Mathematics of data

Irène Waldspurger

waldspurger@ceremade.dauphine.fr

September and October 2026

Contents

Chapter 1

Inverse problems

Some parts of this chapter follow [Gzyl and Velásquez, 2011].

What you should know / be able to do after this chapter

- Know the definition of the term "inverse problem", and a few examples.
- Know the definition of "uniqueness" and "stability" in the context of inverse problems.
- For a linear problem, determine whether it is stable or not by looking at the singular values of the corresponding matrix.
- Be able to explain the definition of the following terms: "Bayes estimator", "maximum a posteriori estimator", "data fidelity term" and "regularizer".
- Be able to compute the Bayes or maximum a posteriori estimator in simple cases (notably for linear problems).

1.1 Definition

An *inverse problem* consists in identifying a (possibly complicated) object from a set of observations[1]. For instance, if we are given (two-dimensional)

[1]Here, we will call *observation* any procedure which, from the object, produces an outcome.

photographs of a building, viewed from different angles, reconstructing a three-dimensional model of the building is an inverse problem. Here, the "object" is the 3D shape of the building and the set of observations is the set of photographs.

Mathematically, these problems are formalized as follows. Let E be the set of possible *objects*, and F the set of possible *observations*. The *observation procedure* is described by a function $M : E \to F$ (which may be deterministic or include some randomness). An inverse problem is, given some observation $y \in F$,

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

Oftentimes, y is not exactly equal to $M(x_{true})$, where x_{true} is the true underlying object which we want to observe. It can notably be contaminated with measurement noise. In this case, we will see that the equality "$M(x) = y$" should rather be replaced with an approximate equality, "$M(x) \approx y$", for a definition of "$\approx$" which must of course be specified.

Remark 1.1

The notion of *inverse problem* is often opposed to the notion of *direct problem*. A direct problem is the converse of an inverse problem: assuming the object and the observation procedure are known, compute the observations. 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, which amounts to solving a specific partial differential equation. Here, the object is the fluid, and the observation procedure is "let it flow for one minute, then look at it".

Remark 1.2

Some mathematicians call *inverse problems* a family of problems which is actually a subclass of inverse problems, according to our general definition. This family typically consists of inverse problems involving a PDE on a domain and measurements on the boundary.
For instance, in the so-called *Calderón problem* [Salo, 2008], the goal is to identify the electrical conductivity γ of a domain $\Omega \subset \mathbb{R}^d$. For

any $f \in H^{1/2}(\partial\Omega)$ (which represents an arbitrary electric potential on the boundary), we let u_f be the unique weak solution in $H^1(\Omega)$ to

$$\begin{aligned} \operatorname{div}(\gamma \nabla u) &= 0 \text{ in } \Omega, \\ u &= f \text{ on } \partial\Omega. \end{aligned}$$

We assume that, for any f, we can measure the normal derivative of u_f at the boundary of Ω,

$$\Lambda_\gamma f \stackrel{def}{=} \gamma \left. \frac{\partial u_f}{\partial \nu} \right|_{\partial\Omega}.$$

Can we identify γ?

First examples The most well-studied inverse problem is *denoising*, where $E = F$ is a vector space, $M = \mathrm{Id}_E$, and the measurements y, instead of being exact, are contaminated with noise. For instance, one may assume that y is the sum of the true object x_{true} and some random noise:

$$y = x_{true} + e,$$

where e is a random element in E, which may or may not be assumed independent from x.

Another classical example is *deconvolution*. Here, $E, F = L^2(\mathbb{R}^d)$ (or, possibly, some other vector spaces of functions from $\mathbb{R}^d$ to $\mathbb{R}$) and, for a fixed $g \in L^1(\mathbb{R}^d)$, the operator M is defined as

$$\begin{array}{rccc} M & : & L^2(\mathbb{R}^d) & \to & L^2(\mathbb{R}^d) \\ & & f & \to & f \star g \stackrel{def}{=} \left(x \in \mathbb{R}^d \to \int_{\mathbb{R}^d} f(z) g(x-z) dz\right). \end{array}$$

An important application field where deconvolution problems appear is optics. An optical instrument, like a microscope or telescope, usually creates some blur in the resulting image. In first approximation, the blur can be modeled as the convolution with a localized function g.

