grad(x)
    gd_path.append(x.copy())

# ---- Heavy-ball momentum -------------------------------------------------
hb_path = [x0.copy()]
x_prev, x = x0.copy(), x0.copy()
for k in range(n_iter):
    update = -alpha * grad(x)
    if k > 0:
        update += beta * (x - x_prev)
    x_prev, x = x, x + update
    hb_path.append(x.copy())

gd_path, hb_path = np.array(gd_path), np.array(hb_path)

# ---- Figures ---------------------------------------------------
fig, (ax_ct, ax_loss) = plt.subplots(1, 2, figsize=(12, 5))

# Contours + trajectories (left figure)
x1 = np.linspace(-5, 5, 200)
x2 = np.linspace(-5, 5, 200)
X1, X2 = np.meshgrid(x1, x2)
Z = 0.5 * (A[0,0]*X1**2 + 2*A[0,1]*X1*X2 + A[1,1]*X2**2)

```

```python
levels = np.logspace(-1, 3, 30)
ax_ct.contour(X1, X2, Z, levels=levels, alpha=0.5)
ax_ct.plot(gd_path[:,0], gd_path[:,1], 'o-', label='Gradient Descent')
ax_ct.plot(hb_path[:,0], hb_path[:,1], 's--', label='Heavy-ball')
ax_ct.scatter(0, 0, c='red', marker='*', s=100, label='Minimum')
ax_ct.set_xlabel(r'$x_1$')
ax_ct.set_ylabel(r'$x_2$')
ax_ct.set_title('Trajectories on quadratic contours')
ax_ct.legend()
ax_ct.axis('equal')

# Loss curves (right figure)
gd_loss = [f(p) for p in gd_path]
hb_loss = [f(p) for p in hb_path]

ax_loss.plot(gd_loss, 'o-', label='GD')
ax_loss.plot(hb_loss, 's--', label='HB')
ax_loss.set_yscale('log')
ax_loss.set_xlabel('Iteration')
ax_loss.set_ylabel('f(x)')
ax_loss.set_title('Loss vs. iteration')
ax_loss.legend()

fig.tight_layout()

