Non-convex inverse problems

Irène Waldspurger

waldspurger@ceremade.dauphine.fr

January to March 2023

Table des matières

Acknowledgements

Many thanks to Claire Boyer and Vincent Duval for having shared their lecture notes with me.

Chapitre 1

Introduction

1.1 Inverse problems

1.1.1 Definition

Given a system and an observation procedure[1], computing the outcome of the observation procedure is called a *direct problem*. For instance, if we are given a description of a fluid at some instant (viscosity, density, velocity at each point...), predicting how the fluid will be one minute later is a direct problem. Here, the system is the fluid, and the observation procedure is "let it flow for one minute, then look at it".

An *inverse problem* is the converse : given the result of the observation procedure, and the knowledge of this procedure, can we identify the system ? For instance, if we are given (two-dimensional) photographs of a building, viewed from several angles, reconstructing a three-dimensional model of the building is an inverse problem. Here, the system is the 3D shape of the building, and the observation procedure is "take a set of photographs from several angles".

Mathematically, these problems are formalized as follows. Let E be a set modelling the possible *systems*, and F a set modelling the possible *observations*. The *observation procedure* is described by a function $M : E \to F$. An

[1] The words "system" and "observation procedure" must be understood in a general sense. A *system* is any complex object of interest, and an *observation procedure* is any process which, given the system, produces some outcome.

inverse problem is, given some observation $y \in F$,

$$\boxed{\text{find } x \in E \text{ such that } M(x) = y.} \qquad \text{(Inverse)}$$

1.1.2 Theoretical aspects

From a purely theoretical perspective, the main two questions regarding an inverse problem are *uniqueness* and *stability*.

- Uniqueness : Is the solution of Problem (Inverse) unique ? This question is crucial, since, if the solution is not unique, it is impossible to recover the system of interest with certainty.
- Stability : If y is not exactly known, but only available up to some error, what will the solution(s) of Problem (Inverse) look like ? Will it be close to the "true" solution, the one we would have obtained if there had been no error on y ? This is also crucial : in real life, exact measurements are never available.

The question of *existence* (for an arbitrary y, does there exist a solution x to Problem (Inverse) ?) is oftentimes also of interest, but we will leave it aside.

Example 1.1 : finite-dimensional linear inverse problem

Let us assume that

- E, F are real finite-dimensional vector spaces : $E = \mathbb{R}^d$ and $F = \mathbb{R}^m$ for some $d, m \in \mathbb{N}^*$;
- $M : E \to F$ is linear, represented by some matrix $A \in \mathbb{R}^{m\times d}$.

Under these assumptions, Problem (Inverse) rewrites as

$$\boxed{\text{find } x \in \mathbb{R}^d \text{ such that } Ax = y.}$$

For a given y, assuming a solution x_* exists, it is *unique* if

$$\{x \in \mathbb{R}^d, Ax = y\} = \{x_*\},$$

that is if and only if $\mathrm{Ker}(A) = \{0\}$ (A is an injective matrix).
We now assume that the solution is unique. Is it *stable* ? In other words, if we replace y by

$$y_\epsilon = y + \epsilon$$

for some “small” $\epsilon \in \mathbb{R}^m$, will the solution x_ϵ be close to x_* ? The notions of “smallness” and “closeness” do not have a precise formal meaning. Depending on the problem, many formalizations are possible. The simplest one is to say that a vector ϵ is *small* if

$$||\epsilon||_2 \ll ||y||_2,$$

and x_ϵ is *close* to x_* if

$$||x_\epsilon - x_*||_2 \ll ||x_*||_2.$$

With these definitions, it is possible to show that the problem is *stable* if the smallest and largest singular values of A satisfy

$$\frac{\lambda_{\max}(A)}{\lambda_{\min}(A)} \approx 1.$$

The ratio $\frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}$ is called *condition number* of A.
For more details, see the exercises.

These questions will not be the subject of the class. In Section 1.3 of this introduction, we will give uniqueness conditions, when possible, for the considered problems. But afterwards, we will most of the time assume that all the problems we consider satisfy uniqueness and stability properties. However, in principle, when facing a new problem, these questions must be the starting point, otherwise we are at risk of working towards the conception of algorithms for solving problems which can actually not be solved.

1.1.3 Our focus : algorithms

In this class, we will be interested in algorithms which allow to solve inverse problems.

In applications, a “good” algorithm is an algorithm which

- works : given a problem, it must output a correct solution ; we can tolerate the algorithm failing once in a while, but the failure rate must be as small as possible ;
- uses as few computational resources as possible : it must be fast (not too many operations) and have a moderate memory footprint.

Here, we will be interested in algorithms for which, moreover,

— these good properties (especially the first one) can be rigorously proved.

This additional requirement tends to be in contradiction with the computational efficiency, in the sense that, oftentimes, the algorithms which work best in practice are difficult to rigorously study. As a consequence, the algorithms we will present in this class will most of the time not be the most well-suited for real applications. They must be considered as toy models for "really usable" algorithms, should ideally retain as many specificities of their "really usable" counterparts as possible, but will inevitably miss some.

Similarly, the hypotheses under which we will establish correctness guarantees for the algorithms will oftentimes be much stronger than what holds in real applications. It is an important but difficult research direction to weaken these hypotheses.

1.2 Convex vs non-convex

All inverse problems can be reformulated as *optimization problems*, that is problems of the following form :

$$\begin{aligned} \text{minimize } & f(x) \\ \text{over all } & x \in H \\ \text{such that } & x \in C_1, \\ & \dots \\ & x \in C_S. \end{aligned} \tag{Opt}$$

Here, $f : H \to \mathbb{R} \cup \{+\infty\}$ can be any *objective* function, over a real or complex vector space H, and $C_1, \dots, C_S$ are subsets of H which model the constraints imposed on the unknown x.

An optimization problem is called *convex* if f is a convex function and $C_1, \dots, C_S$ are convex sets.

Definition 1.2 : convexity

A function $f : H \to \mathbb{R} \cup \{+\infty\}$ is *convex* if, for any $x_1, x_2 \in H$ and any $s \in [0; 1]$,

$$f((1-s)x_1 + sx_2) \leq (1-s)f(x_1) + sf(x_2). \tag{1.1}$$

A set $C \subset H$ is *convex* if, for any $x_1, x_2 \in C$ and any $s \in [0;1]$, the vector

$$(1-s)x_1 + sx_2$$

is also an element of C.

In first approximation, we can say that convex problems admit efficient algorithms. This is not an absolute rule, since some convex sets or functions are quite difficult to manipulate. However, it is true that many algorithms exist for convex problems, with a behavior which is quite well understood. The situation is very different for the problems we will consider in this class, which are non-convex. For non-convex problems, the existence of algorithms both guaranteed to succeed and running in an reasonable amount of time is an exception.

Intuitively, convexity allows to deduce global information from local one. For instance, if one knows the values at a few points of a convex function f and its gradient, Inequality (1.1) makes it possible to compute upper and lower bounds on f, and hence obtain an approximation of its minimum. One can then query the values at other points to refine the approximation. This is illustrated on Figures 1.1a and 1.1b. But if the function is not convex, the knowledge of its values at a few points does not bring information about the values at other points and, in particular, does not bring information on its minimum. This is illustrated on Figures 1.1c and 1.1d. This is what makes non-convex optimization much more difficult than convex optimization.

This difficulty is a fundamental property of non-convex problems : if we do not have good algorithms able to solve any non-convex problem, it is not because we have not discovered these good algorithms yet. It is because good algorithms do not exist.[2] As a consequence, in this class, we will not try to propose algorithms able to solve all problems of a given non-convex family : this is hopeless. At best, our algorithms will be able to solve "a large part" of problems of the family.

[2] In particular, many families of non-convex problems have been proved to be NP-difficult. This means that, unless P=NP, there exists no algorithm able to solve all problems in the family with a time complexity at most polynomial in their dimension.

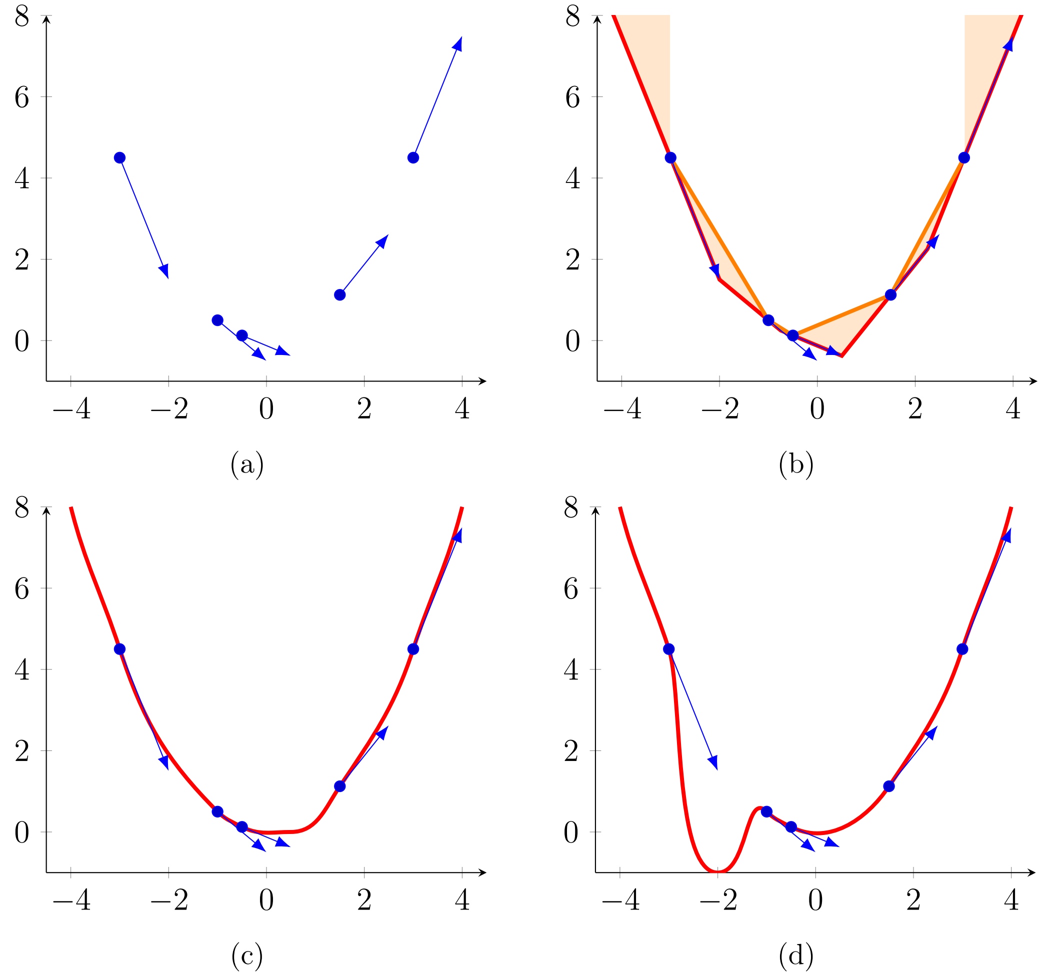

FIGURE 1.1 – (a) Representation of the values and derivatives of a function $f : \mathbb{R} \to \mathbb{R}$ at a few points. (b) Upper and lower bounds on f one can deduce from the knowledge of these values and derivatives if f is convex. (c) A non-convex function compatible with these values and derivatives. (d) Another non-convex function compatible with these values and derivatives.

1.3 Non-convex inverse problems : examples

Let us now present a few examples of non-convex inverse problems which we will encounter during this class.

1.3.1 Sparse recovery - compressed sensing

The first example is very well-known, and we will not spend much time on it since it is the subject of another class. However, as it is the simplest example of some properties we will describe, it is impossible not to mention it at all.

This problem is called *sparse recovery* or *compressed sensing*. It consists in recovering a vector $x \in \mathbb{R}^d$ from linear measurements

$$y \overset{def}{=} Ax \in \mathbb{R}^m,$$

where $A \in \mathbb{R}^{m\times n}$ is some known matrix. If $m \geq n$ and A is injective, this problem reduces to inverting A. But here, m is much smaller than n, which means that A is not injective and, without further information, y does not uniquely determine x. We must therefore assume some additional "structure" on x : we assume that x is *sparse*, that is, it has a small number of non-zero coordinates. More specifically, we assume that, for some $k \in \mathbb{N}^*$ much smaller than n,

$$||x||_0 \leq k,$$

where $||x||_0 = \mathrm{Card}\{i \leq d, x_i \neq 0\}$. (This quantity is often called the *ℓ^0-norm*, although it is not a norm, since it is not homogeneous.)

To summarize, the problem can be written as

$$\boxed{\begin{aligned} &\text{recover } x \in \mathbb{R}^d \\ &\text{such that } Ax = y, \\ &\text{and } ||x||_0 \leq k. \end{aligned}} \tag{CS}$$

It is non-convex because the set $\{x, ||x||_0 \leq k\}$ is non-convex.

Sometimes, the unknown x is not directly sparse, but only sparse when represented in some adequate basis, or after some adequate linear transformation. In this case, the condition "$||x||_0 \leq k$" must be replaced with "$||\Phi x||_0 \leq k$", where Φ encodes the basis or linear transformation.

This problem is notably natural in image processing, since many natural images enjoy a sparsity structure. Photos, for instance, are well-known to be approximately sparse when represented in a *wavelet basis*.

For compressed sensing, uniqueness of the reconstruction can be guaranteed through a condition on the kernel of A.

Proposition 1.3 : unique recovery for compressed sensing

We assume that $\mathrm{Ker}(A)$ does not contain a vector X such that $||X||_0 \leq 2k$.
Then, if Problem (CS) has a solution, this solution is unique.

Démonstration. Let us assume, by contradiction, that Problem (CS) has two distinct solutions $X_1, X_2 \in \mathbb{R}^d$. Then

$$A(X_1 - X_2) = AX_1 - AX_2 = y - y = 0,$$

so $X_1 - X_2$ belongs to $\mathrm{Ker}(A)$. And

$$||X_1 - X_2||_0 \leq ||X_1||_0 + ||X_2||_0 \leq 2k,$$

which contradicts the assumption. □

From this proposition, one can show that, if $m \geq 2k$, then almost all matrices A guarantee unique recovery of the underlying sparse vector. Under a stronger condition on A, one can also establish stability recovery guarantees (see for instance the introductory article [Candès and Wakin, 2008]).

1.3.2 Low rank matrix recovery

In low-rank matrix recovery, the goal is also to recover an object from linear measurements. This time, the “object” is a matrix $X \in \mathbb{R}^{d_1 \times d_2}$ (or $X \in \mathbb{C}^{d_1 \times d_2}$). As in the case of compressed sensing, there are not enough linear measurements to uniquely determine X without additinal information, but we do have some additional information on X : it is low-rank. This yields the problem

$$\boxed{\begin{aligned} &\text{recover } X \in \mathbb{R}^{d_1 \times d_2} \\ &\text{such that } \mathcal{L}(X) = y, \\ &\text{and } \mathrm{rank}(X) \leq r. \end{aligned}} \tag{Low rank}$$

Here, $\mathcal{L} : \mathbb{R}^{d_1\times d_2} \to \mathbb{R}^m$ is the linear measurement operator and r is a given upper bound on the rank of the matrix. In some applications, it is relevant to assume that $d_1 = d_2$ and that X is semidefinite positive : $X \succeq 0$.

This problem is sometimes called *matrix sensing*, especially when $\mathcal{L}$ is a random operator. A uniqueness result similar to Proposition (1.3) holds.

Proposition 1.4 : uniqueness for low-rank matrix recovery

We assume that $\mathrm{Ker}(\mathcal{L})$ does not contain a matrix X such that

$$\mathrm{rank}(X) \leq 2r.$$

Then, if Problem (Low rank) has a solution, this solution is unique.

The proof of the proposition is identical to Proposition 1.3. From this proposition, one can show (but it is not easy) that the solution of Problem (Low rank), when it exists, is unique, for almost all operators $\mathcal{L}$, provided that (in the case where $2r \leq \min(d_1, d_2)$)

$$m \geq 2r(d_1 + d_2 - 2r).$$

When r is small (of order 1, for instance), this means that we can hope to recover the "true" matrix X with a number of linear measurements much smaller than what we would need if we did not know X to be low-rank (in this case, we would need $m \geq \dim(\mathbb{R}^{d_1\times d_2}) = d_1 d_2$, which is much larger that $2r(d_1 + d_2 - 2r)$ if $r \ll \min(d_1, d_2)$).

Matrix completion Several special cases of Problem (Low rank) are of particular interest, and form subfamilies of inverse problems with their own applications and theoretical characteristics. The first one is *matrix completion*. In this case, the linear measurements available on X are the knowledge of a few coefficients :

$$\boxed{\begin{gathered}\text{recover } X \in \mathbb{R}^{d_1\times d_2}\\ \text{such that } X_{ij} = y_{ij}, \forall (i,j) \in \Omega\\ \text{and } \mathrm{rank}(X) \leq r.\end{gathered}} \qquad \text{(Matrix completion)}$$

Here, $\Omega \subset \{1, \dots, d_1\} \times \{1, \dots, d_2\}$ contains the indices of available coefficients.

The most popular application is the so-called "*Netflix* problem". In this application, X represents the opinion of users on films : the coefficient X_{ij} is an "affinity score" between User i and Film j (it represents how much User i would like Film j). It is reasonable to assume that X is low-rank : this models the similarities between the users, and between the films (e.g. if User 1 and 2 have the same opinion on Films $1, 2, 3, 4$, it is plausible that they also have essentially the same opinion on Film 5). The available coefficients X_{ij} correspond to pairs (i, j) for which User i has watched Film j and sent the corresponding score to the film distribution platform. The other coefficients are not available, but the platform would like to guess them, so as to be able to propose relevant film suggestions to their users. Guessing the non-available coefficients exactly amounts to solving Problem (Matrix completion).

Phase retrieval Another special case of Problem (Low rank) which we will discuss in length in this course is *phase retrieval*.

At first sight, phase retrieval problems have nothing to do with matrices and low-rankness. They are problems of the following general form

$$\boxed{\begin{aligned} &\text{recover } x \in \mathbb{C}^d \\ &\text{such that } |L_j(x)| = y_j, \forall j \leq m. \end{aligned}} \tag{Phase retrieval}$$

Here, $L_1, \dots, L_m : \mathbb{C}^d \to \mathbb{C}$ are known linear operators, the notation "$|.|$" stands for the usual complex modulus, and $y_1, \dots, y_m$ are given.

The main motivations for studying phase retrieval come from the field of imaging. Indeed, it is much easier to record the intensity (that is, the modulus, in an adequate mathematical model) of an electromagnetic wave than its phase. It is therefore frequent to have to recover an object from modulus-only measurements. Oftentimes, these measurements can specifically be described by a Fourier transform (because, under some assumptions, the diffraction pattern of an object is the Fourier transform of its characteristic function), but not always. Phase retrieval is also of interest for audio processing.

Remark

For any $x \in \mathbb{C}^d$ and $u \in \mathbb{C}$ such that $|u| = 1$, it holds

$$|L_j(ux)| = |uL_j(x)| = |u|\,|L_j(x)| = |L_j(x)|, \quad \forall j \leq m.$$

Therefore, the sole knowledge of $(y_j = |L_j(x)|)_{j\leq m}$ can never allow to exactly recover x. There is always a *global phase ambiguity* : x cannot be distinguished from ux.
This is in general not harmful in applications, and we will be satisfied if we can recover x up to a global phase.

Given specific linear forms L_j, it is in general difficult to determine if the (Phase retrieval) problem satisfies the uniqueness and stability properties. However, it is known that uniqueness holds "in principle" as soon as m is larger than (roughly) $4d$.

Proposition 1.5 : [Conca, Edidin, Hering, and Vinzant, 2015]

Let us assume that $m \geq 4d-4$. Then, for almost all linear maps $L_1,\dots,L_m : \mathbb{C}^d \to \mathbb{C}$, it holds that, for all $x, x' \in \mathbb{C}^d$,

$$\big(|L_j(x)| = |L_j(x')|, \forall j \leq m\big) \quad \Rightarrow \quad \big(\exists u \in \mathbb{C}, |u| = 1, x = ux'\big).$$

With a slightly larger m, stability also "generically" holds.