Observe that these two examples, as well as the Calderón problem from Remark 1.2, are linear: $M : E \to F$ is a linear map. Non-linear examples are provided in Section 1.4.

1.2 Uniqueness and stability

Let us for the moment consider a general problem of the form (Inverse) (with the exact equality "$M(x) = y$"), and discuss typical difficulties which may arise.

One may first wonder about the *existence* of solutions: for an arbitrary y, does there always exist a solution x to Problem (Inverse)? If we restrict ourselves to vectors y which are the outcome of a real measurement process (that is, of the form $y = M(x)$ for some x), the answer is obviously yes. But if some errors have occured in the process, the answer may not be obvious anymore.

Most importantly, assuming that a solution exists, to what extent does it coincide with the true object that we want to recover? A first, basic requirement for this is that the solution of Problem (Inverse) should be unique. Indeed, if there are two $x_1, x_2 \in E$ such that

$$M(x_1) = M(x_2) = y,$$

then we have no way to determine whether the true object is x_1 or x_2 (or something else). In other words, we require that M is injective. If this is the case, we say that Problem (Inverse) satisfies that *uniqueness* property.

But uniqueness is not enough. As we have already said, y is generally contaminated by noise, and not exactly equal to $M(x_{true})$. Nevertheless, we want the solution x to Problem (Inverse) to be close to x_{true}. This is called the *stability* property. There are several sensible, but not equivalent, ways to translate this informal requirement to a formal property. A standard one is to say that Problem (Inverse) is *stable* if there exists constants $C_1, C_2 > 0$ such that

$$\forall x_1, x_2 \in E, \quad C_1||x_1 - x_2||_F \leq ||M(x_1) - M(x_2)||_E \leq C_2||x_1 - x_2||_F, \quad (1.1)$$

and $\frac{C_2}{C_1}$ is "not too large" (say at most 10). Here, $||.||_E$ and $||.||_F$ are norms on E and F.[2]

When Problem (Inverse) satisfies uniqueness and stability, the natural next step is to design a concrete algorithm to compute its solution x, and this may be difficult as well. This is not our topic right now, but several algorithmic examples will probably be discussed later in this course.

[2]These norms must in principle be carefully chosen according to the physical structure of the concrete underlying problem. Some choices may reflect better than others the desired properties of the solutions.

1.2.1 Uniqueness and stability for linear problems

When the considered inverse problem is linear, uniqueness and stability can be characterized using standard linear algebra notions. For simplicity, we restrict ourselves to finite-dimensional problems. More precisely,

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

Under these assumptions, Problem (Inverse) rewrites as

$$\text{find } x \in \mathbb{R}^d \text{ such that } Ax = y. \qquad \text{(Linear inverse)}$$

For a given y, assuming a solution x_* exists, it is *unique* if and only if A is an injective matrix.

We now assume that the solution is unique. Is it *stable*? This property can be characterized using the singular value decomposition of A.

Proposition 1.3 : singular value decomposition

For any matrix $A \in \mathbb{R}^{m \times d}$, there exists $U \in O(m), V \in O(d)$ and $\lambda_1, \dots, \lambda_{\min(m,d)} \geq 0$ such that

$$A = UDV^T,$$

where D is the $m \times d$ diagonal matrix with coefficients $\lambda_1, \dots, \lambda_{\min(m,d)}$, i.e.

$$D = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \lambda_{\min(m,d)} & \\ & & & 0 \\ \vdots & & & \vdots \\ 0 & \dots & & 0 \end{pmatrix} \begin{matrix} \uparrow \\ \\ \\ \\ \\ \downarrow \end{matrix} m.$$

$$\underbrace{\qquad\qquad\qquad}_{d}$$

A triplet (U, D, V) is called a *singular value decomposition* of A. The numbers $\lambda_1, \dots, \lambda_{\min(m,d)}$ are called the *singular values* of A. They are uniquely defined up to permutation.

Proof. See Appendix A. □

Theorem 1.4

Let $\lambda_1, \dots, \lambda_{\min(m,d)}$ be the singular values of A, in nonincreasing order: $\lambda_1 \geq \cdots \geq \lambda_{\min(m,d)}$. Problem (Linear inverse) is stable (using the definition of stability from Equation (1.1)) if and only if

$$\frac{\lambda_1}{\lambda_{\min(m,d)}} \leq 10.$$

Proof. We first admit the following proposition, and use it to establish the theorem.