plt.show()
```

Proof sketch for quadratic functions We will study the convergence of Polyak's momentum for a quadratic function $f : \mathbb{R}^d \to \mathbb{R}$ of the type $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{H}\mathbf{x} - \mathbf{b}^\top \mathbf{x}$ where $\mathbf{b} \in \mathbb{R}^d$ and $\mathbf{H} \in \mathbb{R}^{d \times d}$ is assumed to be non-singular. This function has a unique minimizer $\mathbf{x}^* = \mathbf{H}^{-1}\mathbf{b}$.

The proof strategy is to analyze the composite error vector,

$$\mathbf{w}_k := \begin{bmatrix} \mathbf{x}_k - \mathbf{x}^* \\ \mathbf{x}_{k-1} - \mathbf{x}^* \end{bmatrix} \tag{9.4}$$

and show that the following recursion holds: $\mathbf{w}_{k+1} = \mathbf{G}\mathbf{w}_k$ where $\mathbf{G}$ is a matrix whose form will be defined shortly. We will then bound the maximum eigenvalue of $\mathbf{G}$ to establish convergence.

Note that for the quadratic function defined above, the gradient is equal to

$$\nabla f(\mathbf{x}) = \mathbf{H}\mathbf{x} - \mathbf{b} = \mathbf{H}(\mathbf{x} - \mathbf{x}^*), \tag{9.5}$$

where we used $\mathbf{x}^* = \mathbf{H}^{-1}\mathbf{b}$.

Considering the error vector over two consecutive iterations, we get

$$\begin{bmatrix}\mathbf{x}_{k+1}-\mathbf{x}^* \\ \mathbf{x}_k-\mathbf{x}^*\end{bmatrix} = \begin{bmatrix}(\mathbf{x}_k-\mathbf{x}^*)-\eta\mathbf{H}(\mathbf{x}_k-\mathbf{x}^*)+\beta((\mathbf{x}_k-\mathbf{x}^*)-(\mathbf{x}_{k-1}-\mathbf{x}^*)) \\ \mathbf{x}_k-\mathbf{x}^*\end{bmatrix}$$
$$= \begin{bmatrix}\mathbf{I}-\eta\mathbf{H}+\beta\mathbf{I} & -\beta\mathbf{I} \\ \mathbf{I} & 0\end{bmatrix}\begin{bmatrix}\mathbf{x}_k-\mathbf{x}^* \\ \mathbf{x}_{k-1}-\mathbf{x}^*\end{bmatrix} \tag{9.6}$$

By defining

$$\mathbf{w}_k := \begin{bmatrix}\mathbf{x}_{k+1}-\mathbf{x}^* \\ \mathbf{x}_k-\mathbf{x}^*\end{bmatrix}, \qquad \mathbf{G} := \begin{bmatrix}\mathbf{I}-\eta\mathbf{H}+\beta\mathbf{I} & -\beta\mathbf{I} \\ \mathbf{I} & 0\end{bmatrix}, \tag{9.7}$$

we get a recursion of the form $\mathbf{w}_k = \mathbf{G}\mathbf{w}_{k-1}$. In order to establish convergence, we will need to characterize the spectrum of the matrix $\mathbf{G}$. Indeed, we have

$$\|\mathbf{w}_k\|_2 \le \|\mathbf{G}^k\|_2\|\mathbf{w}_0\|_2, \tag{9.8}$$

where $\|\mathbf{G}\|_2$ is the operator norm defined as

$$\|\mathbf{G}\|_2 = \sup\left\{\frac{\|\mathbf{G}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} : \mathbf{x}\in\mathbb{R}^d \text{ with } \mathbf{x}\neq 0\right\}$$
$$= s_1(\mathbf{G}), \tag{9.9}$$

where $s_1(\mathbf{G})$ is the largest singular value of $\mathbf{G}$.

By Lemma 11 in Foucart (2012), given $\mathbf{A}\in\mathbb{R}^{d\times d}$, and $\epsilon > 0$, there exists a matrix norm $\|\cdot\|$ such that

$$\|\mathbf{A}\| \le \rho(\mathbf{A}) + \epsilon, \tag{9.10}$$

where $\rho(\mathbf{A}) = \max\{|\lambda| : \lambda \text{ eigenvalue of } \mathbf{A}\}$ (spectral radius of $\mathbf{A}$).

Asymptotically [1] (as $k\to\infty$, one can show (see Theorem 12 in Foucart (2012)) that

$$\|\mathbf{w}_k\|_2 = \mathcal{O}(\rho(\mathbf{G})^k). \tag{9.11}$$

At this stage, we therefore need to estimate the eigenvalues of $\mathbf{G}$. To do so, we will proceed by writing the eigenvalue decomposition of $\mathbf{H}$ as $\mathbf{H} = \mathbf{U}\Sigma\mathbf{U}^\top$ where $\Sigma = \operatorname{diag}(\lambda_1,\dots\lambda_d)$ where $\lambda_1 \ge \dots \ge \lambda_d$.

Using a permutation matrix Π [2], we can transform the matrix $\mathbf{G}$ as

$$\mathbf{G} = \begin{bmatrix}\mathbf{G}_1 & 0 & \dots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \dots & \mathbf{G}_d\end{bmatrix}, \tag{9.12}$$

[1] A non-asymptotic version of the analysis can be derived using Theorem 5 by Wang et al. (2021)

[2] The permutation matrix Π is defined as $\Pi_{ij} = \begin{cases}1 & i \text{ odd }, j = i \\ 1 & i \text{ even }, j = 2n+i \\ 0 & \text{else}\end{cases}$. Note that permutation matrices preserve eigenvalues.

where

$$\mathbf{G}_i := \begin{bmatrix} 1+\beta-\eta\lambda_i & -\beta \\ 1 & 0 \end{bmatrix} \tag{9.13}$$

Since the matrix $\mathbf{G}$ is a block diagonal matrix, we have $\|\mathbf{G}\| \leq \max_i \|\mathbf{G}_i\|$. Therefore, the problem is now simplified to bounding the spectral radii of the individual blocks $\mathbf{G}_i$, for $i = 1, 2, \ldots, d$. The two eigenvalues u_1 and u_2 of $\mathbf{G}_i$ are the roots of the quadratic:

$$q(u) := u^2 - (1+\beta-\eta\lambda_i)u + \beta = 0, \tag{9.14}$$

which take different values depending on the discriminant $\Delta := (1+\beta-\eta\lambda_i)^2 - 4\beta$, namely

$$\begin{aligned} u_{1,2} &= \frac{1}{2}(1+\beta-\eta\lambda_i) \pm i\sqrt{(1+\beta-\eta\lambda_i)^2 - 4\beta} \text{ if } \Delta < 0 \\ u_{1,2} &= \frac{1}{2}(1+\beta-\eta\lambda_i) \pm \sqrt{(1+\beta-\eta\lambda_i)^2 - 4\beta} \text{ if } \Delta > 0 \end{aligned} \tag{9.15}$$

Considering the case where $\Delta < 0$, then one can check that the 2 eigenvalues u_1 and u_2 are complex conjugates and are equal in absolute value, i.e.

$$|u_1| = |u_2| = \frac{1}{4}\sqrt{(1+\beta-\eta\lambda_i)^2 + |(1+\beta-\eta\lambda_i)^2 - 4\beta|} = \sqrt{\beta}. \tag{9.16}$$

We here only consider the case $\Delta > 0$ for which the two roots are real. We then need to check for what values of η and β are the eigenvalues of $\mathbf{G}$ less than 1. By choosing these values to explicitly minimize the maximum eigenvalues of $\mathbf{G}$, we get

$$\eta = \frac{4}{L}\frac{1}{(1+1/\sqrt{\kappa})^2} \quad \beta = \left(1 - \frac{2}{\sqrt{\kappa}+1}\right)^2, \tag{9.17}$$

which leads to a linear rate of convergence with rate $\sqrt{\beta} = 1 - \frac{2}{\sqrt{\kappa}+1}$. A detailed analysis can be found in Polyak (1964).

Main takeaway The important part of this analysis is that the rate of convergence is linear (as for gradient descent on strongly-convex functions) but it has a significantly better dependency in terms of the condition number: it depends on $\sqrt{\kappa}$ instead of κ. Recall that the condition number is greater (or equal) than 1, so the appearance of the square root is a significant improvement for ill-conditioned problems with a large κ.

9.2 Accelerated Gradient Descent (AGD)

Another famous accelerated method is known as Nesterov's optimal method whose update step is given by

$$\begin{aligned} \mathbf{y}_k &= \mathbf{x}_k + \gamma(\mathbf{x}_k - \mathbf{x}_{k-1}) \\ \mathbf{x}_{k+1} &= \mathbf{y}_k - \eta\nabla f(\mathbf{y}_k), \end{aligned}$$

where $\gamma > 0$ controls the strength of the momentum term.

Comparison with Heavy ball We illustrate the difference between Heavy ball (Polyak's momentum) and Nesterov Accelerated Gradient (Nesterov's momentum) in Figure 9.3.

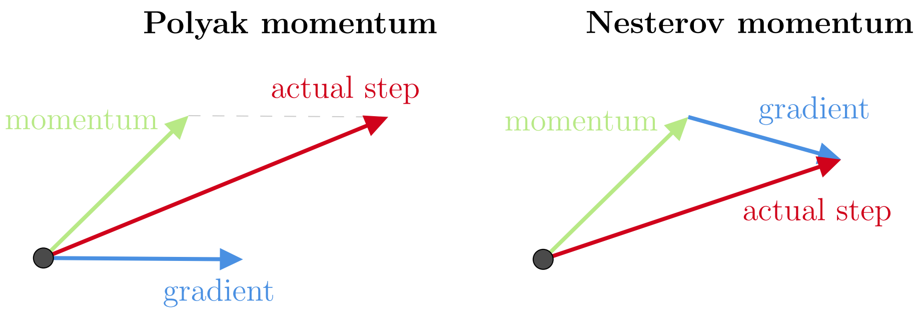


Figure 9.3: Difference between Heavy ball (Polyak's momentum) and Nesterov Accelerated Gradient (Nesterov's momentum).

Convergence rate We will see that for smooth convex and smooth strongly-convex functions, the iteration complexity of AGD is $\mathcal{O}\left(\frac{1}{\sqrt{\epsilon}}\right)$ and $\mathcal{O}\left(\sqrt{\kappa}\log\frac{1}{\epsilon}\right)$ respectively. This is faster than the rates of gradient descent summarized in Table 9.1.

9.2.1 Proof for convex functions

The following result was initially proven by Nesterov (1983) but we follow the proof derived in Beck and Teboulle (2009), see also this blog post: `https://blogs.princeton.edu/imabandit/2013/04/01/acceleratedgradientdescent/`. Although the individual steps followed in the proof are not particularly difficult to follow, the intuition behind the overall proof is difficult to grasp. One can nevertheless recognize a similar telescoping argument to what we have used in earlier proofs of convergence, except that it is now applied to two consecutive iterations (due to the momentum term involving the previous iteration as well).

Theorem 53 (Nesterov (1983))**.** *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a convex and ℓ-smooth function, then Nesterov's Accelerated Gradient Descent satisfies*

$$f(\mathbf{y}_k) - f(\mathbf{x}^*) \leq \frac{2\ell\|\mathbf{x}_1 - \mathbf{x}^*\|^2}{k^2}.$$

To prove this result, we will need the following lemma that was proved earlier in the class.

Lemma 54. *If f has an ℓ-Lipschitz-continuous gradient, then*

$$f(\mathbf{y}) \leq f(\mathbf{x}) + (\mathbf{y}-\mathbf{x})^\top \nabla f(\mathbf{x}) + \frac{\ell}{2}\|\mathbf{y}-\mathbf{x}\|^2 \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{9.18}$$

Proof Theorem 53. By convexity, we have

$$\begin{aligned}
&f\left(\mathbf{x}-\frac{1}{\ell}\nabla f(\mathbf{x})\right)-f(\mathbf{y})\\
&\leq f\left(\mathbf{x}-\frac{1}{\ell}\nabla f(\mathbf{x})\right)-f(\mathbf{x})+\nabla f(\mathbf{x})^\top(\mathbf{x}-\mathbf{y})\\
&\overset{(i)}{\leq}\nabla f(\mathbf{x})^\top\left(\mathbf{x}-\frac{1}{\ell}\nabla f(\mathbf{x})-\mathbf{x}\right)+\frac{\ell}{2}\left\|\mathbf{x}-\frac{1}{\ell}\nabla f(\mathbf{x})-\mathbf{x}\right\|^2+\nabla f(\mathbf{x})^\top(\mathbf{x}-\mathbf{y})\\
&=-\frac{1}{2\ell}\|\nabla f(\mathbf{x})\|^2+\nabla f(\mathbf{x})^\top(\mathbf{x}-\mathbf{y}),
\end{aligned}$$

where (i) uses Lemma 54.

By applying this inequality to $\mathbf{x}=\mathbf{x}_s$ and $\mathbf{y}=\mathbf{y}_s$, we obtain

$$\begin{aligned}
f(\mathbf{y}_{s+1})-f(\mathbf{y}_s)&=f\left(\mathbf{x}_s-\frac{1}{\ell}\nabla f(\mathbf{x}_s)\right)-f(\mathbf{y}_s)\\
&\leq-\frac{1}{2\ell}\|\nabla f(\mathbf{x}_s)\|^2+\nabla f(\mathbf{x}_s)^\top(\mathbf{x}_s-\mathbf{y}_s)\\
&=-\frac{\ell}{2}\|\mathbf{y}_{s+1}-\mathbf{x}_s\|^2-\ell(\mathbf{y}_{s+1}-\mathbf{x}_s)^\top(\mathbf{x}_s-\mathbf{y}_s). \qquad (9.19)
\end{aligned}$$

Similarly we apply it to $\mathbf{x}=\mathbf{x}_s$ and $\mathbf{y}=\mathbf{x}^*$ which gives

$$f(\mathbf{y}_{s+1})-f(\mathbf{x}^*)\leq-\frac{\ell}{2}\|\mathbf{y}_{s+1}-\mathbf{x}_s\|^2-\ell(\mathbf{y}_{s+1}-\mathbf{x}_s)^\top(\mathbf{x}_s-\mathbf{x}^*). \qquad (9.20)$$

Now multiplying Eq. (9.19) by (λ_s-1) and adding the result to Eq. (9.20), we obtain with $\delta_s=f(\mathbf{y}_s)-f(\mathbf{x}^*)$,

$$\lambda_s\delta_{s+1}-(\lambda_s-1)\delta_s\leq-\frac{\ell}{2}\lambda_s\|\mathbf{y}_{s+1}-\mathbf{x}_s\|^2-\ell(\mathbf{y}_{s+1}-\mathbf{x}_s)^\top(\lambda_s\mathbf{x}_s-(\lambda_s-1)\mathbf{y}_s-\mathbf{x}^*).$$

Multiplying this inequality by λ_s and using that by definition $\lambda_{s-1}^2=\lambda_s^2-\lambda_s$ one obtains

$$\begin{aligned}
&\lambda_s^2\delta_{s+1}-\lambda_{s-1}^2\delta_s\\
&\leq-\frac{\ell}{2}\Big(\|\lambda_s(\mathbf{y}_{s+1}-\mathbf{x}_s)\|^2+2\lambda_s(\mathbf{y}_{s+1}-\mathbf{x}_s)^\top(\lambda_s\mathbf{x}_s-(\lambda_s-1)\mathbf{y}_s-\mathbf{x}^*)\Big). \qquad (9.21)
\end{aligned}$$

Now one can verify that

$$\begin{aligned}
&\|\lambda_s(\mathbf{y}_{s+1}-\mathbf{x}_s)\|^2+2\lambda_s(\mathbf{y}_{s+1}-\mathbf{x}_s)^\top(\lambda_s\mathbf{x}_s-(\lambda_s-1)\mathbf{y}_s-\mathbf{x}^*)\\
&=\|\lambda_s\mathbf{y}_{s+1}-(\lambda_s-1)\mathbf{y}_s-\mathbf{x}^*\|^2-\|\lambda_s\mathbf{x}_s-(\lambda_s-1)\mathbf{y}_s-\mathbf{x}^*\|^2. \qquad (9.22)
\end{aligned}$$

Next remark that, by definition, one has

$$\begin{aligned}
&\mathbf{x}_{s+1}=\mathbf{y}_{s+1}+\gamma_s(\mathbf{y}_s-\mathbf{y}_{s+1})\\
&\Leftrightarrow\lambda_{s+1}\mathbf{x}_{s+1}=\lambda_{s+1}\mathbf{y}_{s+1}+(1-\lambda_s)(\mathbf{y}_s-\mathbf{y}_{s+1})\\
&\Leftrightarrow\lambda_{s+1}\mathbf{x}_{s+1}-(\lambda_{s+1}-1)\mathbf{y}_{s+1}=\lambda_s\mathbf{y}_{s+1}-(\lambda_s-1)\mathbf{y}_s. \qquad (9.23)
\end{aligned}$$

Putting together Eqs. (9.21)-(9.23) one gets with $\mathbf{u}_s = \lambda_s \mathbf{x}_s - (\lambda_s - 1)\mathbf{y}_s - \mathbf{x}^*$,

$$\lambda_s^2 \delta_{s+1} - \lambda_{s-1}^2 \delta_s^2 \leq \frac{\ell}{2}\Big(\|\mathbf{u}_s\|^2 - \|\mathbf{u}_{s+1}\|^2\Big).$$

Summing these inequalities from $s = 1$ to $s = k-1$, we obtain:

$$\delta_k \leq \frac{\ell}{2\lambda_{k-1}^2}\|\mathbf{u}_1\|^2.$$

By induction, it is easy to see that $\lambda_{k-1} \geq \frac{k}{2}$ which concludes the proof.

□

Importantly, the bound in Theorem 53 matches the known lower bound, which means that the bound is tight up to constants.

9.2.2 Proof in strongly-convex case

Theorem 55 (Nesterov (1983))**.** *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a μ-strongly-convex and ℓ-smooth function, then Nesterov's Accelerated Gradient Descent with $\eta = \frac{1}{\ell}$ and $\gamma = 1 - \frac{1}{\sqrt{\kappa}}$ satisfies*

$$f(\mathbf{y}_k) - f(\mathbf{x}^*) \leq \left(1 - \frac{1}{\sqrt{\kappa}}\right)\ell\|\mathbf{x}_1 - \mathbf{x}^*\|^2.$$

Proof. See Chapter 3.7 in Bubeck et al. (2015). □

As in the convex case, the bound in the strongly-convex case matches the known lower bound.

Non-convex functions When it comes to acceleration for non-convex functions, the situation becomes more complex compared to convex functions. Non-convex optimization is generally more challenging due to the presence of multiple local minima, saddle points, and other complexities which we will discuss later. Accelerated methods that work well for convex problems may not guarantee similar advantages in the non-convex case. In practice, they are however often used to train the parameters of deep neural networks (despite the non-convexity of this problem).

9.3 Exercise: Accelerated Gradient Descent

Problem 1 (Convergence of GD on quadratic functions):

Consider the quadratic function $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{A}\mathbf{x}$ with $\mathbf{A}$ being symmetric and having bounded minimum and maximum eigenvalues: $\lambda_{\min}(\mathbf{A}) \geq \mu$ and $\|\mathbf{A}\| \leq L$.

1. Show that the gradient descent update can be written as

$$\mathbf{x}_{k+1} = \mathbf{V}(\mathbf{I} - \eta\mathbf{\Lambda})^k \mathbf{V}^\top \mathbf{x}_1,$$

where $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top$ (eigen-decomposition).

2. What can you conclude about the convergence rate as a function of the eigenvalues?

3. Choose $\eta = \frac{1}{L}$. What is the convergence rate of gradient descent?

Problem 2 (Convergence of AGD on quadratic functions):

Consider the quadratic function $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{A}\mathbf{x}$ with $\mathbf{A}$ being symmetric and having bounded minimum and maximum eigenvalues: $\lambda_{\min}(\mathbf{A}) \geq \eta$ and $\|\mathbf{A}\| \leq L$.

1. Write AGD as a recursion of the form $\mathbf{w}_k = \mathbf{G}\mathbf{w}_{k-1}$ and give the explicit form of the $\mathbf{G}$ matrix.

2. Following a similar analysis to Heavy ball (see lecture notes), write $\mathbf{G}$ as a block matrix and derive the eigenvalues of the $\mathbf{G}_i$ matrices that compose the blocks of $\mathbf{G}$.

3. Bonus (more difficult): Derive a rate of convergence for AGD based on your previous calculations. Compare this rate to the rate of GD obtained in Problem 1.

Problem 3 (Lyapunov analysis of AGD):

We optimize a function $f : \mathbb{R}^d \to \mathbb{R}$ that is convex and ℓ-smooth using Accelerated Gradient Descent (AGD) whose update is

$$\begin{aligned}\mathbf{y}_k &= \mathbf{x}_k + \gamma(\mathbf{x}_k - \mathbf{x}_{k-1})\\ \mathbf{x}_{k+1} &= \mathbf{y}_k - \eta\nabla f(\mathbf{y}_k).\end{aligned}$$

Define the following Lyapunov function $L_k := f(\mathbf{x}_k) + \frac{1}{2\eta}\|\mathbf{x}_k - \mathbf{x}_{k-1}\|^2$. Prove that for $\eta \leq \frac{1}{\ell}$ and $\gamma \in [0, 1]$,

$$L_{k+1} - L_k \leq -\frac{1-\gamma^2}{2\eta}\|\mathbf{x}_k - \mathbf{x}_{k-1}\|^2.$$

Problem 4 (Programming exercise):

Fill in the `TODOs` in the associated Jupyter Notebook.

Chapter 10

Adaptive Methods

10.1 Polyak's Step Size

In this chapter, we study the *Polyak step size* for gradient descent, which uses knowledge of the optimal value f^* to adaptively choose the learning rate. We first introduce the algorithm and then study its convergence properties in the convex, L-smooth setting.

10.1.1 Algorithm

The *Polyak step size* is an **adaptive learning-rate rule** for (stochastic) gradient descent that requires *no manual tuning.* At iteration t it sets the step-size as

$$\eta_t = \frac{f(\mathbf{x}_t) - f^*}{\|\nabla f(\mathbf{x}_t)\|^2},$$

i.e. the largest step along the negative gradient that would drive the first-order Taylor model exactly down to the optimum value f^* (exercise: try to prove this claim). Because η_t is computed from quantities already available during the update, it automatically scales to local curvature and noise. Classical analyses show that it achieves the optimal $\mathcal{O}\left(\frac{1}{T}\right)$ convergence for smooth convex objectives, and a linear rate whenever a Polyak–Łojasiewicz or strong-convexity condition holds. Modern variants extend the idea to momentum methods, constraints, and fully stochastic settings (often called `SPS`), see Loizou et al. (2021).

Algorithm 2 Gradient Descent with Polyak Step Size

