Optimisation Convexe: Algorithmes et Applications en Apprentissage

M2 Statistique, Modélisation et Science des données
Université Claude Bernard Lyon 1

Filippo Santambrogio

Contents

Updated on November 21, 2023.

1 Introduction to Optimization

1.1 Min and inf, max and sup

Definition 1.1. *Given a non-empty set $E \subset \mathbb{R}$ we say that a number $a_0 \in \mathbb{R}$ is the minimum of E (and we write $a_0 = \min E$) if it satisfies the two following properties:*

a) for every $a \in E$ we have $a \geq a_0$,

b) $a_0 \in E$.

We say instead that a value $\ell \in \bar{\mathbb{R}} = [-\infty, +\infty]$ is the infimum of E (and we write $\ell = \inf E$) if it satisfies the two following properties:

a) for every $a \in E$ we have $a \geq \ell$,

b) for every $\ell' > \ell$ there exists $a \in E$ such that $a < \ell'$ (in other words, ℓ is the maximal value which satisfies the previous property).

The existence of the infimum is a consequence of the contruction of the set of real numbers $\mathbb{R}$. Note that the inf is never $+\infty$, except for the emptyset (or, if we considered $E \subset \bar{\mathbb{R}}$, in the case $E = \{+\infty\}$). On the contrary, the minimum does not always exist, think at $E = (0, \infty)$. One can check that the minimum exists if and only if we have $\inf E \in E$ (i.e. the inf satisfies the second property for the min).

Moreover, one can also characterize the inf in this other way

a) for every $a \in E$ we have $a \geq \ell$,

b) there exists a sequence of elements $a_n \in E$ such that $a_n \to \ell$.

In practice, we never look for the minimum (or inf) of a set of numbers which is already well-known, but for the min or inf of the set of values that a certain function takes on a certain set, i.e. we take $E = \{f(x) \ : \ x \in A\}$, for a given function $f : A \to \mathbb{R}$. The set A, the domain of the function, can be more or less arbitrary, but it is important that the function takes values into $\mathbb{R}$. this is due to the fact that we want an order on the target set, and we cannot optimize functions which are complex)valued, vector-valued, fruit-valued... It is possible – and sometimes useful – to consider functions valued into $\mathbb{R} \cup \{+\infty\}$. Unless f is the constant function $+\infty$ then its inf is for sure in $\mathbb{R} \cup \{-\infty\}$. In general we do not consider the minimization of a function which takes somewhere the value $-\infty$, since it would be trivial. All these considerations can of course be done for maximization instead of minimization, and we could define the maximum and the supremum.

Example. *Suppose that we want to build the wing of an airplane and that we describe our choice in terms of a certain number of parameters $(x^1, x^2, \dots, x^N)$ standing for its length, width at different locations of the wing, its density a diferent locations, etc. Suppose that $f(x)$ stands for the construction cost of a wing of type $x = (x1, x^2, \dots, x^N) \in \mathbb{R}^N$, and that $g(x)$ stands for a certain performance of the wing. We could be interested in solving*

$$\min\{f(x) \ : \ x \in A\}, \qquad \textit{where } A = \{x \in \mathbb{R}^N \ : \ g(x) \geq c\}$$

for a certain constant c. It is also possible to replace the constraint with a penalization, looking at something like

$$\min\{f(x) - \lambda g(x) \ : \ x \in \mathbb{R}^N\},$$

and the choice of the parameter $\lambda > 0$ is a delicate matter.

It is important in optimization not to get confused between the minimal value and the minimizer of a function (this distinction does not exist when we minimize a set of values but, as we said, in practice we always minimizes the value of a function f over a set A). We denote by $\min_A f$ the minimal value, which is a number. Such number, if it exists, is always unique. We denote by $\operatorname{argmin}_{x \in A} f(x)$ the minimizer, which could be unique or not (many points could give the same minimal value).

1.2 Example of optimization problems in data sciences

A typical problem in data analysis is the following: we have two sets, $\mathcal{X}$ and $\mathcal{Y}$, and we obtain data in the form of pairs $(x, y) \in \mathcal{X} \times \mathcal{Y}$. We can imagine that Y is a set whose goal is to classify points of X. For instance, he set X is composed of images of animals and we would like to decide whether the image represents a cat, a dog or a dolphin, i.e. $\mathcal{Y} = \{\text{cat,dog,dolphin,other}\}$. Or X could be again a set of images, and we would just like to decide whether the image represents a cat or not, so that we could use $\mathcal{Y} = \{-1, 1\}$ for a binary choice, or $Y = [-1, 1]$ if we want to insert different degrees of certainty in the answer ($y > 0$ means "most likely a cat", while $y < 0$ "most likely not a cat"). We suppose that pairs of data $(x, y) \in \mathcal{X} \times \mathcal{Y}$ can occurr with a certain probability distribution $\pi \in \mathcal{P}(\mathcal{X} \times \mathcal{Y})$, which we do not know. We also assume that the correspondence between x and y should be (but it is not because of errors, noise, ambiguities in the interpretation...) deterministic, i.e. $y = f(x)$. We look for this function $f : \mathcal{X} \to \mathcal{Y}$. To do it we fix a *loss function* $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ which acts like a distance: $L(y, y') \geq 0$ and $L(y, y') = 0$ if and only if $y = y'$. Then, we would like to solve

$$\min_f \mathbb{E}_{(X,Y)\sim\pi}[L(f(X), Y)].$$

Yet, as we do not know π we replace it with its empirical version, i.e. with $\pi_N = \frac{1}{N}\sum_{i=1}^{N} \delta_{(x_i,y_i)}$, where the points $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ are the observations we have collected. We then look at

$$\min_f \sum_N L(f(x_i), y_i).$$

This problem has to be considered either under additional constraints on f (for instance: we impose f to be linear, or to be a polynomial of a certain fixed degree, or to belong to a certain paramterized class, the latter being what is actually done in neural networks) or adding a penalization on f, called a regularization. Indeed, if we plot the points (x_i, y_i) on a plane, we are looking for a function f which best interpolates or approximates these data. If the x_i are distinct it is always possible to find a function f such that $f(x_i) = y_i$ for every i, but this can produce a very nasty function f, different from what we expect, and too much subject to outliers coming from possible measurement errors. For instance, if $\mathcal{X}$ is a finite-dimensional vector space and $\mathcal{Y} = \mathbb{R}$, we can consider linear functions of the form $f(x) = v \cdot x + a$ and the problem

$$\min_{a,v} \sum_N |(a + v \cdot x_i) - y_i|^2$$

is called linear regression. It consists in finding the best affine function approximating the data. As an example of regularizing term, if $\mathcal{X} = [0, M] \subset \mathbb{R}$, we could instead consider

$$\min_{f:[0,M]\to\mathbb{R}} \sum_N |f(x_i) - y_i|^2 + \int_0^M |f''(x)|^2 \,\mathrm{d}x,$$

where f is not forced to be affine, but its second derivative (i.e. a measure of how much it is not affine) is penalized.

The regularization term can also be used to impose sparsity, and not only "regularity". For instance, one could hope to write y as an affine function of x using few variables of x... If we define for $p > 0$ the function $A_p : \mathbb{R}^N \to \mathbb{R}$ via $A_p(x) := \sum_i |x^i|^p$ (a function which coincides, for $p \geq 1$, with $||x||^p_{\ell^p}$; this involves the ℓ^p norm $||x||_p := (\sum_i |x^i|^p)^{1/p}$), we see that we have $\lim_{p\to 0^+} A_p(x) = \#\{i : xi \neq 0\}$. This limit is sometimes called, by abuse of language, the ℓ^0 norm of x; we can also denote it by A_0. We could be interested in solving

$$\min_{a,v} \sum_N |(a + v \cdot x_i) - y_i|^2 + A_0(v).$$

Before adding A_0 this optimization problem was quadratic, and hence convex (see below), which was a great advantage. A common trick to have a simpler problem to consider is to replace A_0 with A_1, which is a convex function, but it is the closest to A_0 among the A_p which are convex. In this case we would have

$$\min_{a,v} \sum_N |(a + v \cdot x_i) - y_i|^2 + ||v||_1.$$

Another common problem data analysis involving the ℓ^1 norm, is the following:

$$\min_x ||Ax - y||^2 + ||x||_1.$$

Here the question is slightly different: we try to describe the observations y_i as obtained from uknown parameters x via linear combinations, through a procedure, described by the matrix A, which is known, i.e. we assume $y \sim Ax$. We look for the values of x, and we want them to be sparse.

1.3 Existence and optimality conditions

The most common way to prove that a function admits a minimizer on a certain set is based on the classic Weierstrass Theorem, possibly replacing continuity with semicontinuity.

Definition 1.2. *On a metric space X, a function $f : X \to \mathbb{R} \cup \{+\infty\}$ is said to be lower semicontinuous (l.s.c. in short) if for every sequence $x_n \to x$ we have $f(x) \leq \liminf_n f(x_n)$. A function $f : X \to \mathbb{R} \cup \{-\infty\}$ is said to be upper-semicontinuous (u.s.c. in short) if for every sequence $x_n \to x$ we have $f(x) \geq \limsup_n f(x_n)$.*

We also remind the following:

Definition 1.3. *A metric space X is said to be compact if from any sequence x_n we can extract a converging subsequence $x_{n_k} \to x \in X$.*

One of the main theorems in optimization is:

Theorem 1.4. *If $f : X \to \mathbb{R} \cup \{+\infty\}$ is lower semicontinuous and X is compact, then there exists $\bar{x} \in X$ such that $f(\bar{x}) = \min\{f(x) : x \in X\}$.*

Proof. Define $\ell := \inf\{f(x) : x \in X\} \in \mathbb{R} \cup \{-\infty\}$ ($\ell = +\infty$ only if f is identically $+\infty$, but in this case any point in X minimizes f). By definition there exists a minimizing sequence x_n, i.e. points in X such that $f(x_n) \to \ell$. By compactness we can assume $x_n \to \bar{x}$. By lower semicontinuity, we have $f(\bar{x}) \leq \liminf_n f(x_n) = \ell$. On the other hand, we have $f(\bar{x}) \geq \ell$ since ℓ is the infimum. This proves $\ell = f(\bar{x}) \in \mathbb{R}$ and this value is the minimum of f, realized at $\bar{x}$. □

The compactness of X can be replaced by a lighter assumption: we just need that there exists a value M for which $\{x \in X : f(x) \leq M\}$ is at the same time non-empty and compact. Indeed, it is then possible to restrict the minization to this set. In particular this happens in the case where $X = \mathbb{R}^N$ and $\lim_{||x|| \to \infty} f(x) = +\infty$. In this case we say that f is *coercive.*

We recall this well-known statement:

Theorem 1.5. *Suppose that f is C^1, that x_0 is a of f on E, and that x_0 is in the interior of E (there exists a radius $r > 0$ such that $B(x_0, r) \subset E$). Then $\nabla f(x_0) = 0$.*

The gradient of f is not only used to check whether a point x_0 satisfies or not the optimality conditions above, but also to move from a point in the directin of the minimizer. Indeed, the vector ∇f represents the direction in which f increases the most, and the vector $-\nabla f$ the one in which it decreases the most. Unfortunately, any use of the gradient in order to find a minimizer is limited by the fact that for arbitrary functions there could be points where the gradient vanishes without being minimizers, and that using the gradient (or even higher-order derivatives) will only provide local informations, so that we will never be able to see the difference between a global minimizer and a local minimizer (i.e. a point x_0 such that there exists $r > 0$ such that x_0 minimizes f over the ball $B(x_0, r)$). The only case where everything goes well is the case of convex functions.

1.4 Convex functions

Definition 1.6. *A function $f : \mathbb{R}^n \to \mathbb{R}$ is called convex if, for all $x, y \in \mathbb{R}^n$ and all $t \in [0, 1]$, we have*

$$f((1-t)x + ty) \leq (1-t)f(x) + tf(y).$$

We say that f is strictly convex if the same inequality holds strictly whenever $x \neq y$ and $t \in]0, 1[$.

In dimension 1, we have the following characterizations.

Proposition 1.7. *If f is C^1, it is convex if and only if f' is an increasing function, and if and only if we have the following inequality*

$$f(y) \geq f(x) + f'(x)(y - x)$$

for all x, y.

If f is C^2, then it is convex if and only if $f'' \geq 0$.

In higher dimensions, it becomes

Proposition 1.8. *If f is C^1, it is convex if and only if ∇f satisfies the inequality*

$$(\nabla f(x) - \nabla f(y)) \cdot (x - y) \geq 0$$

for all x, y, and if and only if we have the following inequality

$$f(y) \geq f(x) + \nabla f(x) \cdot (y - x)$$

for all x, y.

If f is C^2, then it is convex if and only if $D^2 f \geq 0$ in the sense of symmetric matrices (which is equivalent to having non-negative eigenvalues).

We also provide the definition of uniformly convex or elliptic functions.

Definition 1.9. *A function $f : \mathbb{R}^n \to \mathbb{R}$ is called uniformly convex or elliptic if there exists $\alpha > 0$ such that $x \mapsto f(x) - \frac{\alpha}{2}||x||^2$ is a convex function (in this case, we say it is α-elliptic).*

We obtain the following characterizations:

Proposition 1.10. *If f is C^1, it is α-elliptic if and only if ∇f satisfies the inequality*

$$(\nabla f(x) - \nabla f(y)) \cdot (x - y) \geq \alpha ||x - y||^2$$

for all x, y, and if and only if we have the following inequality

$$f(y) \geq f(x) + \nabla f(x) \cdot (y - x) + \frac{\alpha}{2}||x - y||^2$$

for all x, y.

