Optimization and Computational Linear Algebra for Data Science
Lecture 10: Linear regression

Léo Miolane · leo.miolane@gmail.com

November 17, 2019

Warning: *This material is not meant to be lecture notes. It only gathers the main concepts and results from the lecture, without any additional explanation, motivation, examples, figures...*

1 Least squares

Assume that we are given points $a_i = (a_{i,1}, \dots, a_{i,d}) \in \mathbb{R}^d$ with labels $y_i \in \mathbb{R}$ for $i = 1 \dots n$. We aim at finding a vector $x \in \mathbb{R}^d$ such that

$$y_i \simeq \langle a_i, x \rangle = \sum_{j=1}^{d} a_{i,j} x_j, \qquad \text{for} \quad i = 1 \dots n.$$

If we denote by A the $n \times d$ matrix whose rows are $a_1, \dots, a_n$, i.e. $A_{i,j} = a_{i,j}$, we are looking for some x such that $Ax \simeq y$.

In general, solutions to $Ax = y$ may not exist: there is no reason for y to belong to $\mathrm{Im}(A)$, especially when $n > d$. (Exercise: why?) Moreover, the measurements may contain some noise. Therefore one is rather interested by solving

$$\text{minimize} \quad \|Ax - y\|^2, \quad \text{with respect to} \quad x \in \mathbb{R}^d. \tag{1}$$

The function $f : x \mapsto \|Ax - y\|^2$ is convex (Exercise: why?) and differentiable. Hence x is solution of (1) if and only if $\nabla f(x) = 0$. Compute

$$f(x) = (Ax - y)^\mathsf{T}(Ax - y) = x^\mathsf{T} A^\mathsf{T} A x - 2y^\mathsf{T} A x + \|y\|^2.$$

Hence $\nabla f(x) = 2A^\mathsf{T} A x - 2A^\mathsf{T} y$. We conclude

$$x \ \text{ is solution of } \ (1) \qquad \Longleftrightarrow \qquad A^\mathsf{T} A x = A^\mathsf{T} y.$$

If $A^\mathsf{T} A$ is invertible there is a unique minimizer $x^* = (A^\mathsf{T} A)^{-1} A^\mathsf{T} y$. In general $A^\mathsf{T} A$ may not be invertible. We see that the solutions of (1) are the solutions of the linear system $A^\mathsf{T} A x = A^\mathsf{T} y$. Let $A = U \Sigma V^\mathsf{T}$ be the singular value decomposition of A. Using the SVD, the linear system can be rewritten as

$$V \Sigma^\mathsf{T} \Sigma V^\mathsf{T} x = V \Sigma^\mathsf{T} U^\mathsf{T} y \qquad \text{which is equivalent to} \qquad \Sigma^\mathsf{T} \Sigma V^\mathsf{T} x = \Sigma^\mathsf{T} U^\mathsf{T} y \tag{2}$$

because V is invertible (recall that V is orthogonal). We will now introduce:

Definition 1.1 (*Moore-Penrose pseudo-inverse*)

The matrix $A^\dagger \overset{\text{def}}{=} V \Sigma' U^\mathsf{T}$ is called the (Moore-Penrose) pseudo-inverse of A, where Σ' is the $d \times n$ matrix given by

$$\Sigma'_{i,i} = \begin{cases} 1/\Sigma_{i,i} & \text{if } \Sigma_{i,i} \neq 0, \\ 0 & \text{otherwise,} \end{cases}$$

and $\Sigma'_{i,j} = 0$ for $i \neq j$.