```
Require: Smooth convex $f : \mathbb{R}^d \to \mathbb{R}$ with minimizer $\mathbf{x}^*$ such that $f(\mathbf{x}^*) = f^*$,
    Lipschitz gradient constant $L$.
Initialize $\mathbf{x}_0 \in \mathbb{R}^d$.
for $t = 0, 1, 2, \dots$ do
    Compute gradient $\mathbf{g}_t = \nabla f(\mathbf{x}_t)$.
    Set
        $$\eta_t = \frac{f(\mathbf{x}_t) - f^*}{\|\mathbf{g}_t\|^2}.$$
    Update
        $$\mathbf{x}_{t+1} = \mathbf{x}_t \; - \; \eta_t \, \mathbf{g}_t.$$
end for
Return $\bar{\mathbf{x}} = \text{argmin}_{t \le T}\{f(\mathbf{x}_t)\}$
```

10.1.2 Convergence Analysis

We will make the following two standard assumptions:

Assumption 2. *The function f is convex and L-smooth:*

$$f(\mathbf{y}) \le f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \tfrac{L}{2}\|\mathbf{y} - \mathbf{x}\|^2, \quad \forall \mathbf{x}, \mathbf{y}.$$

Assumption 3. *The level set $\{\mathbf{x} : f(\mathbf{x}) \le f(\mathbf{x}_0)\}$ has bounded diameter D, i.e. $\|\mathbf{x} - \mathbf{x}^*\| \le D$ whenever $f(\mathbf{x}) \le f(\mathbf{x}_0)$.*

Lemma 56 (Distance recursion (Hazan and Kakade, 2019, Lemma 1))**.** *Let $d_t = \|\mathbf{x}_t - \mathbf{x}^*\|$. For any step size $\eta_t > 0$, the iterates produced by Algorithm 2 satisfy*

$$d_{t+1}^2 \le d_t^2 - 2\eta_t \left(f(\mathbf{x}_t) - f^*\right) + \eta_t^2 \|\mathbf{g}_t\|^2.$$

Proof. Expanding $\|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2$, we get

$$\begin{aligned} d_{t+1}^2 &= \left\| \mathbf{x}_{t+1} - \mathbf{x}^* \right\|^2 \\ &= \left\| \mathbf{x}_t - \eta_t \mathbf{g}_t - \mathbf{x}^* \right\|^2 \\ &= d_t^2 - 2\eta_t \, \mathbf{g}_t^\top \big(\mathbf{x}_t - \mathbf{x}^*\big) + \eta_t^2 \, \|\mathbf{g}_t\|^2 \\ &\le d_t^2 - 2\eta_t (f(\mathbf{x}_t) - f(\mathbf{x}^*)) + \eta_t^2 \, \|\mathbf{g}_t\|^2, \end{aligned}$$

where we used convexity in the last inequality.

□

Lemma 57 (Descent under smoothness)**.** *Assume f is L-smooth. With the Polyak step size $\eta_t = (f(\mathbf{x}_t) - f^*)/\|\mathbf{g}_t\|^2$,*

$$d_{t+1}^2 \le d_t^2 - \frac{(f(\mathbf{x}_t) - f^*)}{2L}.$$

Consequently $\dfrac{1}{T}\displaystyle\sum_{t=0}^{T-1}(f(\mathbf{x}_t) - f^*) \le \dfrac{2L\, d_0^2}{T}$.

Proof sketch. Lemma 56 combined with the definition of η_t gives $d_{t+1}^2 \le d_t^2 - (f(\mathbf{x}_t) - f^*)^2/\|\mathbf{g}_t\|^2$. For a L-smooth function $\frac{1}{2L}\|\mathbf{g}_t\|^2 \le (f(\mathbf{x}_t) - f^*)$ ((Hazan and Kakade, 2019, Eq. (2))), so $(f(\mathbf{x}_t) - f^*)^2/\|\mathbf{g}_t\|^2 \ge (f(\mathbf{x}_t) - f^*)/(2L)$. Sum the inequality over t and telescope. □

Theorem 58 (Best-iterate bound (smooth convex))**.** *After T iterations of Algorithm 2 on a L-smooth convex f, the returned point $\bar{\mathbf{x}}$ satisfies*

$$f(\bar{\mathbf{x}}) - f(\mathbf{x}^*) \le \frac{2L\,\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{T}.$$

Proof. By definition $\bar{\mathbf{x}}$ achieves the minimum value among $\{\mathbf{x}_t\}_{t=0}^{T-1}$, so Lemma 57 yields $f(\bar{\mathbf{x}}) - f(\mathbf{x}^*) \le \frac{1}{T}\sum_{t=0}^{T-1}(f(\mathbf{x}_t) - f^*) \le \frac{2L\,\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{T}$. □

In summary, adaptively selecting the step size yields an $\mathcal{O}\big(1/T\big)$ convergence rate for smooth convex objectives. A caveat is that Polyak's step-size assumes knowledge of the optimal value f^* which can be estimated in some applications, but it is not always accessible in practice.

10.2 Adagrad

A key parameter in first-order methods such as gradient descent is the *step size* (or learning rate). A poorly chosen constant step size can lead to:

- **Slow convergence**, if the step size is too small.
- **Divergence** or oscillation, if the step size is too large.
- **Sensitivity to feature scaling**: different coordinates may have gradients of very different magnitudes.

To address these issues, *adaptive step size* (also called adaptive gradient) methods adjust the learning rate per coordinate based on the history of gradients. This allows:

- Smaller steps in frequently updated directions.
- Larger steps in infrequent or sparse dimensions.

- Reduced need for manual tuning of a global learning rate.

Adagrad Duchi et al. (2011) is one of the earliest and most influential adaptive-gradient algorithms. It adjusts the learning rate coordinate-wise, scaling each step according to the cumulative history of past gradients. This update rule can also be viewed as a diagonal, quasi-second-order precondition, an interpretation we will discuss shortly.

10.2.1 The Adagrad Algorithm

Online convex optimization (OCO). To study the convergence properties of Adagrad, we will use the OCO framework which models a sequential game between a learner and an adversary. On each round, the learner must pick a point inside a known convex decision set $\mathcal{X}$ *before* seeing that round's loss landscape. Immediately after the choice is made, the adversary unveils a convex loss function over $\mathcal{X}$, the learner pays the corresponding loss, and (typically) receives first-order feedback such as a subgradient. The learner's objective is not to minimize the unpredictable losses themselves, but to keep its *regret*—the gap between its cumulative loss and that of the best single fixed decision in hindsight—as small as possible. Achieving sub-linear regret guarantees that, on average, the learner's performance converges to that of the best fixed point, no matter how the losses are chosen.

Definition 34 (Online Convex Optimization)**.** *Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a convex set. At each round $t = 1, 2, \ldots, T$:*

1. *The learner chooses $\mathbf{x}_t \in \mathcal{X}$.*
2. *An adversary reveals a convex loss function $f_t : \mathcal{X} \to \mathbb{R}$.*
3. *The learner incurs loss $f_t(\mathbf{x}_t)$ and observes subgradient $\mathbf{g}_t \in \partial f_t(\mathbf{x}_t)$.*

The goal is to minimize regret

$$R(T) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}).$$

Adagrad: full-matrix version In the full-matrix version of Adagrad, we accumulate the outer product of the gradients:

$$\mathbf{G}_t = \sum_{\tau=1}^{t} \mathbf{g}_\tau \mathbf{g}_\tau^\top$$

Then the parameter update rule becomes:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \cdot (\mathbf{G}_t + \epsilon \mathbf{I})^{-1/2} \mathbf{g}_t$$

Note that $\mathbf{G}_t \in \mathbb{R}^{d \times d}$ accumulates the full outer products of past gradients, so the update scales each direction according to the observed inter-parameter correlations. The term $\epsilon \mathbf{I}$ in $(\mathbf{G}_t + \epsilon \mathbf{I})^{-1/2}$ serves as a regulariser to ensure that the inverse square root is well-defined and numerically stable. If one were to substitute $\mathbf{G}_t$ with the exact Hessian, the update would coincide with a preconditioned Newton step, which demonstrates that the algorithm is similar to a quasi-second-order method.

Algorithm 3 Adagrad (full-matrix version)