If f is C^2, then it is α-elliptic if and only if $D^2 f \geq \alpha I$ in the sense of symmetric matrices (i.e., all its eigenvalues are greater than or equal to α).

It is worth noting that every elliptic function is strictly convex.

For the minimization of convex functions, we have:

Proposition 1.11. *If f is a C^1 convex function, then a point $\bar{x}$ minimizes f if and only if $\nabla f(\bar{x}) = 0$. If f is strictly convex, then the minimum point, if it exists, is unique. If f is elliptic, then the minimum point exists and is unique.*

Indeed, while strict convexity is sufficient for the uniqueness of the minimizer, it is not sufficient for existence (consider $f(x) = e^x$). However, ellipticity is sufficient for existence because every elliptic function is bounded from below by a parabola, hence it tends to infinity at infinity.

2 Gradient descent

Given a C^1 function $f : \mathbb{R}^N \to \mathbb{R}$ we want to find its minimum point and its minimal value (provided they exist), following the opposite direction of the gradient. A possibility could be to the follow the flow of the evolution equation

$$x'(t) = -\nabla f(x(t))$$

which exactly means to follow the steepest descent lines of f and, hopefully, converge to the minimizer.

The above differential equation is useful to have an idea of the qualitative behavior of what we want to do, but in practice one has to use a disretized algorithm.

2.1 Unconstrained fixed-step gradient algorithm

We consider the simplest optimization algorithm, the fixed-step gradient descent: given a point x_0, we define an iterated sequence by taking $x_{k+1} = x_k - \tau \nabla f(x_k)$.

We have the following theorem.

Theorem 2.1. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a C^2 function with $\alpha \mathrm{I} \leq D^2 f(x) \leq L\mathrm{I}$ for two constants $L \geq \alpha > 0$. Suppose $\tau \in]0, \frac{2}{L}[$. Then, the sequence defined by the fixed-step gradient descent algorithm converges to the unique minimizer $\bar{x}$ of f, and we have*

$$||x_k - \bar{x}|| \leq ||x_0 - \bar{x}||\lambda^k$$

where the number λ is given by $\lambda = \max\{1 - \tau\alpha, \tau L - 1\} < 1$.

Note that the value of τ that minimizes the value of λ is $\tau = \frac{2}{\alpha+L}$, which gives $\lambda = \frac{L-\alpha}{L+\alpha}$.

The proof is based on the following preliminaries

- The Banach contraction principle: in a complete metric space X any map $F : X \to X$ such that there exists $\lambda \in (0, 1)$ wth $d(F(x), F(y)) \leq \lambda d(x, y)$ for every x, y (such a map is called a contraction, and being a contraction means being Lipschitz continuous with a Lipschitz constant strictly smaller than 1) admits a unique fixed point $\bar{x}$, and for every point x_0 the sequence defined by $x_{k+1} = F(x_k)$ converges to $\bar{x}$, with $d(x_k, \bar{x}) \leq \lambda^k d(x_0, \bar{x})$.
- The Lischitz constant of a C^1 map $F : \mathbb{R}^n \to \mathbb{R}^m$ is given by $\sup_x |||DF(x)|||$, where the norm $|||A|||$ for a matrix A is defined as

$$|||A||| := \sup_{h \neq 0} \frac{||Ah||}{||h||}.$$

- When $n = m$ and A is symmetric, we have $|||A||| = \max\{|\lambda_1|, \dots, |\lambda_n|\} = \max\{-\lambda_1, \lambda_n\}$, if $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ are the eignevalues of A.

We are now able to prove the above theorem.

Proof. Consider the map $F : \mathbb{R}^n \to \mathbb{R}^n$ given by $F(x) = x - \tau\nabla f(x)$. We have $DF(x) = \mathrm{I} - D^2 f(x)$, which is a symmetric matrix. Since we suppose $\alpha\mathrm{I} \leq D^2 f(x) \leq L\mathrm{I}$ we have $(1 - \tau L)\mathrm{I} \leq DF \leq (1 - \tau\alpha)\mathrm{I}$, so that the eigenvaues of DF are between $1 - \tau L$ and $1 - \tau\alpha$. Hence, $|||DF||| \leq \max\{1 - \tau\alpha, \tau L - 1\} := \lambda$. Our assumption guarantees $0 < \lambda < 1$, so that F is a contraction, and the space $\mathbb{R}^n$ is complete. Hence x_k converges to $\bar{x}$, a point characterized by $F(\bar{x}) = \bar{x}$, and we have the claimed exponential convergence. The point $\bar{x}$ satisfies $\bar{-}\tau\nabla f(\bar{x}) = \bar{x}$, so that we have $\nabla f(\bar{x}) = 0$ and, since f is convex, $\bar{x}$ is the minimizer of f. □

Note that the above proof provides the convergenc eof the points x_k to the optimizer $\bar{x}$. We can also look at the value of the function, and define $\varepsilon_k := f(x_k) - f(\bar{x})$, which is how much x_k is not optimal. Using the fact that f is locally Lipschitz continuous (f is C^1 and on bounded sets C^1 functions have bounded derivatives) we find $\varepsilon_k \leq C||x_k - \bar{x}||$, hence $\varepsilon_k \leq C\lambda^k$. Actually, we con obtain better. Indeed, close to the point $\bar{x}$ the Lipschitz constant of f is small, since $\nabla f(\bar{x}) = 0$. Using a second-order Taylor expansion we have

$$f(x) \leq f(y) + \nabla f(y) \cdot (x - y) + \frac{L}{2}||x - y||^2.$$

Applying this to $x = x_k$ and $y = \bar{x}$ we obtain

$$\varepsilon_k \leq \frac{L}{2}||x_k - \bar{x}||^2 \leq C\lambda^{2k}.$$

2.2 Projected gradient algorithm

We consider now constrained minimization problems, such as

$$\min\{f(x) \ : \ x \in K\},$$

where $K \subset \mathbb{R}^n$ is a closed convex set and f is a C^1 convex function. First of all, we establish the optimality conditions.

Proposition 2.2. *A point $x_0 \in K$ is a a solution of* $\min\{f(x) \ : \ x \in K\}$ *if and only if it satisfies*

$$\nabla f(x_0) \cdot (x - x_0) \geq 0 \quad \textit{for all } x \in K. \tag{2.1}$$

Proof. First, suppose that x_0 satisfies (2.1). Then, using $f(x) \geq f(x_0) + \nabla f(x_0) \cdot (x - x_0)$ we obtain $f(x) \geq f(x_0)$ for all $x \in K$, which proves the minimality of x_0. This part of the proof uses the convexity of f.

Then, suppose that x_0 is a minimizer, take $x \in K$ and define $x_t := (1 - t)x_0 + tx$ for $t \in [0, 1]$. The points x_t belong to K because K is convex. Hence, we have $f(x_t) \geq f(x_0)$, i.e. the function $[0, 1] \ni t \mapsto f(x_t)$ is minimal for $t = 0$. Its derivative is given by $\nabla f(x_t) \cdot (x - x_0)$ and at $t = 0$ it should be non-negative by minimality. □

A particular constrained optimization problem is that of the projection onto K.

Proposition 2.3. *Given a point $x_0 \in \mathbb{R}^n$ and a closed and convex set $K \subset \mathbb{R}^n$ we consider the problem*

$$\min\{||x - x_0||^2 \ : \ x \in K\}.$$

This problem admits a unique solution, that we call projection of x_0 onto K and denote by $P_K[x_0]$. Moreover, a point $x_1 \in K$ is the projection of x_0 onto K if and only if we have

$$(x - x_1) \cdot (x_1 - x_0) \geq 0 \quad \textit{for all } x \in K. \tag{2.2}$$

Finally, the map $\mathbb{R}^n \ni x_0 \mapsto P_K[x_0] \in K$ is 1-Lipschitz.

Proof. The existence of the minimizer is due to the fact that $x \mapsto ||x-x_0||^2$ is coercive. Its uniqueness is due to the fact that the same function is strictly convex. Setting $f(x) = \frac{1}{2}||x - x_0||^2$ and using $\nabla f(x) = x - x_0$ we obtain the characterization of the minimizer as a particular case of Proposition 2.2. Now, consider $x_0, y_0 \in \mathbb{R}^n$ and $x_1 = P_K[x_0], y_1 = P_K[y_0]$. We have, using twice (2.2)

$$(y_1 - x_1) \cdot (x_1 - x_0) \geq 0 \quad (x_1 - y_1) \cdot (y_1 - y_0) \geq 0.$$

If we sum these two inequalities we obtain

$$(y_1 - x_1) \cdot (x_1 - x_0 - y_1 + y_0) \geq 0,$$

which can be re-written as

$$||y_1 - x_1||^2 \leq (y_1 - x_1) \cdot (y_0 - x_0).$$

Using the Cauchy-Schwartz inequality (the scalar product is smaller than the product of the norms) we obtain

$$||y_1 - x_1||^2 \leq ||y_1 - x_1|| \cdot ||y_0 - x_0||,$$

which gives $||y_1 - x_1|| \leq ||y_0 - x_0||$, and the claim. □

We can now consider a variant of the fixed-step gradient algorithm which takes into account the constraint. The goal is to solve $\min\{f(x) \ : \ x \in K\}$. The algorithm is called *projected gradient* and uses an iterative seuence defined as

$$x_{k+1} = P_K[x_k - \tau \nabla f(x_k)].$$

Theorem 2.4. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a C^2 function with $\alpha \mathrm{I} \leq D^2 f(x) \leq L\mathrm{I}$ for two constants $L \geq \alpha > 0$. Suppose $\tau \in]0, \frac{2}{L}[$. Then, the sequence defined by the projected gradient algorithm converges to the unique minimizer $\bar{x}$ of f, and we have*

$$||x_k - \bar{x}|| \leq ||x_0 - \bar{x}||\lambda^k$$

where the number λ is given by $\lambda = \max\{1 - \tau\alpha, \tau L - 1\} < 1$.

Proof. The proof is based on the same idea as in Theorem 2.1. We consider the map $F(x) = P_K[x - \tau\nabla f(x)]$ and prove it is a contraction. Since P_K is 1-Lipschitz, and the map inside P_K was already proven to be a contraction under these very same assumptions, this is easy. Then, we obtain $||x_k - \bar{x}|| \leq ||x_0 - \bar{x}||\lambda^k$, but $\bar{x}$ is not (yet) the minimizer of f over K, but the fixed point of F. Hence, from $\bar{x} = P_K[\bar{x} - \tau\nabla f(\bar{x})]$ we deduce, using (2.2)

$$(x - \bar{x}) \cdot [\bar{x} - (\bar{x} - \tau\nabla f(\bar{x})] \geq 0 \quad \text{for all } x \in K.$$

This can be re-written as

$$(x - \bar{x}) \cdot \nabla f(\bar{x}) \geq 0 \quad \text{for all } x \in K,$$

which is exactly the condition for the optimality of $\bar{x}$. □

We proved exponential convergence of x_k to $\bar{x}$, we could wonder about ε_k. Differently from the unconstrained case we cannot use here the fact $\nabla f(\bar{x}) = 0$, so we cannot obtain a rate λ^{2k}. We can only use the Lipschitz behavior of f around $\bar{x}$ and obtain $\varepsilon_k \leq C\lambda^k$.

We note that the above algorithm is only useful when computing the projection P_K is "easy" (if possible, we would like an explicit formula for the projection ; if computing it, which means solving an optimization problem at each step, requires to run another algorithm to approximate the solution, then it is not necessarily a good idea to use the projected gradient). In particular *never use the projected gradient to compute a projection !* Why? since you would need a formula to compute the projection itself, which is ewactly what you try to find.

Example. *Among the sets for which computing the projection is easy we mention the rectangles: $K = [a_1, b_1] \times \cdots \times [a_n, b_n]$, and the ball $K = \overline{B(0, R)}$. In the first case the projection is given by*

$$(P_K[x])_i = \max\{\min\{x_i, b_i\}, a_i\}$$

and in the second case we have

$$P_K[x] = \frac{x}{\max\{1, |x|/R\}}.$$

2.3 Slower convergence and acceleration for non-elliptic smooth convex functions

If in the fixed-step gradient algorithm we use a function f which is convex but not elliptic, i.e. $\alpha = 0$, then we are not guaranteed that the map g given by $g(x) = x - \tau\nabla f(x)$ is a contraction. Hence, we have no proof of the exponential convergence of x_k to $\bar{x}$. Yet, if we suppose $D^2 f \leq L\mathrm{I}$, we can obtain a proof of convergence of ε_k to 0, even if much slower (and with no clue about the rate of convergence of x_k to $\bar{x}$.

In order to have an idea of what happens we first consider the continuous-in-time equation $x'(t) = -\nabla f(x(t))$ instead of the fixed-step gradient algorithm. We compute

$$\frac{d}{dt}\left(\frac{1}{2}||x(t) - \bar{x}||^2\right) = (x(t) - \bar{x}) \cdot x'(t) = -(x(t) - \bar{x}) \cdot \nabla f(x(t)) \leq f(\bar{x}) - f(x(t)) := -\varepsilon(t),$$

where we define $\varepsilon(t)$ as the optimality error of $x(t)$, i.e. $\varepsilon(t) = f(x(t)) - f(\bar{x})$. We can see that $\varepsilon(t)$ is non-increasing in t since

$$\frac{d}{dt}\left(f(x(t))\right) = \nabla f(x(t)) \cdot x'(t) = -||\nabla f(x(t))||^2 \leq 0.$$

Hence we have

$$T\varepsilon(T) \leq \int_0^T \varepsilon(t)dt \leq \int_0^T -\frac{d}{dt}\left(\frac{1}{2}||x(t) - \bar{x}||^2\right)dt = \frac{1}{2}||x(0) - \bar{x}||^2 - \frac{1}{2}||x(T) - \bar{x}||^2 \leq \frac{1}{2}||x(0) - \bar{x}||^2,$$

which provides $\varepsilon(T) \leq C/T$. In this computation we do not need the constant L, i.e. we do not need $D^2 f$ to be bounded from above neither. Yet, in order to handle the discrete case, we will need it. We try to mimick the same principle in the proof of the statement below.