Proposition 1.5

Let $C \geq 0$ be fixed.
The following two properties are equivalent:

1. $\forall x_1, x_2 \in E, C||x_1 - x_2|| \leq ||Ax_1 - Ax_2||$;
2. $C \leq \lambda_{\min(m,d)}$.

Similarly, the following two properties are equivalent:

1. $\forall x_1, x_2 \in E, ||Ax_1 - Ax_2|| \leq C||x_1 - x_2||$;
2. $\lambda_1 \leq C$.

Using this propostion, we see that, if there exist $C_1, C_2 > 0$ satisfying Equation (1.1), then

$$\frac{C_2}{C_1} \geq \frac{\lambda_1}{\lambda_{\min(m,d)}},$$

so if $\frac{C_2}{C_1} \leq 10$, then $\frac{\lambda_1}{\lambda_{\min(m,d)}} \leq 10$ as well.

Conversely, if $\frac{\lambda_1}{\lambda_{\min(m,d)}} \leq 10$, then Equation (1.1) holds, with $C_1 = \lambda_{\min(m,d)}$ and $C_2 = \lambda_1$, and

$$\frac{C_2}{C_1} = \frac{\lambda_1}{\lambda_{\min(m,d)}} \leq 10.$$

Now, we prove the proposition. We can focus on the first equivalence: the second one is identical. Let $A = UDV^T$ be the singular value decomposition

of A. For any $x_1, x_2 \in E$,

$$\begin{aligned} & C||x_1 - x_2|| \leq ||Ax_1 - Ax_2|| \\ \iff & C||x_1 - x_2|| \leq ||U(DV^T x_1 - DV^T x_2)|| \\ \iff & C||V^T x_1 - V^T x_2|| \leq ||DV^T x_1 - DV^T x_2||. \end{aligned}$$

Therefore, using the change of variable $(x_1, x_2) \to (V^T x_1, V^T x_2)$, we see that the first property is equivalent to

$$\forall x_1, x_2 \in E, C||x_1 - x_2|| \leq ||Dx_1 - Dx_2||.$$

As a consequence, if this property is true, it must be true for $x_1 = e_{\min(m,d)}$ (the $\min(m,d)$-th vector of the canonical basis) and $x_2 = 0$, which implies

$$C = C||x_1 - x_2|| \leq ||Dx_1 - Dx_2|| = ||\lambda_{\min(m,d)} e_{\min(m,d)}|| = \lambda_{\min(m,d)}.$$

Conversely, if the property is not true, there must exist $x_1, x_2 \in E$ such that

$$C||x_1 - x_2|| > ||Dx_1 - Dx_2||.$$

For such x_1, x_2,

$$\begin{aligned} C^2||x_1 - x_2||^2 > ||Dx_1 - Dx_2||^2 & \\ &= \sum_{k=1}^{\min(m,d)} \lambda_k^2 (x_1 - x_2)_k^2 \\ &\geq \sum_{k=1}^{\min(m,d)} \lambda_{\min(m,d)}^2 (x_1 - x_2)_k^2 \\ &= \lambda_{\min(m,d)}^2 ||x_1 - x_2||^2, \end{aligned}$$

so $C > \lambda_{\min(m,d)}$. □

Example 1.6 : deconvolution

Let d, m be equal. Let $g \in \mathbb{R}^d$ be arbitrary. We consider for M the

(discrete) circular convolution operator associated to g, i.e.

$$\forall x \in \mathbb{R}^d, M(x) = g \star x = \left(\sum_{l=0}^{d-1} g_l x_{k-l} \right)_{k=0,\dots,d-1}.$$

(In the above formula, vectors are indexed starting at 0, and indices are considered modulo d.)
Let $F = \left(e^{-2\pi i k l}\right)_{0 \leq k,l \leq d-1}$ be the matrix associated to the discrete Fourier transform. The Fourier transform turns convolution into a product:

$$\forall x \in \mathbb{R}^d, \widehat{M(x)} = (\hat{g}_k \hat{x}_k)_{k=0,\dots,d-1}.$$