Let us now explain why phase retrieval is a special case of low-rank matrix recovery. Recovering $x \in \mathbb{C}^d$ up to a global phase is equivalent to recovering

$$X \overset{def}{=} xx^* = \begin{pmatrix} |x_1|^2 & x_1\overline{x}_2 & \dots & x_1\overline{x}_d \\ x_2\overline{x}_1 & |x_2|^2 & \dots & x_2\overline{x}_d \\ \vdots & & \ddots & \vdots \\ x_d\overline{x}_1 & & \dots & |x_d|^2 \end{pmatrix}.$$

Indeed, X can be computed from x (even up to a global phase : $(ux)(ux)^* = u\overline{u}xx^* = xx^*$ if $|u| = 1$) and x can be computed up to a global phase from X by extracting the only eigenvector of X with non-zero eigenvalue.

Remark

A matrix $X \in \mathbb{C}^{d\times d}$ can be written as $X = xx^*$ for some $x \in \mathbb{C}^d$ if and only if

$$X \succeq 0 \quad \text{and} \quad \operatorname{rank}(X) \leq 1.$$

Démonstration. For any $x \in \mathbb{C}^d$, the matrix xx^* is Hermitian, and semidefinite positive :

$$\forall z \in \mathbb{C}^d, \quad z^*(xx^*)z = |z^*x|^2 \geq 0.$$

It has rank at most 1 because $\mathrm{Range}(xx^*) = \mathrm{Vect}\{x\}$.

Conversely, if $X \succeq 0$ and $\mathrm{rank}(X) \leq 1$, then X can be diagonalized in an orthogonal basis $(z_1, \dots, z_d)$ (as all Hermitian matrices) :

$$X = \sum_{k=1}^{d} \lambda_k z_k z_k^* \quad \text{with } \lambda_1 \geq \dots \geq \lambda_d \text{ the eigenvalues.}$$

All the eigenvalues are nonnegative, since $X \succeq 0$. Since $\mathrm{rank}(X) \leq 1$, they are all 0, except possibly the first one, so

$$X = \lambda_1 z_1 z_1^* = (\sqrt{\lambda_1} z_1)(\sqrt{\lambda_1} z_1)^*,$$

so it can be written as $X = xx^*$ with $x = \sqrt{\lambda_1} z_1$. □

In addition, for any j, knowing $|L_j(x)|$ is equivalent to knowing $|L_j(x)|^2$. Denoting v_j the vector such that $L_j = \langle v_j, . \rangle$, we have

$$\begin{aligned} |L_j(x)|^2 &= \langle v_j, x \rangle \, \overline{\langle v_j, x \rangle} \\ &= (v_j^* x)(x^* v_j) \\ &= v_j^* X v_j. \end{aligned}$$

Consequently, Problem (Phase retrieval) is equivalent to

$$\boxed{\begin{aligned} &\text{recover } X \in \mathbb{C}^{d \times d} \\ &\text{such that } v_j^* X v_j = y_j^2, \forall j \leq m, \\ &X \succeq 0, \\ &\mathrm{rank}(X) \leq 1. \end{aligned}} \tag{Matrix PR}$$

This is, as announced, a low rank matrix recovery problem.

1.3.3 Machine learning

In a machine learning task, the goal is to predict some output y given some input x. For instance, the input can be a photograph, and the output the name of the objects represented on the photograph, or the input can be a low-quality audio signal and the output the corresponding high-quality signal. We denote P the “perfect” prediction function, which to an input x maps the correct

$$y = P(x).$$

The predictor P is unknown and must be learned from the available input-output examples $(x_1, y_1), \dots, (x_n, y_n)$. This leads to the problem

$$\boxed{\begin{gathered} \text{find } P \in \mathcal{H} \\ \text{such that } P(x_k) = y_k, \forall k \leq n, \end{gathered}} \tag{ML}$$

where $\mathcal{H}$ is a well-chosen class of functions ($\mathcal{H}$ can for instance be the set of linear maps, or the set of neural networks with a given architecture).

The questions raised by Problem (ML) are quite different from the ones raised by the other inverse problems we have seen. Indeed, it often happens that the perfect predictor P is not in the chosen set $\mathcal{H}$, in which case the problem may not have an exact solution, only an approximate one. In addition, if $\mathcal{H}$ is a bit sophisticated, there are typically several (and even many) elements $P \in \mathcal{H}$ such that $P(x_k) = y_k$ for all k (in other words, the uniqueness property does not hold). All these elements P yield the same predictions for the available inputs $x_1, \dots, x_n$, but may differ significantly on unseen examples. It is therefore important to choose, among these P, the one which has the best chances to perform well on unseen examples. [3]

1.3.4 Other examples

Dictionary learning In this problem, one is given a set of “interesting” signals $y_1, \dots, y_m \in \mathbb{R}^d$ (e.g. patches of natural photographs or of medical images), and the goal is to learn a good “representation” for them, under the form of a *dictionary*. A dictionary is a set of elements $a_1, \dots, a_M \in \mathbb{R}^d$, usually called *atoms*, such that any signal y_k can be written as a linear combination of a small number of atoms :

$$y_k = \sum_{l=1}^{M} \lambda_l^{(k)} a_l \quad \text{such that } ||\lambda^{(k)}||_0 \text{ is small.}$$

[3] This is called the *generalization* problem.

We write the dictionary in matricial form by concatenating the atoms into a single matrix :

$$A = \begin{pmatrix} a_1 & a_2 & \dots & a_M \end{pmatrix}$$

Finding the dictionary A consists in solving the following problem

$$\boxed{\begin{aligned} &\text{find } A \in \mathbb{R}^{d\times M}, \lambda^{(1)}, \dots, \lambda^{(m)} \in \mathbb{R}^M \\ &\text{such that } A\lambda^{(k)} = y_k, \forall k \leq m, \\ &||\lambda^{(k)}||_0 \leq S, \end{aligned}} \qquad \text{(Dictionary learning)}$$

where S is an a priori bound on the number of atoms involved in the decomposition of each signal y_k.

Super-resolution *Super-resolution* is a general term, which covers all problems where one tries to recover a “sharp” signal from a “blurred” version. In this paragraph, we present the simplest possible model for such a problem.

The signal we aim at identifying is a collection of point masses in $[0;1[$. The positions of the masses are $\tau_1, \dots, \tau_S$ and their weights are $a_1, \dots, a_S$. This signal can be represented by a measure

$$\mu = \sum_{s=1}^{S} a_s \delta_{\tau_s} \in \mathcal{M}([0;1[),$$

where $\mathcal{M}([0;1[)$ is the set of signed (or even complex-valued, if $a_1, \dots, a_S$ are complex) finite Borel measures on $[0;1[$ and, for any s, δ_{τ_s} is the dirac at position τ_s.[4]

The information we have to identify our point masses, the “blurred” version of the signal, is modelled as the set of low-frequency coefficients of the Fourier transform of μ : for all $k = -N, \dots, N$, we have access to

$$\hat{\mu}[k] = \int_0^1 e^{-2\pi i k t} d\mu(t) \left(= \sum_{s=1}^{S} a_s e^{-2\pi i k \tau_s} \right).$$

[4] that is to say, δ_{τ_s} is the measure such that, for any measurable $E \subset [0;1[$, $\mu(E) = 1$ if $\tau_s \in E$ and $\mu(E) = 0$ otherwise.

If we call $y_{-N}, \dots, y_N$ the known Fourier coefficients, the problem can be written as

$$\boxed{\begin{array}{c}\text{find } \mu \in \mathcal{M}([0;1[) \\ \text{such that } \hat{\mu}[k] = y_k, \forall k = -N, \dots, N, \\ \text{and } \mu \text{ is a sum of } S \text{ diracs.}\end{array}} \qquad \text{(Super-resolution)}$$

This problem can be seen as a continuous version of compressed sensing (Problem (CS)). The unknown, instead of a finite-dimensional vector, is a measure on $[0;1[$, but it must still be recovered from linear measurements, and satisfies a sparsity constaint (it is the sum of at most S diracs).

Chapitre 2

Convexification

As discussed in the introduction, non-convexity is a major hurdle for numerically solving inverse problems. Simple local search algorithms are at risk of getting stuck in poor local optima. A possible strategy to overcome this difficulty is to approximate the non-convex problem with a convex one. This convex approximation is called a *convex relaxation*. Since numerically solving a convex problem is in general doable, we can in general solve the approximation. At first sight, there is no reason why solving this approximation would provide useful information towards solving the non-convex problem. But surprisingly, it turns out that, in many situations, the convex approximation has the same solution as the original non-convex problem ! One then says that relaxation is *tight*. When this happens, it yields a convenient method for solving the non-convex problem. This general scheme is depicted on Figure 2.1.

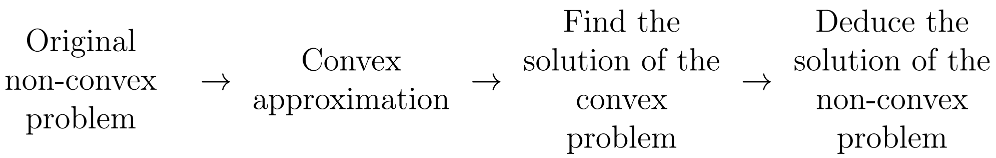
Original non-convex problem
→
Convex approximation
→
Find the solution of the convex problem
→
Deduce the solution of the non-convex problem

FIGURE 2.1 – Principle of convexified algorithms, when relaxation is tight.

2.1 The basis : compressed sensing

2.1.1 Convexification : principle

The model example for this chapter, which serves as a basis for other problems, is compressed sensing.

$$\boxed{\begin{aligned} &\text{recover } x \in \mathbb{R}^d \\ &\text{such that } Ax = y, \\ &\text{and } ||x||_0 \leq k. \end{aligned}} \tag{CS}$$

When the problem has a unique solution, it is the vector of minimal ℓ^0-norm among the vectors x such that $Ax = y$. This allows to reformulate the problem as

$$\boxed{\begin{aligned} &\text{minimize } ||x||_0 \\ &\text{for } x \in \mathbb{R}^d \\ &\text{such that } Ax = y. \end{aligned}} \tag{$||.||_0$ min}$$

The set $\{x \in \mathbb{R}^d, Ax = y\}$ is convex. The non-convex part of the problem is the objective function $||.||_0$. To make the problem convex, we replace the ℓ^0-norm with the ℓ^1-norm :

$$||x||_1 = \sum_{i=1}^{d} |x_i|,$$

which leads to the following *convex* problem :

$$\boxed{\begin{aligned} &\text{minimize } ||x||_1 \\ &\text{for } x \in \mathbb{R}^d \\ &\text{such that } Ax = y. \end{aligned}} \tag{Basis Pursuit}$$

2.1.2 Intuition

The reason why the ℓ^1-norm is a good choice of a convex approximation for the ℓ^0-norm is the following proposition.

Proposition 2.1 : extremal points of the ℓ^1-ball

The extremal points [a] of the unit ℓ^1-ball $\{x \in \mathbb{R}^d, ||x||_1 \leq 1\}$ are the vectors with exactly one non-zero coordinate, equal to -1 or 1.

[a] An *extremal point* of a convex set C is a point y which cannot be written as

$$y = (1-\theta)z_1 + \theta z_2$$

for $z_1, z_2 \in C$ different from y and $\theta \in [0;1]$.

Démonstration. Exercise. □

This proposition says that the extremal points of the unit ℓ^1-ball are the maximally sparse non-zero vectors. If the vector x_* we are trying to recover through Problem (CS) is sparse, then it is a linear combination of a small number of extremal points of the ℓ^1-ball, which can be geometrically interpreted as the fact that it belongs to a "corner" of the ℓ^1-ball

$$B_{\ell^1,x_*} \overset{def}{=} \{x \in \mathbb{R}^d, ||x||_1 \leq ||x_*||_1\}.$$

The convex approximation (Basis Pursuit) has a unique minimizer equal to x_* if and only if

$$\begin{aligned} &\nexists x \in \mathbb{R}^d \text{ such that } Ax = y = Ax_* \text{ and } ||x||_1 \leq ||x_*||_1 \\ \iff \quad & B_{\ell^1,x_*} \cap \{x \in \mathbb{R}^d, Ax = Ax_*\} = \{x_*\}. \end{aligned}$$

And the intersection of B_{ℓ^1,x_*} and an affine space containing x_* has much more chances to be the singleton $\{x_*\}$ if x_* is in a "corner" of B_{ℓ^1,x_*} (very crudely, if x_* is in a "corner", then, in the neighborhood of x_*, B_{ℓ^1,x_*} occupies only a small fraction of the space ; it is therefore easier not to intersect it when considering an affine space going through x_*). This is depicted on Figure 2.2.

2.1.3 Tightness guarantees under restricted isometry

The convex problem (Basis Pursuit) can be traced back to at least the 70's. Since then, many researchers have proposed conditions on x_* and A under which the relaxation is tight (that is, the solutions of (Basis Pursuit) and (CS) are the same). A major progress (due notably to Candès, Donoho, Romberg and Tao) on this subject was, around twenty years ago, the introduction of the so-called *Restricted Isometry Property*, which is a simple

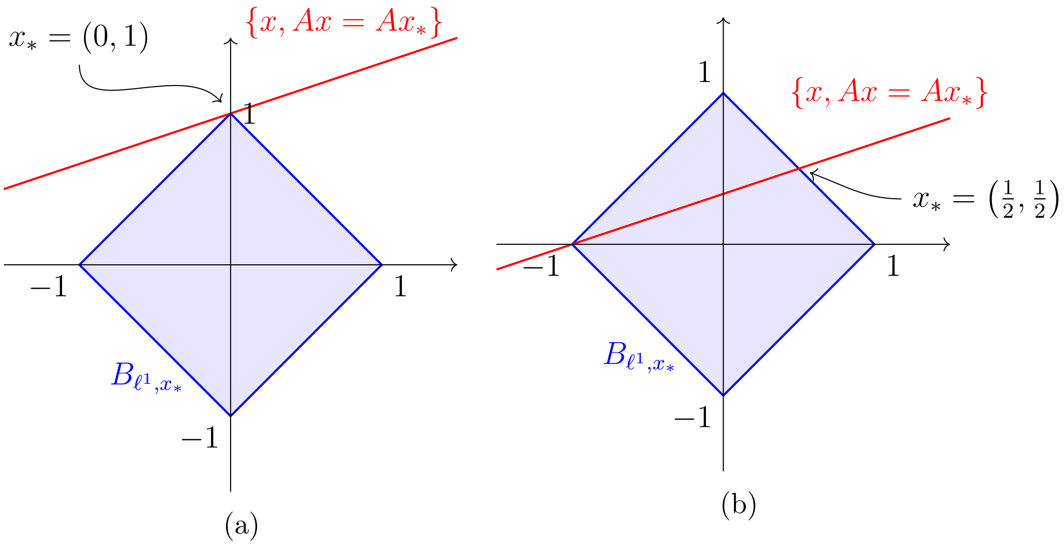

FIGURE 2.2 – Representation of B_{ℓ^1,x_*} and $\{x \in \mathbb{R}^2, Ax = Ax_*\}$ for $A = \begin{pmatrix} 1 & -3 \end{pmatrix}$ in two situations : (a) when $x_* = (0,1)$ is sparse ; (b) when $x_* = \left(\frac{1}{2},\frac{1}{2}\right)$ is not sparse. Observe that $B_{\ell^1,x_*} \cap \{x, Ax = Ax_*\}$ is a singleton in the first case, but not in the second one.

assumption on A under which it is possible to guarantee tightness without imposing stringent conditions on x_*.

Definition 2.2 : restricted isometry

Let $A \in \mathbb{R}^{m\times d}$ be a matrix. For any $k \in \{1,\dots,d\}$, we define the k-restricted isometry constant δ_k of A as the smallest real number such that

$$(1-\delta_k)||z||_2 \leq ||Az||_2 \leq (1+\delta_k)||z||_2$$

for all vectors $z \in \mathbb{R}^d$ with at most k non-zero coordinates.

Tightness of the convex relaxation (Basis Pursuit) under a restricted isometry condition is guaranteed by the following theorem.

Theorem 2.3

Let $A \in \mathbb{R}^{m\times d}$ be a matrix. For some $k \in \{1, \dots, d\}$, we assume that its $4k$-restricted isometry constant satisfies

$$\delta_{4k} < \frac{1}{4}. \tag{2.4}$$

For any $x_* \in \mathbb{R}^d$ with at most k non-zero coordinates, Problem (Basis Pursuit) with $y = Ax_*$ has a unique solution, which is x_*.

Under the same condition, it is moreover possible to prove a stability result for the convex relaxation : if y is "close" to Ax_*, then the solution of a slight modification of (Basis Pursuit) is "close" to x_*. The proof of Theorem 2.3 is the subject of an exercise, which follows [Candès, Romberg, and Tao, 2006].

Let us keep in mind that the restricted isometry property is a *sufficient* but not *necessary* condition for the correctness of the basis pursuit approach : there are matrices A for which condition (2.4) does *not* hold and, nevertheless, Problems (CS) and (Basis Pursuit) have the same solution. However, it turns out that many natural matrices A satisfy the condition, hence Theorem 2.3 explains the success of the basis pursuit approximation in several interesting situations. The following theorem provides the simplest example of matrices with the restricted isometry property : matrices chosen at random according to a normal distribution (with high probability).

Theorem 2.4 : [Candès and Tao, 2005]

Let $c > 0$ be some explicit constant, whose value we will not give here. We assume that $A \in \mathbb{R}^{m\times d}$ is generated at random according to a normal distribution [a]. If

$$ck\log(d/k) \leq m,$$

Condition (2.4) holds with high probability. [b]

[a] that is, each coefficient of A is chosen independently at random according to a normal law $\mathcal{N}(0, 1/m)$.

[b] *With high probability* means that it holds with probability at least $1 - e^{-\alpha m}$ for some constant $\alpha > 0$.

This theorem, combined with Theorem 2.3, shows that convexification allows to recovery k-sparse vectors from $O(k\log(d/k))$ linear measurements, which is surprisingly few. Indeed, if the indices of the non-zero coordinates of the vector were known, k linear measurements would be necessary. Not knowing the indices only increases this number by a logarithmic factor.

2.2 Low-rank matrix recovery

After compressed sensing, convexification techniques have been developed for other non-convex problems. In particular, an important part of the theory developed for compressed sensing can be transposed to low-rank matrix recovery.

2.2.1 Convexification : principle

We recall the general form of a low-rank matrix recovery problem.

$$\boxed{\begin{aligned}\text{recover } & X \in \mathbb{R}^{d_1\times d_2}\\ \text{such that } & \mathcal{L}(X) = y,\\ \text{and } & \operatorname{rank}(X) \leq r,\end{aligned}} \tag{Low rank}$$

which, when the solution is unique, is equivalent to

$$\boxed{\begin{aligned}\text{minimize } & \operatorname{rank}(X)\\ \text{for } & X \in \mathbb{R}^{d_1\times d_2}\\ \text{such that } & \mathcal{L}(X) = y.\end{aligned}} \tag{Rank min}$$

In the same way as, in the case of compressed sensing, we have approximated the ℓ^0-norm with the ℓ^1-norm, we can replace the non-convex rank functional with a convex approximation. For the rank, the most reasonable convex approximation is the *nuclear norm*.

Definition 2.5 : nuclear norm

For any $X \in \mathbb{R}^{d_1 \times d_2}$, the *nuclear norm* of X is

$$||X||_* = \sum_{k=1}^{\min(d_1,d_2)} \lambda_k(X),$$

where $\lambda_1(X), \dots, \lambda_{\min(d_1,d_2)}(X)$ are the singular values of X.[a]
If $d_1 = d_2$ and $X \succeq 0$, this definition can be simplified :

$$||X||_* = \mathrm{Tr}(X).$$

[a] Readers who are not familiar with the singular value decomposition are encouraged to do the first exercise of the exercise sheet.