Proposition 2.5. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a C^2 function with $D^2 f(x) \leq L\mathrm{I}$, which admits a minimizer $\bar{x}$. Suppose $\tau \in (0, \frac{1}{L})$. Then, the sequence defined by the fixed-step gradient descent algorithm satisfies $\varepsilon_{k+1} \leq C/k$, where $C = \frac{||x_0 - \bar{x}||^2}{2\tau}$.*

Proof. Let us consider

$$\begin{aligned} \frac{1}{2}||x_{k+1}\bar{x}||^2 - \frac{1}{2}||x_k - \bar{x}||^2 &= (x_{k+1} - x_k) \cdot \left(\frac{x_{k+1} + x_k}{2} - \bar{x}\right) && (2.3) \\ &= -\tau\nabla f(x_k) \cdot \left((x_k - \bar{x}) - \frac{\tau}{2}\nabla f(x_k))\right) && (2.4) \\ &\leq -\tau\varepsilon_k + \frac{\tau^2}{2}||\nabla f(x_k)||^2. && (2.5) \end{aligned}$$

We now consider how much f decreases from one step to the other. We have

$$f(x_{k+1}) \leq f(x_k) + \nabla f(x_k)\dot{(}x_{k+1} - x_k) + \frac{L}{2}||x_{k+1} - x_k||^2 = f(x_k) - \tau||\nabla f(x_k)||^2 + \frac{\tau^2 L}{2}||\nabla f(x_k)||^2.$$

Hence, f decreases at each step if $\tau^2 L \leq 2\tau$, i.e. $\tau \leq 2/L$. If we suppose $\tau \leq \frac{1}{L}$ we havve more: we obtain $-\tau + \frac{\tau^2 L}{2} \leq \frac{\tau}{2}$, hence

$$\varepsilon_{k+1} \leq \varepsilon_k - \frac{\tau}{2}||\nabla f(x_k)||^2,$$

which can be combined with (2.3). We then obtain

$$\frac{1}{2}||x_{k+1}\bar{x}||^2 - \frac{1}{2}||x_k - \bar{x}||^2 \leq -\tau\varepsilon_{k+1}$$

and, moreover, ε_k is non-increasing in k. Hence we have

$$\tau k\varepsilon_{k+1} \leq \tau\sum_{i=0}^{k}\varepsilon_{i+1} \leq \sum_{i=0}^{k}\frac{1}{2}||x_i - \bar{x}||^2 - \frac{1}{2}||x_{i+1} - \bar{x}||^2 \leq \frac{1}{2}||x_{0-}\bar{x}||^2,$$

which is the claim. □

Since C/k is much worse than λ^k, people have looked for ways to *accelerate* the convergence, and a very clever idea was presented by Nesterov in [4]. What Nesterov suggests is to apply at each step the same construction of the fixed-step gradient algorithm, i.e. passing from a point y to $y - \tau\nabla f(y)$, but instead of applying it to $y = x_k$ doing it to another point $y = y_k$, which is a clever combination of x_k and x_{k-1}. From the computational point of view, the cost is approximately the same (one unique computation of the gradient at each iteration), and what [4] proves is that we obain $\varepsilon_k \leq C/k^2$.

We will discuss the algorithm in details and provide a proof of this rate of convergence. We consider the following algorithm:

$$x_{k+1} = y_k - \tau\nabla f(y_k)\ ; \quad y_k = \text{ a linear combination of } x_k \text{ and } x_{k-1} \text{ using coefficients } (t_k)_k.$$

We start from the following estimates: for every x we have

$$f(x) \geq f(y_k) + \nabla f(y_k)\cdot(x - y_k)$$

as well as

$$f(x) \leq f(y_k) + \nabla f(y_k)\cdot(x - y_k) + \frac{L}{2}||x - y_k||^2.$$

We take $x = x_{k+1}$ in the second estimate and subtract from the first, so that we get

$$f(x) - f(x_{k+1}) \geq \nabla f(y_k)\cdot(x - x_{k+1}) - \frac{L}{2}||x_{k+1} - y_k||^2.$$

Then, we use $\nabla f(y_k) = (y_k - x_{k+1})/\tau$, and we choose $\tau = \frac{1}{L}$, so that we obtain

$$f(x) - f(x_{k+1}) \geq L(y_k - x_{k+1})\cdot(x - x_{k+1}) - \frac{L}{2}||x_{k+1} - y_k||^2. \tag{2.6}$$

We apply (2.6) to $x = x_k$ and obtain

$$\frac{2}{L}\left(\varepsilon_k - \varepsilon_{k+1}\right) \geq 2(y_k - x_{k+1})\cdot(x_k - x_{k+1}) - ||x_{k+1} - y_k||^2. \tag{2.7}$$

We also apply (2.6) to $x = \bar{x}$ and obtain

$$-\frac{2}{L}\varepsilon_{k+1} \geq 2(y_k - x_{k+1})\cdot(\bar{x} - x_{k+1}) - ||x_{k+1} - y_k||^2. \tag{2.8}$$

We multiply (2.7) by $t_{k+1} - 1$ and add it to (2.8) and obtain

$$\frac{2}{L}\left((t_{k+1} - 1)\varepsilon_k - t_{k+1}\varepsilon_{k+1}\right) \geq 2(y_k - x_{k+1})\cdot((t_{k+1} - 1)x_k + \bar{x} - t_{k+1}x_{k+1}) - t_{k+1}||x_{k+1} - y_k||^2.$$

We multiply everything by t_{k+1} and we assume that we have $t_{k+1}^2 - t_{k+1} \leq t_k^2$, so that we obtain

$$\frac{2}{L}\left(t_k^2\varepsilon_k - t_{k+1}^2\varepsilon_{k+1}\right) \geq 2t_{k+1}(y_k - x_{k+1}) \cdot ((t_{k+1}-1)x_k + \bar{x} - t_{k+1}x_{k+1}) - t_{k+1}^2||x_{k+1} - y_k||^2.$$

Using $2a \cdot b - ||a||^2 = -||a-b||^2 + ||b||^2$ with $a = t_{k+1}(y_k - x_{k+1})$ and $b = (t_{k+1}-1)x_k + \bar{x} - t_{k+1}x_{k+1}$ we can re-write the right-hand side and obtain

$$\frac{2}{L}\left(t_k^2\varepsilon_k - t_{k+1}^2\varepsilon_{k+1}\right) \geq ||(t_{k+1}-1)x_k + \bar{x} - t_{k+1}x_{k+1}||^2 - ||(t_{k+1}-1)x_k + \bar{x} - t_{k+1}y_k||^2.$$

We would like the right-hand side to give rise to a telescopic sum. We would like the last term to be equal to the previous one when exchanging $k+1$ with k. More precisely, if we set $u_k := (t_k - 1)x_{k-1} - t_kx_k + \bar{x}$ we would like to write the right-hand side above as $||u_{k+1}||^2 - ||u_k||^2$. For this, we need to impose

$$(t_{k+1}-1)x_k + \bar{x} - t_{k+1}y_k = (t_k - 1)x_{k-1} - t_kx_k + \bar{x}.$$

This requires to choose

$$y_k = x_k + \frac{t_k - 1}{t_{k+1}}(x_k - x_{k-1}).$$

This explains the precise choice of the point y_k, and the choice of the coefficients t_k has to comply with the condition $t_{k+1}^2 - t_{k+1} \leq t_k^2$. We then obtain the following.

Proposition 2.6. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a C^2 function with $D^2f(x) \leq LI$, which admits a minimizer $\bar{x}$. Suppose $\tau = \frac{1}{L}$. Fix a sequence $(t_k)_k$ such that $t_0 = 0$ and $t_{k+1}^2 - t_{k+1} \leq t_k^2$ and a point x_0. Define $x_{-1} = x_0$ and*

$$y_k = x_k + \frac{t_k - 1}{t_{k+1}}(x_k - x_{k-1}); x_{k+1} = y_k - \tau\nabla f(y_k)$$

Setting $\varepsilon_k = f(x_k) - f(\bar{x})$ we then obtain

$$\varepsilon_k \leq \frac{C}{t_k^2} \quad \text{where } C = \frac{L}{2}||x_0 - \bar{x}||^2.$$

Proof. Applying all the computations that we just described we obtain the inequality

$$\frac{2}{L}\left(t_k^2\varepsilon_k - t_{k+1}^2\varepsilon_{k+1}\right) \geq ||u_{k+1}||^2 - ||u_k||^2,$$

where $u_k := (t_k - 1)x_{k-1} - t_kx_k + \bar{x}$. We take the sum of these inequalities for k ranging from 0 to N and obtain

$$\frac{2}{L}\left(t_0^2\varepsilon_0 - t_{N+1}^2\varepsilon_{N+1}\right) \geq ||u_{N+1}||^2 - ||u_0||^2 \geq -||u_0||^2.$$

We use $t_0 = 0$ to get rid of the first term, change the sign and multiply by $L/2$ and obtain

$$t_{N+1}^2\varepsilon_{N+1} \leq \frac{L}{2}||u_0||^2.$$

We conclude using $u_0 := (t_0 - 1)x_{-1} - t_0x_0 + \bar{x}$, so that $||u_0|| = ||x_{-1} - \bar{x}|| = ||x_0 - \bar{x}||$. □

In the previous statement, we can use $t_k = (k+1)/2$ for $k \geq 1$, since we can easily check that we have

$$t_{k+1}^2 - t_{k+1} = \frac{(k+2)^2}{4} - \frac{k+2}{2} = \frac{k^2 + 2k}{4} \leq \frac{k^2 + 2k + 1}{4} = t_k^2.$$

Another reasonable choice is the one which saturates the inequality $t_{k+1}^2 - t_{k+1} \leq t_k^2$ making it an equality, i.e.

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}.$$

Anyway, in all these cases we have $t_k = O(k)$ and we obtain thus $\varepsilon_k \leq C/k^2$.

3 Non-smooth optimization

In the last section we removed the assumption that the objective function should be elliptic, but still we kept the assumption that it has to be smooth (i.e. ∇f should be Lipschitz continuous, $D^2 f$ bounded from above). We want now to consider the case where f is not necessarily smooth. In particular, convex functions could be non-differentiable.

3.1 Subdifferential and subgradient descent

One of the main tools to deal with non-differentiable convex functions is the deinition of subdifferential.

Definition 3.1. *Given $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ and a point $x_0 \in \mathbb{R}^n$ we define the subdifferential of f at x_0 as the set $\partial f(x_0) \subset \mathbb{R}^n$ defined as follows*

$$\partial f(x_0) := \{v \in \mathbb{R}^n \ : \ f(x) \geq f(x_0) + v \cdot (x - x_0) \ \textit{for all } x \in \mathbb{R}^n\}.$$

The elements of the subdifferential are called subgradients.

The above definition is inspired by the property which is satisfied when taking $v = \nabla f(x_0)$. Among the main properties of the subdifferential we cite the following.

- The subdifferential $\partial f(x)$ is always a closed and convex set, whatever is f.
- If f is l.s.c., the graph of the subdifferential multi-valued map is closed: take a sequence $x_n \to x_0$ and a sequence v_n with $v_n \to v$ and $v_n \in \partial f(x_n)$. Then $v \in \partial f(x_0)$. This is useful to obtain stability properties for the subdifferential.
- When dealing with arbitrary functions f, the subdifferential is in most cases empty, as there is no reason that the inequality defining $v \in \partial f(x_0)$ is satisfied for x very far from x_0. The situation is completely different when dealing with convex functions, which is the standard case where subdifferentials are defined and used. In this case we can prove that $\partial f(x_0)$ is never empty if x_0 lies in the interior of the set $\{f < +\infty\}$ (note that outside $\{f < +\infty\}$ the subdifferential of a proper function is clearly empty). In particular, for real-valued convex functions, the subdifferential is never empty.
- If f is convex and differentiable at a point x_0, then $\partial f(x_0) = \{\nabla f(x_0)\}$.
- A point x_0 solves $\min\{f(x) \ : \ x \in \mathcal{X}\}$ if and only if we have $0 \in \partial f(x_0)$.
- The subdifferential satisfies the monotonicity property

  $$v_i \in \partial f(x_i) \text{ for } i = 1, 2 \ \Rightarrow \ \langle v_1 - v_2, x_1 - x_2 \rangle \geq 0.$$

  This can be seen by writing

  $$f(x_1) \geq f(x_2) + v_2 \cdot (x_1 - x_2), \quad f(x_2) \geq f(x_1) + v_1 \cdot (x_2 - x_1),$$

  and summing up the two inequalities.

Example. *Several examples can be considered*

- *For the function $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = |x|$ we have*

$$\partial f(x) = \begin{cases} \{1\} & \textit{if } x > 0, \\ \{-1\} & \textit{if } x < 0, \\ [-1, 1] & \textit{if } x = 0. \end{cases}$$

- *For the function $f : \mathbb{R}^n \to \mathbb{R}$ given by $f(x) = ||x||$ we have*

$$\partial f(x) = \begin{cases} \{\frac{x}{||x||}\} & \textit{if } x \neq 0, \\ \overline{B(0,1)} & \textit{if } x = 0. \end{cases}$$

- *Different situations can occur at the boundary of $\{f < +\infty\}$. If we take for instance the proper function f defined via*

  $$f(x) = \begin{cases} x^2 & \textit{if } x \geq 0, \\ +\infty & \textit{if } x < 0, \end{cases}$$

  we see that we have $\partial f(0) = [-\infty, 0]$ so that the subdifferential can be "fat" on these boundary points. If we take, instead, the proper function f defined via

  $$f(x) = \begin{cases} -\sqrt{x} & \textit{if } x \geq 0, \\ +\infty & \textit{if } x < 0, \end{cases}$$

  we see that we have $\partial f(0) = \emptyset$, a fact related to the infinite slope of f at 0: of course, infinite slope can only appear at boundary points.