```
Require: Learning rate $\eta > 0$, domain $\mathcal{X}$.
1: Initialize $\mathbf{x}_1 \in \mathcal{X}$, and $\mathbf{G}_0 = 0 \in \mathbb{R}^{d\times d}$ (diagonal matrix).
2: for $t = 1$ to $T$ do
3:     Compute subgradient $\mathbf{g}_t \in \partial f_t(\mathbf{x}_t)$.
4:     Update the accumulated squared gradients:
           $$\mathbf{G}_t = \mathbf{G}_{t-1} + \mathbf{g}_t\mathbf{g}_t^\top.$$
5:     Update the iterate:
           $$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}^{\mathbf{G}_t^{1/2}}\big(\mathbf{x}_t \;-\; \eta\,\mathbf{G}_t^{-1/2}\,\mathbf{g}_t\big),$$
       where $\Pi_{\mathcal{X}}^{\mathbf{A}}(\mathbf{y}) = \arg\min_{\mathbf{x}\in\mathcal{X}} \|\mathbf{x}-\mathbf{y}\|_{\mathbf{A}}$ with $\|\mathbf{v}\|_{\mathbf{A}} = \sqrt{\mathbf{v}^\top\mathbf{A}\mathbf{v}}$.
6: end for
```

Remark about Computational cost The full version of Adagrad is computationally expensive for large models due to the $\mathcal{O}(d^2)$ cost of storage and inversion. It is therefore common to consider a diagonal version, which is introduced in Algorithm 4.

Algorithm 4 AdaGrad (diagonal)

```
Require: Learning rate $\eta > 0$, domain $\mathcal{X}$, horizon $T$.
Initialize $\mathbf{x}_1 \in \mathcal{X}$ and $\mathbf{G}_0 = 0 \in \mathbb{R}^{d \times d}$ (diagonal).
for $t = 1$ to $T$ do
    Obtain gradient estimate $\mathbf{g}_t \in \mathbb{R}^d$.
    Accumulate squared gradients:
        $$\mathbf{G}_t = \mathbf{G}_{t-1} + \operatorname{diag}\big(\mathbf{g}_t \odot \mathbf{g}_t\big).$$
    Update the iterate:
        $$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}^{\mathbf{G}_t^{1/2}}\big(\mathbf{x}_t - \eta\, \mathbf{G}_t^{-1/2} \mathbf{g}_t\big),$$
    where $\Pi_{\mathcal{X}}^{\mathbf{A}}(\mathbf{y}) = \arg\min_{\mathbf{x} \in \mathcal{X}} \|\mathbf{x} - \mathbf{y}\|_{\mathbf{A}}$, $\|\mathbf{v}\|_{\mathbf{A}} = \sqrt{\mathbf{v}^\top \mathbf{A} \mathbf{v}}$.
end for
```

Diagonal case: The matrix $\mathbf{G}_t$ is such that

$$G_{t,ii} = \sum_{\tau=1}^{t} g_{\tau,i}^2, \qquad x_{t+1,i} = x_{t,i} - \frac{\eta\, g_{t,i}}{\sqrt{\sum_{\tau=1}^{t} g_{\tau,i}^2}}. \tag{10.1}$$

Algorithm 5 AdaGrad-Scalar