Proof of the last assertion in the definition. If $d_1 = d_2$ and $X \succeq 0$, the matrix X can be diagonalized in an orthonormal basis and has nonnegative eigenvalues $\mu_1, \dots, \mu_{d_1}$: there exists $U \in O_{d_1}(\mathbb{R})$ such that

$$X = U \begin{pmatrix} \mu_1 & & \\ & \ddots & \\ & & \mu_{d_1} \end{pmatrix} U^T.$$

This equality is the singular value decomposition of X : $\mu_1, \dots, \mu_{d_1}$ are the singular values of X, so

$$\begin{aligned}
||X||_* &= \sum_{k=1}^{d_1} |\mu_k| \\
&= \sum_{k=1}^{d_1} \mu_k \\
&= \mathrm{Tr} \begin{pmatrix} \mu_1 & & \\ & \ddots & \\ & & \mu_{d_1} \end{pmatrix} \\
&= \mathrm{Tr} \left(\begin{pmatrix} \mu_1 & & \\ & \ddots & \\ & & \mu_{d_1} \end{pmatrix} U^T U \right) \qquad \text{as } U^T U = I_{d_1} \\
&= \mathrm{Tr} \left(U \begin{pmatrix} \mu_1 & & \\ & \ddots & \\ & & \mu_{d_1} \end{pmatrix} U^T \right) \\
&= \mathrm{Tr}(X).
\end{aligned}$$

□

The motivation for using the nuclear norm is the following proposition, which is a matricial analogue of Proposition 2.1.

Proposition 2.6

The extremal points of the nuclear norm unit ball

$$\{X \in \mathbb{R}^{d_1 \times d_2}, ||X||_* \leq 1\}$$

are the exactly the matrices with unit Frobenius norm [a] and rank 1.

[a] The Frobenius norm is $||X||_F = \left(\sum_{\substack{1 \leq k_2 \leq d_2 \\ 1 \leq k_1 \leq d_1}} X_{k_1,k_2}^2 \right)^{1/2}$.

Démonstration. Exercise. □

Replacing the rank with the nuclear norm in the non-convex problem (Rank min), we arrive at the following convex approximation :

$$\boxed{\begin{aligned} &\text{minimize } ||X||_* \\ &\quad \text{for } X \in \mathbb{R}^{d_1 \times d_2} \\ &\text{such that } \mathcal{L}(X) = y. \end{aligned}} \tag{Nuclear min}$$

2.2.2 Tightness guarantees under restricted isometry

As in the case of compressed sensing, the nuclear norm relaxation is often tight (that is, the solution of (Nuclear min) is the same as the one of Problem (Low rank)), meaning that solving the convex problem actually solves the non-convex one. A lot of work has been devoted to finding classes of operators $\mathcal{L}$ for which this phenomenon provably happens. A simple particular property under which tightness necessarily holds is a matricial analogue of restricted isometry.

Definition 2.7 : restricted isometry for matrices

Let $\mathcal{L} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^m$ be a linear operator.
For any $r \in \{1, \dots, \min(d_1, d_2)\}$, the r-restricted isometry constant δ_r of $\mathcal{L}$ is the smallest real number such that

$$(1 - \delta_r)||X||_F \leq ||\mathcal{L}(X)||_2 \leq (1 + \delta_r)||X||_F$$

for all matrices $X \in \mathbb{R}^{d_1 \times d_2}$ with rank at most r.

Theorem 2.8 : [Recht, Fazel, and Parrilo, 2010]

Let $\mathcal{L} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^m$ be a linear operator. We assume that, for some $r \in \{1, \dots, \min(d_1, d_2)\}$, its $5r$-restricted isometry constant satisfies

$$\delta_{5r} < \frac{1}{10}. \tag{2.8}$$

For any $X_* \in \mathbb{R}^{d_1 \times d_2}$ with rank at most r, Problem (Nuclear min) with $y = \mathcal{L}(X_*)$ has a unique solution, which is X_*.

The proof of this result is an adaptation to matrices of the proof of Theorem 2.3 proposed in [Candès, Romberg, and Tao, 2006]. It is the subject of an exercise.

As in the case of compressed sensing, it can be shown that, if we choose an operator $\mathcal{L}$ at random (according to the simplest possible distribution), it satisfies the restricted isometry property (2.8) with high probability.

Theorem 2.9 : [Candès and Plan, 2011]

Let us assume that $\mathcal{L}$ is of the form

$$\mathcal{L} : X \in \mathbb{R}^{d_1 \times d_2} \to \left(\mathrm{Tr}(A_k X^T)\right)_{k=1,\dots,m},$$

where $A_1, \dots, A_m$ are chosen independently according to standard normal distributions (that is, each coordinate of each A_k is chosen independently according to the law $\mathcal{N}(0, 1/m)$).
There exists a constant $c > 0$ such that, if

$$m \geq cr(d_1 + d_2),$$

then Condition (2.8) holds with high probability [a].

[a] that is, with probability at least $1 - e^{-\alpha m}$ for some $\alpha > 0$.

It can be checked that the set of rank r matrices has "dimension" [1]

$$r(d_1 + d_2 - r).$$

As a consequence, there is no hope to recover a rank r-matrix from less that $r(d_1+d_2-r)$ linear measurements. The combination of Theorems 2.8 and 2.9 therefore guarantees that, when $\mathcal{L}$ follows a normal law, solving the convex (Nuclear min) problem allows to recover a low-rank matrix from a number of measurements which is only a logarithmic factor away from optimal.

2.2.3 Problems without restricted isometry

Phase retrieval

Other linear operators than the ones considered in Theorem 2.9 can be shown to satisfy Condition (2.8). Unfortunately, there are also many natural operators which do not satisfy it. In particular, the linear operators arising in the problems of phase retrieval and matrix completion (Problems (Matrix PR) and (Matrix completion)) do not. These problems can still be solved through convexification techniques, but proving the correctness of the approach must be done with other tools than restricted isometry.

Let us first discuss phase retrieval. We recall below the general form of a phase retrieval problem (Problem (Phase retrieval)) and its reformulation as a low-rank matrix recovery problem (Problem (Matrix PR)).

(Phase retrieval - original)

$$\begin{aligned} &\text{find } x \in \mathbb{C}^d \\ &\text{s.t. } |L_j(x)| = y_j, \forall j \leq m. \end{aligned}$$

(Phase retrieval - matricial)

$$\begin{aligned} &\text{find } X \in \mathbb{C}^{d\times d} \\ &\text{s.t. } v_j^* X v_j = y_j^2, \forall j \leq m, \\ &\qquad X \succeq 0, \\ &\qquad \text{rank}(X) \leq 1. \end{aligned}$$

[1] We use quotes because this set is neither a vector space nor a smooth submanifold of $\mathbb{R}^{d_1\times d_2}$, hence formally talking about the "dimension" of this set requires a careful definition of the notion.

We apply to the matricial formulation the same strategy we have seen for general low-rank matrix recovery problems : we replace the rank functional with the nuclear norm. Since the matrix X we are looking for must be semidefinite positive, its nuclear norm is equal to its trace (see Definition 2.5), which results in the following convex relaxation :

$$\boxed{\begin{aligned} &\text{minimize } \mathrm{Tr}(X) \\ &\quad \text{for all } X \in \mathbb{C}^{d\times d}, \\ &\text{such that } v_j^* X v_j = y_j^2, \forall j \leq m, \\ &\qquad\qquad X \succeq 0. \end{aligned}} \tag{PhaseLift}$$

This relaxation has been introduced in [Chai, Moscoso, and Papanicolaou, 2011] and [Candès, Eldar, Strohmer, and Voroninski, 2011]. The name (PhaseLift) comes from the second article.

For "interesting" values of m and families of measurement vectors $v_1, \dots, v_m \in \mathbb{C}^d$, the linear operator

$$\mathcal{L} : X \in \mathbb{C}^{d\times d} \quad \to \quad (v_j^* X v_j)_{1\leq j\leq m} \in \mathbb{R}^m$$

does generally not satisfy a restricted isometry property.[2] Nevertheless, it does not prevent the convex relaxation (PhaseLift) from being oftentimes tight. This tightness can be numerically observed for many families of measurements vectors, and has been rigorously proven for a few ones. The simplest case is when $v_1, \dots, v_m$ are chosen at random according to normal laws ; in this case, tightness is guaranteed by the following theorem.

[2] Here is a crude and oversimplified idea of why it does not, in the case where $v_1, \dots, v_m$ are chosen independently at random according to standard normal distributions and $m = O(d)$. The operator $\mathcal{L}$ depends quadratically on each v_j (by comparison, in Theorem 2.9, it depends linearly on each A_k). Therefore, it behaves somewhat similarly to a sequence of *squared* Gaussian variables (rather than a sequence of plain Gaussian variables as in Theorem 2.9). Squared Gaussian variables have much more frequent high values than plain Gaussian ones, hence their *concentration* properties are less good : they deviate more from their average expected behavior, hence there are a few directions along which $\mathcal{L}$ dilates distances much more than along the other ones, which prevents it from being an approximate isometry.

Theorem 2.10 : [Candès and Li, 2014]

Let us assume that $v_1, \dots, v_m$ are chosen independently at random in $\mathbb{C}^d$, following normal distributions. Let $x_0 \in \mathbb{C}^d$ be a vector. We consider the convex problem (PhaseLift) with $y_j = |\langle x_0, v_j \rangle|$ for all j. There exists a constant $c > 0$ such that, if

$$m \geq cd,$$

then the relaxation provided by (PhaseLift) is tight with high probability[a] : it has a unique solution, the same as Problem (Matrix PR), that is $X = x_0 x_0^*$.

[a] that is, with probability at least $1 - e^{-\gamma m}$ for some constant $\gamma > 0$,

We do not provide the proof of this result. The one proposed in [Candès and Li, 2014] relies on the notion of *dual certificate*, which we will introduce later. Another one, from [Chen, Chi, and Goldsmith, 2015], uses a restricted isometry property, but for different norms than in Definition 2.7.

Remark

The semidefinite positiveness constraint, by itself, tends to encourage solutions of optimization problems to have small rank. Therefore, in Problem (PhaseLift), the trace minimization is not always necessary. The bare feasibility problem

$$\boxed{\begin{gathered}\text{find } X \in \mathbb{C}^{d \times d}, \\ \text{such that } v_j^* X v_j = y_j^2, \forall j \leq m, \\ X \succeq 0.\end{gathered}}$$

is already a good convex relaxation for the original non-convex problem (in the sense that it satisfies similar guarantees as the ones stated for (PhaseLift) in Theorem 2.10).

Matrix completion

We recall the problem of matrix completion :

$$\boxed{\begin{gathered}\text{recover } X \in \mathbb{R}^{d_1 \times d_2}\\ \text{such that } X_{ij} = y_{ij}, \forall (i,j) \in \Omega\\ \text{and } \operatorname{rank}(X) \leq r.\end{gathered}} \qquad \text{(Matrix completion)}$$

We obtain a convex relaxation by following the same principle as before : we replace the minimization of the rank with the minimization of the nuclear norm.

$$\boxed{\begin{gathered}\text{minimize } ||X||_*\\ \text{for all } X \in \mathbb{R}^{d_1 \times d_2}\\ \text{such that } X_{ij} = y_{ij}, \forall (i,j) \in \Omega.\end{gathered}} \qquad \text{(Convex MC)}$$

The linear operator

$$\mathcal{L} : X \in \mathbb{R}^{d_1 \times d_2} \quad \to \quad (X_{ij})_{(i,j)\in\Omega} \in \mathbb{R}^{\mathrm{Card}(\Omega)}$$

does not satisfy a restricted isometry property similar to Condition (2.8). Indeed, for any $(k,l) \notin \Omega$, the matrix $e_{k,l}$ whose coefficients are all zero, except the (k,l)-th one which is 1, satisfies

$$\mathcal{L}(e_{k,l}) = 0.$$

As $\operatorname{rank}(e_{k,l}) = 1$, the r-restricted isometry constant of $\mathcal{L}$ can never be below 1, for any $r \in \{1, \dots, \min(d_1, d_2)\}$.

Besides showing the absence of restricted isometry, this remark points at a fundamental limitation in matrix completion : matrices which are "too concentrated" on a few coefficients cannot be recovered from a subset of coefficients ; the matrix $e_{k,l}$ cannot be recovered from the observation of its coefficients with indices in Ω. Only matrices whose coefficients are sufficiently "spread out" can be recovered : they must not have very large coefficients, and the largest coefficients must also not be aligned on a row or a column. This can be formalized through the so-called *incoherence condition.*

Definition 2.11 : incoherence condition

Let $X \in \mathbb{R}^{d_1 \times d_2}$ be a rank r matrix. It can be written as

$$X = U \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_r \end{pmatrix} V,$$

where $U \in \mathbb{R}^{d_1 \times r}$ has orthonormal columns, and $V \in \mathbb{R}^{r \times d_2}$ has orthonormal rows. [a]

We say that X satisfies the *incoherence condition* with parameter $\mu_0 > 0$ if

$$|U_{k,l}| \le \sqrt{\frac{\mu_0}{d_1}}, \quad \forall k \le d_1, l \le r, \tag{2.12a}$$

$$\text{and } |V_{k,l}| \le \sqrt{\frac{\mu_0}{d_2}}, \quad \forall k \le r, l \le d_2. \tag{2.12b}$$

[a] This can be deduced from the singular value decomposition.

Remark

For any $l \le r$, the l-th column of U has unit norm, hence at least one of its d_1 coordinates satisfies

$$|U_{k,l}| \ge \sqrt{\frac{1}{d_1}},$$

and equality holds only if all coordinates are equal to $\sqrt{\frac{1}{d_1}}$ in absolute value. Hence, Condition (2.12a) holds with $\mu_0 = 1$ if and only if the coordinates of U are "maximally spread out", that is they are all equal (in absolute value). Similarly, Condition (2.12b) holds with $\mu_0 = 1$ if and only if all coordinates of V are equal in absolue value.

This intuitively justifies the scalings $\sqrt{\frac{1}{d_1}}$ and $\sqrt{\frac{1}{d_2}}$ in Equations (2.12a) and (2.12b).

Theorem 2.12 : [Chen, 2015]

We assume that Ω is chosen by selecting each pair $(k,l) \in \{1,\ldots,d_1\} \times \{1,\ldots,d_2\}$ independently at random, with some probability $p > 0$.
Let $X_* \in \mathbb{R}^{d_1 \times d_2}$ be a matrix satisfying the incoherence condition with some parameter $\mu_0 > 0$.
There exists a constant $c > 0$ such that, if

$$p \geq c\frac{\mu_0 r(d_1+d_2)\log^2(d_1+d_2)}{d_1 d_2},$$

then, with high probability, [a] the convex relaxation (Convex MC) (with $y_{ij} = X_{*ij}$ for all $(i,j) \in \Omega$) is tight : it has a unique solution, which is X_*, the same as (Matrix completion).

[a] that is, with probability at least $1-(d_1+d_2)^{-\alpha}$ for some $\alpha > 0$,

Remark

The condition $p \geq c\frac{\mu_0 r(d_1+d_2)\log^2(d_1+d_2)}{d_1 d_2}$ means that the cardinal of Ω is of order

$$p d_1 d_2 \geq c\mu_0 r(d_1+d_2)\log^2(d_1+d_2).$$

Therefore, when μ_0 is of order 1, the unknown matrix can be recovered through convex programming using a number of measurements which is only a logarithmic factor larger than the optimum (see the discussion after Theorem 2.9).

2.3 Super-resolution

We recall the super-resolution problem presented in the introduction, where one must recover a sum of a few diracs in $[0;1[$ from its low-frequency Fourier coefficients.

$$\boxed{\begin{gathered}\text{find } \mu \in \mathcal{M}([0;1[)\\ \text{such that } \hat{\mu}[k] = y_k, \forall k = -N,\ldots,N,\\ \text{and } \mu \text{ is a sum of } S \text{ diracs.}\end{gathered}} \qquad \text{(Super-resolution)}$$

Here, $\mathcal{M}([0;1[)$ is the set of complex-valued finite Borel measures on $[0;1[$.

2.3.1 Convexification through the total variation norm

A reasonable convex approximation for Problem (Super-resolution) can be proposed using the analogy between super-resolution and compressed sensing.

In compressed sensing, the unknowns are sparse vectors of $\mathbb{R}^d$, that is, they can be written as

$$x = \sum_{s=1}^{k} x_{i_s} e_{i_s},$$

where k is an integer much smaller than d, $i_1, \dots, i_k$ are the indices of the non-zero coordinates of x and, for each j, $e_j \in \mathbb{R}^d$ is the j-th vector of the canonical basis. [3] We have seen that a good convex approximation of the non-convex compressed sensing problem, (Basis Pursuit), is obtained by replacing the non-convex ℓ^0-norm with

$$||x||_1 = \sum_{s=1}^{k} |x_{i_s}|.$$

In super-resolution, the unknowns are sparse measures over $[0;1[$. They can be written as

$$\mu = \sum_{s=1}^{S} a_s \delta_{\tau_s},$$

where S is an integer, $\tau_1, \dots, \tau_s \in [0;1[$ are the positions of the diracs and $a_1, \dots, a_s \in \mathbb{R}$ are coefficients. Here, the diracs δ_{τ_s} play the roles of the canonical vectors e_{i_s}. Therefore, is seems reasonable to approximate the non-convex requirement that μ is a sum of S diracs using the following analogue of the ℓ^1-norm :

$$||\mu||_{\text{analogue-}\ell^1} = \sum_{s=1}^{S} |a_s|.$$

This analogue of the ℓ^1-norm happens to coincide with a standard norm of finite measures, called *total variation*. The exact definition of this norm follows. Since we will not explicitely use it in most of the rest of the section,

[3] i.e. the vector whose coordinates are all 0, except the j-th one, which is 1

readers who are not familiar with measure theory are encouraged to skip it. Readers who, on the contrary, are familiar with measure theory and want to know more about total variation are encouraged to read [Rudin, 1987, Chapter 6], notably Theorem 6.19.

Definition 2.13 : total variation

For any complex-valued finite measure $\mu \in \mathcal{M}([0;1[)$, its *total variation norm* is

$$||\mu||_{TV} = \sup_{(E_1,\dots,E_N)\in\Pi} \sum_{s=1}^{N} |\mu(E_s)|,$$

where Π is the set of all finite partitions of $[0;1[$:

$$\begin{aligned}\Pi = \{(E_1,\dots,E_N) \text{ for } N \in \mathbb{N}^*, E_1,\dots,E_N \text{ measurable},\\ \text{such that } E_1 \cup \dots \cup E_N = [0;1[,\\ E_i \cap E_j = \emptyset, \forall i \neq j\}.\end{aligned}$$

Proposition 2.14 : equivalent definition of total variation

A definition equivalent to the previous one is

$$\begin{aligned}||\mu||_{TV} = \sup \Bigg\{ \mathrm{Re}\left(\int_0^1 f(t)d\mu(t)\right), |f(t)| \leq 1, \forall t \in [0;1],\\ f : [0;1] \to \mathbb{C} \text{ is continuous} \Bigg\}.\end{aligned}$$

In addition, if the supremum is attained by a function f, it must hold

$$\mathrm{Supp}(\mu) \subset \{t \in [0;1[, |f(t)| = 1\}\,. \tag{2.14}$$

The total variation norm shares a property similar with Propositions 2.1 and 2.6 : its extremal points are the diracs.

Replacing the "sum of S diracs" constraint with the minimization of the total variation norm, we arrive at the following problem, proposed in [de Castro and Gamboa, 2012] :

$$\boxed{\begin{aligned} &\text{minimize } ||\mu||_{TV} \\ &\quad\text{for } \mu \in \mathcal{M}([0;1[), \\ &\text{such that } \hat{\mu}[k] = y_k, \forall k = -N, \dots, N. \end{aligned}} \tag{Min TV}$$

Remark

Although Problem (Min TV) is convex, it cannot be solved with standard solvers as easily as the other convex problems we have encountered so far. Indeed, the unknown μ belongs to an infinite-dimensional vector space $\mathcal{M}([0;1[)$, hence does not admit a convenient representation allowing manipulation on a computer. By contrast, in the problems we have seen until now, the unknowns were vectors or matrices, with a finite number of coefficients.
The main three approaches to numerically solve Problem (Min TV) are the following ones.

— Showing that it is equivalent to a (finite-dimensional) semidefinite problem (see the exercises for details) : this approach has the drawback to strongly rely on the properties of the Fourier coefficients. Therefore, it cannot be generalized to more complex super-resolution problems than (Super-resolution).

— Discretizing the set of measures : one approximates a measure on $[0;1[$ by a measure on

$$\left\{\frac{0}{N}, \frac{1}{N}, \dots, \frac{N-1}{N}\right\}$$

for some large integer N. The approximation can be represented by a finite number of coefficients.

— Applying a general convex or non-convex solver directly to the infinite-dimensional problem. If the intermediate solver iterates turn out to be finite sums of diracs, they can actually be represented by a finite number of parameters (although this number may grow with the iteration count).