We also prove the following fact.

Proposition 3.2. *Take two convex functions f_1 and f_2, and suppose that f_1 is differentiable at a point x_0. Set $f = f_1 + f_2$. Then we have*

$$v \in \partial f(x_0) \Leftrightarrow v - \nabla f_1(x_0) \in \partial f_2(x_0).$$

In other words, $\partial f(x_0) = \nabla f_1(x_0) + \partial f_2(x_0)$. As a consequence, x_0 minimizes f if and only if $-\nabla f_1(x_0) \in \partial f_2(x_0)$.

Proof. If $v = \nabla f_1(x_0) + w$ with $w \in \partial f_2(x_0)$, then clearly we have $v \in \partial f(x_0)$. On the other hand, suppose $v \in \partial f(x_0)$ and take a point x_1. Define $x_t := (1-t)x_0 + tx_1$. Using $x_t - x_0 = t(x_1 - x_0)$, for $t \in [0,1]$ we have

$$f_1(x_t) + f_2(x_t) \geq f_1(x_0) + f_2(x_0) + tv \cdot (x_1 - x_0).$$

This can be re-written as

$$\frac{f_2(x_t) - f_2(x_0)}{t} \geq v \cdot (x_1 - x_0) - \frac{f_1(x_t) - f_1(x_0)}{t}.$$

Since the incremental ratios of convex functions are increasing we also obtain

$$\frac{f_2(x_1) - f_2(x_0)}{1} \geq \frac{f_2(x_t) - f_2(x_0)}{t} \geq v \cdot (x_1 - x_0) - \frac{f_1(x_t) - f_1(x_0)}{t}.$$

Taking the limit $t \to 0$ we then get

$$\frac{f_2(x_1) - f_2(x_0)}{1} \geq v \cdot (x_1 - x_0) - \nabla f(x_0) \cdot (x_1 - x_0),$$

which shows $v - \nabla f_1(x_0) \in \partial f_2(x_0)$. □

A reasonable algorithm that one could imagine to replace the gradient descent inthe case of non-differentiable functions could be the following: given x_k take $v_k \in \partial f(x_k)$ and then update $x_{k+1} = x_k - \tau v_k$. The problem with this algorithm is that at the point $\bar{x}$ which minimizes f we have $0 \in \partial f(\bar{x})$, but close to this point the vectors which belong to the sudifferential are not necessarily small. This is what happens in the case of the modulus or the norm functions, where outside the origin all vectors in teh subdifferential have unit norm. This prevents the sequence to converge, since we would have $||x_{k+1} - x_k|| \geq c\tau$. The correct version of the algorithm requires to use a non-constant step τ_k. The convergence result (that we do not prove) is the following.

Theorem 3.3. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function which admits a minimizer and $(\tau_k)_k$ a sequence such that $\tau_k > 0$, $\sum\limits_k \tau_k = +\infty$, $\sum\limits_k \tau_k^2 < +\infty$. Then, any sequence (x_k, v_k) satisfying*

$$v_k \in \partial f(x_k), \quad v_k \neq 0, \quad \text{and} \quad x_{k+1} = x_k - \tau_k \frac{v_k}{||v_k||}$$

is such that $x_k \to \bar{x}$, where $\bar{x}$ is a minimizer of f.

The proof of this fact can be found in Theorem 9.3 of [2].

From the computational point of view The drawback of this algorithm is that it imposes to choose a variable stepsiez τ_k which should tend to 0, and this implies that the algorithm is necessarily slow (it can be proven that it cannot be as fast as the fides-step gradient algorithm, i.e. that we cannot have $||x_k - \bar{x}|| \leq C\lambda^k$ for $\lambda < 1$. This explains the quest for alternative algorithms for non-smooth optimization, based on different principles.

3.2 Proximal methods

Let us think at what we actually do in the fixed-step algorithm. If we want to minimize a function f at each step we take a point x_k, compute $\nabla f(x_k)$, and tehn move to $x_{k+1} = x_k - \tau\nabla f(x_k)$. We can also say that the point x_{k+1} is characterized by

$$x_{k+1} = \operatorname*{argmin}_x \tilde{f}(x) := f(x_k) + \nabla f(x_k) \cdot (x - x_k) + \frac{1}{2\tau}||x - x_k||^2,$$

i.e. x_{k+1} minimizes a quadratic function $\tilde{f}$ that is chosen as an approximation of f around x_k. The function $\tilde{f}$ recalls a Taylor expansion of f, but the second-order term $\frac{1}{2}D^2 f(x_k)(x - x_k) \cdot (x - x_k)$ has been replaced with a simpler second-order term. If we had used $\tilde{f}(x) := f(x_k) + \nabla f(x_k) \cdot (x - x_k) + \frac{1}{2}D^2 f(x_k)(x - x_k) \cdot (x - x_k)$ we would have found as an algorithm the Newton's method for finding the solution of $\nabla f = 0$ (indeed, expanding the function f at order two is the same as expanding ∇f at order one). Yet, Newton's method requires to invert $D^2 f(x_k)$ and requires high regularity to converge, which is not the first goal here. Another interesting point of not choosing the exact 2nd order Taylor expansion is that it allows to obtain $\tilde{f} \geq f$. This works as soon as $\tau \leq \frac{1}{L}$, which is an assumption we already used many times. The advantage of having $\tilde{f} \geq f$ is that the optimality of x_{k+1} provides $f(x_{k+1}) \leq \tilde{f}(x_{k+1}) \leq \tilde{f}(x_k) = f(x_k$, so that the algorithm let f decrease.

Now, the question is: what to do if f is not smooth? take for instance a function f of the form $f = f_1 + g$, with f_1 smooth but g only convex. In this case a good choice of $\tilde{f}$ is given by

$$\tilde{f}(x) := f_1(x_k) + \nabla f_1(x_k) \cdot (x - x_k) + \frac{1}{2\tau}||x - x_k||^2 + g(x),$$

where we only replace with a second-order polynomial the smooth part f_1. Minimizing $\tilde{f}$ isequivalent to solving

$$\min g(x) + \frac{||x - (x_k - \tau\nabla f_1(x_k))||^2}{2\tau},$$

which means applying a proximal operator. These operators are defined here.

Definition 3.4. *Given $g : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ a convex function we define the proximal operator of g as follows:*

$$\text{Prox}_{\tau,g}[x_0] := \operatorname*{argmin}_x g(x) + \frac{||x - x_0||^2}{2\tau},$$

where $\tau > 0$ is a parameter. Note that the minimizer of $g(x) + \frac{||x-x_0||^2}{2\tau}$ exists (because g is convex, hence bounded from below by an affine function, and adding a quadratic penalization we obtain a coercive function) and is unique (because g is convex, and adding a quadratic penalization we obtain a strictly convex function).

Among the main properties of the proximal operator we cite the following.

- A point y_0 equals $\mathrm{Prox}_{\tau,g}[x_0]$ if and only if $-\frac{y_0-x_0}{\tau} \in \partial g(y_0)$.
- For any $\tau > 0$, the map $x_0 \mapsto \mathrm{Prox}_{\tau,g}[x_0]$ is 1-Lipschitz. This can be seen by taking x_0, x_1 and $y_i = \mathrm{Prox}_{\tau,g}[x_i]$. We then write $-\frac{y_i-x_i}{\tau} \in \partial g(y_i)$ and use the monotonicity property of the subdifferential, thus obtaining

  $$(x - 1 - y_1 - x_0 + y_0) \cdot (y_1 - y_0) \geq 0.$$

  This can be written as

  $$||y_1 - y_0||^2 \leq (x_0 - x_1) \cdot (y_1 - y_0) \leq ||y_1 - y_0|| \cdot ||x_1 - x_0||$$

  and allows to obtain $||y_1 - y_0|| \leq ||x_1 - x_0||$.
- In the particular case where g is the function given by

  $$g(x) = \begin{cases} 0 & \text{if } x \in K \\ +\infty & \text{if } x \notin K, \end{cases}$$

  for a convex set $K \subset \mathbb{R}^n$ (this function is called indicator function of K, in convex analysis, and denoted by I_K), we have $\mathrm{Prox}_{\tau,g} = P_K$ for every $\tau > 0$.

The use of the proximal operator allows to define a new algorithm, different from the subgradient descent algorithm, in order to attack non-smooth optimization problems. The algorithm, called proximal gradient algorithm, will work as follows: when minimizing $f_1 + g$, we take $\tau > 0$ and define an iterative sequence via

$$x_{k+1} = \mathrm{Prox}_{\tau,g}[x_k - \tau \nabla f_1(x_k)].$$

We then have the following theorem.

Theorem 3.5. *Let $f_1 : \mathbb{R}^n \to \mathbb{R}$ be a C^2 function with $\alpha \mathrm{I} \leq D^2 f(x) \leq L\mathrm{I}$ for two constants $L \geq \alpha > 0$, and $g : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ a convex function. Suppose $\tau \in]0, \frac{2}{L}[$. Then, the sequence defined by the proximal gradient algorithm converges to the unique minimizer $\bar{x}$ of $f := f_1 + g$, and we have*

$$||x_k - \bar{x}|| \leq ||x_0 - \bar{x}|| \lambda^k$$

where the number λ is given by $\lambda = \max\{1 - \tau\alpha, \tau L - 1\} < 1$.

Proof. The idea is the same as for the projected gradient algorithm. We define $F : \mathbb{R}^n \to \mathbb{R}^n$ via $F(x) = \mathrm{Prox}_{\tau,g}[x - \tau \nabla f_1(x)]$; we check that F is a contraction, since $x \mapsto x - \tau \nabla f(x)$ is $\lambda-$Lipchitz, and $\mathrm{Prox}_{\tau,g}$ is $1-$Lipschitz. Then the sequence converges exponentially to $\bar{x}$, defined as the fixed point of F. We now need to prove that $\bar{x}$ is the minimizer of f. The condition $\bar{x} = \mathrm{Prox}_{\tau,g}[\bar{x} - \tau \nabla f_1(\bar{x})]$ implies

$$-\frac{\bar{x} - (\bar{x} - \tau \nabla f_1(\bar{x}))}{\tau} \in \partial g(\bar{x}),$$

which means $-\nabla f_1(\bar{x}) \in \partial g(\bar{x})$. this is exactly the condition to minimize $f_1 + g$ □

As for the projected gradient algorithm (which is a particular case of the proximal algorithm, when taking $g = I_K$), this procedure is only useful if we know how to compute the proximal operator explicitly. This is the case, for instance, of a convex function widely used in application to data sciences, i.e. $g(x) = ||x||_1 = \sum_i |x^i|$ (please note that the index i is a coordinate here, and not the iteration step of the algorithm), as we can see in the following example.

Example. *Suppose $n = 1$ and take $g(x) = |x|$. Given $x_0 \in \mathbb{R}$, it is clear that $\mathrm{Prox}_{\tau,g}[x_0]$ and x_0 shoud have the same sign (because, otherwise, changing the sign would make the point closer to x_0 without changing the value of g). Suppose for simplicity $x_0 \geq 0$. We can then find $\mathrm{Prox}_{\tau,g}[x_0]$ by solving $\min_{x\geq 0} x + \frac{|x-x_0|^2}{2\tau}$. Differentiatng, we get $1 + (x - x_0)/\tau$. The derivative vanishes at $x = x_0 - \tau$. If such a point is in the domain $x \geq 0$, then it is the minimizer. Otherwise the mnimizer is 0. So the solution of the minimization problem is given by $(x_0 - \tau)_+$. More generally, we have $\mathrm{Prox}_{\tau,g}[x_0] = \mathrm{sign}(x_0)(|x_0| - \tau)_+$. This procedure is called shrinkage-thresholding since we shrink x_0 moving it towards the origin, with a threshold effect (if closer than τ, we stop at the origin).*

In higer dimension the same construction can be done componentwise: every time that a function g is separable (i.e. it is a sum of functions of different components) we can exploit the fact that also the quadratic penalization is separable and treat separately each variable. We then obtain, for $g(x) = ||x||_1$,

$$(\mathrm{Prox}_{\tau,g}[x_0])^i = \mathrm{sign}(x_0^i)(|x_0^i| - \tau)_+.$$

In particular, it turns out that it is quite easy to use this algorithm to solve $\min f_1(x) + ||x||_1$ when f_1 is smooth and elliptic. On the other hand, a typical example would be to take $f_1(x) = ||Ax - b||_2^2$, which in many cases is not elliptic. In this case we can adapt the results of Section 2.3 (but we won't prove them) and obtain the following.

Proposition 3.6. *Let $f_1 : \mathbb{R}^n \to \mathbb{R}$ be a C^2 function with $D^2 f(x) \leq L\mathrm{I}$ which admits a minimizer $\bar{x}$, and let $g : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a convex function. Take $\tau = 1/L$. Then, the sequence defined by the proximal gradient algorithm satisfies $\varepsilon_{k+1} \leq C/k$, where $\varepsilon_k := f(x_k) - f(\bar{x})$ and $C = L||x_0 - \bar{x}||^2$.*

If, instead, we fix a sequence $(t_k)_k$ such that $t_0 = 0$ and $t_{k+1}^2 - t_{k+1} \leq t_k^2$, we take a point x_0, define $x_{-1} = x_0$, and take a sequence (x_k, y_k) such that

$$y_k = x_k + \frac{t_k - 1}{t_{k+1}}(x_k - x_{k-1}); x_{k+1} = \mathrm{Prox}_{\tau,g}[y_k - \tau\nabla f(y_k)],$$

setting $\varepsilon_k = f(x_k) - f(\bar{x})$ we then obtain

$$\varepsilon_k \leq \frac{C}{t_k^2} \quad \text{where } C = \frac{L}{2}||x_0 - \bar{x}||^2.$$

These results are proven, for instance, in [1].