Therefore, the matrix A associated to M is

$$A = F^{-1} \begin{pmatrix} \hat{g}_0 & & \\ & \ddots & \\ & & \hat{g}_d \end{pmatrix} F = \frac{F^*}{\sqrt{n}} \begin{pmatrix} n\hat{g}_0 & & \\ & \ddots & \\ & & n\hat{g}_{d-1} \end{pmatrix} \frac{F}{\sqrt{n}}.$$

The inverse problem

$$\text{find } x \in \mathbb{R}^d \text{ such that } x \star g = y$$

satisfies the uniqueness property if and only if A is injective, i.e.

$$\hat{g}_k \neq 0, \forall k = 0, \dots, d-1.$$

What about stability? As $\frac{F}{\sqrt{n}}$ is a unitary matrix[a], the singular values of A are the same as the singular values of

$$\begin{pmatrix} n\hat{g}_0 & & \\ & \ddots & \\ & & n\hat{g}_{d-1} \end{pmatrix},$$

which are $n|\hat{g}_0|, n|\hat{g}_1|, \dots, n|\hat{g}_d|$, since

$$\begin{pmatrix} n\hat{g}_0 & & \\ & \ddots & \\ & & n\hat{g}_{d-1} \end{pmatrix} = \begin{pmatrix} e^{i \operatorname{phase}(\hat{g}_0)} & & \\ & \ddots & \\ & & e^{i \operatorname{phase}(\hat{g}_{d-1})} \end{pmatrix} \begin{pmatrix} n|\hat{g}_0| & & \\ & \ddots & \\ & & n|\hat{g}_{d-1}| \end{pmatrix} (I_d)^T.$$

From Theorem 1.4, the deconvolution problem is thus stable if

$$\frac{\max_k |\hat{g}_k|}{\min_k |\hat{g}_k|} \leq 10.$$

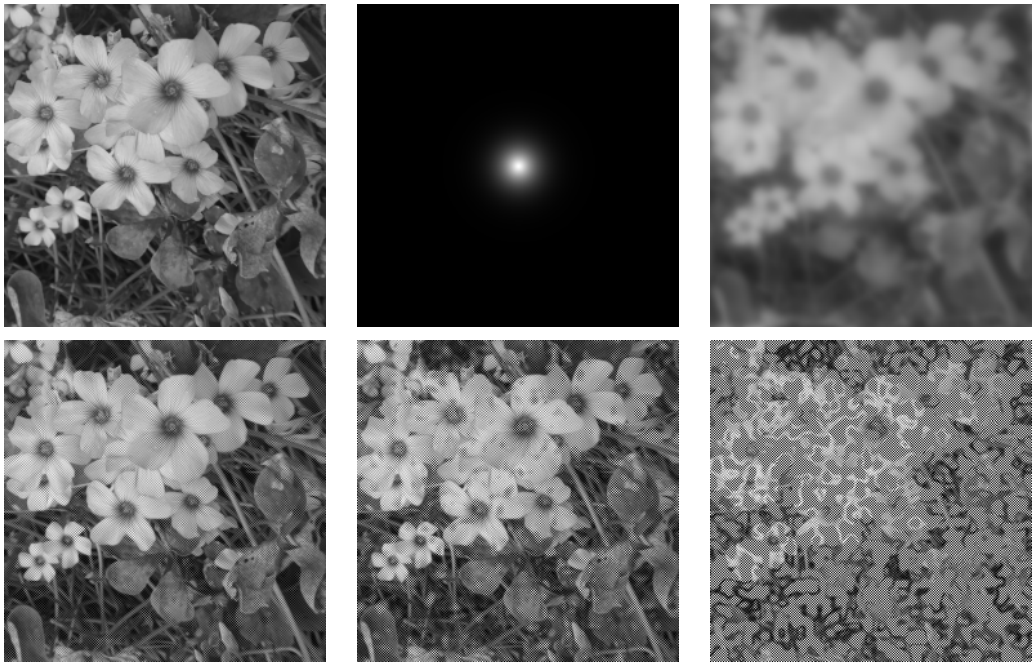

Figure 1.1: Example of a deconvolution problem.
Top row: a signal $x_{true} \in \mathbb{R}^{250\times250}$, a blurring filter $g \in \mathbb{R}^{250\times250}$ (in the Fourier domain) and their convolution $x_{true} \star g$.
Bottom row: the solution x to (Inverse) for $y = x_{true} \star g + e$, where e is a realization of a standard random vector, with norm chosen so that $||e|| = 10^{-8}||x_{true} \star g||$ for the left image, $||e|| = 3.10^{-8}||x_{true} \star g||$ for the middle image and $||e|| = 10^{-7}||x_{true} \star g||$ for the right image.