```
Require: Learning rate $\eta > 0$, domain $\mathcal{X}$, horizon $T$.
Initialize $\mathbf{x}_1 \in \mathcal{X}$ and $s_0 = 0$.
for $t = 1$ to $T$ do
    Obtain gradient estimate $\mathbf{g}_t \in \mathbb{R}^d$.
    Accumulate squared-gradient norm:
        $$s_t = s_{t-1} + \|\mathbf{g}_t\|^2.$$
    Update the iterate:
        $$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}\big(\mathbf{x}_t - \eta\, s_t^{-1/2}\, \mathbf{g}_t\big).$$
end for
```

Scalar case: The matrix $\mathbf{G}_t$ is such that:

$$s_t = \sum_{\tau=1}^{t} \|\mathbf{g}_\tau\|^2, \qquad \mathbf{x}_{t+1} = \mathbf{x}_t - \frac{\eta\, \mathbf{g}_t}{\sqrt{\sum_{\tau=1}^{t} \|\mathbf{g}_\tau\|^2}}. \tag{10.2}$$

10.2.2 Convergence Analysis

We will prove that Adagrad achieves a *sublinear* regret bound, implying that the average regret vanishes as $T \to \infty$.

Key Lemma We start with a lemma that will be required to bound the regret.

Lemma 59. *Let $a_1, \dots, a_T$ be nonnegative scalars, and define $S_t = \sum_{i=1}^{t} a_i$. Then*

$$\sum_{t=1}^{T} \frac{a_t}{\sqrt{S_t}} \le 2\sqrt{S_T}.$$

Proof. Observe

$$\sqrt{S_t} \;-\; \sqrt{S_{t-1}} = \frac{S_t - S_{t-1}}{\sqrt{S_t} + \sqrt{S_{t-1}}} = \frac{a_t}{\sqrt{S_t} + \sqrt{S_{t-1}}} \;\geq\; \frac{a_t}{2\sqrt{S_t}}.$$

Summing both sides from $t = 1$ to T gives

$$\sum_{t=1}^{T} \frac{a_t}{\sqrt{S_t}} \leq 2 \sum_{t=1}^{T} \big(\sqrt{S_t} - \sqrt{S_{t-1}}\big) \leq 2\sqrt{S_T}.$$

□

Regret Bound Next, we state a bound on the regret bound of Adagrad:

Theorem 60 (Adagrad Regret Bound)**.** *Assume each f_t is convex, and for all t and all coordinates i, $|g_{t,i}| \leq G_i$. Let $D_\infty = \max_{\mathbf{x},\mathbf{y}\in\mathcal{X}} \|\mathbf{x} - \mathbf{y}\|_\infty$. Then the regret of Algorithm 3 satisfies*

$$R(T) \leq \frac{D_\infty^2}{2\eta} \sum_{i=1}^{d} \sqrt{\sum_{t=1}^{T} g_{t,i}^2} \;+\; \eta \sum_{i=1}^{d} \sqrt{\sum_{t=1}^{T} g_{t,i}^2}.$$

In particular, choosing $\eta = D_\infty$ gives

$$R(T) \leq 2\, D_\infty \sum_{i=1}^{d} \sqrt{\sum_{t=1}^{T} g_{t,i}^2} \leq 2\, D_\infty \sum_{i=1}^{d} G_i \sqrt{T} = O\big(d\sqrt{T}\big).$$

Proof. Define the weighted distance

$$D_t^2 = (\mathbf{x}_t - \mathbf{x}^*)^\top \mathbf{G}_t^{1/2} (\mathbf{x}_t - \mathbf{x}^*),$$

where $\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathcal{X}} \sum f_t(\mathbf{x})$. Using the update

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}^{\mathbf{G}_t^{1/2}} \big(\mathbf{x}_t - \eta\, \mathbf{G}_t^{-1/2} \mathbf{g}_t\big)$$

and non-expansiveness of the projection[1],

$$D_{t+1}^2 \leq \big\|\mathbf{x}_t - \eta \mathbf{G}_t^{-1/2} \mathbf{g}_t - \mathbf{x}^*\big\|_{\mathbf{G}_t^{1/2}}^2 = D_t^2 - 2\eta\, \mathbf{g}_t^\top (\mathbf{x}_t - \mathbf{x}^*) + \eta^2 \, \|\mathbf{g}_t\|_{\mathbf{G}_t^{-1/2}}^2.$$

Rearranging and summing over $t = 1, \ldots, T$ gives

$$2\eta \sum_{t=1}^{T} \mathbf{g}_t^\top (\mathbf{x}_t - \mathbf{x}^*) \leq D_1^2 - D_{T+1}^2 \;+\; \eta^2 \sum_{t=1}^{T} \|\mathbf{g}_t\|_{\mathbf{G}_t^{-1/2}}^2.$$

[1] Recall that $\big\|\Pi_{\mathcal{X}}^H(\mathbf{a}) - \Pi_{\mathcal{X}}^H(\mathbf{b})\big\|_H \leq \big\|\mathbf{a} - \mathbf{b}\big\|_H$

By convexity, $f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*) \le \mathbf{g}_t^\top(\mathbf{x}_t - \mathbf{x}^*)$. Hence

$$R(T) = \sum_{t=1}^{T} \big(f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*)\big) \le \frac{D_1^2}{2\eta} + \frac{\eta}{2}\sum_{t=1}^{T} \|\mathbf{g}_t\|^2_{\mathbf{G}_t^{-1/2}}.$$

Now observe that using Eq. (10.1),

$$\|\mathbf{g}_t\|^2_{\mathbf{G}_t^{-1/2}} = \sum_{i=1}^{d} \mathbf{G}_{t,ii}^{-1/2} g_{t,i}^2 = \sum_{i=1}^{d} \frac{g_{t,i}^2}{\sqrt{\sum_{s=1}^{t} g_{s,i}^2}},$$

and apply Lemma 59 to each coordinate to get

$$\sum_{t=1}^{T} \|\mathbf{g}_t\|^2_{\mathbf{G}_t^{-1/2}} \le 2\sum_{i=1}^{d} \sqrt{\sum_{t=1}^{T} g_{t,i}^2}.$$

In order to bound the term D_1^2, recall that

$$D_1^2 = (\mathbf{x}_1 - \mathbf{x}^*)^\top \mathbf{G}_1^{1/2} (\mathbf{x}_1 - \mathbf{x}^*), \qquad \mathbf{G}_1^{1/2} = \operatorname{diag}\big(|g_{1,1}|, \dots, |g_{1,d}|\big).$$

Because both $\mathbf{x}_1$ and $\mathbf{x}^*$ lie in the feasible set $\mathcal{X}$,

$$|x_{1,i} - x_i^*| \le \|\mathbf{x}_1 - \mathbf{x}^*\|_\infty \le D_\infty \quad \text{for every } i,$$

so $(x_{1,i} - x_i^*)^2 \le D_\infty^2$. Therefore

$$D_1^2 = \sum_{i=1}^{d} |g_{1,i}|(x_{1,i} - x_i^*)^2 \le D_\infty^2 \sum_{i=1}^{d} |g_{1,i}|.$$

Since $|g_{1,i}| = \sqrt{g_{1,i}^2} \le \sqrt{\sum_{t=1}^{T} g_{t,i}^2}$,

$$\boxed{D_1^2 \le D_\infty^2 \sum_{i=1}^{d} \sqrt{\sum_{t=1}^{T} g_{t,i}^2}}.$$

Combining terms yields the stated bound. □

10.2.3 Bounding $f(\mathbf{x}_t) - f^*$ in the (Stochastic) Convex Setting

We now convert our online-regret analysis into a direct bound on the suboptimality $\mathbb{E}[f(\bar{\mathbf{x}}_T)] - f(\mathbf{x}^*)$ (and hence on the best iterate), where

$$f(x) = \mathbb{E}_t[\, f_t(\mathbf{x})\,]$$

is a convex objective and $\mathbf{g}_t = \nabla f_t(\mathbf{x}_t)$ are unbiased estimates of $\nabla f(\mathbf{x}_t)$.

From Regret to Suboptimality Recall that the regret after T rounds satisfies

$$R(T) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}^*) \le B,$$

where (from the previous theorem)

$$B = \frac{D_\infty^2}{2\eta} \sum_{i=1}^{d} \sqrt{\sum_{t=1}^{T} g_{t,i}^2} + \frac{\eta}{2} \sum_{i=1}^{d} \sqrt{\sum_{t=1}^{T} g_{t,i}^2}.$$

Since $\{f_t\}$ are i.i.d. samples of f, taking expectations gives

$$\mathbb{E}\Big[\sum_{t=1}^{T} f(\mathbf{x}_t)\Big] - T\, f(\mathbf{x}^*) \le \mathbb{E}[B].$$

Dividing by T yields a bound on the *average iterate* $\bar{\mathbf{x}}_T = \frac{1}{T}\sum_{t=1}^{T} \mathbf{x}_t$ by convexity of f:

$$\mathbb{E}\big[f(\bar{\mathbf{x}}_T)\big] - f(\mathbf{x}^*) \le \frac{\mathbb{E}[B]}{T}.$$

Moreover, since $\min_{1 \le t \le T} f(\mathbf{x}_t) \le f(\bar{\mathbf{x}}_T)$, the same bound applies to the best iterate.

Concrete Rate Choose the stepsize $\eta = D_\infty$. Then

$$\mathbb{E}[B] = D_\infty \sum_{i=1}^{d} \sqrt{\sum_{t=1}^{T} \mathbb{E}\big[g_{t,i}^2\big]} \le D_\infty \sum_{i=1}^{d} G_i \sqrt{T},$$

where $|g_{t,i}| \le G_i$. Hence

$$\mathbb{E}\big[f(\bar{\mathbf{x}}_T)\big] - f(\mathbf{x}^*) \le \frac{D_\infty \sum_{i=1}^{d} G_i \sqrt{T}}{T} = D_\infty \sum_{i=1}^{d} G_i \, \frac{1}{\sqrt{T}} = O\big(d\, T^{-1/2}\big).$$

Therefore, Adagrad enjoys an $O(1/\sqrt{T})$ convergence rate on the average (or best) iterate:

$$\boxed{\mathbb{E}\big[f(\bar{\mathbf{x}}_T)\big] - f(\mathbf{x}^*) \le D_\infty \Big(\sum_{i=1}^{d} G_i\Big) \frac{1}{\sqrt{T}}.}$$

10.3 Adam

Adam (**Ada**ptive **M**oment estimation) Kingma and Ba (2014) combines the diagonal preconditioning idea of Adagrad with exponentially-weighted moving averages of past gradients (”momentum”). At every iteration it keeps *two* state vectors:

- **First moment (mean)** $\mathbf{m}_t \approx \mathbb{E}[\mathbf{g}_t]$,
- **Second moment (uncentered variance)** $\mathbf{v}_t \approx \mathbb{E}[\mathbf{g}_t \odot \mathbf{g}_t]$,

both updated with an exponential decay controlled by hyperparameters $\beta_1, \beta_2 \in [0, 1)$. Bias-correction terms make the estimators unbiased during the first few steps.

10.3.1 Algorithm

Algorithm 6 Adam (Kingma & Ba, 2015). Note that all operations on vectors are element-wise.