4 Convex duality and algorithms using duality

We consider here some tools from convex analysis, with the goal to associate with some classes of optimization problems a corresponding dual problem, which could be useful to develop algorithms to solve the primal one exploiting the dual one.

4.1 Fenchel-Legendre transform

Definition 4.1. *We say that a function valued in $\mathbb{R} \cup \{+\infty\}$ is proper if it is not identically equal to $+\infty$. The set $\{f < +\infty\}$ is called the domain of f.*

Definition 4.2. *Given a proper function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ we define its Fenchel-Legendre transform $f^* : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ via*

$$f^*(\xi) := \sup_x \xi \cdot x - f(x).$$

We observe that we trivially have $f^*(0) = -\inf_{\mathbb{R}^n} f$.

We note that f^*, as a sup of affine functions, is both convex and l.s.c., as these two notions are stable by sup. We indeed have the following lemma.

Lemma 4.3. *Given a family of functions $f_\alpha : X \to \mathbb{R} \cup \{+\infty\}$, define $f(x) := \sup_\alpha f_\alpha(x)$. Then, if the functions f_α are all l.s.c. f is also l.s.c. if they are all convex, f is also convex.*

Proof. Assume that the functions f_α are all l.s.c. Take $x_n \to x$ and write

$$f_\alpha(x) \le \liminf_n f_\alpha(x_n) \le \liminf_n f(x_n).$$

It is then enough to take the sup over α in the left hand side in order to obtain

$$f(x) \le \liminf_n f(x_n),$$

which is the desired result.

For convexity, assume that the functions f_α are all convex, and take x, y and write

$$f_\alpha((1-t)x + ty) \le (1-t)f_\alpha(x) + tf_\alpha(y) \le (1-t)f(x) + tf(y).$$

Again, taking the sup over α in the left hand side provides the desired inequality. □

We then note that, actually, not only any sup of affine functions is convex and l.s.c. but any convex and l.s.c. function is indeed a sup of affine functions. A rough proof of this fact can be obtained by considering for every point $x \in \mathbb{R}^n$ the (or a) tangent to the graph of f at $(x, f(x))$. This actually works well whenever f is finite-valued, while some attention has to be paid to the points where f takes the value $+\infty$, or to the boundary of the domain of f in the general case. By the way, the assumption that f is l.s.c. is useless when f is finite-valued, since finite convex functions are always locally Lipschitz and hence continuous.

We then consider the following characterization.

Proposition 4.4. *Consider a function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ which is proper, convex, and l.s.c. Then*

a) there exists $g : \mathcal{X}^* \to \mathbb{R} \cup \{+\infty\}$ such that $f = g^*$;

b) we have $f^{**} = f$.

Proof. Once that we know that f is a sup of affine functions we can write

$$f(x) = \sup_\alpha \xi_\alpha \cdot x \ + c_\alpha$$

for a family of indices α. We then set $c(\xi) := \sup\{c_\alpha : \xi_\alpha = \xi\}$. The set in the sup can be empty, which would mean $c(\xi) = -\infty$. Anyway, the sup is always finite: fix a point x_0 with $f(x_0) < +\infty$ and use since $c_\alpha \le f(x_0) - \langle \xi, x_0 \rangle$. We then define $g = -c$ and we see $f = g^*$.

Finally, before proving $f = f^{**}$ we prove that for any function f we have $f \ge f^{**}$ even if f is not convex or l.s.c. Indeed, we have $f^*(\xi) + f(x) \ge \xi \cdot x$ which allows to write $f(x) \ge \xi \cdot x - f^*(\xi)$, an inequality true for every ξ. Taking the sup over ξ we obtain $f \ge f^{**}$. We now want to prove that this inequality is an equality if f is convex and l.s.c. We write $f = g^*$ and transform this into $f^* = g^{**}$. We then have $f^* \le g$ and, transforming this inequality (which changes its sign), $f^{**} \ge g^* = f$, which proves $f^{**} = f$. □

A nice connection between Fenchel-Legendre transforms and subdifferentials is the following.

Proposition 4.5. *If f is convex and l.s.c., the subdifferentials of f and f^* are related through*

$$\xi \in \partial f(x) \Leftrightarrow x \in \partial f^*(\xi) \Leftrightarrow f(x) + f^*(\xi) = \xi \cdot x$$

Proof. Once we know that for convex and l.s.c. functions we have $f^{**} = f$, it is enough to prove $\xi \in \partial f(x) \Leftrightarrow f(x) + f^*(\xi) = \xi\dot{x}$ since then, by symmetry, we can also obtain $x \in \partial f^*(\xi) \Leftrightarrow f(x) + f^*(\xi) = \xi \cdot x$. We now look at the definition of subdifferential, and we have

$$\begin{aligned}\xi \in \partial f(x) \quad &\Leftrightarrow \quad \text{for every } y \in \mathcal{X} \text{ we have } f(y) \geq f(x) + \xi \cdot (y - x)\\ &\Leftrightarrow \quad \text{for every } y \in \mathcal{X} \text{ we have } \xi \cdot x - f(x) \geq \xi \cdot y - f(y)\\ &\Leftrightarrow \quad \xi \cdot x - f(x) \geq \sup_y \xi \cdot y - f(y)\\ &\Leftrightarrow \quad \xi \cdot x - f(x) \geq f^*(\xi).\end{aligned}$$

This shows that $\xi \in \partial f(x)$ is equivalent to $\xi \cdot x \geq f(x) + f^*(\xi)$, which is in turn equivalent to to $\xi \cdot x = f(x) + f^*(\xi)$, since the opposite inequality is always true by definition of f^*. □

A corollary of the above fact is the following; take two proper, convex and l.s.c. conjugate functions f and f^* (with $f = f^{**}$); then f is a real-valued C^1 function on $\mathbb{R}^n$ if and only if f^* is strictly convex and superlinear. This can be seen in the following way: f is a real-valued C^1 function on $\mathbb{R}^n$ if for every x we find a unique ξ in the set $\partial f(x)$; on the other hand, f^* is strictly convex if for every x we find at most one point ξ such that $x \in \partial f^*(\xi)$, and superlinear if for every x we find at least one point ξ such that $x \in \partial f^*(\xi)$.

All the above properties can be verified on the following examples.

Example. *Given $p > 1$, consider $f(x) = \frac{1}{p}||x||^p$. Then we have $f^*(\xi) = \frac{1}{q}||\xi||^q$, where q is the dual exponent of p, characterized by $\frac{1}{p} + \frac{1}{q} = 1$. For $p = 1$, setting $f(x) = ||x||$, we have*

$$f^*(\xi) = \begin{cases} 0 & \text{if } ||\xi|| \leq 1,\\ +\infty & \text{it not.}\end{cases}$$

Let us consider the last case in the one-dimensional setting (i.e. $f(x) = |x|$, $x \in \mathbb{R}$). In this case we have $\partial f(x) \supset \{1\}$ for every $x \geq 0$, and every $x \geq 0$ is such that $x \in \partial f^*(1)$. All the other cases in the relations described in Proposition 4.5 can be easily checked.

We also discuss a connection between proximal operators and Legendre transforms.

Proposition 4.6. *We have the following equality*

$$\mathrm{Prox}_{1,g} + \mathrm{Prox}_{1,g^*} = id.$$

Proof. Take a point y and call $x := \mathrm{Prox}_{1,g}[y]$. The point x is characterized by $y - x \in \partial g(x)$. This implies, using Proposition 4.5, that we also have $x \in \partial g^*(y - x)$. Setting $z = y - x$ we can write this as $y - z \in \partial g^*(z)$, i.e. $z = \mathrm{Prox}_{1,g^*}[y]$. The condition $x + z = y$ allows to conclude, as we wanted to prove $\mathrm{Prox}_{1,g}[y] + \mathrm{Prox}_{1,g^*}[y] = y$. □

4.2 Dual problems

In this section we introduce the notion of dual problem of a convex optimization problem through an inf-sup exchange procedure. This often requires to write possible constraints as a sup penalization.

We start as an example from the following problem:

$$(\mathbf{P}) \quad \min\left\{f(x) \ : \ x \in \mathbb{R}^n \ g_i(x) \leq c_i \ \ i = 1, \dots, m\right\},$$

where the functions f, g_i $(i = 1, \dots, m)$ are convex. We also consider the function $g : \mathbb{R}^n \to \mathbb{R}^m$ whose components are the functions g_i and we write $g \leq c$ (when writing inequalities involving vectors, we mean componentwise inequalities).

We then observe that we have

$$\sup_{\lambda\in\mathbb{R}^m_+} \lambda\cdot(g(x)-c) = \begin{cases} 0 & \text{if } g(x)\le c,\\ +\infty & \text{if not.}\end{cases}$$

Since we can always replace a constraint in an optimization problem by adding a function which takes the value 0 if it is satisfied and $+\infty$ if not, the above optimization problem is equivalent to

$$\min_{x\in\mathbb{R}^n}\sup_{\lambda\in\mathbb{R}^m_+} f(x)+\lambda\cdot(g(x)-c).$$

We get now to a problem of the form

$$\inf_{x\in X}\sup_{\lambda\in\Lambda} L(x,\lambda)$$

where, in this case, $L(x,\lambda) := f(x)+\lambda\cdot(g(x)-c)$.

This is an inf-sup problem, and we can associate with it a second optimization problem, obtained by switching the order of the inf and the sup. We can consider

$$\sup_{\lambda\in\Lambda}\inf_{x\in X} L(x,\lambda),$$

which means maximizing over λ the function G obtained as the value of the inf over x: $G(\lambda) := \inf_{x\in X} L(x,\lambda)$. We denote by F the function obtained by maximizing first in λ, i.e. $F(x) := \sup_{\lambda\in\Lambda} L(x,\lambda)$. In this very example the function F coincides with f if the constraint is met, otherwise it is $+\infty$.

We would like the two above optimization problems ("inf sup" and "sup inf") to be related to each other, and for instance their values to be the same.

Given an arbitrary function L the values of inf sup and of sup inf are in general different, as we can see from this very simple example: take $X=\Lambda=\{\pm1\}$ and $L(x,\lambda)=\lambda x$. In this case we have $\inf\sup = 1 > \sup\inf = -1$. Here the two values are different, and the inf sup is larger than the sup inf. Actually, this inequality is always true. Indeed, for arbitrary x,λ we have by definition of F, G

$$F(x)\ge L(x,\lambda)\ge G(\lambda).$$

Ignoring the term in the middle we see that all the values of F are larger than all the values of G, hence $\inf F\ge\sup G$.

We then want to discuss the connection between minimizing F, maximizing G, and finding saddle points of L.

Definition 4.7. *Given a function $L : X\times\Lambda\to\overline{R}$ we say that a pair (x_0,λ_0) is a saddle point of L if $x\mapsto L(x,\lambda_0)$ is minimal for $x=x_0$ and $\lambda\mapsto L(x_0,\lambda)$ is maximal for $\lambda=\lambda_0$.*

We have the following theorem.

Theorem 4.8. *Suppose $\inf F=\sup G$. Then the two following conditions are equivalent:*

a) (x_0,λ_0) is a saddle point of L,

b) x_0 is a minimizer for F and λ_0 a maximizer for G.

Proof. If (x_0,λ_0) is a saddle point of L, this means $G(\lambda_0)=L(x_0,\lambda_0)$ (since $x\mapsto L(x,\lambda_0)$ is minimal for $x=x_0$) and $F(x_0)=L(x_0,\lambda_0)$ (since $\lambda\mapsto L(x_0,\lambda)$ is maximal for $\lambda=\lambda_0$). Hence we have $F(x_0)=G(\lambda_0)$ but we also know that for every x we have $F(x)\ge G(\lambda_0)$, hence x_0 is a minimizer for F, and for every λ we have $G(\lambda)\le F(x_0)$, hence λ_0 a maximizer for G.

If on the contrary x_0 is a minimizer for F and λ_0 a maximizer for G, then we have

$$\inf F=F(x_0)\ge L(x_0,\lambda_0)\ge G(\lambda_0)=\sup G=\inf F,$$

and all inequalities are equalities. Hence, we obtain $F(x_0)\ge L(x_0,\lambda_0)$, thus $\lambda\mapsto L(x_0,\lambda)$ is maximal for $\lambda=\lambda_0$, and $G(\lambda_0)=L(x_0,\lambda_0)$, thus $x\mapsto L(x,\lambda_0)$ is minimal for $x=x_0$. Then (x_0,λ_0) is a saddle point of L. □

We note that in the above statement, the implication 1. implies 2. does not need the assumption $\inf F = \sup G$.