When g has the effect of blurring the signal x, the coefficients of $\hat{g}$ are typically negligible, except at low frequencies ($k \approx 0$ or $k \approx d$). Therefore, the inequality above has no chance to hold. An example can be found on Figure 1.1, where it is clear that the problem is so instable that the solution of (Inverse) is typically very different from the true signal, even for very low values of noise.

[a]Note that, here, the matrices have complex coefficients, while Proposition 1.3 was about real matrices. However, everything in Proposition 1.3 is true for complex matrices as well, replacing the orthogonal matrices with unitary ones

1.3 Using priors

Ici, j'écrirai une introduction qui explique qu'à cause de l'instabilité, c'est important d'utiliser un prior.

1.3.1 Bayes estimator

In the Bayesian framework, one assumes that the unknown signal $x \in E$ is a realization of a random vector X, while the observation $y \approx M(x)$ is a realization of Y (where Y is of course not independent from X). The law of X is called the *prior* on the signal.

Let us imagine that we have chosen a function $L : E^2 \to \mathbb{R}$, called a *loss*, such that $L(x, x')$ quantifies how problematic it is to reconstruct an object x' in place of the true object x. When reconstructing an approximation of x from y, it seems a natural goal to try to minimize this loss, in expectation. Therefore, when given some $y \in F$, one should estimate x as $\hat{x}(y)$, with

$$\hat{x}(y) \in \operatorname{argmin}_{x'} \mathbb{E}(L(X, x')|Y = y).$$

An estimator $\hat{x}$ which satisfies this property is called a *Bayes estimator*.

Quadratic loss A particularly important case is when E is a Hilbert space and L is the quadratic loss:

$$\forall x, x', \quad L(x, x') = \frac{1}{2}||x - x'||^2.$$

A Bayes estimator must be such that, for all y, $\hat{x}(y)$ belongs to

$$\begin{aligned}
&\operatorname{argmin}_{x'} \frac{1}{2}\mathbb{E}\left(||X - x'||^2|Y = y\right) \\
&\quad = \operatorname{argmin}_{x'} \frac{1}{2}\left(\mathbb{E}\left(||X||^2|Y = y\right) - 2\mathbb{E}\left(\langle X, x'\rangle |Y = y\right) + ||x'||^2\right) \\
&\quad = \operatorname{argmin}_{x'} \frac{1}{2}\left(||\mathbb{E}(X|Y = y)||^2 - 2\mathbb{E}\left(\langle X, x'\rangle |Y = y\right) + ||x'||^2\right) \\
&\qquad \text{(the first term does not depend on } x' \text{ and thus does not} \\
&\qquad \text{change the argmin)} \\
&\quad = \operatorname{argmin}_{x'} \frac{1}{2}\left(||\mathbb{E}(X|Y = y)||^2 - 2\left\langle \mathbb{E}\left(X|Y = y\right), x'\right\rangle + ||x'||^2\right)
\end{aligned}$$

$$= \text{argmin}_{x'} \frac{1}{2}||\mathbb{E}(X|Y=y) - x'||^2$$
$$= \{\mathbb{E}(X|Y=y)\}.$$

Therefore, $\hat{x}(y) = \mathbb{E}(X|Y=y)$ (which is the *posterior mean* of X given y).

When X, Y do not have very simple distributions, this estimator is often hard to compute. In addition, its computation requires the laws of X, Y to be known, which is rarely the case in practice. Nevertheless, its value in very simple cases, as the one from the following example, already provides some intuition on what a reasonable estimator should look like.