```
Require: Initial point $\mathbf{x}_0 \in \mathbb{R}^d$, stepsize $\alpha > 0$, exponential-decay rates $\beta_1, \beta_2 \in [0, 1)$, numerical stabiliser $\varepsilon > 0$.
$\mathbf{m}_0 \leftarrow 0, \quad \mathbf{v}_0 \leftarrow 0$
for $t = 1, 2, \dots, T$ do
    Obtain (sub)gradient $\mathbf{g}_t = \nabla f_t(\mathbf{x}_{t-1})$
    Update biased moments
        $\mathbf{m}_t = \beta_1 \mathbf{m}_{t-1} + (1 - \beta_1)\,\mathbf{g}_t, \qquad \mathbf{v}_t = \beta_2 \mathbf{v}_{t-1} + (1 - \beta_2)\,\mathbf{g}_t \odot \mathbf{g}_t$
    Bias-correct
        $\hat{\mathbf{m}}_t = \frac{\mathbf{m}_t}{1 - \beta_1^t}, \qquad \hat{\mathbf{v}}_t = \frac{\mathbf{v}_t}{1 - \beta_2^t}$
    Coordinate-wise update
        $\mathbf{x}_t = \mathbf{x}_{t-1} \; - \; \alpha \frac{\hat{\mathbf{m}}_t}{\sqrt{\hat{\mathbf{v}}_t} \; + \; \varepsilon}$
end for
return $\mathbf{x}_T$
```

Default hyper-parameters. In deep-learning practice the choices $\alpha = 10^{-3}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\varepsilon = 10^{-8}$ work reliably across a wide range of tasks; tuning usually focuses on α.

10.3.2 Convergence (one-sentence summary)

For *convex* objectives and bounded gradients, Adam with a diminishing stepsize $\alpha_t = \alpha/\sqrt{t}$ and fixed $\beta_1, \beta_2 < 1$ enjoys an $\mathcal{O}(\sqrt{T})$ regret bound, hence $\mathcal{O}(1/\sqrt{T})$ convergence of the averaged iterate; however, with a constant stepsize the original Adam may *fail* to converge, and variants such as AMSGrad Reddi et al. (2019) or AdamW restore guarantees while preserving the empirical benefits.

Chapter 11

Non-convex Optimization

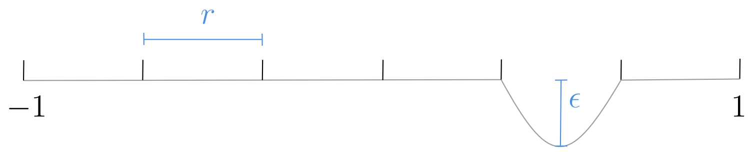
Acknowledgements: *Part of these lecture notes are based on the Optimization lecture notes from Prof. Chi Jin (Princeton).*

11.1 Hardness of Optimizing Non-convex Functions

Gradient descent has become a key algorithm in many areas of science. Its convergence properties are well understood for convex functions, which arise in many machine learning applications (Nesterov, 2003). However, non-convex functions are also ubiquitous and have recently drawn a lot of interest due to the growing success of deep neural networks. Yet, non-convex functions are extremely hard to optimize due to the presence of saddle points and local minima which are not global optima (Dauphin et al., 2014; Choromanska et al., 2015). In fact, the work of Hillar and Lim (2013) showed that even a degree four polynomial can be NP-hard to optimize. Instead of aiming for a global minimizer, we will thus seek a local optimum of the objective.

Theorem 61. *For all $\ell \in \mathbb{R}_+$, and for any algorithm $\mathcal{A}$, there exists an ℓ-smooth function f on $[-1,1]^d$ such that $\mathcal{A}$ has to make at least $\left(\frac{\ell}{\epsilon}\right)^{\Omega(d)}$ gradient queries to guarantee that the output $\bar{\mathbf{x}}$ will satisfy $f(\bar{\mathbf{x}}) < f(\mathbf{x}^*) + \epsilon$ with probability $\geq \frac{1}{2}$, where $\mathbf{x}^*$ is the global minimum of f.*

Proof. The proof consists in finding one hard function that satisfies the required condition. To do so, we define a function g on the interval $[-1,1]^d$, which we subdivide into multiple (hyper-)rectangles with side length $r > 0$. The function $g : [-1,1]^d \to \mathbb{R}$ is a constant on all the intervals except for one where it has a dip of depth ϵ, see the illustration below in the 1-dimensional case.



It is clear that the minimum $\bar{\mathbf{x}}$ has to lie in the special interval. The question we need to answer is therefore how many queries are required to find this special block? The answer turns out to be $\left(\frac{2}{r}\right)^{\Omega(d)}$. Choosing $r = 2\sqrt{\frac{\epsilon}{\ell}}$, we conclude that the number of queries is $\left(\sqrt{\frac{\ell}{\epsilon}}\right)^{\Omega(d)}$.

□

11.2 Saddle points

Types of critical points Recall the following theorem (discussed in Chapter 1):

Theorem 62. *Assume $f \in C^2(\mathbb{R}^d)$. Then $\nabla f(\mathbf{x}^*) = 0$ and $\nabla^2 f(\mathbf{x}^*)$ is positive semidefinite (i.e. $\mathbf{s}^\top \nabla^2 f(\mathbf{x}^*)\mathbf{s} \geq 0$ for all $\mathbf{s} \in \mathbb{R}^d$) implies that $\mathbf{x}^*$ is a local minimizer of f.*

Based on the previous theorem, we will differentiate between different types of critical points, leading us to the following definition. We will denote by $\lambda_{\min}(\mathbf{A})$ the smallest eigenvalue of the matrix $\mathbf{A}$.

Definition 35. *A critical point $\mathbf{x}^*$ of a function $f : \mathbb{R}^d \to \mathbb{R}$ is a* strict *saddle point if $\lambda_{\min}(\nabla^2 f(\mathbf{x}^*)) < 0$.*

If the eigenvalues are instead zero, one will talk about non-strict or degenerate saddle points. We show some examples in Figure 11.1.

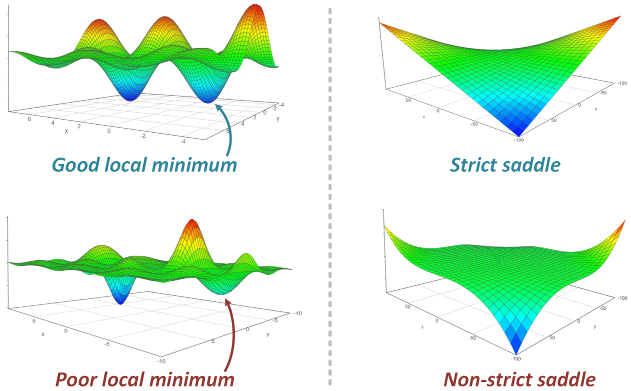


Figure 11.1: Illustration of possible saddle points in 2 dimensions. (Top left) Good local minimum (and local maxima). (Top right) Strict saddle point with $\lambda_1 < 0$ and $\lambda_2 > 0$. (Bottom left) Poor local minimum (i.e. not the lowest function value). (Bottom right) Non-strict saddle point where $\lambda_1 = 0$ and $\lambda_2 = 0$.
While the non-strict saddle is flat, the strict saddle has one direction along which the curvature is strictly negative. The presence of such a direction gives gradient descent the possibility of escaping the saddle point.
Source: `https://www.offconvex.org/2018/11/07/optimization-beyond-landscape/`

Convergence behavior of GD on a quadratic saddle We will start with a somewhat simpler problem discussed in Lee et al. (2016) where we consider a quadratic function $f : \mathbb{R}^d \to \mathbb{R}$ of the form $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{H}\mathbf{x}$ where $\mathbf{H} \in \mathbb{R}^{d \times d}$. This function obviously has a unique critical point at zero (check using first-order optimality).

W.l.o.g., we will assume that $\mathbf{H}$ is diagonal (else it can be diagonalized), i.e. $\mathbf{H} = \text{diag}(\lambda_1, \dots, \lambda_d)$. We will also consider the case where we have both positive and negative eigenvalues: $\lambda_1, \dots, \lambda_p > 0$ and $\lambda_{p+1}, \dots, \lambda_d < 0$ (for some $p \leq d$), which means that the unique critical point at zero is a saddle point. An example of such a function in 2 dimensions is shown in Figure 11.2.

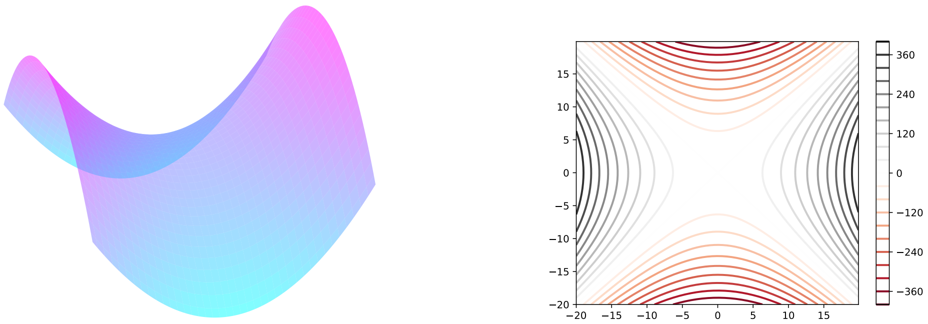


Figure 11.2: Illustration of the function $f(\mathbf{x}) = x_1^2 - x_2^2$ (left) with its contour lines (right).

Recall that gradient descent initialized at $\mathbf{x}_0$ has the following iterates:

$$\mathbf{x}_k = (\mathbf{I} - \eta\mathbf{H})^k \mathbf{x}_0 = \sum_{i=1}^{d} (1 - \eta\lambda_i)^k (\mathbf{e}_i^\top \mathbf{x}_0)\mathbf{e}_i, \tag{11.1}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{e}_i$ denotes the i-th standard basis vector of $\mathbb{R}^d$. We will choose the step size η to satisfy the following condition: $\eta < \frac{1}{L}$, where $L = \max_i |\lambda_i|$.

We need to consider two cases to determine whether $\mathbf{x}_k$ converges:

1. For $i \leq p$, $\lambda_i > 0$, thus $0 < (1 - \eta\lambda_i) < 1$,
2. For $i > p$, $\lambda_i < 0$, thus $(1 - \eta\lambda_i) > 1$.

Now comes an important point that will determine the convergence behavior of gradient descent. If the initial point $\mathbf{x}_0 \in E_s := \text{span}(\mathbf{e}_1, \dots, \mathbf{e}_p)$, then $\mathbf{x}_k$ converges to the saddle point at zero since $\forall i \leq p$, we have $\lim_{k\to\infty}(1 - \eta\lambda_i)^k \to 0$. However, if $\mathbf{x}_0$ has a single component outside of E_s, then $\mathbf{x}_k$ will diverge to ∞ since we then have at least one component i for which $\lim_{k\to\infty}(1 - \eta\lambda_i)^k \to \infty$. Since we typically choose $\mathbf{x}_0$ at random, it has zero probability of being in E_s (which is a finite set).

Worst-case guarantees We have seen that gradient descent will not move away from a stationary point if started there, or if $\mathbf{x}_0 \in E_s$. In order to establish worst-case convergence guarantees, we will have to modify gradient descent in order to add some degree of randomness to avoid $\mathbf{x}_0 \in E_s$. One could either use random initialization as discussed earlier or modify gradient descent to add some random perturbation to the updates as done, for instance, in Ge et al. (2015); Jin et al. (2017). We will discuss this in more detail later on. For now, we will focus on another approach that provides second-order convergence guarantees by exploiting curvature information. At a saddle point, the gradient is zero but the Hessian is negative definite. We will see that the eigenvector corresponding to $\lambda_{\min}$ in fact provides a way to minimize the function. Before we do so, we briefly discuss how this eigenvector direction can be extracted in the first place.

11.3 Eigenvector Problem

The problem we consider in this section is as follows. We are given a symmetric PSD matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$ whose eigenvalues are denoted by $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d$, with corresponding eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d$. Our goal is to find the top eigenvector $\mathbf{v}_1$.

A well-known method to find the top eigenvector of a PSD matrix $\mathbf{A}$ is the power method. As shown in Algorithm 7, it simply starts from a random point $\mathbf{w}_0$ and iteratively performs the update $\mathbf{w}_{k+1} = \frac{\mathbf{A}\mathbf{w}_k}{\|\mathbf{A}\mathbf{w}_k\|}$.

Algorithm 7 Power method.