2.3.2 No restricted isometry property

As in the case of compressed sensing and low-rank matrix recovery, the solution of the convex (Min TV) problem is often the same as the solution of the original non-convex (Super-resolution) problem. A natural first idea to rigorously establish this fact is to ask whether some analogue of restricted isometry property (Definition 2.2) holds. One could define, in the context of super-resolution, the S-restricted isometry constant as the smallest real number $\delta \geq 0$ such that

$$(1-\delta)||\mu||_{\text{simili } \ell^2} \leq ||(\hat{\mu}[-N],\dots,\hat{\mu}[N])||_2 \leq (1+\delta)||\mu||_{\text{simili } \ell^2} \tag{2.16}$$

for all measures μ which are a sum of S diracs, where $||.||_{\text{simili } \ell^2}$ is a norm which should mimic, on the set of measures, the ℓ^2-norm of vectors.

Unfortunately, this definition does not lead to a useful quantity : a number δ satisfying Equation (2.16) is necessarily at least 1 (for $S \geq 2$). Indeed, let $(x_n)_{n\in\mathbb{N}}$ be a sequence of strictly positive numbers going to 0. For any n, we set

$$\mu_n = \delta_0 - \delta_{x_n}.$$

For any k,

$$\begin{aligned}\hat{\mu}_n[k] &= \int_0^1 e^{-2\pi ikt} d\mu_n(t)\\ &= e^0 - e^{-2\pi ikx_n}\\ &\to e^0 - e^0 = 0 \text{ when } n \to +\infty.\end{aligned}$$

Therefore, if we apply Equation (2.16) to $\mu = \mu_n$ and let n go to infinity, we see that either $\delta \geq 1$ or

$$||\mu_n||_{\text{simili } \ell^2} \overset{n\to+\infty}{\longrightarrow} 0,$$

which is in contradiction with the fact that the norm $||.||_{\text{simili } \ell^2}$ should be somewhat similar to the ℓ^2-norm.

2.3.3 Correctness via dual certificates

Since no restricted isometry property holds, proving equality between the solutions of (Super-resolution) and (Min TV) must rely on other arguments. The most common one is to use *duality theory*.

A few words on general duality theory

Let us consider a general convex optimization problem, as in Section 1.2 :

$$\boxed{\begin{array}{rl} \text{minimize} & f(x) \\ \text{over all} & x \in H \\ \text{such that} & x \in C_1, \\ & \dots \\ & x \in C_S, \end{array}} \tag{Primal}$$

where H is a real or complex vector space, $f : H \to \mathbb{R} \cup \{+\infty\}$ is a convex function and $C_1, \dots, C_S \subset H$ are convex sets. [4]

When discussing duality, the problem of interest is called the *primal problem*. Duality theory is a general method to associate to Problem (Primal) another convex problem

$$\boxed{\begin{array}{rl} \text{maximize} & g(y) \\ \text{over all} & y \in E \\ \text{such that} & y \in D_1, \\ & \dots \\ & y \in D_T, \end{array}} \tag{Dual}$$

where E is a vector space, $D_1, \dots, D_T$ are convex sets and $g : E \to \mathbb{R} \cup \{-\infty\}$ is a *concave* function. [5]

The method which constructs Problem (Dual) from Problem (Primal) ensures that

$$\max \text{ (Dual)} \leq \min \text{ (Primal)},$$

where min (Primal) and max (Dual) respectively denote the optimal values of Problems (Primal) and (Dual). Additionally, under relatively weak as-

[4] Actually, duality theory applies when the sets C_k have a specific form (they are sublevel sets of convex functions), but most commonly encountered constraint sets can be written under this form.

[5] A function g is concave if $-g$ is convex. Maximizing a concave function g is equivalent to minimizing the convex function $-g$; therefore, maximizing a *concave* function is a *convex* problem.

sumptions on $C_1, \dots, C_S$, the inequality is actually an equality[6] :

$$\min \text{ (Primal)} = \max \text{ (Dual)}. \tag{2.19}$$

When interested in solving Problem (Primal), it is quite useful to consider Problem (Dual) because it provides a way to certify that a candidate minimizer x_* of Problem (Primal) is really a minimizer. Without looking at Problem (Dual), proving that x_* is a minimizer is not easy : it a priori requires to consider *all* elements $x \in C_1 \cap \dots \cap C_S$ and show that none of them yields a smaller value of f. But if we can exhibit a candidate maximizer y_* for Problem (Dual) and verify that

$$f(x_*) = g(y_*),$$

then, from the definition of min (Primal) and max (Dual),

$$\min \text{ (Primal)} \leq f(x_*) = g(y_*) \leq \max \text{ (Dual)}$$

so, from Equation (2.19), the inequalities are equalities :

$$\min \text{ (Primal)} = f(x_*) = g(y_*) = \max \text{ (Dual)}.$$

The first of these equalities guarantees that x_* is a minimizer of Problem (Primal).

In these notes, we do not explain the general method to construct a dual problem from a primal one, but we present the construction and its consequences in the specific case of Problem (Min TV).

Dual of total variation minimization

To construct the dual of Problem (Min TV), we must first rewrite the problem as a "min-max problem".

To begin with, Problem (Min TV) can be very slightly reformulated as

$$\boxed{\begin{gathered}\min_{\mu \in \mathcal{M}([0;1[)} ||\mu||_{TV} \\ \text{under the constraint} \quad y - \hat{\mu}[-N:N] = 0.\end{gathered}}$$

[6] This equality is called *strong duality*.

The first step of the rewriting is to incorporate the constraint $y - \hat{\mu}[-N : N] = 0$ into the objective : the problem above has the same optimal value and minimizers as

$$\boxed{\min_{\mu \in \mathcal{M}([0;1[)} \underbrace{||\mu||_{TV} + 1_{y-\hat{\mu}[-N:N]=0}}_{\overset{def}{=} f_1(\mu)},}$$

where, for any vector $v \in \mathbb{C}^n$, $1_{v=0}$ is defined as

$$\begin{aligned} 1_{v=0} \quad &= 0 \qquad \text{if } v = 0, \\ &= +\infty \quad \text{otherwise.} \end{aligned}$$

From the proposition which follows (see Section A.1 in appendix for the proof),

$$f_1(\mu) = \max_{z \in \mathbb{C}^{2N+1}} ||\mu||_{TV} + \mathrm{Re}\,\langle z, y - \hat{\mu}[-N : N]\rangle\,.$$

Proposition 2.15

For any $v \in \mathbb{C}^n$,

$$1_{v=0} = \max_{z \in \mathbb{C}^n} \mathrm{Re}\,\langle z, v\rangle\,.$$

For any $z \in \mathbb{C}^{2N+1}$,

$$\begin{aligned} \mathrm{Re}\,\langle z, y - \hat{\mu}[-N : N]\rangle &= \mathrm{Re}\,\langle z, y\rangle - \mathrm{Re}\,\langle z, \hat{\mu}[-N : N]\rangle \\ &= \mathrm{Re}\,\langle z, y\rangle - \mathrm{Re}\left(\sum_{k=-N}^{N} \overline{z_k}\hat{\mu}[k]\right) \\ &= \mathrm{Re}\,\langle z, y\rangle - \mathrm{Re}\left(\sum_{k=-N}^{N} \overline{z_k}\int_0^1 e^{-2\pi ikt} d\mu(t)\right) \\ &= \mathrm{Re}\,\langle z, y\rangle - \mathrm{Re}\int_0^1 \left(\sum_{k=-N}^{N} \overline{z_k} e^{-2\pi ikt}\right) d\mu(t), \end{aligned}$$

hence

$$f_1(\mu) = \max_{z \in \mathbb{C}^{2N+1}} \underbrace{||\mu||_{TV} - \mathrm{Re}\int_0^1 \left(\sum_{k=-N}^{N} \overline{z_k} e^{-2\pi ikt}\right) d\mu(t) + \mathrm{Re}\,\langle z, y\rangle\,.}_{\overset{def}{=} F(\mu, z)}$$

Consequently, Problem (Min TV) has the same optimal value and minimizers as

$$\boxed{\min_{\mu\in\mathcal{M}([0;1[)} \max_{z\in\mathbb{C}^{2N+1}} F(\mu,z).} \tag{Primal min-max}$$

This is the reformulation we needed for the primal (Min TV) problem. Now, we define the dual problem by simply switching the minimum and maximum :

$$\boxed{\max_{z\in\mathbb{C}^{2N+1}} \underbrace{\min_{\mu\in\mathcal{M}([0;1[)} F(\mu,z)}_{\overset{def}{=}\ f_2(z)}.} \tag{Dual max-min}$$

The minimization over μ in the definition of f_2 has an explicit solution, given in Proposition 2.16 (see Section A.2 in appendix for a proof) :

$$\begin{aligned} f_2(z) &= \min_{\mu\in\mathcal{M}([0;1[)} ||\mu||_{TV} - \mathrm{Re}\int_0^1 \left(\sum_{k=-N}^{N} \overline{z_k} e^{-2\pi ikt}\right) d\mu(t) + \mathrm{Re}\,\langle z,y\rangle \\ &= \mathrm{Re}\,\langle z,y\rangle - 1_{\left|\sum_{k=-N}^{N} z_k e^{2\pi ikt}\right|\le 1,\forall t}, \end{aligned}$$

where $1_{\left|\sum_{k=-N}^{N} z_k e^{2\pi ikt}\right|\le 1,\forall t} = 0$ if $\left|\sum_{k=-N}^{N} z_k e^{2\pi ikt}\right| \le 1, \forall t \in \mathbb{R}$ and $+\infty$ otherwise.

Proposition 2.16

For any continuous function $f : [0;1] \to \mathbb{C}$,

$$\min_{\mu\in\mathcal{M}([0;1[)} \left(||\mu||_{TV} - \mathrm{Re}\int_0^1 f(t)d\mu(t)\right) \begin{array}{ll} = 0 & \text{if } |f(t)| \le 1, \forall t \in [0;1], \\ = -\infty & \text{otherwise.} \end{array}$$

In addition, if a minimizer μ exists, it satisfies

$$\mathrm{Supp}(\mu) \subset \{t \in [0;1[, |f(t)| = 1\}.$$

Problem (Dual max-min) can therefore be rewritten as

$$\boxed{\begin{aligned} &\text{maximize } \operatorname{Re}\langle z, y\rangle \\ &\qquad \text{for } z \in \mathbb{C}^{2N+1} \\ &\text{such that } \left|\sum_{k=-N}^{N} z_k e^{2\pi i k t}\right| \leq 1, \forall t \in \mathbb{R}. \end{aligned}} \tag{Dual TV}$$

By construction,

$$\begin{aligned} \min \text{ (Min TV)} &= \min_{\mu\in\mathcal{M}([0;1[)} f_1(\mu) \\ &= \min_{\mu\in\mathcal{M}([0;1[)} \max_{z\in\mathbb{C}^{2N+1}} F(\mu, z) \\ &\geq \min_{\mu\in\mathcal{M}([0;1[)} \max_{z\in\mathbb{C}^{2N+1}} \min_{\nu\in\mathcal{M}([0;1[)} F(\nu, z) \\ &= \max_{z\in\mathbb{C}^{2N+1}} \min_{\nu\in\mathcal{M}([0;1[)} F(\nu, z) \\ &= \max_{z\in\mathbb{C}^{2N+1}} f_2(z) \\ &= \max \text{ (Dual TV)}, \end{aligned}$$

that is, the optimal value of (Dual TV) is necessarily smaller than the optimal value of (Min TV). Actually, the two values are equal, as stated in the following theorem. This is the *strong duality* property (Equation (2.19) in the previous paragraph), which will be our crucial tool to establish tightness of the convex relaxation (Min TV).

Theorem 2.17 : strong duality for TV minimization

Problem (Min TV) has at least one minimizer μ_*, and Problem (Dual TV) has at least one maximizer z_*.
The two problems have the same optimal value :

$$||\mu_*||_{TV} = \operatorname{Re}\langle z_*, y\rangle .$$

This theorem admits an elementary, but tricky proof, given in Section A.3 of the appendix.

It states that the primal and dual problems, (Min TV) and (Dual TV) are equivalent in the sense that they have the same optimal value. It turns out that, in addition, the solutions of one problem can be partly characterized from the solutions of the other one. This is the content of the following proposition, which we will need in the next paragraph and whose proof is in Section A.4.

Proposition 2.18

Let μ_* be a minimizer of Problem (Min TV) and z_* a maximizer of Problem (Dual TV). It holds :

$$\mathrm{Supp}(\mu_*) \subset \left\{ t \in [0;1[, \left| \sum_{k=-N}^{N} z_{*k} e^{2\pi i k t} \right| = 1 \right\}.$$

Correctness guarantees

Let us summarize what we have said so far in this section. We want to recover a measure μ_0, which is a sum of S diracs ($\mu_0 = \sum_{s=1}^{S} a_s \delta_{\tau_s}$), from its Fourier coefficients. This is the non-convex problem (Super-resolution). In Subsection 2.3.1, we have introduced the convex relaxation (Min TV). Our objective is now to prove that, at least under some suitable assumptions on μ_0, the relaxation is tight : Problem (Min TV) (with $y = \hat{\mu}_0[-N : N]$) has a single solution, which is μ_0.

To prove that μ_0 is the solution of Problem (Min TV), we use the dual problem (Dual TV). More specifically, we construct a so-called *dual certificate* : a feasible point z for Problem (Dual TV) satisfying

$$||\mu_0||_{TV} = \mathrm{Re}\,\langle z, y \rangle . \tag{2.23}$$

The existence of a dual certificate proves that μ_0 is a minimizer of Problem (Min TV) : indeed,

$$\min \text{ (Min TV)} \leq ||\mu_0||_{TV} = \mathrm{Re}\,\langle z, y \rangle \leq \max \text{ (Dual TV)}$$

and, from Theorem 2.17, min (Min TV) = max (Dual TV), so the two inequalities above are actually equalities. In particular,

$$\min \text{ (Min TV)} = ||\mu_0||_{TV},$$

so μ_0 is a minimizer of (Min TV). If, in addition, the dual certificate satisfies

$$\left| \sum_{k=-N}^{N} z_k e^{2\pi i k t} \right| < 1 \text{ for all } t \in [0;1[\backslash \mathrm{Supp}(\mu_0), \tag{2.24}$$

then we can add that μ_0 is the *only* minimizer of (Min TV). Indeed, any minimizer μ_* must then satisfy, from Proposition 2.18,

$$\mathrm{Supp}(\mu_*) \subset \left\{ t \in [0;1[, \left| \sum_{k=-N}^{N} z_k e^{2\pi ikt} \right| = 1 \right\} \subset \mathrm{Supp}(\mu_0) = \{\tau_1, \dots, \tau_S\}.$$

Consequently, μ_* is of the form $\mu_* = \sum_{s=1}^S a_{*s}\delta_{\tau_s}$, for some coefficients $a_{*1}, \dots, a_{*S} \in \mathbb{C}$. It can be checked that the map

$$\begin{array}{cccc} L: & \mathbb{C}^S & \to & \mathbb{C}^{2N+1} \\ & (x_1, \dots, x_S) & \to & \left(\widehat{\sum_{s=1}^S x_s \delta_{\tau_s}}[k] \right)_{-N\le k\le N} \end{array}$$

is injective if $S \le 2N+1$[7]. As $L(a_1, \dots, a_S) = \hat{\mu}_0[-N:N] = y = \hat{\mu}_*[-N:N] = L(a_{*1}, \dots, a_{*S})$, it means that $a_{*s} = a_s$ for any s.

Under which conditions does there exist a dual certificate ? The following theorem states that, when μ_0 is nonnegative, it always exists. In particular, in this case, μ_0 is the only solution of the convex relaxation (Min TV) ; the relaxation is tight.

Theorem 2.19 : tightness for nonnegative measures [de Castro and Gamboa, 2012]

Let $\mu_0 = \sum_{s=1}^S a_s\delta_{\tau_s}$ be a nonnegative measure (that is, $a_s \in \mathbb{R}^+$ for all $s \le S$).
If $S \le N$, there exists a dual certificate z as in Equation (2.23), satisfying the additional condition (2.24).

Démonstration. In this proof, for any vector $z \in \mathbb{C}^{2N+1}$, we denote P_z the associated trigonometric polynomial

$$P_z(e^{2\pi it}) = \sum_{k=-N}^{N} z_k e^{2\pi ikt}.$$

We recall $||\mu_0||_{TV} = \sum_{s=1}^S |a_s|$ and, from the same computation as the one following Proposition 2.15, for any z,

$$\mathrm{Re}\,\langle z, y\rangle = \mathrm{Re}\,\langle z, \hat{\mu}_0[-N:N]\rangle$$

[7] Its matrix, in a canonical basis, is a so-called *Vandermonde matrix*, whose determinant has a simple explicit expression, and cannot be zero if the τ_s are distinct.

$$\begin{aligned}
&= \operatorname{Re} \int_0^1 \left(\sum_{k=-N}^{N} \overline{z_k} e^{-2\pi i k t} \right) d\mu_0(t) \\
&= \sum_{s=1}^{S} a_s \left(\sum_{k=-N}^{N} \overline{z_k} e^{-2\pi i k \tau_s} \right) \\
&= \sum_{s=1}^{S} a_s \overline{P_z(e^{2\pi i \tau_s})}.
\end{aligned}$$

Equation (2.23) can be rewritten as

$$\sum_{s=1}^{S} |a_s| = \sum_{s=1}^{S} a_s \overline{P_z(e^{2\pi i \tau_s})}.$$

Since the a_s are nonnegative, this equality notably holds if

$$P_z(e^{2\pi i \tau_s}) = 1, \quad \forall s = 1, \dots, S. \tag{2.25}$$

To find a dual certificate satisfying Equation (2.24), we must therefore only find $z \in \mathbb{C}^{2N+1}$ satisfying Equation (2.25) such that

$$\left| P_z(e^{2\pi i t}) \right| < 1, \quad \forall t \in [0; 1[\backslash \{\tau_1, \dots, \tau_S\}. \tag{2.26}$$

(Note that a vector satisfying these two conditions is automatically a feasible point of Problem (Dual TV).)

Let $\epsilon > 0$ be a small constant (to be chosen later). We define a trigonometric polynomial

$$\begin{aligned}
Q^\epsilon(e^{2\pi i t}) &= 1 - \epsilon \prod_{s=1}^{S} \left| e^{2\pi i t} - e^{2\pi i \tau_s} \right|^2 \\
&= 1 - \epsilon \prod_{s=1}^{S} \left(e^{2\pi i t} - e^{2\pi i \tau_s} \right) \left(e^{-2\pi i t} - e^{-2\pi i \tau_s} \right).
\end{aligned}$$

If ϵ is small enough, we have for all $t \in [0; 1[$

$$Q^\epsilon(e^{2\pi i t}) = 1 - \epsilon \prod_{s=1}^{S} \left| e^{2\pi i t} - e^{2\pi i \tau_s} \right|^2 \in [0; 1].$$

We fix such an ϵ and define $z \in \mathbb{C}^{2N+1}$ such that

$$Q^\epsilon = P_z.$$

If $S \leq N$, such a vector z exists. It satisfies the desired conditions : Equation (2.25) is true because, for any $s \leq S$,

$$P_s(e^{2\pi i \tau_s}) = Q^\epsilon(e^{2\pi i \tau_s}) = 1 - \epsilon \prod_{s'=1}^{S} \left| e^{2\pi i \tau_s} - e^{2\pi i \tau_{s'}} \right|^2 = 1.$$

Equation (2.26) is also true because, from the choice of ϵ, $P_z(e^{2\pi i t})$ is in $[0;1]$ for all $t \in [0;1[$. It is exactly equal to 1 if and only if $\prod_{s=1}^{S} \left| e^{2\pi i t} - e^{2\pi i \tau_s} \right|^2 = 0$, that is if and only if

$$t \in \{\tau_1, \dots, \tau_S\}.$$

Said otherwise, $|P_z(e^{2\pi i t})| < 1$ for all $t \in [0;1[\setminus\{\tau_1, \dots, \tau_S\}$. □

And when μ_0 is not nonnegative ? In this case, a dual certificate also exists, provided that the diracs in μ_0 are sufficiently well *separated*. Separation is defined as the minimal distance between any two τ_s, where the distance is considered[8] modulo 1 : we define

$$\Delta(\mu_0) = \min_{s \neq s'} \mathrm{dist}(\tau_s, \tau_{s'}),$$
$$\text{where } \mathrm{dist}(\tau_s, \tau_{s'}) = \min_{n \in \mathbb{Z}} |\tau_s - \tau_{s'} - n|.$$

Theorem 2.20 : tightness for well-separated diracs [Candès and Fernandez-Granda, 2014]

Let $N \in \mathbb{N}^*$ be fixed and large enough[a]. If μ_0 is a measure with separation

$$\Delta(\mu_0) \geq \frac{2}{N},$$

then a dual certificate satisfying the additional condition (2.24) exists. As a consequence, the convex relaxation (Min TV) is tight.