Example 1.7

Let us assume that $E = \mathbb{R}^m$, for some $m \in \mathbb{N}^*$, and X follows a normal distribution: $X \sim \mathcal{N}(0, I_m)$.
We consider the denoising problem, where $M = \text{Id}_{\mathbb{R}^m}$, and the observation y is a realization of the random variable

$$Y = X + Z,$$

where Z follows a normal distribution independent from X, $Z \sim \mathcal{N}(0, \sigma^2 I_m)$, for some $\sigma > 0$.

To compute the posterior mean, we observe that $X' \overset{def}{=} X - \frac{Y}{1+\sigma^2}$ is a Gaussian random variable (and jointly Gaussian with Y), whose expectation is 0, and such that

$$\begin{aligned}\mathbb{E}(X'Y^T) &= \mathbb{E}(XY^T) - \frac{\mathbb{E}(YY^T)}{1+\sigma^2} \\ &= \mathbb{E}(XX^T) - \frac{(1+\sigma^2)I_m}{1+\sigma^2} \\ &= 0.\end{aligned}$$

Therefore, X' is independent from Y, and

$$\mathbb{E}(X|Y) = \mathbb{E}\left(\frac{Y}{1+\sigma^2} + X' \middle| Y\right) = \frac{Y}{1+\sigma^2}.$$

This entails that the Bayes estimator is

$$\hat{x}(y) = \frac{y}{1+\sigma^2}, \quad \forall y \in \mathbb{R}^m.$$

1.3.2 Maximum a Posteriori estimator

Let us stay in the Bayesian setting, but assume in addition that $E = \mathbb{R}^m$, $F = \mathbb{R}^d$, and all involved laws have densities. We denote $p_X : \mathbb{R}^m \to \mathbb{R}, p_Y : \mathbb{R}^d \to \mathbb{R}, p_{X,Y} : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}$ the respective densities of $X, Y, (X, Y)$.

À partir de là, le texte est un copié-collé de notes issues d'un autre cours ; je vais le reprendre.

Let us imagine that we want to recover some vector $x \in \mathbb{R}^d$ from an observation $y \in \mathbb{R}^m$ which depends on x in a possibly non-deterministic manner. We assume that we know the probability distribution of x in $\mathbb{R}^d$, which is a law with density $p : \mathbb{R}^d \to \mathbb{R}$ (called a *prior*), and the distribution of y given x, which has density $p(.|x)$. The joint law of (x, y) is a probability distribution with density

$$\begin{array}{rccc} p & : & \mathbb{R}^d \times \mathbb{R}^m & \to & \mathbb{R} \\ & & (x, y) & \to & p(y|x)p(x). \end{array}$$

Given some specific realization y_* of y, the MAP estimator is the signal x which has maximum probability conditioned to the fact that $y = y_*$. In formulas,

$$\begin{aligned} x_{MAP} &\in \operatorname*{argmax}_{x \in \mathbb{R}^d} p(x|y = y_*) \\ &= \operatorname*{argmax}_{x \in \mathbb{R}^d} \frac{p(x, y_*)}{p(y_*)} \\ &= \operatorname*{argmax}_{x \in \mathbb{R}^d} p(x, y_*) \\ &= \operatorname*{argmax}_{x \in \mathbb{R}^d} p(y_*|x)p(x). \end{aligned}$$

1.3.3 Regularization

If we replace the last function in the argmax by its negative logarithm (and the maximization by a minimization), we can also write

$$x_{MAP} \in \operatorname*{argmin}_{x \in \mathbb{R}^d} \underbrace{(-\log p(y_*|x))}_{\stackrel{def}{=} L(y_*, x)} + \underbrace{(-\log p(x))}_{\stackrel{def}{=} R(x)}. \tag{1.2}$$

The map L is called the *data fidelity term*, while R is the *regularizer*.

In practice, $p(.|.)$ and p are usually not exactly known, so L and R are set to standard functions, known to enforce desirable properties on the signal x and the pair (x, y), even if they are not the negative logarithm of reasonable probability distributions, and x is reconstructed by solving

$$\min_{x \in \mathbb{R}^d} L(y_*, x) + R(x). \tag{1.3}$$

1.4 More examples

Je vais raccourcir cette section. De toute façon, à part peut-être le début, je ne la couvrirai pas en cours.