```
INPUTS : $\mathbf{w}_0$ = random vector, $\mathbf{A}$
for $k = 1$ to $K$ do
    $\mathbf{w}_{k+1} = \frac{\mathbf{A}\mathbf{w}_k}{\|\mathbf{A}\mathbf{w}_k\|}$
end for
```

11.3.1 Asymptotic convergence

First, we use the eigen-decomposition of $\mathbf{A}$ as $\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^\top$, where Λ is a diagonal matrix that contains the eigenvalues λ_i of $\mathbf{A}$, and $\mathbf{V}$ is an orthogonal matrix (i.e. $\mathbf{V}\mathbf{V}^\top = \mathbf{V}^\top\mathbf{V} = \mathbf{I}$). Then, the starting vector $\mathbf{w}_0$ can be written as a linear combination of the columns of $\mathbf{V}$, thus

$$\mathbf{w}_0 = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n,$$

where $c_i \in \mathbb{R}$.

Then one can show that

$$\begin{aligned} \mathbf{w}_k &= \frac{\mathbf{A}^k\mathbf{w}_0}{\|\mathbf{A}^k\mathbf{w}_0\|} && (11.2) \\ &= \left(\frac{\lambda_1}{|\lambda_1|}\right)^k \frac{c_1}{|c_1|}\frac{\mathbf{v}_1}{\|\mathbf{v}_1\|} + r_k, && (11.3) \end{aligned}$$

where r_k is a vector such that $\|r_k\| \to 0$ as $k \to \infty$. The detailed calculation is left as an exercise.

11.3.2 Gap-dependent analysis

Theorem 63. *For any PSD matrix* $\mathbf{A}$ *with largest eigenvalues* λ_1, λ_2 *such that* $\lambda_1 > \lambda_2$*, the iterates of the power method satisfy*

$$\langle \mathbf{v}_1, \mathbf{w}_k \rangle \geq 1 - \left(1 - \frac{\lambda_1 - \lambda_2}{\lambda_1}\right)^{2k} \frac{1}{\langle \mathbf{v}_1, \mathbf{w}_0 \rangle}.$$

Proof. First, note that the output of the power method is given by

$$\begin{aligned} \bar{\mathbf{w}}_k &= \mathbf{A}^k \mathbf{w}_0 \\ \mathbf{w}_k &= \frac{\bar{\mathbf{w}}_k}{\|\bar{\mathbf{w}}_k\|}. \end{aligned}$$

Let P_1 denote the projection onto the space spanned by $\mathbf{v}_1$, and $P_1^\perp$ the projection onto the orthogonal space. Then, we have (recalling that $\|\mathbf{w}_k\|^2 = 1$),

$$\begin{aligned} 1 - \langle \mathbf{v}_1, \mathbf{w}_k \rangle &= \|P_1^\perp \mathbf{w}_k\|^2 \\ &= \frac{\|P_1^\perp \bar{\mathbf{w}}_k\|^2}{\|\bar{\mathbf{w}}_k\|^2} \\ &\leq \frac{\|P_1^\perp \bar{\mathbf{w}}_k\|^2}{\|P_1 \bar{\mathbf{w}}_k\|^2}, \end{aligned}$$

where the last inequality is due to the fact that the projection does not increase distances.

The starting vector $\mathbf{w}_0$ can be written as a linear combination of the columns of $\mathbf{V}$, thus

$$\mathbf{w}_0 = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \cdots + c_d \mathbf{v}_d.$$

Then $\bar{\mathbf{w}}_1$ can be written as

$$\bar{\mathbf{w}}_1 = \lambda_1 c_1 \mathbf{v}_1 + \lambda_2 c_2 \mathbf{v}_2 + \cdots + \lambda_d c_d \mathbf{v}_d,$$

and more generally

$$\bar{\mathbf{w}}_k = \sum_{i=1}^{k} \lambda_i^k c_i \mathbf{v}_i.$$

Therefore,

$$\|P_1^\perp \bar{\mathbf{w}}_k\|^2 = \sum_{i=2}^{k} \lambda_i^{2k} c_i^2 \leq \lambda_2^{2k} \sum_{i=2}^{k} c_i^2 \leq \lambda_2^{2k}$$

(note that the sum starts at $i = 2$), and

$$\|P_1 \bar{\mathbf{w}}_k\|^2 = \lambda_1^{2k} c_1^2.$$

Combining the last two inequalities, we get

$$1 - \langle \mathbf{v}_1, \mathbf{w}_k \rangle \leq \left(\frac{\lambda_2}{\lambda_1}\right)^{2k} \frac{1}{c_1^2} = \left(1 - \frac{\lambda_1 - \lambda_2}{\lambda_1}\right)^{2k} \frac{1}{\langle \mathbf{v}_1, \mathbf{w}_0 \rangle}.$$

□

11.4 Escaping Saddle Points

We will make the following standard assumption:

Assumption 4. *The function f is twice differentiable, L-smooth and its Hessian is L_2 Lipschitz continuous. This implies that:*

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y}-\mathbf{x}) + \frac{1}{2}(\mathbf{y}-\mathbf{x})^\top \nabla^2 f(\mathbf{x})(\mathbf{y}-\mathbf{x}) + \frac{L_2}{6}\|\mathbf{y}-\mathbf{x}\|^3, \quad \forall \mathbf{x}, \mathbf{y}.$$

Algorithm 8 Two–Step Method