Now, one can check that $x = A^\dagger y$ is a solution of (2):

$$\Sigma^\mathsf{T}\Sigma V^\mathsf{T} A^\dagger y = \Sigma^\mathsf{T}\Sigma\Sigma' U^\mathsf{T} y = \Sigma^\mathsf{T} U^\mathsf{T} y.$$

Hence, the set of solutions of the linear system $A^\mathsf{T}Ax = A^\mathsf{T}y$ is $A^\dagger y + \mathrm{Ker}(A^\mathsf{T}A)$. Notice now that (exercise!) we have $\mathrm{Ker}(A^\mathsf{T}A) = \mathrm{Ker}(A)$. We conclude:

Proposition 1.1 (*Least squares*)

The set of solution of the minimization problem $\min_{x\in\mathbb{R}^n} \|Ax - y\|^2$ *is*

$$A^\dagger y + \mathrm{Ker}(A).$$

As a byproduct of our analysis, we deduce:

Corollary 1.1

The set of solution of the linear system $Ax = y$ *is*

- $\emptyset$ *if* $y \notin \mathrm{Im}(A)$.
- $A^\dagger y + \mathrm{Ker}(A)$ *otherwise.*

2 Penalized least squares: Ridge regression and Lasso

When $\mathrm{Ker}(A) \neq \{0\}$ the least squares problem (1) has an infinite number of solutions: which one should we pick?

2.1 Ridge regression

The Ridge regression adds a ℓ_2 penalty to the least square problem in order to "select" a solution of smaller norm, and minimizes

$$\min_{x\in\mathbb{R}^d} \Big\{ \|Ax - y\|^2 + \lambda \|x\|^2 \Big\}, \tag{3}$$

for some penalization parameter $\lambda > 0$.

Exercise 2.1. *Show that* (3) *admits a unique solution given by*

$$x^{\mathrm{Ridge}} = (A^\mathsf{T}A + \lambda \mathrm{Id})^{-1} A^\mathsf{T} y.$$

2.2 Lasso

The Lasso adds a ℓ_1 penalty to the least square problem, and minimizes

$$\frac{1}{2}\|Ax - y\|^2 + \lambda \|x\|_1, \tag{4}$$

for some penalization parameter $\lambda > 0$. Here, the cost function is not strictly convex in general, hence there may be multiple minimizers. However, in many situations (for instance if the entries of A are drawn from a continuous probability distribution) the minimizer x^{Lasso} will be unique [4].

The Lasso has the wonderful property of *feature selection*: the solution x^{Lasso} of (4) is likely to be *sparse* (many coordinates x_j^{Lasso} will be set to 0). In words, the Lasso estimator discards the "useless features" by setting their corresponding coefficient to 0. This is particularly nice for the

interpretability of the results: in many application, each data point a_i has an enormous number of features (d very large), but only a small number of them are useful to predict the label y_i.

We will now give some intuition on this phenomena. First, we have the following

Lemma 2.1

Let x^{Lasso} be a minimizer of the Lasso cost function (4) and let $r = \|x^{\text{Lasso}}\|_1$. Then x^{Lasso} is a solution to the constrained optimization problem:

$$\text{minimize} \quad \frac{1}{2}\|Ax - y\|^2 \quad \text{subject to} \quad \|x\|_1 \leq r.$$

Proof. By contradiction, assume that there exists $x \in \mathbb{R}^d$ such that $\|Ax - y\|^2 < \|Ax^{\text{Lasso}} - y\|^2$ and $\|x\|_1 \leq r = \|x^{\text{Lasso}}\|_1$. Then

$$\frac{1}{2}\|Ax - y\|^2 + \lambda\|x\|_1 < \frac{1}{2}\|Ax^{\text{Lasso}} - y\|^2 + \lambda\|x^{\text{Lasso}}\|_1,$$

which contradicts the fact that x^{Lasso} is a minimizer of (4). □

Lemma 2.1 tells us that a solution x^{Lasso} of the Lasso problem is a minimizer of the quadratic function $\|Ax - y\|^2$ on the ℓ_1-ball $B_{\ell_1}(r) = \{x \,|\, \|x\|_1 \leq r\}$.