1.4.1 Sparse recovery - compressed sensing

Our first example 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 $A \in \mathbb{R}^{m \times d}$ is a known matrix, under the assumption that x is *sparse*. The word *sparse* means that x has a small number of non-zero coordinates: for some $k \in \mathbb{N}^*$ much smaller than d,

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

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

Note that, if $m \geq d$ and A is injective, then this problem can be solved by inverting A; it is not necessary to use the sparsity assumption. This problem is only interesting when m is much smaller than d, in which case A is not injective and, if we were to ignore the sparsity assumption, y would not uniquely determine x.

Assuming that k is known, the problem can be written as

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

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 1.8 : 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.

Proof. 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]).

1.4.2 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

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

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. Given that any $d_1 \times d_2$ matrix has rank at most $\min(d_1, d_2)$, the rank constraint is only useful if $r < \min(d_1, d_2)$. In some applications, it is relevant to assume that $d_1 = d_2$ and 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 (1.8) holds.

Proposition 1.9 : uniqueness for low-rank matrix recovery

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

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

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

The proof of the proposition is identical to Proposition 1.8. 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

$$\begin{aligned} m &\geq 2r(d_1 + d_2 - 2r) \qquad \text{if } 2r \leq \min(d_1, d_2), \\ &\geq d_1 d_2 \qquad \text{if } \min(d_1, d_2) < 2r < 2\min(d_1, d_2). \end{aligned}$$

When r is small (of order 1, for instance), this shows 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 than $2r(d_1 + d_2 - 2r)$ 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 a subset of coefficients:

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

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".[3] 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:[4] 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

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

[3] asked by Netflix in 2006, with a $1,000,000$\$ prize, and declared solved in 2009

[4] Keep however in mind that this assumption is only approximately satisfied by the "true" Netflix affinity scores matrix. On the other hand, the true matrix has additional structure that can be exploited to solve the problem.

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 1.10

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 1.11 : [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. Readers which are not perfectly comfortable with the notions of

Hermitian matrices and of semidefinite positive matrices should first read Appendix **??**.

The crucial ingredient is an adequate change of variable: instead of recovering $x \in \mathbb{C}^d$ up to a global phase, let us try to recover

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

Remark 1.12

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

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

When these conditions hold, x is equal, up to a global phase, to

$$\sqrt{\lambda_1} z_1,$$

where λ_1 is the largest eigenvalue of X, and z_1 any unit-normed eigenvector for this eigenvalue.

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

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

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

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

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

All the eigenvalues are nonnegative, since $X \succeq 0$. Since $\operatorname{rank}(X) \leq 1$, 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 $x = \sqrt{\lambda_1} z_1$. This proves the first part of the remark.

For the second part, let us assume that $X = xx^*$ for some $x \in \mathbb{C}^d$. We have just seen that X is also equal to $\tilde{x}\tilde{x}^*$ for $\tilde{x} = \sqrt{\lambda_1} z_1$. We must simply show that x and $\tilde{x}$ are equal up to a global phase. As

$$\text{Vect}\{x\} = \text{Range}(X) = \text{Vect}\{\tilde{x}\},$$

it holds that x and $\tilde{x}$ are colinear: there exists $u \in \mathbb{C}$ such that $x = u\tilde{x}$. In addition,

$$||x||^2 = \text{Tr}(X) = ||\tilde{x}||^2,$$

hence x and $\tilde{x}$ have the same norm. As $||x|| = |u|\,||\tilde{x}||$, this implies that $|u| = 1$: x and $\tilde{x}$ are equal up to a global phase. □

From the previous remark, it is equivalent to recover x up to a global phase or X. 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.

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

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

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

1.4.3 Other examples

These examples will not be covered in class (except for super-resolution, but later); they are provided for curious readers only.

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

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

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 (i.e. which *generalizes* best)

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:

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

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

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

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.[5]

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

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

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

[5] 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.

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).

Appendix A

Proof of Proposition 1.3

Proof. Let $A \in \mathbb{R}^{m\times d}$ be fixed. Let us assume $m \geq d$. The case $m < d$ can be deduced from the case $m \geq d$ by applying the already established result to $A^T \in \mathbb{R}^{d\times m}$ in place of A.

We first show the existence of the singular value decomposition. Then, we will prove the uniqueness of the singular values.