[a] larger than 128

[8] It is considered modulo 1 because the Fourier transform is 1-periodic : for any τ, the dirac δ_τ has the same Fourier coefficients as $\delta_{\tau+1}, \delta_{\tau+2}, \dots$

The proof of this theorem follows a similar methodology as the proof of Theorem 2.19, but the construction of the dual certificate is significantly more difficult. We do not present it here.

Chapitre 3

Non-convex methods

In the previous chapter, we have presented algorithms relying on convexification techniques and seen that these algorithms

- work well, in the sense that, at least for specific classes of random inverse problems, they succeed with high probability ;
- can be rigorously analyzed, at least in some settings.

These are two of the three properties in our "wishlist" of Subsection 1.1.3. Unfortunately, there is a third property in this wishlist : good algorithms must be reasonably fast. And this property is often not satisfied by convex algorithms.

This is especially true for low-rank matrix recovery problems. A low-rank matrix with dimension $d_1 \times d_2$ and rank r can be parameterized by $r(d_1+d_2)$ parameters : we can write it as

$$X = LR,$$

for some matrices $L \in \mathbb{R}^{d_1\times r}$ and $R \in \mathbb{R}^{r\times d_2}$.[1] We could naively expect that there exist algorithms which only explore the set of matrices with rank

[1] If we write the singular value decomposition $X = U \begin{pmatrix} \mu_1 & & \\ & \ddots & \\ & & \mu_{d_2} \\ 0 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & 0 \end{pmatrix} V$, the matrices $L = U \begin{pmatrix} \sqrt{\mu_1} & & \\ & \ddots & \\ & & \sqrt{\mu_r} \\ 0 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & 0 \end{pmatrix}$ and $R = \begin{pmatrix} \sqrt{\mu_1} & & & 0 & \dots & 0 \\ & \ddots & & \vdots & & \vdots \\ & & \sqrt{\mu_r} & 0 & \dots & 0 \end{pmatrix} V$ satisfy the equality $X = LR$.

r, using this representation with $r(d_1 + d_2)$ parameters, for a cost of order (ideally) $O(r(d_1 + d_2))$ elementary operations at each iteration. But Problem (Nuclear min) is an optimization problem on the whole set of $d_1 \times d_2$ matrices. It does not take advantage of the fact that the underlying matrix we want to recover can be parameterized by $r(d_1 + d_2)$ parameters. Each iteration of the solver typically requires at least $d_1 d_2$ operations (because one needs $d_1 d_2$ operations to simply read each entry of a $d_1 \times d_2$ matrix), and possibly much more.[2] As r is much smaller than $\min(d_1, d_2)$,

$$d_1 d_2 \gg r(d_1 + d_2).$$

For this reason, convexification techniques are oftentimes considered too costly, and other algorithms are preferred, where iterates are low-rank. These algorithms are called *non-convex* because they (explicitely or implicitely) operate on a non-convex set, and do not attempt to introduce hidden or approximate convexity.

The theoretical foundation of non-convex algorithms is generally not as clear as for convex ones. They can contain various heuristic steps, tailored to the problem at hand. Regarding their results, they are not guaranteed to return a meaningful vector at all. In particular, because they optimize over a non-convex set, they can a priori get stuck in local optima and there is no good strategy to tell in advance whether this will happen or not and, if yes, how to avoid it. However, they have been used for a long time and numerical results have shown that they work well in many situations. This has motivated researchers, in the last decade, to develop analysis techniques suited to non-convex algorithms.

3.1 General non-convex optimization

We start with an overview of general non-convex optimization algorithms, to understand what we can expect of the output of a non-convex algorithm : it is generally not guaranteed to find the global optimum of the problem, but it will in principle find at least a *critical point*.

The next subsection defines two versions of this notion : *first-order* and *second-order critical points*. The subsequent two subsections describe the

[2] Some operations on $d_1 \times d_2$ matrices require significantly more than $O(d_1 d_2)$ operations, like singular value decomposition.

convergence guarantees of standard optimization algorithms towards, respectively, first and second-order critical points.

3.1.1 Critical points versus minimizers

To simplify the discussion, let us restrict ourselves to finite-dimensional and *unconstrained* optimization. "Unconstrained" means that, in Problem (Opt), the number of constraint sets C_s is zero :

$$\text{minimize } f(x) \text{ over all } x \in \mathbb{R}^d.$$

We assume that a minimizer exists.

We recall from Section 1.2 that it is in general hopeless to try to find a global minimizer of f if f is not convex : even assuming that f is smooth, this would require to query information on f at all points of a fine grid of $\mathbb{R}^d$ (at least a bounded subset thereof) which is already slow for very small values of d, and quickly becomes unrealistic when d grows.

Thus, what can we expect from a good non-convex optimization algorithm ? It won't be able to find global minimizers with certainty. Can it at least be guaranteed to find a *local* minimizer, if one exists ? It turns out that this is also out of reach : there are functions, even polynomial ones, for which determining whether a point is a local minimum is already NP-difficult. To describe what reasonable non-convex algorithms should output, the good notion is *critical points*.

Critical points are points at which "the derivatives of f satisfy the same properties as at a local minimizer".

Definition 3.1 : critical point

We say that an element x of $\mathbb{R}^d$ is

— a *first-order critical point* of f if $\nabla f(x) = 0$,

— a *second-order critical point* of f if $\nabla f(x) = 0$ and $\text{Hess}\, f(x) \succeq 0$.

Of course, the first notion is well-defined only for differentiable functions f, and the second one only for twice-differentiable ones.

Remark

Local minimizers of f are necessarily second-order critical points, but the converse may not be true. For instance, the map $x \in \mathbb{R} \to x^3 \in \mathbb{R}$ has a second-order critical point at 0, but no local minimizer.

Example 3.2

Let us consider the map

$$\begin{array}{cccc} f: & \mathbb{R}^2 & \to & \mathbb{R} \\ & (x,y) & \to & \frac{x^4}{4} - \frac{x^3}{3} - x^2 + y^2. \end{array}$$

For any $(x,y) \in \mathbb{R}^2$,

$$\nabla f(x,y) = \begin{pmatrix} (x+1)x(x-2) \\ 2y \end{pmatrix}, \quad \mathrm{Hess} f(x,y) = \begin{pmatrix} 3x^2 - 2x - 2 & 0 \\ 0 & 2 \end{pmatrix}.$$

From these expressions, one can check that f has three first-order critical points, which are $(-1,0), (0,0)$ and $(2,0)$.
Among them, only $(-1,0)$ and $(2,0)$ are second-order critical points. Since

$$\mathrm{Hess} f(-1,0) = \begin{pmatrix} 3 & 0 \\ 0 & 2 \end{pmatrix} \succ 0 \quad \text{and} \quad \mathrm{Hess} f(2,0) = \begin{pmatrix} 6 & 0 \\ 0 & 2 \end{pmatrix} \succ 0,$$

both are local minimizers of f. The point $(2,0)$ is a global minimizer while $(-1,0)$ is only a local one.

At this point, the reader may wonder : why bother proving that a given non-convex algorithm always outputs a critical point of the objective function ? What we really want are minimizers of f, not critical points ! For us, the main reason is that knowing that an algorithm returns a critical point for sure is a first step towards analyzing its behavior. Indeed, it allows us to restrict our analysis of possible outputs to the set of critical points. In particular, if the objective function has only a few critical points, it already provides a lot of information on the output.

3.1.2 Finding first-order critical points

Assuming f is differentiable, the most basic optimization algorithm is gradient descent. It defines a sequence of iterates $(x_t)_{t\in\mathbb{N}}$ by

$$x_{t+1} = x_t - \alpha_t \nabla f(x_t), \quad \forall t \in \mathbb{N}.$$

Here, the parameters $\alpha_t > 0$ are called the *stepsizes*.

In this subsection, we are going to see the following results :

- under very weak hypotheses, x_t is an approximate first-order critical point for t large enough (Corollary 3.4) ;
- under slightly stricter (but still weak) hypotheses, x_t actually converges to a first-order critical point when $t \to +\infty$ (Theorem 3.6).

We first need a proposition about the decay of f along the gradient descent trajectory.

Proposition 3.3

We assume that the gradient of f is L-Lipschitz [a] for some $L > 0$: for any $x, y \in \mathbb{R}^d$,

$$||\nabla f(x) - \nabla f(y)||_2 \leq L||x - y||_2.$$

We consider gradient descent with stepsize $\alpha_t = \frac{1}{L}$. [b]
Then, for each $t \in \mathbb{N}$,

$$f(x_{t+1}) \leq f(x_t) - \frac{1}{2L}||\nabla f(x_t)||_2^2.$$

[a] This assumption is often called *L-smoothness*.

[b] Other choices are possible. In practice, L is usually unknown and the stepsizes are chosen using linesearch.

Démonstration. For all $x, h \in \mathbb{R}^d$,

$$\begin{aligned} f(x+h) &= f(x) + \int_0^1 \langle \nabla f(x+th), h \rangle \, dt \\ &= f(x) + \int_0^1 \langle \nabla f(x) + (\nabla f(x+th) - \nabla f(x)), h \rangle \, dt \end{aligned}$$

$$
\begin{aligned}
&= f(x) + \langle \nabla f(x), h\rangle + \int_0^1 \langle \nabla f(x+th) - \nabla f(x), h\rangle \, dt \\
&\leq f(x) + \langle \nabla f(x), h\rangle + \int_0^1 ||\nabla f(x+th) - \nabla f(x)||_2 \, ||h||_2 dt \\
&\qquad\qquad\qquad \text{(by triangular inequality)} \\
&\leq f(x) + \langle \nabla f(x), h\rangle + L \int_0^1 ||h||_2^2 t dt \\
&\qquad\qquad\qquad \text{(as } \nabla f \text{ is } L\text{-Lipschitz)} \\
&= f(x) + \langle \nabla f(x), h\rangle + \frac{L}{2}||h||_2^2.
\end{aligned}
$$

We apply this inequality to $x = x_t$ and $h = -\frac{1}{L}\nabla f(x_t)$:

$$\forall t \in \mathbb{N}, \quad f(x_{t+1}) \leq f(x_t) - \frac{1}{2L}||\nabla f(x_t)||_2^2.$$

□

This property implies that the gradient descent iterates are "asymptotically first-order critical", in the sense that ∇f goes to zero along the sequence.

Corollary 3.4

Under the same assumptions as Proposition 3.3, and recalling that we assume the existence of at least one minimizer of f,

$$||\nabla f(x_t)||_2 \overset{t\to+\infty}{\longrightarrow} 0.$$

Démonstration. For any $T \in \mathbb{N}$, from Proposition 3.3,

$$
\begin{aligned}
\frac{1}{2L}\sum_{t=0}^{T} ||\nabla f(x_t)||_2^2 &\leq \sum_{t=0}^{T} [f(x_t) - f(x_{t+1})] \\
&= f(x_0) - f(x_{T+1}) \\
&\leq f(x_0) - \min f.
\end{aligned}
$$

Consequently, the sum $\sum_{t\in\mathbb{N}} ||\nabla f(x_t)||_2^2$ is convergent, so its terms go to zero. □

Refining the argument, we can moreover give an a priori estimate for the convergence rate of $||\nabla f(x_t)||_2$ towards zero.

Corollary 3.5

We keep the same assumptions as in Corollary 3.4
For any T, if we set $\tilde{x}_T = \text{argmin}\,\{||\nabla f(x)||_2, x \in \{x_0, \dots, x_T\}\}$, this point satisfies

$$||\nabla f(\tilde{x}_T)||_2 \leq \sqrt{\frac{2L(f(x_0) - \min f)}{T+1}}.$$

Démonstration. We have seen in the proof of Corollary 3.4 that, for any T,

$$\sum_{t=0}^{T} ||\nabla f(x_t)||_2^2 \leq 2L(f(x_0) - \min f).$$

Since $||\nabla f(\tilde{x}_T)||_2 \leq ||\nabla f(x_t)||_2$ for any $t \leq T$,

$$(T+1)||\nabla f(\tilde{x}_T)||_2^2 \leq 2L(f(x_0) - \min f),$$

which implies

$$||\nabla f(\tilde{x}_T)||_2 \leq \sqrt{\frac{2L(f(x_0) - \min f)}{T+1}}.$$

□

Another way of stating the above result is that, for fixed Lipschitz constant L and gap $(f(x_0) - \min(f))$, gradient descent needs at most $O\left(\frac{1}{\epsilon^2}\right)$ iterations to find a ϵ-approximate first-order critical point. Let us mention that this convergence rate is optimal : for any algorithm and any ϵ, there is at least one function with the given Lipschitz constant and gap such that the algorithm needs to query at least $O\left(\frac{1}{\epsilon^2}\right)$ values of the function or its derivatives to find an ϵ-approximate first-order critical point [Carmon, Duchi, Hinder, and Sidford, 2020].

We have seen that gradient descent iterates are asymptotically first-order critical. At this stage, a natural question is : do the iterates actually converge a first-order critical point ? For most functions f, the answer is yes. However, there exist a few functions with Lipschitz gradient for which this is not true, [3] so we need additional assumptions to guarantee it.

[3] The iterates may go to infinity if the function is not coercive, or cycle around a large set of critical points.

Theorem 3.6 : convergence of gradient descent iterates

We still assume that the gradient of f is L-Lipschitz, for some $L > 0$. In addition, we make either of the following two assumptions :

— f is coercive [a] and the set of its first-order critical points is discrete ; [b]

— f is coercive and analytic. [c]

We still consider gradient descent with stepsize $\frac{1}{L}$.
The sequence of iterates $(x_t)_{t\in\mathbb{N}}$ converges towards a first-order critical point of f.

[a] A function f is *coercive* if $f(x) \to +\infty$ when $||x||_2 \to +\infty$.

[b] A set E is discrete if, for all $x \in E$, there exists $\epsilon > 0$ such that $E \cap B(x,\epsilon) = \{x\}$.

[c] A function is analytic if it is C^∞ and agrees with its Taylor series in a neighborhood of every point.

Démonstration. We only prove the result for the first assumption. For the second one, the reader is referred to [Absil, Mahony, and Andrews, 2005, Thm 3.2].

From Proposition 3.3, the iterates satisfy

$$f(x_t) \leq f(x_0), \quad \forall t \in \mathbb{N}.$$

We define $A = \left\{x \in \mathbb{R}^d, f(x) \leq f(x_0)\right\}$. It is a closed and bounded set, which contains all points x_t. Consequently, $(x_t)_{t\in\mathbb{N}}$ is bounded, hence has at least one accumulation point.

From Corollary 3.4, and because ∇f is continuous, all accumulation points are first-order critical.

Let $x_{c,1}, \dots, x_{c,S}$ be the first-order critical points in A. There is only a finite number of them because the set of first-order critical points is discrete, hence has finite intersection with every bounded set.

Let us fix

$$\epsilon < \frac{1}{3} \min_{s\neq s'} ||x_{c,s} - x_{c,s'}||_2,$$

$$\mu \overset{def}{=} \min_{x\in A\setminus\left(\bigcup_{s\leq S} B(x_{c,s},\epsilon)\right)} ||\nabla f(x)||_2.$$

We observe that $\mu > 0$; otherwise, f would have a first-order critical point in A, different from all $x_{c,s}$, contradicting the fact that $x_{c,1}, \dots, c_{c,S}$ are *all* critical points of f in A.

From Corollary 3.4, for t large enough, $||\nabla f(x_t)||_2 < \mu$, hence

$$x_t \in \bigcup_{s \leq S} B(x_{c,s}, \epsilon).$$

Also for t large enough, $||x_{t+1} - x_t||_2 = \frac{1}{L}||\nabla f(x_t)||_2 < \epsilon$. Because all balls $B(x_{c,s}, \epsilon)$ are at distance at least ϵ one from each other (from the definition of ϵ), it is impossible that

$$x_t \in B(x_{c,s}, \epsilon) \quad \text{and} \quad x_{t+1} \in B(x_{c,s'}, \epsilon) \text{ for } s' \neq s.$$

Therefore, for t large enough, all iterates belong to the *same* ball $B(x_{c,s}, \epsilon)$. Let s be the index of this ball. All accumulation points of $(x_t)_{t\in\mathbb{N}}$ are first-order critical and the only first-order critical point in $\overline{B(x_{c,s}, \epsilon)}$ is $x_{c,s}$, so $(x_t)_{t\in\mathbb{N}}$ is a bounded sequence with a single accumulation point, which is $x_{c,s}$. Therefore,

$$x_t \overset{t\to+\infty}{\longrightarrow} x_{c,s}.$$

□

Remark

If f satisfies all assumptions of the previous theorem, except that its gradient is not Lipschitz, the conclusion still holds true. However, the stepsize of gradient descent cannot be chosen as $\frac{1}{L}$ (the Lipschitz constant L is not defined). It must be chosen by linesearch.

3.1.3 Finding second-order critical points

In this subsection, we assume that f is twice-differentiable over $\mathbb{R}^d$.

Second-order algorithms

Since the definition of second-order critical points involves the Hessian of f, it seems reasonable that using Hess$f$ during the optimization procedure might help to find a second-order critical point. Such algorithms, which use second-order derivatives, are called *second-order algorithms*. In this paragraph, we present a simplified version of one of them, called *Trust-Region*.

The starting point of this algorithm is that for any $x \in \mathbb{R}^d$,

$$f(x+h) = f(x) + \langle h, \nabla f(x) \rangle + \frac{1}{2} \langle h, \mathrm{Hess} f(x) h \rangle + o(||h||^2). \tag{3.1}$$

In view of this equation, one might be tempted to define the iterates $(x_t)_{t\in\mathbb{N}}$ using a recurrence relation $x_{t+1} = x_t + h_t$, where

$$h_t \in \mathrm{argmin}_{h\in\mathbb{R}^d} \left(f(x_t) + \langle h, \nabla f(x_t) \rangle + \frac{1}{2} \langle h, \mathrm{Hess} f(x_t) h \rangle \right).$$

Unfortunately, this definition makes no sense : when Hess$f$ is not semidefinite positive, the above function is not lower bounded, hence has no minimizer. Even if a minimizer existed, it would only be a sensible choice for x_{t+1} if he belonged to the neighborhood of x_t on which Approximation (3.1) is valid. Therefore, it is best to refine the previous definition as

$$x_{t+1} = x_t + h_t, \tag{3.2a}$$
$$h_t = \underset{||h||\leq R_t}{\mathrm{argmin}} \left(f(x_t) + \langle h, \nabla f(x_t) \rangle + \frac{1}{2} \langle h, \mathrm{Hess}\, f(x_t) h \rangle \right). \tag{3.2b}$$
$$\tag{3.2c}$$

In this definition, R_t is a positive number, the *trust radius*, which is an estimation of the size of the region over which Equation (3.1) provides a good approximation of f.

Theorem 3.7 : convergence of the trust-region method

Let $\epsilon > 0$ be fixed.
We assume that f has at least one minimizer x_* and that Hess f is L_2-Lipschitz for some $L_2 > 0$:

$$\forall x, y, h \in \mathbb{R}^n, \quad ||(\mathrm{Hess}\, f(x) - \mathrm{Hess}\, f(y))h||_2 \leq L_2 ||x-y||_2\, ||h||_2.$$

Let $(x_t)_{t\in\mathbb{N}}$ be defined as in Equations (3.2b) and (3.2a), with $R_t = \frac{\sqrt{\epsilon}}{2L_2}$ for any t.
For any $x_0 \in \mathbb{R}^n$, the algorithm finds an ϵ-approximate second-order critical point in at most $O\left(\frac{L_2^2(f(x_0)-f(x_*))}{\epsilon^{3/2}}\right)$ iterations. More precisely, there exists

$$t \leq c \frac{L_2^2(f(x_0) - f(x_*))}{\epsilon^{3/2}}$$

(for some explicit constant $c > 0$) such that

$$||\nabla f(x_t)||_2 \leq \frac{\epsilon}{L_2} \quad \text{and} \quad \text{Hess} f(x_t) + \sqrt{\epsilon} I_d \succeq 0.$$

Sketch of proof, based on [Ye, 2015]. We admit the following statement : for each t, there exists $\sigma_t \geq 0$ such that

$$(\text{Hess} f(x_t) + \sigma_t I_d)\, h_t = -\nabla f(x_t) \quad \text{and} \quad \text{Hess} f(x_t) + \sigma_t I_d \succeq 0.$$

In addition, if $\sigma_t > 0$, then $||h_t||_2 = R_t$.