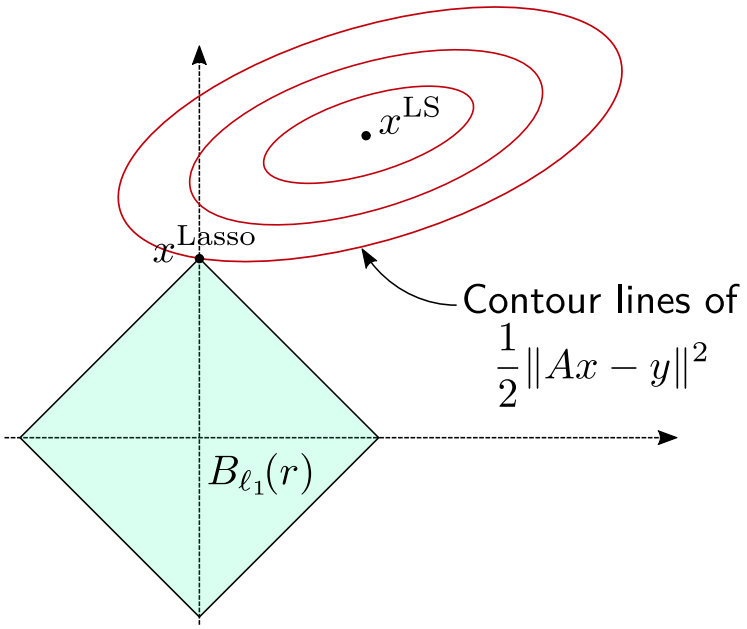

Figure 1

This is illustrated on Figure 1 above. The Lasso solution x^{Lasso} is the first point at which the elliptical contour lines of the function $\frac{1}{2}\|Ax - y\|^2$ hit the ℓ_1-ball $B_{\ell_1}(r)$. Observe that the ℓ_1-ball has corners: x^{Lasso} is therefore very likely to be at one of these corners which is a point where many coordinates are set to zero. Hence, the ℓ_1 penalization tends to produce sparse solutions.

Lasso for orthonormal design. We will show by a computation below that the ℓ_1 penalization induces sparsity, on a toy example. In general there is no closed form formula for the Lasso estimator. It is however possible to derive one in the case where the matrix A has orthonormal

columns: $A^\mathsf{T} A = \mathrm{Id}$. The least square estimator is then given by $x^{\mathrm{LS}} = A^\dagger y = A^\mathsf{T} y$ and the Lasso cost function becomes

$$\frac{1}{2}\|Ax - y\|^2 + \lambda\|x\|_1 = \frac{1}{2}\|x\|^2 - \langle x^{\mathrm{LS}}, x\rangle + \frac{1}{2}\|y\|^2 + \lambda\|x\|_1. \tag{5}$$

The minimizer of (5) is therefore the minimizer of

$$\frac{1}{2}\|x\|^2 - \langle x^{\mathrm{LS}}, x\rangle + \lambda\|x\|_1 = \sum_{j=1}^{d} \frac{x_j^2}{2} - x_j^{\mathrm{LS}} x_j + \lambda|x_j| = \sum_{j=1}^{d} f_{x_j^{\mathrm{LS}}}(x_j),$$

where $f_{x_0}(x) \stackrel{\mathrm{def}}{=} \frac{1}{2}x^2 - x_0 x + \lambda|x|$.

Exercise 2.2. *Show that the function f_{x_0} admits a unique minimizer given by*

$$x^* = \eta(x_0; \lambda),$$

where η denotes the "soft-thresholding" function:

$$\eta(x_0;\lambda) = \begin{cases} x_0 - \lambda & \text{if} \quad x_0 \geq \lambda \\ 0 & \text{if} \quad -\lambda \leq x_0 \leq \lambda \\ x_0 + \lambda & \text{if} \quad x_0 \leq -\lambda. \end{cases}$$

We conclude that the Lasso estimator is given by

$$x_j^{\mathrm{Lasso}} = \eta(x_j^{\mathrm{LS}}; \lambda) \qquad \text{for} \quad j = 1, \ldots, d.$$

This formula confirms the intuition given by Figure 1: the Lasso estimator translates the coefficients of the least-square solution by λ, truncating at 0.