The matrix A^TA is symetric and positive semidefinite. Therefore, there exists $V \in O(d)$ and $\mu_1, \dots, \mu_d \geq 0$ such that

$$A^TA = V \begin{pmatrix} \mu_1 & & \\ & \ddots & \\ & & \mu_d \end{pmatrix} V^T.$$

We fix such $V, \mu_1, \dots, \mu_d$. Up to reordering the columns of V, we can assume that $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_d$.

Let $k \in \{1, \dots, d\}$ be such that $\mu_1, \dots, \mu_k \neq 0$, $\mu_{k+1} = \cdots = \mu_d = 0$. If we write

$$AV = \begin{pmatrix} A_1 & A_2 \end{pmatrix} \text{ with } A_1 \in \mathbb{R}^{m\times k}, A_2 \in \mathbb{R}^{m\times(d-k)},$$

we have that

$$\begin{pmatrix} A_1^TA_1 & A_1^TA_2 \\ A_2^TA_1 & A_2^TA_2 \end{pmatrix} = V^TA^TAV = \begin{pmatrix} \mu_1 & & & & \\ & \ddots & & & \\ & & \mu_k & & \\ & & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{pmatrix}.$$

In particular, $A_2^TA_2 = 0$, so $A_2 = 0$. Also,

$$A_1^TA_1 = \begin{pmatrix} \mu_1 & & \\ & \ddots & \\ & & \mu_k \end{pmatrix},$$

so that, if we define $\tilde{U} = A_1 \begin{pmatrix} \mu_1^{-1/2} & & \\ & \ddots & \\ & & \mu_k^{-1/2} \end{pmatrix}$, then we have that

$$\tilde{U}^T \tilde{U} = I_k,$$

meaning that the columns of $\tilde{U}$ are unit-normed and orthonormal.

Let U be any orthogonal matrix whose first k columns coincide with the columns of $\tilde{U}$.

It holds

$$\begin{aligned} AV &= \begin{pmatrix} A_1 & 0_{m\times(d-k)} \end{pmatrix} \\ &= \left(\tilde{U} \begin{pmatrix} \mu_1^{1/2} & & \\ & \ddots & \\ & & \mu_k^{1/2} \end{pmatrix} \quad 0_{m\times(d-k)} \right) \\ &= U \begin{pmatrix} \mu_1^{1/2} & \dots & 0 \\ & \ddots & \\ & & \mu_k^{1/2} \\ & & 0 \\ \vdots & & \ddots \\ & & 0 \\ & & \vdots \\ 0 & \dots & 0 \end{pmatrix} \\ &= UD \end{aligned}$$

if we set $\lambda_1 = \mu_1^{1/2}, \dots, \lambda_k = \mu_k^{1/2}, \lambda_{k+1} = \dots = \lambda_d = 0$ and define D as in the statement of the proposition. This implies $A = UDV^T$, and we have proved the existence of the singular value decomposition.

To show that the singular values are uniquely defined up to permutation, we note that, if (U, D, V) is a singular value decomposition of A, then

$$A^T A = VD^T DV^T = V \begin{pmatrix} \lambda_1^2 & & \\ & \ddots & \\ & & \lambda_d^2 \end{pmatrix} V^T.$$

Consequently, $\lambda_1^2, \dots, \lambda_d^2$ are the eigenvalues of $A^T A$ (with multiplicity); these numbers are therefore uniquely defined. Since $\lambda_1, \dots, \lambda_d$ are nonnegative, these numbers are also uniquely defined: they are the square roots of the eigenvalues of $A^T A$. □

Bibliography

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.

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

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. 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.

H. Gzyl and Y. Velásquez. *Linear Inverse Problems: The Maximum Entropy Connection (with CD-ROM)*, volume 83. World Scientific, 2011.

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.

K. G. Murty and S. N. Kabadi. Some np-complete problems in quadratic and nonlinear programming. *Mathematical Programming: Series A and B*, 39(2):117–129, 1987.

A.-A. Muşat and N. Boumal. Gradient descent avoids strict saddles with a simple line-search method too. *arXiv preprint arXiv:2507.13804*, 2025.

I. Panageas and G. Piliouras. Gradient descent only converges to minimizers: Non-isolated critical points and invariant regions. In *Innovations in Theoretical Computer Science*, 2017.

M. Salo. Calderón problem. *Lecture Notes*, 2008.

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