We first show that there exists $t \leq \frac{12 L_2^2 (f(x_0) - f(x_*))}{\epsilon^{3/2}} + 1$ such that

$$\sigma_t \leq \frac{\sqrt{\epsilon}}{2}.$$

By contradiction, let us assume that it is not true. Because the Hessian is L_2-Lipschitz, for all $t \leq \frac{12 L_2^2 (f(x_0) - f(x_*))}{\epsilon^{3/2}} + 1$,

$$\begin{aligned}
f(x_{t+1}) &= f(x_t + h_t) \\
&\leq f(x_t) + \langle h_t, \nabla f(x_t) \rangle + \frac{1}{2} \langle h_t, \text{Hess}\, f(x_t) h_t \rangle + \frac{L_2}{6} ||h_t||_2^3 \\
&= f(x_t) - \langle h_t, \text{Hess} f(x_t) h_t + \sigma_t h_t \rangle + \frac{1}{2} \langle h_t, \text{Hess}\, f(x_t) h_t \rangle + \frac{L_2}{6} ||h_t||_2^3 \\
&= f(x_t) - \frac{1}{2} \langle h_t, (\text{Hess} f(x_t) + \sigma_t I_d)\, h_t \rangle - \frac{\sigma_t}{2} R_t^2 + \frac{L_2}{6} R_t^3 \\
&\qquad\qquad (||h_t||_2 = R_t \text{ since } \sigma_t > 0) \\
&\leq f(x_t) - \frac{\sigma_t}{2} R_t^2 + \frac{L_2}{6} R_t^3 \\
&\qquad\qquad (\text{as Hess} f(x_t) + \sigma_t I_d \succeq 0), \\
&< f(x_t) - \frac{\sqrt{\epsilon}}{4} R_t^2 + \frac{L_2}{6} R_t^3 \\
&= f(x_t) - \frac{\epsilon^{3/2}}{12 L_2^2}.
\end{aligned}$$

Therefore, for any $t \leq \frac{12 L_2^2 (f(x_0) - f(x_*))}{\epsilon^{3/2}} + 1$,

$$f(x_0) - f(x_*) \geq f(x_0) - f(x_{t+1})$$

$$> \frac{\epsilon^{3/2}}{12L_2^2}t,$$

which cannot be true for $t = \left\lceil \frac{12L_2^2(f(x_0)-f(x_*))}{\epsilon^{3/2}} \right\rceil$. This contradiction concludes the first part of the proof.

Let $t \leq \frac{12L_2^2(f(x_0)-f(x_*))}{\epsilon^{3/2}} + 1$ be such that $\sigma_t \leq \frac{\sqrt{\epsilon}}{2}$. We show that x_{t+1} is an approximate second-order critical point. First, it holds

$$\begin{aligned}
\text{Hess}f(x_{t+1}) &= \text{Hess}f(x_t + h_t) \\
&\succeq \text{Hess}f(x_t) - L_2||h_t||I_d \\
&= \text{Hess}f(x_t) + \sigma_t I_d - \sigma_t I_d - L_2||h_t||I_d \\
&\succeq -\sigma_t I_d - L_2||h_t||I_d \\
&\succeq -\sqrt{\epsilon} I_d.
\end{aligned}$$

Second, $||\nabla f(x_{t+1}) - \nabla f(x_t) - \text{Hess}f(x_t)h_t||_2 \leq \frac{L_2}{2}||h_t||_2^2$, hence

$$\begin{aligned}
||\nabla f(x_{t+1})||_2 &\leq ||\nabla f(x_t) + \text{Hess}f(x_t)h_t||_2 + \frac{L_2}{2}||h_t||_2^2 \\
&= ||-\sigma_t h_t||_2 + \frac{L_2}{2}||h_t||_2^2 \\
&\leq \frac{\sqrt{\epsilon}}{2}R_t + \frac{L_2}{2}R_t^2 \\
&= \frac{3\epsilon}{8L_2}.
\end{aligned}$$

□

Gradient descent, again

We have seen in Subsection 3.1.2 that, under mild assumptions on f, gradient descent, starting at any point $x_0 \in \mathbb{R}^d$, allows to find an approximate first-order critical point. The same is not true for second-order critical points. For instance, if x_0 is a first-order critical point of f, but not a second-order critical point, then

$$x_0 = x_1 = x_2 = \dots,$$

because $\nabla f(x_0) = 0$, hence gradient descent stays stuck at x_0 and never gets close to a second-order critical point.

Nevertheless, this phenomenon is very rare : for "general" initializations, it does not happen, and gradient descent converges to a second-order critical point.

Theorem 3.8 : [Lee, Simchowitz, Jordan, and Recht, 2016]

We still assume that ∇f is L-Lipschitz, for some $L > 0$. In addition, we assume that

— f has only a finite number of first-order critical points [a] ;

— f is coercive.

We consider gradient descent with constant stepsize $\alpha \in]0; \frac{1}{L}[$.
For almost any x_0, [b] $(x_t)_{t\in\mathbb{N}}$ converges to a second-order critical point.

[a] This hypothesis can be relaxed.

[b] that is, for all x_0 outside a zero-Lebesgue measure set

Remark

The theorem is still true without the Lipschitz assumption on the gradient, if we replace "with constant stepsize $\alpha \in]0; \frac{1}{L}[$" with "for a small enough stepsize α, possibly depending on x_0" or "when the stepsize is chosen by a standard linesearch procedure".

Intuition of proof. Theorem 3.6 shows that the gradient descent iterates $(x_t)_{t\in\mathbb{N}}$ converge to a first-order critical point whatever x_0.

We must show that, if x_{crit} is a first-order but not a second-order critical point of f, then $(x_t)_{t\in\mathbb{N}}$ does not converge to x_{crit}, for almost any x_0. We consider such a critical point ; up to translation, we can assume that it is 0.

We make the (very) simplifying hypothesis that f is quadratic in a ball centered at 0, with radius r_0 :

$$\forall x \in B(0, r_0), \quad f(x) = \frac{1}{2} \langle x, Mx \rangle + \langle x, b \rangle ,$$

for some $n \times n$ symmetric matrix M.

For any $x \in B(0, r_0)$, $\nabla f(x) = Mx + b$. Since 0 is a first-order critical point, we necessarily have $b = 0$. In addition, Hess $f(x) = M$ for any $x \in B(0, r_0)$. The assumption that 0 is not a second-order critical point is then equivalent to the fact that $M \not\succeq 0$.

The matrix M can be diagonalized in an orthonormal basis :

$$M = U^T \begin{pmatrix} \lambda_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \lambda_d \end{pmatrix} U,$$

with $\lambda_1 \geq \dots \geq \lambda_d$ the eigenvalues of M and U an orthonomal matrix. Up to a change of coordinates, we can assume $U = \mathrm{Id}$. Since $M \not\succeq 0$, at least the smallest eigenvalue of M is negative : $\lambda_d < 0$.

If the sequence $(x_t)_{t\in\mathbb{N}}$ of gradient descent iterates converges to $x_{crit} = 0$, then x_t belongs to $B(0, r_0)$ for any t large enough, in which case

$$\begin{aligned} x_{t+1} &= x_t - \alpha \nabla f(x_t) \\ &= x_t - \alpha M x_t \\ &= \begin{pmatrix} (1-\alpha\lambda_1)x_{t,1} \\ \vdots \\ (1-\alpha\lambda_d)x_{t,d} \end{pmatrix}. \end{aligned}$$

We fix t_0 such that this relation holds for any $t \geq t_0$. Then, for any $s \in \mathbb{N}$,

$$x_{t_0+s} = \begin{pmatrix} (1-\alpha\lambda_1)^s x_{t_0,1} \\ \vdots \\ (1-\alpha\lambda_d)^s x_{t_0,d} \end{pmatrix}.$$

If the sequence converges to 0, all the coordinates of x_{t_0+s} must go to 0 when s goes to $+\infty$ (for any fixed t), which means that

$$\forall k \in \{1, \dots, d\}, \quad (1-\alpha\lambda_k)^s x_{t_0,k} \overset{s\to+\infty}{\to} 0. \tag{3.3}$$

We have said that $\lambda_d < 0$, hence $1 < 1 - \alpha\lambda_d$ and $(1-\alpha\lambda_d)^s \not\to 0$ when $s \to +\infty$. In order for Property (3.3) to hold, we must therefore have

$$x_{t_0,d} = 0.$$

To summarize, we have shown that, if $(x_t)_{t\in\mathbb{N}}$ converges to 0, then, for some t_0,

$$x_{t_0} \in \mathcal{E} \overset{def}{=} \{z \in B(0, r_0) \text{ such that } z_d = 0\}.$$

As a consequence,

$$x_0 \in (\mathrm{Id} - \alpha\nabla f)^{-t_0}(\mathcal{E}).$$

(For any map $g : \mathbb{R}^n \to \mathbb{R}^n$, we define $g^{-t_0}(\mathcal{E})$ as the set of points x such that $g^{t_0}(x) = g \circ \overset{t_0 \text{ times}}{\dots} \circ g(x) \in \mathcal{E}$.) Therefore, the set of initial points x_0 for which gradient descent iterates may converge to 0 is included in

$$\bigcup_{t\in\mathbb{N}} (\mathrm{Id} - \alpha\nabla f)^{-t}(\mathcal{E}).$$

The set $\mathcal{E}$ has zero Lebesgue measure and one can check that $\mathrm{Id} - \alpha\nabla f$ is a diffeomorphism, hence $(\mathrm{Id} - \alpha\nabla f)^{-t}(\mathcal{E})$ has zero Lebesgue measure for any $t \in \mathbb{N}$, and the set of "problematic" initial points also has zero Lebesgue measure. □

3.1.4 Summary

The main messages to remember from this section are :

— it is in general not possible to find *a global minimizer* of a non-convex function (at least in a reasonable amount of time) ;

— however, standard optimization algorithms (like gradient descent or trust-region) are in general able to find at least *a second-order critical point.*

3.2 Examples of non-convex algorithms

In this section, we describe a few non-convex algorithms, to give a quick overview of the principles underlying these methods. We divide them in two categories : "optimization-based methods", which rely on general optimization algorithms, and algorithms "tailored to a problem", which exploit the specific form of the problem at hand.

3.2.1 Optimization-based methods

Here, the principle is to formulate the given problem as a standard optimization problem, on a space with minimal dimension, and apply a standard optimization algorithm. We present it in the context of low-rank matrix recovery problems.

As said at the beginning of this chapter, a matrix $X \in \mathbb{R}^{d_1\times d_2}$ with rank at most r can always be written as

$$X = LR, \quad \text{for some } L \in \mathbb{R}^{d_1\times r}, R \in \mathbb{R}^{r\times d_2},$$

or even, if $d_1 = d_2$ and X is semidefinite positive,

$$X = UU^T, \quad \text{for some } U \in \mathbb{R}^{d_1 \times r}.$$

Conversely, any matrix of this form has rank at most r.[4] This is called a *low-rank factorization* of X.

This allows to rewrite Problem (Low rank) as an optimization problem on $\mathbb{R}^{d_1 \times r} \times \mathbb{R}^{r \times d_2}$.

$$\boxed{\begin{array}{c} \text{recover } X \in \mathbb{R}^{d_1 \times d_2} \\ \text{such that } \mathcal{L}(X) = y, \\ \text{and } \operatorname{rank}(X) \leq r. \end{array}} \tag{Low rank}$$

$$\Updownarrow$$

$$\boxed{\begin{array}{c} \text{find } L \in \mathbb{R}^{d_1 \times r}, R \in \mathbb{R}^{r \times d_2} \\ \text{such that } \mathcal{L}(LR) = y. \end{array}}$$

For any function $f : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ such that $f(a, b) = 0$ if and only if $a = b$, this latter problem is equivalent to

$$\boxed{\begin{array}{c} \text{minimize } f(\mathcal{L}(LR), y) \\ \text{over all } L \in \mathbb{R}^{d_1 \times r}, R \in \mathbb{R}^{r \times d_2}. \end{array}} \tag{Factorized}$$

The simplest and most standard choice for f is

$$f(\mathcal{L}(LR), y) = \frac{1}{2} \left|\left|\mathcal{L}(LR) - y\right|\right|_2^2.$$

Others are possible, depending on the structure of $\mathcal{L}$ and eventual additional assumptions on X, for instance

$$\begin{aligned} f(\mathcal{L}(LR), y) &= \left|\left|\mathcal{L}(LR) - y\right|\right|_1 \\ \text{or} \quad f(\mathcal{L}(LR), y) &= \frac{1}{2} \left|\left|\sqrt{\mathcal{L}(LR)} - \sqrt{y}\right|\right|_2^2. \end{aligned}$$

[4] because $\operatorname{Range}(X) \subset \operatorname{Range}(L)$, which has dimension at most r if L has r columns.

(In the second example, the square root must be understood as the *component-wise square root*. It is of course well-defined only if $\mathcal{L}(LR)$ and y are assumed to have nonnegative coordinates.)

The same principle applies when $X \succeq 0$, leading to

$$\boxed{\begin{array}{c}\text{minimize } f(\mathcal{L}(UU^T), y)\\ \text{over all } U \in \mathbb{R}^{d\times r}.\end{array}} \tag{Sym-factorized}$$

Standard optimization algorithms can be applied to Problems (Factorized) or (Sym-factorized). The simplest choice (and very often the preferred one for theoretical analysis) is of course gradient descent, but many others can be considered ; for (Factorized), alternating minimization is notably also quite common.

While we have focused on low-rank matrix recovery in this subsection, the principle we have described is very general and standard. This is notably the favored approach in deep learning : the predictor one wants to learn is described by a set of parameters, combined together according to a specific network architecture. The learning problem is then formulated as the minimization of a data fidelity term on the set of all possible parameters, which is solved using refinements of gradient descent. The successes obtained in training neural networks this way, despite their non-convexity, have been an important motivation for the research community to better investigate the mechanisms governing the behavior of non-convex optimization algorithms, even in other problems than deep learning.

3.2.2 Problem-specific methods : orthogonal matching pursuit

The optimization-based approach has the advantage of being very general. However, in some settings, the specific properties of the problem suggest other strategies, possibly leading to more natural, simpler to implement or faster algorithms. In this subsection, we describe one example, which is an algorithm for compressed sensing called *Orthogonal matching pursuit (OMP)* (another example - alternating projections for phase retrieval - is described in an exercise). Historically important, Orthogonal matching pursuit is now outperformed by more recent and more sophisticated compressed sensing methods in terms of recovery capacity and speed. However, it has the advantage of being very simple, and can be proved to succeed under similar conditions as (Basis Pursuit) [Tropp and Gilbert, 2007].

We recall that compressed sensing is the following problem :

$$\boxed{\begin{aligned} &\text{recover } x \in \mathbb{R}^d \\ &\text{such that } Ax = y, \\ &\quad \text{and } ||x||_0 \leq k. \end{aligned}} \tag{CS}$$

The difficult part is to recover the support of x, that is, the indices of the non-zero coordinates. Once the support has been recovered, (CS) becomes a simple linear inverse problem. Orthogonal matching pursuit builds on a specific selection procedure for the support. New support elements are iteratively selected. After each selection, the corresponding linear inverse problem, on the current estimated support, is solved. The solution is used to select the following element.

To describe the selection procedure, let us denote $i_1, \dots, i_k$ the elements of the support (this is what we want to find), and $a_1, \dots, a_d$ the columns of A. The equality $Ax = y$ can be written as

$$y = x_{i_1} a_{i_1} + ... + x_{i_d} a_{i_d}.$$

Finding $i_1, \dots, i_d$ amounts to finding a small number of columns of A such that y is a linear combination of these columns. Let us imagine that we have already found the first indices $i_1, \dots, i_t$, and computed the best approximation of y in $\mathrm{Vect}\left\{a_{i_1}, \dots, a_{i_d}\right\}$:

$$z_t = \operatorname{argmin}\left\{||y - z||_2, z \in \mathrm{Vect}\left\{a_{i_1}, \dots, a_{i_d}\right\}\right\}.$$

The principle of the procedure is to choose i_{t+1} such that, for an appropriate $\xi_{t+1} \in \mathbb{R}$, $z_t + \xi_{t+1} a_{i_{t+1}}$ approximates y as well as possible. This is equivalent to choosing i_{t+1} such that the projection of $y - z_t$ onto $\mathbb{R} a_{i_{t+1}}$ has maximal norm, that is,

$$i_{t+1} \in \underset{i \in \{1,\dots,d\} \setminus \{i_1,\dots,i_t\}}{\operatorname{argmax}} \left| \left\langle y - z_t, \frac{a_i}{||a_i||_2} \right\rangle \right|.$$

The resulting algorithm is summarized in the following pseudo-code.

```
Input: $A \in \mathbb{R}^{m \times d}, y \in \mathbb{R}^m, k \in \mathbb{N}$
Set $x_0 = 0$ (initial signal estimate).
Set $z_0 = 0$ (initial approximation of $y$).
for $t = 1, \dots, k$ do
    Choose $i_t \in \underset{i \in \{1,\dots,d\} \setminus \{i_1,\dots,i_{t-1}\}}{\operatorname{argmax}} \left| \left\langle y - z_t, \frac{a_i}{||a_i||_2} \right\rangle \right|$.
    Compute $x_t = \underset{x, \mathrm{Supp}(x) \subset \{i_1,\dots,i_t\}}{\operatorname{argmin}} ||y - Ax||_2$.
    Set $z_t = Ax_t$.
end
return $x_k$
```

Algorithm 1: Orthogonal matching pursuit

3.3 Correctness guarantees

Several proof techniques have been introduced, in the last decade, to establish correctness guarantees for non-convex algorithms. Some directly exploit the specificities of a problem or an algorithm (like [Tropp and Gilbert, 2007] for Orthogonal Matching Pursuit). In these notes, we only give an overview of the most versatile ones, which have been successfully applied to several inverse problems and algorithms.

They have been developed withing the scope of "optimization-based algorithms" (Subsection 3.2.1) ; their principle is simply to study the critical points of the objective function and, possibly, analyze in more detail the behavior of the function in the neighborhood of critical points. They result in two types of correctness guarantees.

- Local convergence results show that the algorithm finds the solution, provided that its starting point is in an (explicit) neighborhood of this solution.

- Global convergence results show that the iterates generated by the algorithm converge to the solution, starting from almost any point.

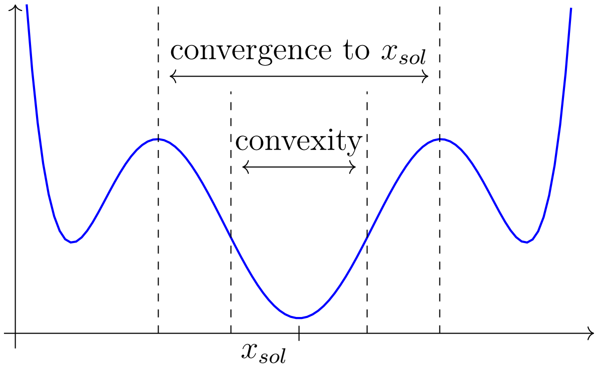

FIGURE 3.1 – a function $F : \mathbb{R} \to \mathbb{R}$, the region around x_{sol} where it is convex, and the interval of starting points from which gradient descent converges to x_{sol}.

3.3.1 Local convergence

Intuition

Let us imagine that we consider a problem (P), with unknown solution x_{sol}, and an algorithm Alg, which takes a starting point x_0 as input and produces a sequence of iterates $(x_t)_{t\in\mathbb{N}}$. A local convergence result for Alg is typically of the following form :

"For any $x_0 \in B(x_{sol}, R)$, the sequence $(x_t)_{t\in\mathbb{N}}$ converges to x_{sol}.",

where $R > 0$ is a convergence radius. The result can also include a statement about the convergence speed towards x_{sol}.