The idea of duality is then that we can solve at the same time the primal problem $\min F$ and the dual problem $\max G$ if we are able to find all saddle points of L. Moreover, we can solve $\min F$ if we are able to solve the dual problem $\max G$, find a maximizer λ_0, and then find all saddle points of L of the form (x_0, λ_0). This requires to solve the optimization problem $\min_x L(x, \lambda_0)$. The interest is that, in many cases, this helps in getting rid of the constraints. For instance, in the case of problem **(P)** where $L(x, \lambda) := f(x) + \lambda \cdot (g(x) - c)$, instead of considering the "hard" optimization problem

$$\min \{f(x) \ : \ x \in \mathbb{R}^n \ g(x) \leq c\},$$

where the constraints could be very annoying, we have to solve

$$\max \{G(\lambda) \ : \ \lambda \geq 0\},$$

which is also a constrained optimization problem (because of $\lambda \geq 0$), but with a much nicer constraint (the set being $\Lambda := \mathbb{R}^m_+$, and the projection on it is explicit: $P_\Lambda(x^1, \dots, x^m) = (x^1_+, \dots, x^m_+)$, which allows to efficiently use the projected gradient algorithm), and then the unconstrained problem $\min_x f(x) + \lambda_0 \cdot (g(x) - c)$. This construction also corresponds to finding the good value of the Lagrange multiplier λ, and the optimality condition $\nabla f(x_0) + \lambda_0 \cdot Dg(x_0) = 0$ is exactly what we expect from the Lagrange multipliers theory.

We need now to prove that, in the case of problem **(P)**, we have $\inf F = \sup G$, at least under suitable assumptions.

Theorem 4.9. *Consider $L(x, \lambda) = f(x) + \lambda_0 \cdot (g(x) - c)$, with $X = \mathbb{R}^n$ and $\Lambda = \mathbb{R}^m_+$, so that we have*

$$F(x) = \begin{cases} f(x) & \text{if } g(x) \leq c, \\ +\infty & \text{if not,} \end{cases}$$

and

$$G(\lambda) := \inf_{x \in \mathbb{R}^n} f(x) + \lambda_0 \cdot (g(x) - c).$$

Suppose that f is a coercive convex function, and that all g_i are convex. Then we have $\inf F = \sup G$.

Proof. We define a function $h : \mathbb{R}^m \to \overline{\mathbb{R}}$ via

$$h(p) := \inf\{f(x) \ : \ g(x) + p \leq c\}.$$

We compute h^*:

$$h^*(\lambda) = \sup_p \lambda \cdot p - h(p) = \sup_{p, x : g(x) + p \leq c} \lambda \cdot p - f(x).$$

We note that we can subract arbitrary positive numbers from the components of p and still satisfy the constraints, so that if λ is not a positive vector the sup on the right-hand side will be $+\infty$, by choosing a vector p with a very negative component correponding to a $\lambda_i < 0$. Then we consider the case where $\lambda \geq 0$ and in this case in the optimization it is convenient to choose p as large as possible, hence $p = c - g(x)$. We then obtain

$$h^*(\lambda) = \begin{cases} \sup_x \lambda \cdot (c - g(x)) - f(x) = -G(\lambda) & \text{if } \lambda \geq 0, \\ +\infty & \text{if not.} \end{cases}$$

We then use $f^*(0) = -\inf f$ applied to h^*, thus obtaining

$$h(0) = \inf F; \quad h^* * (0) = -\inf -G = \sup G,$$

and the proof is concluded if we have $h = h^{**}$. For this, we need to prove that h is convex and l.s.c. Let us start from convexity, and fix p_0, p_1, together with the corresponding minimizers x_0, x_1 (which exist since f is coercive and the set $\{x \,:\, g(x) + p \leq c\}$ is closed), satisfying $h(p_j) = f(x_j)$ and $g(x_j) + P_j \leq c$ for $j = 0, 1$. We then define $x_t := (1-t)x_0 + tx_1$ and $p_t := (1-t)p_0 + tp_1$. For every i we use

$$g_i(x_t) + p_t \leq (1-t)g_i(x_0) + tg_i(x_1) + (1-t)p_0 + tp_1 \leq (1-t)c + tc = c,$$

so that x_t is admissible in the optimization problem defining $h(p_t)$. Then we have

$$h(p_t) \leq f(x_t) \leq (1-t)f(x_0) + tf(x_1) = (1-t)h(p_0) + th(p_1),$$

which proves convexity.

For lower semicontinuity, we consider a sequence $p_n \to p_0$ and the corresponding sequence of optimizers x_n. If $h(p_n) = f(x_n)$ is unbounded there is nothing to prove, otherwise we can extract a converging subsequence from x_n, say $x_{n_k} \to x_0$. The condition $g(x_n) + p_n \leq c$ passes to the limit and gives $g(x_0) + p_0 \leq c$, so that x_0 is admissible in the optimization problem defining $h(p_0)$ and we have

$$h(p_0) \leq f(x_0) \leq \liminf_k f(x_{n_k}) = \liminf_k h(p_{n_k}),$$

which proves the semicontinuity. □

This is a nice application of the notion of Fenchel-Legendre transform to convex duality, but there are cases where this notion appears even more.

Consider for instance the case of linear equality constraints:

$$\min \{f(x) \;:\; x \in \mathbb{R}^n,\, Ax = b\},$$

where A is a linear map between $\mathbb{R}^n$ and $\mathbb{R}^m$, $b \in \mathbb{R}^m$ is a fixed vector and $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a given convex and l.s.c. function. We will denote by A^t the transpose operator of A, a linear mapping defined on $\mathbb{R}^m$, valued into $\mathbb{R}^n$, and characterized by

$$A^t\xi \cdot x = \xi \cdot Ax \text{ for all } \xi \in R^m \text{ and } x \in \mathbb{R}^n.$$

We can see that the above problem is equivalent to

$$\min \left\{ f(x) + \sup_{\xi \in \mathbb{R}^m} \xi \cdot (Ax - b) \; x \in \mathbb{R}^n \right\}.$$

In this case, in order to enforce an equality constraint, we do not need to impose a sign on the dual variable ξ. Setting $L(x, \xi) := f(x) + \xi \cdot (Ax - b)$, we then consider the dual problem

$$\sup_\xi \inf_x L(x, \xi).$$

We can then give a better expression to this new problem, that we will call dual problem. Indeed we have

$$\sup_\xi \inf_x L(x, \xi) = \sup_\xi -\xi \cdot b + \inf_x f(x) + \xi \cdot Ax.$$

We then rewrite $\xi \cdot Ax$ as $A^t\xi \cdot x$ and change the sign in the inf so as to write it as a sup. We do obtain

$$\sup_\xi \inf_x L(x, \xi) = \sup_\xi -\xi \cdot b - \sup_x -f(x) + -A^t\xi \cdot x.$$

We now recognize in the sup over x the form of a Fenchel-Legendre transform and we finally obtain

$$\sup_\xi \inf_x L(x, \xi) = \sup_\xi -\xi \cdot b - f^*(-A^t\xi).$$

This is a convex optimization problem in the variable ξ (the maximization of the sum of a linear functional and the opposite of a convex function, f^*, applied to a linear function of ξ), involving the Legendre transform of the original objective function f.

It is possible to prove in a very similar way to what done in Theorem 4.9 that we indeed have equality between the inf in the rpimal and the sup in the dual, or even to see it as a particular case of Theorem 4.9, writing the equality $Ax = b$ as two inequalities $Ax \leq b$ and $-Ax \leq -b$. We now discuss some consequences and some other variants.

A first consequence concerns the necessary optimality conditions. If x_0 and ξ_0 are optimal, then we have

$$f(x_0) = -\xi_0 \cdot b - f^*(-A^t\xi_0) \quad \text{and} \quad Ax_0 = b.$$

This can be re-written as

$$f(x_0) + f^*(-A^t\xi_0) = -\xi_0 \cdot b = -\xi_0 \cdot Ax_0 = -A^t\xi_0 \cdot x_0,$$

i.e. we have equality in the inequality $f(x) + f^*(y) \geq x \cdot y$. This is equivalent to

$$x_0 \in \partial f^*(-A^t\xi_0) \quad \text{and} \quad -A^t\xi_0 \in \partial f(x_0).$$

We can noteonce more the similarity with Lagrange multipliers, where optimizing a function f under a linear constraint of the form $Ax = b$ can be translated into the fact that ∇f should belong to a subspace, orthogonal to the affine space of the constraints, which is indeed the image of A^t.

the main variant that we want to consider is the following one:

$$\min \{f(x) + g(Ax)\},$$

where $g = I_{\{b\}}$ corresponds to the previous example, or $g = I_K$, for $K = \{y \in \mathbb{R}^m : y \leq b\}$ corresponds to the inequality constrait case.

In the case where we use a generic function g, we do not have constraints to write as a sup, but we can decide to write one of the two functions f or g as a sup thanks to the double Legendre transform. We then set

$$L(x, \xi) := f(x) + \xi \cdot Ax - g^*(\xi)$$

and we easily see that we have

$$\min \{f(x) + g(Ax)\} = \inf_x \sup_\xi L(x, \xi).$$

We then interchange inf and sup thus obtaining the dual problem

$$\begin{aligned}\sup_\xi \inf_x L(x, \xi) = \sup_\xi -g^*(\xi) + \inf_x f(x) + \xi \cdot Ax \\ = \sup_\xi -g^*(\xi) - \sup_x -f(x) + -A^t\xi \cdot x = \sup_\xi -g^*(\xi) - f^*(-A^t\xi).\end{aligned}$$

As we said, the equality constraint $Ax = b$ corresponds to $g = I_{\{b\}}$, and indeed we have $g^*(\xi) = \xi \cdot b$.

The duality between

$$\min \{f(x) + g(Ax) \,:\, x \in \mathcal{X}\} \quad \text{and} \quad \sup \{-g^*(\xi) - f^*(-A^t\xi) \,:\, \xi \in \mathcal{Y}'\}$$

is a classical object in convex analysis and a theorem guaranteeing, under some conditions, that the the values are actually equal is known as Fenchel-Rockafellar theorem. See, for instance, Chapter 6 in [3].

4.3 Uzawa and Augmented Lagrangian algorithms

We consider here how to turn the ideas from convex duality into a way to produce aloritghms whose aim is to solve an optimization problem which is complicated, for instance because of the presence of constraints, but manage to do it by exploiting its dual problem. We will consider three cases:

- the case of linear equality constraints:
$$\min\{f(x)\ :\ Ax=b\};$$
in this case the dual problem is
$$\max\{G(\lambda):=\inf_x f(x)+\lambda\cdot(Ax-b)\ :\ \lambda\in\mathbb{R}^m\};$$
we will not necessarily use the fact that G can be expressed in terms of f^* as we explained above;
- the case of linear inequality constraints:
$$\min\{f(x)\ :\ Ax\leq b\},$$
where, as usual, the inequality is to be considered component-wise; in this case the dual problem is
$$\max\{G(\lambda):=\inf_x f(x)+\lambda\cdot(Ax-b)\ :\ \lambda\in\mathbb{R}^m_+\};$$
- the case of non-linear inequality constraints:
$$\min\{f(x)\ :\ g_i(x)\leq c_i \text{ for } i=1,\dots,m\};$$
in this case the dual problem is
$$\max\{G(\lambda):=\inf_x f(x)+\sum_{i=1}^m\lambda^i(g_i(x)-c_i)\ :\ \lambda\in\mathbb{R}^m\}.$$

We do not consider non-linear inequality constraints because this would give raise to non-convex optimization problems.

The general idea is to run a fixed-step gradient algorithm on G. It is then necessary compute $\nabla G(\lambda)$. In order to do this, we exploit a general idea for the differentiation of functions expressed as the result of on optimization problem: $G(\lambda)=\min_x L(x,\lambda)$. In this case we compute $\nabla_\lambda L(x,\lambda)$ and we obtain, for fixed λ, a vector depending on x; we then need to choose the poit $x=x(\lambda)$ which is the one which realizes the minimal value in the definition of $G(\lambda)$. It is possible to prove that, if such a point is unique and L is C^1, then G is also differentiable at λ and we have $\nabla G(\lambda)=\nabla_\lambda L(x(\lambda),\lambda)$; moreover, independently of the uniqueness of $x(\lambda)$, if G is differentiable at λ we necessarily have $\nabla G(\lambda)=\nabla_\lambda L(x(\lambda),\lambda)$, which also implies that if many minimizers exist, their gradients should be the same.

Then, in the case of linear equality constraints the algorithms reads as follows: we define a sequence (λ_k,x_k) with
$$\lambda_0=0,\quad x_k:=\operatorname*{argmin}_x f(x)+\lambda_k\cdot(Ax-b),\quad \lambda_{k+1}:=\lambda_k+\tau(Ax_k-b),$$
where $\tau>0$ is a fixed step, the vector Ax_k-b represents $\nabla G(\lambda_k)$, and the positive sign in front of τ is due to the fact that G is maximized and not minimized. If we can prove convergence, then the sequence x_k converges to the minimizer of the primal problem, and the sequence λ_k to the maximizer of the dual.

For the case of linear inequality constraints the only difference is given by the update in λ, as we should impose the positivity constraints. We then define

$$\lambda_0 = 0, \quad x_k := \operatorname*{argmin}_x f(x) + \lambda_k \cdot (Ax - b), \quad \lambda_{k+1} := (\lambda_k + \tau(Ax_k - b))_+,$$

where the positive part is to be applied componentwise and corresponds to projecting on the set $\mathbb{R}^m_+$. In this case, we are running a projected gradient algorithm.

Finally, in the case of non-linear inequality constraints the only difference is given by the update in λ, as we should impose the positivity constraints. We then define

$$\lambda_0 = 0, \quad x_k := \operatorname*{argmin}_x f(x) + \sum_i \lambda_k^i (g_i(x) - c_i), \quad \lambda_{k+1}^i := (\lambda_k^i + \tau(g_i(x) - c_i))_+.$$

These algorithms, in particular in the last case, are called *Uzawa algorithm.*

We provide a proof of convergence in the case of linear equalities or inequalities.