```
INPUTS : $\mathbf{x}_1 \in \mathbb{R}^n$, $\eta \in \left(0, \frac{2}{L}\right)$, $\beta \in \left(0, \frac{3\gamma}{L_2}\right)$
for $k = 1, 2, \ldots$ do
    if $\lambda_k \geq 0$ then
        $\mathbf{d}_k = 0$
    else
        choose $\mathbf{d}_k$ satisfying (2a)
    end if
    $\hat{\mathbf{x}}_k = \mathbf{x}_k + \beta \mathbf{d}_k$
    if $\mathbf{d}_k = 0$ and $\nabla f(\mathbf{x}_k) = 0$ then
        return $\mathbf{x}_k$
    end if
    $\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \nabla f(\mathbf{x}_k) + \beta \mathbf{d}_k$
end for
```

Consider Algorithm 8 (Curtis and Robinson, 2019) that alternates between gradient descent steps and negative curvature steps using fixed stepsizes for each. At a given iterate $\mathbf{x}_k \in \mathbb{R}^n$, let λ_k denote the left-most eigenvalue of $\mathbf{H}_k$. If $\lambda_k \geq 0$ (i.e., $\mathbf{H}_k \succeq 0$), the algorithm sets $\mathbf{d}_k \leftarrow 0$; otherwise, $\mathbf{d}_k$ is computed so that

$$\mathbf{d}_k^\top \mathbf{H}_k \, \mathbf{d}_k \leq \gamma \lambda_k \, \|\mathbf{d}_k\|_2^2 < 0, \tag{2a}$$

$$\nabla f(\mathbf{x}_k)^\top \mathbf{d}_k \leq 0, \tag{2b}$$

$$\|\mathbf{d}_k\|_2 \leq |\lambda_k|, \tag{2c}$$

for some $\gamma \in (0, 1]$ independent of k. In particular, we will choose $\mathbf{d}_k$ to be an eigenvector corresponding to the leftmost eigenvalue λ_k scaled so that $\nabla f(\mathbf{x}_k)^\top \mathbf{d}_k \leq 0$ and $\|\mathbf{d}_k\| = |\lambda_k|$.

Algorithm 1 terminates finitely only if, for some $k \in \mathbb{N}$, $\mathbf{d}_k = 0$ and $\nabla f(\mathbf{x}_k) = 0$. This can occur only when $\lambda_k \geq 0$ and, since $\mathbf{d}_k = 0$ then yields $\hat{\mathbf{x}}_k = \mathbf{x}_k$, when $\nabla f(\mathbf{x}_k) = 0$. These represent the desired conclusions for this case.

Otherwise, if Algorithm 1 does not terminate finitely, fix an arbitrary $k \in \mathbb{N}$. If $k \notin \mathcal{D}$, then $\mathbf{d}_k = 0$ and $\hat{\mathbf{x}}_k = \mathbf{x}_k$, so that $f(\hat{\mathbf{x}}_k) = f(\mathbf{x}_k)$. If $k \in \mathcal{D}$, we have $\mathbf{d}_k \neq 0$ and $\lambda_k < 0$, and, by (2),

$$\begin{aligned} f(\hat{\mathbf{x}}_k) &\leq f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^\top (L\mathbf{d}_k) + \tfrac{1}{2}(L\mathbf{d}_k)^\top \mathbf{H}_k (L\mathbf{d}_k) + \tfrac{1}{6} L_2 \|L\mathbf{d}_k\|_2^3 \\ &\leq f(\mathbf{x}_k) + \tfrac{1}{2} L^2 \gamma \lambda_k \|\mathbf{d}_k\|_2^2 + \tfrac{1}{6} L_2 L^3 |\lambda_k|^3 \\ &= f(\mathbf{x}_k) - \tfrac{1}{2} L^2 \big(\gamma - \tfrac{1}{3} L_2 L\big) |\lambda_k|^3 \\ &= f(\mathbf{x}_k) - c_1(L) |\lambda_k|^3, \end{aligned}$$

where $c_1(L) = \frac{1}{2} L^2 \big(\gamma - \frac{1}{3} L_2 L\big) > 0$. Therefore we see that a single step of the algorithm decreases the objective function, which implies asymptotic convergence. Deriving a precise rate is more possible, although it typically requires adding noise to the gradient Jin et al. (2017).

Bibliography

Yossi Arjevani, Yair Carmon, John C Duchi, Dylan J Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *arXiv preprint arXiv:1912.02365*, 2019.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

Jonathan Borwein and Adrian Lewis. *Convex Analysis.* Springer, 2006.

Stephen Boyd, Lin Xiao, and Almir Mutapcic. Subgradient methods. *lecture notes of EE392o, Stanford University, Autumn Quarter*, 2004:2004–2005, 2003.

Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.

Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *Artificial intelligence and statistics*, pages 192–204. PMLR, 2015.

Frank E Curtis and Daniel P Robinson. Exploiting negative curvature in deterministic and stochastic optimization. *Mathematical Programming*, 176:69–94, 2019.

Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. *Advances in neural information processing systems*, 27, 2014.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Cong Fang, Zhouchen Lin, and Tong Zhang. Sharp analysis for nonconvex sgd escaping from saddle points. In *Conference on Learning Theory*, pages 1192–1234. PMLR, 2019.

Simon Foucart. Lecture 6: Matrix norms and spectral radii. *lecture notes for the course NSTP187 at Drexel University, Philadelphia, PA, Fall*, 2012, 2012.

Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on learning theory*, pages 797–842. PMLR, 2015.

Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

Elad Hazan and Sham Kakade. Revisiting the polyak step size. *arXiv preprint arXiv:1905.00313*, 2019.

Christopher J Hillar and Lek-Heng Lim. Most tensor problems are np-hard. *Journal of the ACM (JACM)*, 60(6):1–39, 2013.

Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pages 1724–1732. PMLR, 2017.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 795–811. Springer, 2016.

Ahmed Khaled and Peter Richtárik. Better theory for sgd in the nonconvex world. *arXiv preprint arXiv:2002.03329*, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Simen Kvaal, Ulf Ekström, Andrew M Teale, and Trygve Helgaker. Differentiable but exact formulation of density-functional theory. *The Journal of chemical physics*, 140(18): 18A518, 2014.

Jason D Lee, Max Simchowitz, Michael I Jordan, and Benjamin Recht. Gradient descent only converges to minimizers. In *Conference on learning theory*, pages 1246–1257. PMLR, 2016.

Claude Lemaréchal and Claudia Sagastizábal. Practical aspects of the moreau–yosida regularization: Theoretical preliminaries. *SIAM journal on optimization*, 7(2):367–385, 1997.

Hongzhou Lin, Julien Mairal, and Zaid Harchaoui. Catalyst acceleration for first-order convex optimization: from theory to practice. *Journal of Machine Learning Research*, 18 (1):7854–7907, 2018.

Nicolas Loizou, Sharan Vaswani, Issam Hadj Laradji, and Simon Lacoste-Julien. Stochastic polyak step-size for sgd: An adaptive learning rate for fast convergence. In *International Conference on Artificial Intelligence and Statistics*, pages 1306–1314. PMLR, 2021.

Jean-Jacques Moreau. Proximité et dualité dans un espace hilbertien. *Bulletin de la Société mathématique de France*, 93:273–299, 1965.

Eric Moulines and Francis Bach. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. *Advances in neural information processing systems*, 24, 2011.

James R Munkres. *Topology*, volume 2. Prentice Hall Upper Saddle River, 2000.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

Yu E Nesterov. A method for solving the convex programming problem with convergence rate $o\bigl(\frac{1}{k^2}\bigr)$. In *Dokl. Akad. Nauk SSSR,*, volume 269, pages 543–547, 1983.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 1999.

Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, 4(5):1–17, 1964.

Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. *arXiv preprint arXiv:1109.5647*, 2011.

Sashank J Reddi, Ahmed Hefny, Suvrit Sra, Barnabas Poczos, and Alex Smola. Stochastic variance reduction for nonconvex optimization. In *International conference on machine learning*, pages 314–323. PMLR, 2016.

Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, 2019.

Walter Rudin et al. *Principles of mathematical analysis*, volume 3. McGraw-hill New York, 1976.

Jun-Kun Wang, Chi-Heng Lin, and Jacob D Abernethy. A modular analysis of provable acceleration via polyak's momentum: Training a wide relu network and a deep linear network. In *International Conference on Machine Learning*, pages 10816–10827. PMLR, 2021.