Intuitively, why is it reasonable to expect that a non-convex algorithm enjoys local convergence guarantees ? Let us assume that Alg is "optimization-based" (Subsection 3.2.1) : x_{sol} is the global minimizer of some non-convex objective function F, and Alg attempts to find it by running a standard optimization method on F. If F is C^2, and if Hess$F(x_{sol})$ is definite positive (which is the most frequent situation when x_{sol} is an isolated global minimizer), then F is convex in the neighborhood of x_{sol}. When initialized in this neighborhood, Alg should behave as if ran on a globally convex function, hence converge to x_{sol}. This provides an argument for the existence of a region around x_{sol} where Alg converges to x_{sol} (which may of course be larger than the convexity region). See Figure 3.1 for an illustration.

Note that a local convergence result is more or less interesting, depending on the value of R (the radius of the convergence region). If R is too small, then choosing an initial point in $B(x_{sol}, R)$ is not a significantly easier problem than finding x_{sol} itself, so the result is not of much use.

Example : phase retrieval by Wirtinger Flow

A simple example of a local convergence result comes from [Candès, Li, and Soltanolkotabi, 2015], and is about the so-called *Wirtinger Flow* algorithm for phase retrieval.

Wirtinger Flow is an "optimization-based" algorithm. Let us describe it. We recall the general form of a phase retrieval problem (denoting $v_j \in \mathbb{C}^d$ the vectors associated to the linear forms : $L_j = \langle v_j, . \rangle$) :

$$\boxed{\begin{array}{c}\text{recover } x \in \mathbb{C}^d \\ \text{such that } |\langle v_j, x\rangle| = y_j, \forall j \leq m.\end{array}} \qquad \text{(Phase retrieval)}$$

A vector $x \in \mathbb{C}^d$ solves the problem if and only if, for all j,

$$\left(|\langle v_j, x\rangle|^2 = y_j^2\right) \quad \Longleftrightarrow \quad \left(\left(|\langle v_j, x\rangle|^2 - y_j^2\right)^2 = 0\right).$$

Therefore, solving Problem (Phase retrieval) amounts to finding a global minimizer of

$$\begin{array}{rccc} f: & \mathbb{C}^d & \to & \mathbb{R} \\ & x & \to & \frac{1}{2m}\sum_{j=1}^m \left(|\langle v_j, x\rangle|^2 - y_j^2\right)^2. \end{array}$$

The Wirtinger Flow algorithm attempts to find a minimizer by gradient descent, starting at an arbitrary point x_0 :

$$\begin{aligned} x_{t+1} &= x_t - \mu \nabla f(x_t) \\ &= x_t - \mu\left(\frac{1}{m}\sum_{j=1}^m \left(|\langle v_j, x_t\rangle|^2 - y_j^2\right)\langle v_j, x_t\rangle v_j\right), \quad \forall t \in \mathbb{N}. \end{aligned}$$

Theorem 3.9 : local convergence for Wirtinger Flow [Candès, Li, and Soltanolkotabi, 2015]

Let us assume that $v_1, \dots, v_m$ are chosen independently at random in $\mathbb{C}^d$, following standard normal distributions. Let $x_{sol} \in \mathbb{C}^d$ be any vector.
There exists a constant $c > 0$ such that, if

$$m \geq cd \log(d),$$

then, with high probability, [a] for any

$$x_0 \in B\left(x_{sol}, \frac{1}{8}||x_{sol}||_2\right),$$

the sequence $(x_t)_{t\in\mathbb{N}}$ converges to x_{sol} (up to a global phase) at a linear rate if the stepsize μ is small enough.

[a] that is, with probability at least $1 - \frac{c}{d^2}$,

Vague proof idea. Directly analyzing the function

$$f : x \in \mathbb{C}^d \to \frac{1}{2m}\sum_{j=1}^m \left(|\langle v_j, x\rangle|^2 - y_j^2\right)^2$$
$$= \frac{1}{2m}\sum_{j=1}^m \left(|\langle v_j, x\rangle|^2 - |\langle v_j, x_{sol}\rangle|^2\right)^2$$

is difficult. To make it easier, we observe that, for fixed x, $f(x)$ is the average of m random components, with the same distribution :

$$\frac{1}{2}\left(|\langle v_j, x\rangle|^2 - |\langle v_j, x_{sol}\rangle|^2\right)^2, \quad j = 1, \dots, m.$$

(To be clear : here, x and x_{sol} are fixed vectors. The randomness lies in the measurement vectors v_j, which follow standard Gaussian laws, independently one from each other.)

By the law of large numbers, we may expect that $f(x)$ is close to the expectation of the components :

$$f(x) \approx \mathbb{E}_{v_1,\dots,v_m} f(x)$$

$$= \mathbb{E}_{v_1}\left[\frac{1}{2}\left(|\langle v_1, x\rangle|^2 - |\langle v_1, x_{sol}\rangle|^2\right)^2\right]$$
$$= (||x||^2 - ||x_{sol}||^2)^2 + ||x||^2||x_{sol}||^2 - |\langle x, x_{sol}\rangle|^2.$$

The expectation is a much simpler function, and gradient descent on $\mathbb{E}f$ can be analyzed with elementary linear algebra (see the exercises for the analysis of local convergence of gradient descent on a different but similar objective function). This provides the backbone of a proof strategy :

1. prove that gradient descent on $\mathbb{E}f$ converges linearly to x_{sol} for all $x_0 \in B\left(x_{sol}, \frac{1}{8}||x_{sol}||_2\right)$;
2. prove that f (and its derivatives) are sufficiently close to $\mathbb{E}f$ so that the proof for $\mathbb{E}f$ also applies to f.

The second part is called a *concentration* property. There exist well-established statistical tools to prove such properties (*concentration inequalities*).[5] □

A refinement of Theorem 3.9, described in [Ma, Wang, Chi, and Chen, 2018], is to consider for the local convergence region a set which is not a ball but has a more complicated shape. The advantage of choosing the region in a more subtle way is that this allows to ensure that f and its derivatives possess some nice properties over the region, which they do not possess over a ball, and which allow to establish faster convergence rates to x_{sol} for Wirtinger Flow.[6] The drawback is that proving that the gradient descent iterates do not leave the region becomes much more difficult.[7]

Remark : initialization in the local convergence region

As hinted at before, a local convergence result is useful only if there exists a procedure to find an initial point in the region of local convergence. Theorem 3.9, for instance, must come together with a procedure

[5] This is where the assumption that $v_1, \dots, v_m$ are independent and Gaussian is crucial. Gaussian variables have better concentration properties than non-Gaussian ones, and establishing concentration for non-independent variables is far more difficult than for independent ones.

[6] Specifically, Hess$f$ is bounded over the region, with bounds independent from d, m.

[7] When the region is a ball, as in Theorem 3.9, the negative gradient points towards the interior of the ball. This directly implies that gradient descent iterates stay inside the ball if the stepsize is small enough. This is not true anymore if the local convergence region has a more complicated shape.

to find a point

$$x_0 \in B\left(x_{sol}, \frac{1}{8}||x_{sol}||_2\right). \tag{3.9}$$

When the vectors v_j follow a normal distrubtion, such a procedure exists. One can choose x_0 as the main eigenvector of a suitable matrix built from the measurement vectors v_j and measurements y_j, and it satisfies Property (3.9). This strategy, called *spectral initialization*, has been introduced in [Netrapalli, Jain, and Sanghavi, 2013].

3.3.2 Global convergence

Differently from "local" results, global convergence guarantees state that, for most initializations, even far away from the true solution, the sequence of iterates generated by the algorithm converges to the correct solution of the inverse problem. This is usually more relevant than local guarantees from the point of view of applications, since the situations where good initial points are available (like in Remark 3.3.1) are rare.

Let us discuss the most frequent case, where the algorithm consists in minimizing a well-chosen non-convex objective function f, whose global minimizers are the solutions of the problem, using a standard optimization method (gradient descent, for instance). The standard optimization method is usually guaranteed to return a second-order critical point. Therefore, showing that it returns a global minimizer of f amounts to showing that the basins of attraction of the non-globally optimal second-order critical points occupy a small volume in the space of all possible initial points (so that "most" initial points one can choose are outside these basins).

Quantifying the size of the basins is generally difficult. Indeed, denoting $\mathcal{T}$ the iteration operator (for instance, $\mathcal{T} : x \to x - \alpha\nabla f(x)$ for gradient descent), it requires to study $\lim_{t\to+\infty} \mathcal{T}^t(x_0)$ as a function of x_0. But since $\mathcal{T}$ is generally a relatively complicated operator, it is usually already difficult to get a precise understanding of $\mathcal{T}^2$; the limit of $\mathcal{T}^t$ is out of reach.

A particular case is generally slightly easier : when there are no non-globally optimal second-order critical points. For instance, this happens for the Wirtinger Flow objective function considered in the previous subsection ; it allows to improve the local guarantees of Theorem 3.9 to global ones. This is the content of the following theorem.

Theorem 3.10 : global convergence for Wirtinger Flow [Sun, Qu, and Wright, 2018]

Let us assume that $v_1, \dots, v_m$ are chosen independently at random in $\mathbb{C}^d$, following standard normal distributions. Let $x_{sol} \in \mathbb{C}^d$ be any vector.

There exists a constant $c > 0$ such that, if

$$m \geq cd \log^3(d),$$

then, with high probability,[a] the only second-order critical points of the Wirtinger Flow objective

$$\begin{array}{rccc} f: & \mathbb{C}^d & \to & \mathbb{R} \\ & x & \to & \frac{1}{2m}\sum_{j=1}^m \left(|\langle v_j, x\rangle|^2 - y_j^2 \right)^2 \end{array}$$

are its global minimizers.

As a consequence, provided that the stepsize is small enough, the Wirtinger Flow iterates converge to a solution of the phase retrieval problem for almost any initial point x_0.

[a] that is, with probability at least $1 - \frac{c}{m}$,

Remark

Compared to Theorem 3.9, the main improvement in Theorem 3.10 is of course that the guarantees are *global* and not *local*. This comes with two drawbacks :

- the number of measurements has to be higher : $m \geq cd \log^3(d)$ versus $m \geq cd \log(d)$;
- most importantly, Theorem 3.10 provides no guarantee on the convergence rate.

The proof idea of Theorem 3.10 is somewhat similar in spirit to Theorem 3.9, although the detail of the computations is different. One shows that $f, \nabla f, \text{Hess} f$ are "close" to their expectation, at least in some well-chosen regions of $\mathbb{C}^d$, and only along some directions for the Hessian. It allows to show that the second-order critical points of f are the same as the second-order critical points of $\mathbb{E}f$. The latter are easy to compute and turn out to be exactly the solutions of the phase retrieval problem.

The non-existence of non-globally optimal second-order critical points has been established for several problems and algorithms other than phase retrieval with Wirtinger Flow. However, there are also non-convex algorithms which appear to work well but for which non-globally optimal second-order critical points exist. For these algorithms, it is sometimes possible to show that these critical points necessarily belong to some small explicit[8] open set $\mathcal{E}_{bad}$ and that, for most initial points, the sequence of iterates generated by the algorithm never enters $\mathcal{E}_{bad}$. This guarantees that the algorithm does not converge to a non-globally optimal second-order critical point. However, this strategy requires sophisticated statistical arguments, and, at the current stage of knowledge, can only be applied to relatively simple objective functions.

[8] By "explicit", we mean that the set has a reasonably simple definition in terms of the parameters and data of the problem.

Chapitre 4

Synchronization

In this short chapter, we discuss a family of non-convex inverse problems we have not encountered so far : synchronization problems. Although other applications exist, we motivate their study by the development of single-particle electron cryomicroscopy, of which we provide a brief description in the first section (based on the review [Singer and Sigworth, 2020]).

Compared to the inverse problems we have discussed until now, synchronization problems present a specificity : the observation $M(x)$ is not a deterministic, but a random function of the unknown object x. This prevents reconstructing x with certainty. However, as we will see, an approximate reconstruction is possible, and the best achievable approximation quality can be precisely quantified.

4.1 Single-particle electron cryomicroscopy

Electron cryomicroscopy (cryo-EM) is an imaging technique for biological molecules, where a specimen containing the particles is frozen and exposed to an electron beam. [1] In single-particle cryo-EM, the sample is a thin sheet of matter, containing many examplars of the molecule of interest, in more or less random orientations. The image obtained by refocusing the electron beam after it has been diffracted by the specimen looks like Figure 4.1. From it, a 3D view of the molecule must be reconstructed. This is difficult, in

[1] Electron beams behave as a wave with very small wavelength, which in priciple allows for a high resolution image. Freezing the specimen helps reducing the amount of damage caused by the beam to the particles (which destroys the sample).

particular because

- the image is very noisy ;
- the orientation of each molecule in the image is unknown ;
- only the two-dimensional projections of the molecules are observed, not the full 3D shapes.

Typically, the image is first segmented into subimages, each subimage containing one molecule. Then, the subimages undergo an averaging procedure : they are grouped into batches by similarity (so that all molecules in a batch ideally have the same orientation), and the mean of each group is computed. This partly removes the noise. Finally, the dominant orientation in each batch and the 3D shape of the molecule are simultaneously reconstructed by iteratively refining a rough initial shape estimate. This process raises many interesting mathematical and algorithmic questions. We focus on a specific part of this process and explain how *phase synchronization* problems can be seen as a (very simplified) mathematical model for this part.

4.2 Definition of the problem

The orientation of a 3D molecule can be described by an element of $SO(3)$ [2], which represents the movement of the 3D molecule compared to a reference orientation. For each batch k, let $R_k \in SO(3)$ be the element describing the corresponding orientation.

For each pair of batches (k, k'), it is possible ([Singer and Shkolnisky, 2011]) to estimate the orientation difference between the batches

$$Y_{k,k'} \overset{def}{=} R_k R_{k'}^{-1}.$$

From this information, how to estimate all the R_k ? This is called a $SO(3)$-*synchronization problem.*

A simpler problem is when the orientations are not 3-dimensional, but 2-dimensional. They can then simply be described by unitary complex numbers $(z_k)_{1\leq k\leq n}$. We call $\mathbb{C}_1$ the set of unitary complex numbers. Observing the orientation difference between batches k and k' corresponds to observing

$$y_{k,k'} \overset{def}{=} z_k z_{k'}^{-1} = z_k \overline{z_{k'}}.$$

[2] The *special orthogonal group* $SO(3)$ is the set of linear isometries of $\mathbb{R}^3$ with determinant 1.

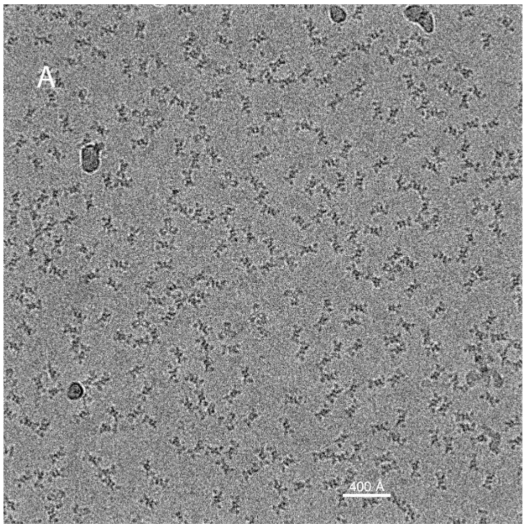

FIGURE 4.1 – Image obtained in the imaging process of the so-called PaaZ enzyme [Sathyanarayanan, Cannone, Gakhar, Katagihallimath, Sowdhamini, Ramaswamy, and Vinothkumar, 2019].

Given that $y_{k,k} = 1$ and $y_{k,k'} = \overline{y_{k,k'}}$ for all k, k', we can assume that we simply observe $y_{k,k'}$ for $k < k'$.

Recovering $(z_k)_{1\le k\le n}$ from $(y_{k,k'})_{1\le k<k'\le n}$ is only possible up to a global phase ambiguity since, for each $\phi \in \mathbb{R}$ and all $(k,k') \in \{1,\dots,n\}^2$,

$$\left(z_k e^{i\phi}\right)\left(\overline{z_{k'} e^{i\phi}}\right) = z_k \overline{z_{k'}} e^{i\phi} e^{-i\phi} = z_k \overline{z_{k'}}.$$

Up to this ambiguity, recovery is easy when the $y_{k,k'}$ are exactly observed. Indeed, we can assume that $z_n = 1$ (this is always true up to a global phase) and then, for any $k < n$,

$$z_k = z_k \overline{1} = y_{k,n}.$$

We don't even need all the $(y_{k,k'})_{1\le k<k'\le n}$ to recover z!

However, in cryo-EM, the images are so noisy that the estimations of the orientation differences $Y_{k,k'}$ ($y_{k,k'}$ in the simplified two-dimensional model) are of poor quality. To model this, let us assume our actual observations are not the $y_{k,k'} = z_k \overline{z_{k'}}$, but

$$w_{k,k'} = z_k \overline{z_{k'}} + e_{k,k'},$$

where $(e_{k,k'})_{1\leq k<k'\leq n}$ are independent noises, each with Gaussian distribution on $\mathbb{C}$ of variance

$$\sigma^2 \overset{def}{=} \mathbb{E}|e_{k,k'}|^2.$$

In the end, the problem we consider is

$$\boxed{\begin{aligned} &\text{recover } z \in \mathbb{C}_1^n \\ &\text{from } (w_{k,k'})_{1\leq k<k'\leq n}, \\ &\text{knowing that } w_{k,k'} \sim \mathcal{N}_{\mathbb{C}}(z_k\overline{z_{k'}}, \sigma^2), \quad \forall k, k'. \end{aligned}} \tag{phase sync}$$

4.3 Optimal statistical estimator

As the measurements are noisy, we cannot hope to exactly recover the unknown z. However, at least if σ is small, we should be able to recover an approximation of z. What is the best approximation quality that we can expect, depending on n and σ ?

In this section, we consider this question without imposing any practicality constraint on the recovery procedure. Let Est be any estimator, that is to say a function

$$\text{Est} : \mathbb{C}^{\frac{n(n-1)}{2}} \to \mathbb{C}_1^n$$

which, given $(w_{k,k'})_{1\leq k<k'\leq n}$ as input, outputs some estimation $\text{Est}(z)$ of z. For some fixed z, the average ℓ^2 error [3] of the estimator Est is

$$\mathbb{E}\left(\min_{a\in\mathbb{C}_1} ||\text{Est}(z) - az||_2^2\right).$$

In this definition, the minimisation over $a \in \mathbb{C}_1$ accounts for the global phase ambiguity. The expectation is over the randomness present in the $w_{k,k'}$: since the $w_{k,k'}$ are random functions of z and $\text{Est}(z)$ depends on z through the $w_{k,k'}$, $\text{Est}(z)$ is a random variable.

We quantify the performance of an estimator by its worst-case ℓ^2 error :

$$\max\text{error}(\text{Est}) \overset{def}{=} \max_{z\in\mathbb{C}_1^n} \mathbb{E}\left(\min_{a\in\mathbb{C}_1} ||\text{Est}(z) - az||_2^2\right).$$

[3] We could of course evaluate the error with respect to another norm that the ℓ^2 one, but the analysis would probably be much more difficult.

The following theorem describes the best possible performance achievable by any estimator.

Theorem 4.1 : minimax estimation for phase synchronization [Gao and Zhang, 2021]

We consider Problem (phase sync) when n goes to infinity. The variance σ^2 is allowed to depend on n ; we denote it σ_n^2 to make the dependency visible. We assume that $\sigma_n = o(\sqrt{n})$.
For any estimator Est,

$$\max \operatorname{error}(\mathrm{Est}) \geq (1 - o(1))\frac{\sigma_n^2}{2}.$$

Moreover, there exists an estimator for which equality is attained :

$$\max \operatorname{error}(\mathrm{Est}) = (1 - o(1))\frac{\sigma_n^2}{2}.$$

Remark

The trivial estimator which returns $z_1 = z_2 = \cdots = z_n = 1$ has performance

$$\max \operatorname{error}(\mathrm{Est}) = 2n.$$

Theorem 4.1 states that, when $\sigma_n = o(\sqrt{n})$, it is possible to recover a "better than trivial" approximation of z.
Observe that, for any k, k', $|z_k\overline{z_{k'}}| = 1$ while the noise has magnitude $|e_{k,k'}|$ of order σ_n. Therefore, the assumption $\sigma_n = o(\sqrt{n})$ allows for the noise magnitude to be much larger than the signal.