Theorem 4.10. *If f is $\alpha-$elliptic and $\tau < 2\alpha/|||A|||^2$ then the sequence x_k defined by the Uzawa algorithms in the case of linear equality constraints converges to the unique solution $\bar{x}$ of the primal problem.*

Proof. Using Lagrange multipliers, the point $\bar{x}$ is such that there exists a vector $\bar{\lambda}$ such that $\nabla f(\bar{x}) + A^t\bar{\lambda} = 0$ and $A\bar{x} = b$. Moroever, using the optimality conditions at each step of the Uzawa algorithm we have $\nabla f(x_k) + A^t\lambda_k = 0$. We then compute

$$||\lambda_{k+1} - \bar{\lambda}||^2 = ||\lambda_k - \bar{\lambda}||^2 + 2\tau(Ax_k - b) \cdot (\lambda_k - \bar{\lambda}) + \tau^2||Ax_k - b||^2.$$

Replacing b with $A\bar{x}$ this gives

$$||\lambda_{k+1} - \bar{\lambda}||^2 \leq ||\lambda_k - \bar{\lambda}||^2 + 2\tau A(x_k - \bar{x}) \cdot (\lambda_k - \bar{\lambda}) + \tau^2|||A|||^2||x_k - \bar{x}||^2.$$

We then use

$$A(x_k - \bar{x}) \cdot (\lambda_k - \bar{\lambda}) = (x_k - \bar{x}) \cdot A^t(\lambda_k - \bar{\lambda}) = -(x_k - \bar{x}) \cdot (\nabla f(x_k) - \nabla f(\bar{x})) \leq -\alpha||x_k - \bar{x}||^2$$

and obtain

$$||\lambda_{k+1} - \bar{\lambda}||^2 \leq ||\lambda_k - \bar{\lambda}||^2 - (2\tau\alpha - \tau^2|||A|||^2)||x_k - \bar{x}||^2.$$

If $2\tau\alpha - \tau^2|||A|||^2 > 0$ this shows that the series of the terms $||x_k - \bar{x}||^2$ is bounded above by a telescopic series, and hence converges. We deduce $||x_k - \bar{x}||^2 \to 0$. □

The variat for the case of inequality constraints is the following:

Theorem 4.11. *If f is $\alpha-$elliptic and $\tau < 2\alpha/|||A|||^2$ then the sequence x_k defined by the Uzawa algorithms in the case of linear inequality constraints converges to the unique solution $\bar{x}$ of the primal problem.*

Proof. First, let us assume that the point $\bar{x}$ is such that there exists a vector $\bar{\lambda} \in \mathbb{R}^m_+$ such that $\nabla f(\bar{x}) + A^t\bar{\lambda} = 0$, $A\bar{x} \leq b$ and $\bar{\lambda} \cdot (A\bar{x} - b) = 0$. This means

$$P_{\mathbb{R}^m_+}[\bar{\lambda} + \tau(A\bar{x} - b)] = \bar{\lambda}.$$

Indeed, for each component i such that $\bar{\lambda}^i = 0$ we have $(\bar{\lambda} + \tau(A\bar{x} - b))^i \leq 0$ and for each component i such that $\bar{\lambda}^i > 0$ we have $(\bar{\lambda} + \tau(A\bar{x} - b))^i = \bar{\lambda}^i$.

We then use this condition together with the fact that projections are 1-Lipschitz to obtain

$$||\lambda_{k+1} - \bar{\lambda}||^2 = ||P_{\mathbb{R}^m_+}[\lambda_k + \tau(Ax_k - b)] - P_{\mathbb{R}^m_+}[\bar{\lambda} + \tau(A\bar{x} - b)]||^2 \leq ||\lambda_k + \tau(Ax_k - b) - \bar{\lambda} - \tau(A\bar{x} - b)||$$

and then the computation goes almost as in the previous theorem. Indeed, we obtain

$$||\lambda_{k+1}-\bar\lambda||^2 \le ||\lambda_k-\bar\lambda||^2+\tau^2|||A|||^2||x_k-\bar x||^2+2\tau A(x_k-\bar x)\cdot(\lambda_k-\bar\lambda),$$

so that we have

$$||\lambda_{k+1}-\bar\lambda||^2 \le ||\lambda_k-\bar\lambda||^2-(2\tau\alpha-\tau^2|||A|||^2)||x_k-\bar x||^2.$$

Why can we say that we have a vector $\bar\lambda \in \mathbb{R}^m_+$ such that $\nabla f(\bar x)+A^t\bar\lambda=0$, $A\bar x\le b$ and $\bar\lambda\cdot(A\bar x-b)=0$? if we use Lagrange multipliers we easily get $\nabla f(\bar x)+A^t\bar\lambda=0$, $A\bar x\le b$ and, for each component i such that $(A\bar x-b)^i<0$, we have $\bar\lambda^i=0$ (since such a constraint would not be active). The only remaining fact to prove is that we can assume $\bar\lambda\ge 0$. This can be seen as a particular case of Kuhn-Tucker conditions (Lagrange multipliers with sign constraints), or as a consequence of the duality result (but we need in this case to prove that the dual problem admits a solution, which will be $\bar\lambda$). □

We finish this section with a variant of the Uzawa algorithm for equality constraints. We keep in mind that the goal of such an algorithm is to find the saddle points of the Lagrangian $L(x,\lambda)=f(x)+\lambda\cdot(Ax-b)$. The variant we consider, called *Augmented Lagrangian*, considers instead the Lagrangian function $\tilde L(x,\lambda):=f(x)+\lambda\cdot(Ax-b)+\frac{\tau}{2}||Ax-b||^2$. The important point is that the saddle points of L and of $\tilde L$ are the same. Indeed, the sadle points of L are characterized by

$$\begin{cases}\nabla f(x)+A^t\lambda=0,\\ Ax-b=0,\end{cases}$$

while those of $\tilde L$ by

$$\begin{cases}\nabla f(x)+A^t\lambda+\tau A^t(Ax-b)=0,\\ Ax-b=0,\end{cases}$$

which has the same sets of solutions since the second equation imposes that the extra term in the first vanishes.

The algorithm then becomes looking for a sequence (λ_k,x_k) with

$$\lambda_0=0,\quad x_k:=\operatorname*{argmin}_x f(x)+\lambda_k\cdot(Ax-b)+\frac{\tau}{2}||Ax-b||^2,\quad \lambda_{k+1}:=\lambda_k+\tau(Ax_k-b).$$

The extra quadratic term in the minimization defining x_k makes the function to be minimized more convex, and eases the use of fast-converging fixed step gradient algorithms for this. We omit the proof of convergence, but it could be done without the ellipticity assumption on f (coercivity would be enough to obtain convergence, up to a subsequence, to a saddle point for L) .

5 Stochastic gradient descent

We consider here the case where we adapt the procedure of a gradient descent algorithm in order to include random effects. More precisely, we consider an iterative algorithm of the form

$$X_{k+1}=X_k-\tau_kY_x,$$

where Y_k is any random variable such that $\mathbb{E}[Y_k|X_k]=\nabla f(X_k)$. Even if we take $X_0=x_0$ to be a deterministic initial point (for instance $x_0=0$), starting from the first iterations the algorithm will give as an output a random variable and not a precise value. The goal is to provide conditions such that X_k or some other r.v. built out of the sequence $(X_k)_k$ converges (for instance in L^2, or a.s.) to the minimizer $\bar x$ of f.

Before proving any convergence result, let us analyze some examples of application.

Example. *Suppose that we have $f(x) := \mathbb{E}[g(x,\omega)]$, where ω is a random variable. Under suitable assumptions so as to apply the differentiation under the integral sign we have $\nabla f(x) = \mathbb{E}[\nabla_x g(x,\omega]$. This means that we can choose $Y_k = \nabla_x g(X_k, \omega_k)$, where $(\omega_k)_k$ is a sequence of independent random variable distributed as ω. In practice, we want to optimize the average value of something which depends on a parameter x and on a random effect ω. At each step we are in a certain x ($= X_k$) and we move to a new point by choosing to follow the opposite gradient computed according to one realization of ω instead of the average over all ωs. This is particularly useful when the distribution of the variable ω is not actually known but we only know a sequence of samples ω_k.*

Example. *Suppose that f is a detrministic function of the form $f(x) = \sum_{i=1}^{n} f_i(x)$. We can consider it as an expectation of the form $f(x) := \mathbb{E}[g(x,\omega)]$, where ω is a r.v. uniformly distributed over $\{1,\dots,n\}$, and $g(x,\omega) = nf_\omega(x)$. We can then move from X_k to X_{k+1} using the gradient of one only function f_i choosing the index i at random. It is also possible to select once for all an order (a permutation) of the indices at random, and then cycle accoring to this order.*

Example. *It is also possible to exploit the fact that a gradient is a sum of vectors of the form $\frac{\partial f}{\partial x_i} e_i$, where the vectors e_i are the vectors of the canonical basis. In this case we use $Y_k = n\frac{\partial f}{\partial x_i}(X_k)e_i$, where the index i is uniformly drawn from $\{1,\dots,n\}$.*

We suppose f to be convex. The main computation that we do is the following

$$\mathbb{E}[||X_{k+1} - \bar{x}||^2] = \mathbb{E}[||X_k - \bar{x}||^2] - 2\tau_k \mathbb{E}[Y_k \cdot (X_k - \bar{x})] + \tau_k^2 \mathbb{E}[||Y_k||^2].$$

We then use $\mathbb{E}[Y_k|X_k] = \nabla f(X_k)$ and assume $\mathbb{E}[||Y_k||^2] \leq M$. Suppose $D^2 f \geq \alpha I$. We will distinguish later the case $\alpha > 0$ and $\alpha = 0$. We use the inequality

$$\min f = f(\bar{x}) \geq f(x) + \nabla f(x) \cdot (\bar{x} - x) + \frac{\alpha}{2}||\bar{x} - x||^2$$

and apply it to $x = X_k$, and take expectations. In this way we obtain

$$\mathbb{E}[||X_{k+1} - \bar{x}||^2] \leq (1 - \tau_k\alpha)\mathbb{E}[||X_k - \bar{x}||^2] - 2\tau_k \mathbb{E}[f(X_k) - \min f] + \tau_k^2 M. \tag{5.1}$$

We then state the two following results:

Theorem 5.1. *Suppose that f is convex; let Y_k satisfy $\mathbb{E}[Y_k|X_k] = \nabla f(X_k)$ and $\mathbb{E}[||Y_k||^2] \leq M$. Let the sequence of random variables X_k be defined via the stochastic gradient algorithm as described above. Define*

$$\gamma_k := \sum_{j=0}^{k} \tau_j \quad \text{and} \quad \tilde{X}_k := \frac{\sum_{j=0}^{k} \tau_j X_j}{\gamma_k}.$$

Then we have

$$\mathbb{E}[f(\tilde{X}_k) - \min f] \leq \frac{\mathbb{E}[||X_0 - \bar{x}||^2] + M \sum_{j=0}^{k} \tau_j^2}{2\gamma_k}$$

and, in particular, if $\tau_k = \frac{1}{\sqrt{k+1}}$, we obtain $\mathbb{E}[f(\tilde{X}_k) - \min f] \leq\approx \frac{C + \log k}{\sqrt{k}}$.

Proof. We use (5.1) with $\alpha = 0$, sum over k. This gives

$$2\sum_{j=0}^{k} \tau_k \mathbb{E}[f(X_j) - \min f] \leq \mathbb{E}[||X_0 - \bar{x}||^2] - \mathbb{E}[||X_{k+1} - \bar{x}||^2] + M\sum_{j=0}^{k} \tau_j^2.$$

We divide by γ_k, ignore the positive term $\mathbb{E}[||X_{k+1}-\bar{x}||^2]$, and use $f(\tilde{X}_k) \leq \sum\limits_{j=0^k \frac{}{\tau j}} \gamma_k f(X_j)$, which is a consequence of the convexity of f, since $\tilde{X}_k$ is a convex combination of the variables X_j. This gives the desired estimate.

We can then choose suitably the coefficients τ_j so as to make the right-hand side tend to 0. If we choose $\tau_j \approx j^{-1/2}$ we have $\sum\limits_{j=0}^{k} \tau_j^2 \approx \sum\limits_{j=0}^{k} \frac{1}{j} \approx \log k$ and $\sum\limits_{j=0}^{k} \tau_j \approx \sum\limits_{j=0}^{k} \frac{1}{\sqrt{j}} \approx \sqrt{k}$, which gives the result. □

It is possible to check that other choices of τ_j of the form $\tau_j \approx j^-\alpha$ provide a less good result, since we would have in this case $\sum\limits_{j=0}^{k} \tau_j^2 \approx k^{1-2\alpha}$ and $\sum\limits_{j=0}^{k} \tau_j \approx k^{1-\alpha}$. In case $\alpha < 1/2$, the ratio would be of the order of $k^{-\alpha}$, which is a power strictly worse that $k^{-1/2}$ and hence of $\frac{C+\log k}{\sqrt{k}}$. In case $\alpha > 1/2$ the dominant term of the numerator would be the constant one and the ratio would be of order $k^{\alpha-1}$. Again, the exponent would be worse than $-1/2$.

In the case $\alpha > 0$ it is possible to give a differnt result, in terms of $\mathbb{E}[||X_k - \bar{x}||]$ instead of the value of the function. This is not surprising, if we think at the case of the gradient descent and of the different results that we obtain in the elliptic case or in the case where f is only convex.