4.4 Spectral estimator

In the previous section, we have established the best possible performance of *any* estimator for the phase synchronization problem. Now, the question is : is there a practical algorithm which achieves this performance ? The answer is yes, and it turns out that even a very simple one, called *spectral estimator*, does the job.[4]

To define the spectral estimator, we arrange the observations $(w_{k,k'})_{1\leq k<k'\leq n}$ into a matrix

$$W = \begin{pmatrix} 1 & w_{1,2} & \dots & w_{1,n} \\ \overline{w_{1,2}} & 1 & \dots & w_{2,n} \\ \vdots & & & \vdots \\ \overline{w_{1,n}} & & \dots & 1 \end{pmatrix} \in \mathbb{C}^{n\times n}.$$

Observe that, when there is no noise ($\sigma = 0$),

$$W = \begin{pmatrix} |z_1|^2 & z_1\overline{z_2} & \dots & z_1\overline{z_n} \\ z_2\overline{z_1} & 1 & \dots & z_2\overline{z_n} \\ \vdots & & & \vdots \\ z_n\overline{z_1} & & \dots & 1 \end{pmatrix} = zz^*,$$

so that z is the only eigenvector of W corresponding to a non-zero eigenvalue. This suggests an estimation procedure for z, even when $\sigma > 0$: we call u the main eigenvector of W, and consider the estimator $\hat{z} \in \mathbb{C}_1^n$ such that

$$\hat{z}_k = \frac{u_k}{|u_k|}, \qquad \forall k = 1, \dots, n.$$

Theorem 4.2 : performance of the spectral estimator [Zhang, 2022]

We keep the hypotheses of Theorem 4.1.
The spectral estimator we just defined has the following performance :

$$\max \text{error}(\text{Est}) = (1 - o(1))\frac{\sigma_n^2}{2}.$$

Idea of proof. We denote E the matrix of errors :

$$E = \begin{pmatrix} 0 & e_{1,2} & \dots & e_{1,n} \\ \overline{e_{1,2}} & 0 & \dots & e_{2,n} \\ \vdots & & & \vdots \\ \overline{e_{1,n}} & & \dots & 0 \end{pmatrix},$$

[4] It is not always the case that the best possible performance can be achieved with a practical algorithm. There are problems where we have strong hints that no polynomial-time method can approach the best performance ; this is called a *statistical-to-computational* gap.

so that $W = zz^* + E$.

The first part of the proof is to provide a simple approximation for u, the main eigenvector of W, as a function of z and E : for some constant $c > 0$,

$$\left|\left|u - \frac{Wz}{||Wz||_2}\right|\right|_2 \leq c\frac{\sigma_n^2 + \sigma_n}{n}(\ll 1).$$

This inequality can be interpreted as a first-order Taylor expansion of u : the vector Wz is the first iterate of the power iteration method on W with starting point z. We admit this inequality. It yields, for each k,

$$\frac{u_k}{|u_k|} \approx \frac{(Wz)_k}{|(Wz)_k|}.$$

We have $Wz = ||z||^2 z + Ez = nz + Ez$, so for each $k \leq n$,

$$(Wz)_k = n\left(z_k + \frac{(Ez)_k}{n}\right),$$

and

$$\frac{u_k}{|u_k|} \approx \frac{z_k + \frac{(Ez)_k}{n}}{\left|z_k + \frac{(Ez)_k}{n}\right|}.$$

The entries of E are Gaussian random variables with variance σ_n^2, except the diagonal ones which are 0, so each $\frac{(Ez)_k}{n}$ is a Gaussian random variable with variance $\sigma_n^2\left(\frac{n-1}{n^2}\right) \approx \frac{\sigma_n^2}{n} \ll 1$. Therefore, most of the time, $\left|\frac{(Ez)_k}{n}\right| = O\left(\frac{\sqrt{\sigma_n}}{n}\right) \ll 1$, and we can compute the first-order Taylor expansion of the previous quantity :

$$\frac{u_k}{|u_k|} \approx z_k\left(1 + i\,\mathrm{Im}\left(\frac{(Ez)_k\overline{z_k}}{n}\right)\right).$$

The term $\mathrm{Im}\left(\frac{(Ez)_k\overline{z_k}}{n}\right)$ is a real Gaussian random variable with variance approximately $\frac{\sigma_n^2}{2n}$, so

$$\mathbb{E}\left|\frac{u_k}{|u_k|} - z_k\right|^2 \approx \frac{\sigma_n^2}{2n}.$$

Summing over k yields

$$\mathbb{E}||\hat{z} - z||_2^2 \approx \frac{\sigma_n^2}{2}.$$

As formulated, this proof is not rigorous because the approximate equality signs hide error terms which should be precisely computed and upper bounded, but it can be turned into a rigorous proof.

□

Annexe A

Additional proofs

A.1 Proof of Proposition 2.15

Démonstration. If $v = 0$, then $1_{v=0} = 0 = \max_{z\in\mathbb{C}^n} 0 = \max_{z\in\mathbb{C}^n} \operatorname{Re}\langle z, v\rangle$.

If $v \neq 0$, then, for any $t \in \mathbb{R}$,

$$\max_{z\in\mathbb{C}^n} \operatorname{Re}\langle z, v\rangle \geq \operatorname{Re}\left\langle t\frac{v}{||v||_2^2}, v\right\rangle = t.$$

Therefore, $\max_{z\in\mathbb{C}^n} \operatorname{Re}\langle z, v\rangle = +\infty = 1_{v=0}$. □

A.2 Proof of Proposition 2.16

Démonstration. Let $f : [0; 1] \to \mathbb{C}$ be a continuous function.

Let us first assume that there exists $t_0 \in [0; 1]$ for which $|f(t_0)| > 1$. As f is continuous, we can assume that $t_0 < 1$. Let ϕ be the argument of $f(t_0)$, so that $e^{-i\phi} f(t_0) = |f(t_0)|$. For an arbitrary $r \in \mathbb{R}^+$, let us set

$$\mu = re^{-i\phi}\delta_{t_0}.$$

We have

$$\begin{aligned} ||\mu||_{TV} - \operatorname{Re}\int_0^1 f(t)d\mu(t) &= r - re^{-i\phi}f(t_0) \\ &= r(1 - |f(t_0)|). \end{aligned}$$

Therefore, for any $r \in \mathbb{R}^+$,

$$\min_{\mu \in \mathcal{M}([0;1[)} \left(||\mu||_{TV} - \operatorname{Re} \int_0^1 f(t) d\mu(t) \right) \leq r(1 - |f(t_0)|).$$

By letting r go to infinity, we get

$$\min_{\mu \in \mathcal{M}([0;1[)} \left(||\mu||_{TV} - \operatorname{Re} \int_0^1 f(t) d\mu(t) \right) = -\infty.$$

Let us now assume that $|f(t)| \leq 1$ for all $t \in [0;1]$. Then, from the equivalent definition of total variation in Proposition 2.14, for all $\mu \in \mathcal{M}([0;1[)$, $\operatorname{Re} \int_0^1 f(t) d\mu(t) \leq ||\mu||_{TV}$, meaning that

$$||\mu||_{TV} - \operatorname{Re} \int_0^1 f(t) d\mu(t) \geq 0.$$

For $\mu = 0$, we have

$$||\mu||_{TV} - \operatorname{Re} \int_0^1 f(t) d\mu(t) = 0.$$

Therefore,

$$\min_{\mu \in \mathcal{M}([0;1[)} \left(||\mu||_{TV} - \operatorname{Re} \int_0^1 f(t) d\mu(t) \right) = 0.$$

Finally, let us prove the property about the support of minimizers. Let μ be a minimizer. Since $||\mu||_{TV} - \operatorname{Re} \int_0^1 f(t) d\mu(t) \neq -\infty$, it means that we are in the case where $|f| \leq 1$, and $||\mu||_{TV} - \operatorname{Re} \int_0^1 f(t) d\mu(t) = 0$. As a consequence, using Proposition 2.14 for the first equality,

$$\begin{aligned} \sup \left\{ \operatorname{Re} \int_0^1 f(t) d\mu(t), |f(t)| \leq 1, \forall t, f \text{ continuous} \right\} &= ||\mu||_{TV} \\ &= \operatorname{Re} \int_0^1 f(t) d\mu(t). \end{aligned}$$

This means that f attains the supremum of Proposition 2.14, hence Equation (2.14) holds :

$$\operatorname{Supp}(\mu) \subset \{t \in [0;1[, |f(t)| = 1\}.$$

□

A.3 Proof of Theorem 2.17

Démonstration. Problem (Dual TV) consists in maximizing a continuous function on a non-empty compact set (the feasible set is a closed and bounded subset of $\mathbb{C}^{2N+1}$) ; a maximizer exists. Proving the existence of a minimizer for Problem (Min TV) also relies on a compactness argument [1], but, as this argument requires some knowledge of functional analysis, we simply admit it.

Let z_* and μ_* be respectively a maximizer and a minimizer. Let us show that $||\mu_*||_{TV} = \mathrm{Re}\,\langle z_*, y\rangle$. This can be deduced from general results in convex optimization, but we give here an elementary proof, adapted from [Boyd and Vandenberghe, 2004, Paragraph 5.3.2].

We define two subsets of $\mathbb{C}^{2N+1} \times \mathbb{R}$:

$$A = \{(y_{-N} - \hat{\mu}[-N], \dots, y_N - \hat{\mu}[N], t) \text{ such that } \mu \in \mathcal{M}([0;1[), t \geq ||\mu||_{TV}\},$$
$$B = \{(0, \dots, 0, u) \text{ such that } u < ||\mu_*||_{TV}\}.$$

One can check that these subsets are convex and non-empty. They are also disjoint : $A \cap B = \emptyset$. Indeed, for any $\mu \in \mathcal{M}([0;1[)$ and $t \geq ||\mu||_{TV}$, either

$$(y_{-N} - \hat{\mu}[-N], \dots, y_N - \hat{\mu}[N]) \neq (0, \dots, 0)$$

or $(y_{-N} - \hat{\mu}[-N], \dots, y_N - \hat{\mu}[N]) = (0, \dots, 0)$ and then, since μ is a feasible point for Problem (Min TV),

$$||\mu||_{TV} \geq ||\mu_*||_{TV},$$

which implies $t \not< ||\mu_*||_{TV}$. In any case,

$$(y_{-N} - \hat{\mu}[-N], \dots, y_N - \hat{\mu}[N], t) \notin B.$$

From a convex separation theorem, there exists a non-zero $\zeta = (z, r) \in \mathbb{C}^{2N+1} \times \mathbb{R}$ and $\alpha \in \mathbb{R}$ such that

$$\forall (a,t) \in A, \quad \alpha \leq \mathrm{Re}\,\langle z, a\rangle + rt, \tag{A.1a}$$
$$\forall (b,u) \in B, \quad \mathrm{Re}\,\langle z, b\rangle + ru \leq \alpha. \tag{A.1b}$$

Let such a ζ be fixed. Equation (A.1b) can be rewritten as

$$\forall u < ||\mu_*||_{TV}, \quad ru \leq \alpha,$$

[1] For any $M > 0$, $\{\mu \in \mathcal{M}([0;1[), ||\mu||_{TV} \leq M\}$ is compact for some adequate topology.

which is equivalent to

$$r \geq 0 \text{ and } \alpha \geq r||\mu_*||_{TV}.$$

Let us first assume that $r > 0$ (in which case the previous equation implies $\frac{\alpha}{r} \geq ||\mu_*||_{TV}$). Then, from Equation (A.1a), it holds for any $\mu \in \mathcal{M}([0;1[)$ that

$$\begin{aligned} \forall t \geq ||\mu||_{TV}, \quad & \alpha \leq \operatorname{Re}\langle z, y - \hat{\mu}[-N:N]\rangle + rt, \\ \iff \quad & \alpha \leq \operatorname{Re}\langle z, y - \hat{\mu}[-N:N]\rangle + r||\mu||_{TV}, \\ \iff \quad & \frac{\alpha}{r} \leq \operatorname{Re}\left\langle \frac{z}{r}, y - \hat{\mu}[-N:N]\right\rangle + ||\mu||_{TV}. \end{aligned}$$

Since this holds true for all measures μ, it implies, setting $\tilde{z} = \frac{z}{r}$,

$$\begin{aligned} ||\mu_*||_{TV} \leq \frac{\alpha}{r} \leq & \inf_{\mu \in \mathcal{M}([0;1[)} \operatorname{Re}\langle \tilde{z}, y - \hat{\mu}[-N:N]\rangle + ||\mu||_{TV} \\ & = \inf_{\mu \in \mathcal{M}([0;1[)} ||\mu||_{TV} - \operatorname{Re}\int_0^1 \left(\sum_{k=-N}^{N} \overline{\tilde{z}_k} e^{-2\pi ikt}\right) d\mu(t) + \operatorname{Re}\langle \tilde{z}, y\rangle \\ & = \operatorname{Re}\langle \tilde{z}, y\rangle \text{ if } \left|\sum_{k=-N}^{N} \tilde{z}_k e^{2\pi ikt}\right| \leq 1, \forall t \in \mathbb{R} \\ & = -\infty \text{ otherwise, from Proposition 2.16.} \end{aligned}$$

As $||\mu_*||_{TV} \not\leq -\infty$, this means that $\left|\sum_{k=-N}^{N} \tilde{z}_k e^{2\pi ikt}\right| \leq 1, \forall t \in \mathbb{R}$ and $||\mu_*||_{TV} \leq \operatorname{Re}\langle \tilde{z}, y\rangle$. Consequently, $\tilde{z}$ is a feasible point for Problem (Dual TV), and

$$||\mu_*||_{TV} \leq \operatorname{Re}\langle \tilde{z}, y\rangle \leq \max \text{ (Dual)}.$$

But the construction of the dual guarantees that

$$\max \text{ (Dual)} \leq \min \text{ (Primal)} = ||\mu_*||_{TV}.$$

Therefore, we have the equality $||\mu_*||_{TV} = \max \text{ (Dual)} = \operatorname{Re}\langle z_*, y\rangle$.

Finally, let us show that it is impossible to have $r = 0$, meaning that our assumption that $r > 0$ is true. By contradiction, assuming $r = 0$, it holds from Equation (A.1a)

$$\alpha \leq \inf_{\mu \in \mathcal{M}([0;1[)} \operatorname{Re}\langle z, y - \hat{\mu}[-N:N]\rangle$$

$$= \langle z, y\rangle + \inf_{\mu\in\mathcal{M}([0;1[)} \mathrm{Re}\int_0^1 \left(-\sum_{k=-N}^{N} \overline{z_k} e^{-2\pi ikt}\right) d\mu(t).$$

The previous infimum is $-\infty$ if $\sum_{k=-N}^{N} \overline{z_k} e^{-2\pi ikt} \neq 0$ for at least one value of t. Since $\alpha \not\leq -\infty$, it must hold

$$\sum_{k=-N}^{N} \overline{z_k} e^{-2\pi ikt} = 0, \quad \forall t\in\mathbb{R},$$

hence $z = 0$, which contradicts the fact that $\zeta = (z, r)$ is non-zero. □

A.4 Proof of Proposition 2.18

Démonstration. The proof consists in studying the optimality conditions of the primal and dual problems. Using the same notation as in the reasoning which led to the definition of Problem (Dual TV),

$$\begin{aligned}
\min \text{ (Min TV)} &= f_1(\mu_*)\\
&= \max_{z\in\mathbb{C}^{2N+1}} F(\mu_*, z)\\
&\geq F(\mu_*, z_*)\\
&\geq \min_{\mu\in\mathcal{M}([0;1[)} F(\mu, z_*)\\
&= f_2(z_*)\\
&= \max \text{ (Dual TV)},
\end{aligned}$$

The equality between the optimal primal and dual values implies that the inequalities are equalities. In particular, $F(\mu_*, z_*) = \min_{\mu\in\mathcal{M}([0;1[)} F(\mu, z_*)$, which is to say that μ_* is a minimizer of

$$\mu\in\mathcal{M}([0;1[) \quad\rightarrow\quad ||\mu||_{TV} - \mathrm{Re}\int_0^1 \left(\sum_{k=-N}^{N} \overline{z_k} e^{-2\pi ikt}\right) d\mu(t).$$

The conclusion therefore follows from Proposition 2.16.

□

Bibliographie

P.-A. Absil, R. Mahony, and B. Andrews. Convergence of the iterates of descent methods for analytic cost functions. *SIAM Journal on Optimization*, 16(2) :531–547, 2005.

S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

E. J. Candès and X. Li. Solving quadratic equations via phaselift when there are about as many equations as unknowns. *Foundations of Computational Mathematics*, 14(5) :1017–1026, 2014.

E. J. Candès and T. Tao. Decoding by linear programming. *IEEE transactions on information theory*, 51(12) :4203–4215, 2005.

E. J. Candès and M. B. Wakin. An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2) :21–30, 2008.

E. J. Candès, J. K. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics : A Journal Issued by the Courant Institute of Mathematical Sciences*, 59(8) :1207–1223, 2006.

E. J. Candès, Y. C. Eldar, T. Strohmer, and V. Voroninski. Phase retrieval via matrix completion. *SIAM Journal on Imaging Sciences*, 6(1) :199–225, 2011.

E. J. Candès and C. Fernandez-Granda. Towards a mathematical theory of super-resolution. *Communications on Pure and Applied Mathematics*, 67 (6) :906–956, 2014.

E. J. Candès and Y. Plan. Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory*, 57(4) :2342–2359, 2011.

E. J. Candès, X. Li, and M. Soltanolkotabi. Phase retrieval via wirtinger flow : Theory and algorithms. *IEEE Transactions on Information Theory*, 61(4) :1985–2007, 2015.

Y. Carmon, J. C. Duchi, O. Hinder, and A. Sidford. Lower bounds for finding stationary points i. *Mathematical Programming*, 184(1-2) :71–120, 2020.

A. Chai, M. Moscoso, and G. Papanicolaou. Array imaging using intensity-only measurements. *Inverse Problems*, 27(1), 2011.

Y. Chen. Incoherence-optimal matrix completion. *IEEE Transactions on Information Theory*, 61(5) :2909–2923, 2015.

Y. Chen, Y. Chi, and A. Goldsmith. Exact and stable covariance estimation from quadratic sampling via convex programming. *IEEE Transactions on Information Theory*, 61(7) :4034–4059, 2015.

A. Conca, D. Edidin, M. Hering, and C. Vinzant. Algebraic characterization of injectivity in phase retrieval. *Applied and Computational Harmonic Analysis*, 32(2) :346–356, 2015.

Y. de Castro and F. Gamboa. Exact reconstruction using Beurling minimal extrapolation. *Journal of Mathematical Analysis and Applications*, 395 : 336–354, 2012.

C. Gao and A. Y. Zhang. Exact minimax estimation for phase synchronization. *IEEE Transactions on Information Theory*, 67(12) :8236–8247, 2021.

J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht. Gradient descent converges to minimizers. In *Proceedings of the Conference on Computational Learning Theory*, 2016.

C. Ma, K. Wang, Y. Chi, and Y. Chen. Implicit regularization in nonconvex statistical estimation : Gradient descent converges linearly for phase retrieval and matrix completion. In *International Conference on Machine Learning*, pages 3345–3354, 2018.

P. Netrapalli, P. Jain, and S. Sanghavi. Phase retrieval using alternating minimization. In *Advances in Neural Information Processing Systems 26*, pages 1796–2804, 2013.

B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3) :471–501, 2010.

W. Rudin. *Real and complex analysis, Third edition.* McGraw-Hill, 1987.

N. Sathyanarayanan, G. Cannone, L. Gakhar, N. Katagihallimath, R. Sowdhamini, S. Ramaswamy, and K. R. Vinothkumar. Molecular basis for metabolite channeling in a ring opening enzyme of the phenylacetate degradation pathway. *Nature communications*, 10(1), 2019.

A. Singer and Y. Shkolnisky. Three-dimensional structure determination from common lines in cryo-em by eigenvectors and semidefinite programming. *SIAM journal on imaging sciences*, 4(2) :543–572, 2011.

A. Singer and F. J. Sigworth. Computational methods for single-particle electron cryomicroscopy. *Annual review of biomedical data science*, 3 : 163–190, 2020.

J. Sun, Q. Qu, and J. Wright. A geometric analysis of phase retrieval. *Foundations of Computational Mathematics*, 18 :1131–1198, 2018.

J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on information theory*, 53(12) :4655–4666, 2007.

Y. Ye. Second order optimization algorithms i, 2015. http ://web.stanford.edu/class/msande311/2017lecture13.pdf.

A. Y. Zhang. Exact minimax optimality of spectral methods in phase synchronization and orthogonal group synchronization. *arXiv preprint arXiv :2209.04962*, 2022.