Theorem 5.2. *Suppose that f is convex with $D^2 f \geq \alpha I$; let Y_k satisfy $\mathbb{E}[Y_k|X_k] = \nabla f(X_k)$ and $\mathbb{E}[||Y_k||^2] \leq M$. Let the sequence of random variables X_k be defined via the stochastic gradient algorithm as described above. Suppose $\sum\limits_k \tau_k = +\infty$ but $\sum\limits_k \tau_k^2 < +\infty$. Then we have*

$$\mathbb{E}[||X_k - \bar{x}||^2] \to 0.$$

When we choose $\tau_k = \frac{1}{(k+1)\alpha}$ we have more precisely

$$\mathbb{E}[||X_k - \bar{x}||^2] \leq\approx \frac{M \log k}{\alpha^2 k}.$$

Proof. We use (5.1) with $\alpha > 0$, and ignore the positive term $\mathbb{E}[f(X_k) - \min f]$. We then obtain, setting $u_k := \mathbb{E}[||X_k - \bar{x}||^2]$;

$$u_{k+1} \leq (1 - \alpha\tau_k)u_k + \tau_k^2 M.$$

We can write this as $\alpha\tau_k u_k \leq u_k - u_{k+1} + \tau_k^2 M$. With our assumptions, the right hand side is summable, since a part is telescopic and we asumed $\sum\limits_k \tau_k^2 < +\infty$. We deduce that we have $\sum\limits_k \tau_k u_k < +\infty$. If we suppose that we have $\liminf u_k > 0$ we havea contradiction with the non-summability of τ_k. Hence there exists a subsequence $u_{k_h} \to 0$. If we fix $\varepsilon > 0$ we can choose h such that $u_{k_h} < \varepsilon/2$ and $M \sum\limits_{k=k_h}^{\infty} \tau_k^2 < \varepsilon_2$. Using $u_n \leq u_m + \sum\limits_{k=n}^{m-1} \tau_k^2 M$, which is true for any $n > m$, we deduce $u_n < \varepsilon$ for every $n > k_h$, which exactly means $\lim\limits_k u_k = 0$.

In the particular case $\tau_k = \frac{1}{(k+1)\alpha}$ we can make an explicit computation. We have

$$u_{k+1} \leq (1 - \frac{1}{k+1})u_k + \frac{M}{\alpha^2(k+1)^2}.$$

Multiplying by $k+1$ we obtain

$$(k+1)u_{k+1} \leq k u_k + \frac{M}{\alpha^2(k+1)}$$

and hence

$$(k+1)u_{k+1} \leq \frac{M}{\alpha^2} \sum_{j=0}^{k} \frac{1}{j+1} \approx \frac{M \log k}{\alpha^2},$$

which proves the claim. □

We can observe that in this estimate there is no depdence on the initial datum and in particular on $\mathbb{E}[||X_0 - \bar{x}||^2]$. Yet, our assumptions imply $\mathbb{E}[||\nabla f(X_k)||^2] \leq M$ and, since f is elliptic, we have $|\nabla f(X_k)||^2 \geq \alpha^2||X_k - \bar{x}||^2$. It is then necessary to choose M large enough so as to bound all the terms of the form $\mathbb{E}[||X_k - \bar{x}||^2]$, and in particular $\mathbb{E}[||X_0 - \bar{x}||^2]$. Hence, this result should rather be understood as an information on the asymptotic rate of convergence rathern than a precise non-asymptotic bound.

We finish this section with few words about the notion of *importance sampling*. Let us stick, for instance, to the example where $f = \sum\limits_i f_i$ and imagine that the orders of magnitude of the different functions f_i and/or of their gradients are not at all comparable. We can write $f = \sum\limits_i f_i$, but also $f = \sum\limits_i \lambda_i \tilde{f}_i$, where $\tilde{f}_i = f_i/\lambda_i$, for arbitrary numbers $\lambda_i > 0$ such that $\sum\limits_i \lambda_i = 1$. Hence, fhte function f is also the expected value of the $\tilde{f}_i$, when we do not use the uniform distribution on the values of i but we use a probability such that $\mathbb{P}(\omega = i) = \lambda_i$. If we then take $Y_k = \nabla \tilde{f}_{\omega_k}(X_k)$ we stille have $\mathbb{E}[Y_k|X_k] = \nabla f(X_k)$ but we have

$$\mathbb{E}[||Y_k||^2|X_k] = \sum_i \lambda_i||\nabla \tilde{f}_i(X_K)||^2 = \sum_i \frac{||\nabla f_i(X_k)||^2}{\lambda_i}.$$

This quantity is not independent of the λ_i and is minimized when each λ_i is proportional to $\nabla f_i(X_k)$. As a consequence, if one has bounds of the form $||\nabla f_i|| \leq L_i$, it could be convenient to use $\lambda_i = \frac{L_i}{\sum\limits_j L_j}$. The expected value of the gradient does not change, but

6 Complementary material

We discuss here some examples of optimization problems coming from data analysis, essentially taken from [3].

6.1 Point clouds separation

Suppose that we have some points $x_i \in \mathbb{R}^n$ for $i = 1, \ldots, m$, in the space and for every point we have a label $y_i = \pm 1$. We want to find a separation between points with $y_i = +1$ and points with $y_i = -1$ so that when new data arrive we know how to classify them. For simplicity, we hope to find a linear separation, i.e. a vector $v \in \mathbb{R}^n$ and a constant $c \in \mathbb{R}$ such that y_i should have the same sign as $v \cdot x_i + c$. We consider all pairs (v, c) such that, for every i, we have $y_i(v \cdot x_i + c) > 0$ and we look for the one which maximizes te distance between the points and the separation subspace $\{v \cdot x + c = 0\}$. This distance is given by $\min\limits_i \frac{|v \cdot x_i + c|}{||v||}$. We would like to solve

$$\max \left\{ \left(\min_i \frac{|v \cdot x_i + c|}{||v||} \right) \;:\; v \in \mathbb{R}^n, c \in \mathbb{R}, y_i(v \cdot x_i + c) > 0 \right\}.$$

We can always replace a pair (v, c) with (tv, tc) for $t > 0$ and nothing changes, so we can add the condition $\min\limits_i |v \cdot x_i + c| = 1$ or, keeping into account the sign condition $\min\limits_i y_i(v \cdot x_i + c) = 1$. In this case, we have to minimize the norm of v or, equivalently, the square of the norm. This becomes

$$\min \left\{ \frac{1}{2}||v||^2 \;:\; v \in \mathbb{R}^n, c \in \mathbb{R} \ \min_i y_i(v \cdot x_i + c) = 1 \right\}.$$

We can also replace $\min\limits_i y_i(v \cdot x_i + c) = 1$ with $\min\limits_i y_i(v \cdot x_i + c) \geq 1$ since it is clear, by rescaling again v and c by a same factor, that the minimizer under the condition $\min\limits_i |v \cdot x_i + c| \geq 1$ would satisfy the equality. The condition $\min_i y_i(v\cdot x_i+c)\geq 1$ can be written as "for every i we have $y_i(v\cdot x_i+c)\geq 1$", so that we get

$$\min\left\{\frac{1}{2}||v||^2\ :\ v\in\mathbb{R}^n, c\in\mathbb{R}\, y_i(v\cdot x_i+c)\geq 1\right\}.$$

How to solve this optimization problem under constraints? the projected gradient algorithm does not seem a good idea since the whole problem indeed consists in projecting 0 onto the constraints, and in general we do not have a formula for the projection. Uzawa's algorithm is a better choice, since we only have to solve a sequence of problems of the form

$$\min\frac{1}{2}||v||^2+\sum_i\lambda_i(1-y_i(v\cdot x_i+c)),$$

for which we can have an explicit expression of the minimizers.

All the analysis above starts from the assumption that a separating hyperplane $\{v\cdot x+c=0\}$ exists. It is not always the case, but it is possible that it "almost" exists in the sense that few outliers are the only exception. In this case a possibility is to solve the following problem:

$$\min\left\{\frac{1}{2}||v||^2+\sum_{i=1}^m|\varepsilon_i|\ :\ v\in\mathbb{R}^n, c\in\mathbb{R}, \varepsilon\in\mathbb{R}^m,\ y_i(v\cdot x_i+c)\geq 1-\varepsilon_i\right\},$$

which consists in admitting violations of the constraint $y_i(v\cdot x_i+c)\geq 1$ but penalizing them. Again, it is possible to solve this problem via the Uzawa algorithm. In this case, for $\lambda_i\geq 0$ one has to solve the problem

$$\min\frac{1}{2}||v||^2+\sum_{i=1}^m|\varepsilon_i|+\sum_i\lambda_i(y_i(1-\varepsilon_i-(v\cdot x_i+c)).$$

If we look at the dependence in ε_i we see that we have $\min_\varepsilon|\varepsilon|-\lambda_i\varepsilon=0$ if $\lambda_i\in[0,1]$ while $\inf_\varepsilon|\varepsilon|-\lambda_i\varepsilon=-\infty$ if $\lambda_i>1$. Hence, the iterations of the Uzawa algorithm require the extra constraint $\lambda_i\in[0,1]$ instead of only $\lambda_i\geq 0$; at every step of the projected gradient algorithm on the variable λ it is then necessary to project onto such a constraint, i.e. taking the positive part and truncating at 1.

6.2 Inverse problems

In many applications we cannot observe directly the parameters $x\in\mathbb{R}^n$ of a model but only their output after applying an operator $A\in M^{n\times m}$ and we want to find a reasonable estimation of x. We want then to solve an equation $Ax=b$; in many cases A is neither injective nor surjective, and it is also possible that, because of noise, the observation b that we have does not belong to the image of A. We are then lead to solve an optimization problem of the form $\min||Ax-b||^2$ and to add possible regularization terms on X in order to select a "better" solution. We list here some observations on this problem.

- If $b\in Im(A)$ and we solve $\min||Ax-b||^2+\varepsilon F(x)$ we obtain (if they exist, which is the case if F is coercive, and if they are unique, which is the case if F is strictly convex) a solution x_ε. Then, if we have a sequence $x_{\varepsilon_j}\to x_0$, we can say that x_0 is a solution of $Ax=b$ which minimizes, among solutions, the quantity F. This is a consequence of Lemma 6.1. If F is strictly convex and coercive the whole sequence x_ε is bounded and converges to x_0.

- When we take $F(x)=||x||^2$ this is a way of finding the solution of $Ax=b$ of minimal norm. Since we have $(A^tAx+\varepsilon I)x_\varepsilon=A^tb$ we can see that x_ε depends linearly on b, so x_0 also depends linearly on b. This defines a map $A^\dagger$ such that $AA^\dagger b=b$ and $A^\dagger b=\text{argmin}\{||x||\ :\ Ax=b\}$. Note that the matrix A^tA is always symmetric positive semidefinite, and becomes positive definite when adding εI.

- If $b \notin ImA$ the problem is invariant if replacing b with $\tilde{b} := P_{ImA}[b]$, since $||Ax-b||^2 = ||Ax-\tilde{b}||^2 + ||\tilde{b}-b||^2$, using the fact that $\tilde{b}-b$ is orthogonal to $Im(A)$.
- We can consider $F(x) = ||x||_1$, which is often used to select solutions of $Ax = b$ which are sparse. As we saw in Section 1.2 we should use a different quantity, that we called $A_0(x)$, but Lemma 6.2 justifies the fact that we use the norm $||x||_1$.
- A problem that we are hence often lead to solve is therefor

  $$\min ||Ax-b||^2 + ||x||_1.$$

  For this problem, the use of proximal gradient descent, maybe in accelerated versions (as in [1], see also Section 3.2), is the best option. Note that when A^tA is not positive definite the function decomposes into $f(x)+g(x)$ with f smooth but no elliptic and g non-smooth, and the convergence rate is $1/k$ (for the non-accelerated version) or $1/k^2$.

Lemma 6.1. *Let F, G be two l.s.c. functionals bounded from below on a space X and $x_\varepsilon \in \operatorname{argmin} G + \varepsilon F$. Suppose $x_{\varepsilon_j} \to x_0$. Then $x_0 \in \operatorname{argmin}\{F(x) \,:\, x \in \operatorname{argmin} G\}$. If either G or F is coercive and the problem $\min\{F(x) \,:\, x \in \operatorname{argmin} G\}$ has a unique solution, then the whole sequence x_ε converges to such a solution.*

Proof. Writing x_ε for x_{ε_j}, for every x we have $G(x_\varepsilon) + \varepsilon \inf F \leq G(x_\varepsilon) + \varepsilon F(x_\varepsilon) \leq G(x) + \varepsilon F(x)$. Passing to the limit as $\varepsilon \to 0$ we obtain $G(x_0) \leq G(x)$. Which shows $x_0 \in \operatorname{argmin} G$. We then choose $x \in \operatorname{argmin} G$. We then obtain

$$G(x) + \varepsilon F(x_\varepsilon) \leq G(x_\varepsilon) + \varepsilon F(x_\varepsilon) \leq G(x) + \varepsilon F(x).$$

This implies $F(x_\varepsilon) \leq F(x)$ and, at the limi, $F(x_0) \leq F(x)$, which shows the claim. □

Lemma 6.2. *Consider the problem*

$$\min\{||x||_1 \,:\, Ax = b\}.$$

This problem admits at least a solution $\bar{x}$ such that $\#\{i \,:\, x_i \neq 0\} \leq \dim(ImA)$.

We do not prove this last result, which van be found as Theorem 8.4 in [?]

Bibliographie

[1] A Beck, M Teboulle A fast iterative shrinkage-thresholding algorithm for linear inverse problems *SIAM journal on imaging sciences* 2 (1), 183–202, 2009

[2] J.F. Bonnans, J.C. Gilbert, C. Lemaréchal and C. Sagastizábal, *Numerical Optimization*, 2nd Edition, Springer-Verlag, Heidelberg, 2006.

[3] G. Carlier, *Classical and Modern Optimization*, World Scientific, London, 2022.

[4] Y.E.Nesterov, A method for solving the convex programming problem with convergence rate $O(1/k^2)$, *Dokl. Akad. Nauk SSSR*,269(1983), pp. 543–547 (in Russian).