3 Norms for matrices, application to matrix completion

Before looking at low-rank matrix estimation and matrix completion, we need to look at different norms we can have for matrices. Recall that $\mathbb{R}^{n\times m}$, the set of $n \times m$ matrices, is a vector space (of dimension nm).

3.1 Matrix norms

The Frobenius norm. The most obvious norm to consider is the equivalent of the ℓ_2 norm, called the Frobenius norm:

Definition 3.1 (*Frobenius norm*)

The Frobenius norm of a matrix $A \in \mathbb{R}^{n\times m}$ is defined as

$$\|A\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m} A_{i,j}^2}.$$

Remark 3.1. *The Frobenius norm is the norm induced by the following inner-product:*

$$\langle A, B\rangle_F = \mathrm{Tr}(A^\mathsf{T} B) = \sum_{i=1}^{n}\sum_{j=1}^{m} A_{i,j}B_{i,j}, \qquad \text{for} \quad A, B \in \mathbb{R}^{n\times m}.$$

Proposition 3.1

Let $A \in \mathbb{R}^{n\times n}$ and let $\sigma_1, \dots, \sigma_{\min(n,m)}$ be the singular values of A. Then

$$\|A\|_F = \sqrt{\sum_{i=1}^{\min(n,m)} \sigma_i^2}.$$

The spectral norm. Another extremely common norm for matrices is the spectral norm, also called the operator norm:

Definition 3.2 (*Spectral norm*)

The spectral norm of a matrix $A \in \mathbb{R}^{n\times m}$ is defined as the maximal singular value of A:

$$\|A\|_{\mathrm{Sp}} = \max_{\substack{x\in\mathbb{R}^m \\ \|x\|=1}} \|Ax\|.$$

Proposition 3.2

$$\|A\|_{\mathrm{Sp}} = \sigma_1(A),$$

where $\sigma_1(A)$ denotes the largest singular value of A.

The nuclear norm. We have seen above that the Frobenius norm of a matrix correspond to the ℓ_2 norm of its singular values and that the spectral norm correspond to the infinity norm of the singular values. The nuclear norm is simply the ℓ_1 norm of its singular values:

Definition 3.3 (*Nuclear norm*)

Let $A \in \mathbb{R}^{n\times m}$ and let $\sigma_1(A), \dots, \sigma_{\min(n,m)}(A)$ be the singular values of A. The nuclear norm of A is defined as:

$$\|A\|_* = \sum_{i=1}^{\min(n,m)} \sigma_i(A).$$

3.2 Application to matrix completion

The matrix completion problem can be formulated as follows. We have a data matrix $M \in \mathbb{R}^{n\times m}$ that we only observe partially. That is we only have access to

$$M_{i,j} \quad \text{for } (i,j) \in \Omega,$$

where $\Omega \subset \{1, \dots, n\} \times \{1, \dots m\}$ is a subset of the complete set of the entries. The goal of matrix completion is to recover the matrix M entirely.

Such problems appears for instance in the context of movie recommendation systems, where there are n users and m movies, and $M_{i,j}$ represent the rating of movie j by the user i. Of course, we do not have access to the entire matrix M since each user has seen/rated only a fraction of all the movies. One would like therefore to complete the entries of M in order to predict if a given user will like or not a movie, which would allow to make recommendation to users.

A very natural assumption is to assume that M has a small rank. This makes a lot of sense for recommendation systems, since it is reasonable to assume that only a few factors influence the preference of an user.

Hence, one would be interested to solve:

$$\text{minimize} \quad \text{rank}(X) \quad \text{subject to} \quad X \in \mathbb{R}^{n\times m} \text{ and } X_{i,j} = M_{i,j} \text{ for all } (i,j) \in \Omega. \tag{6}$$

However, the above minimization problem is in general NP-hard, making it impossible to solve in reasonable time.

The paper [2] proposed to replace the problem (6) by the following convex minimization problem.

$$\text{minimize} \quad \|X\|_* \quad \text{subject to} \quad X \in \mathbb{R}^{n\times m} \text{ and } X_{i,j} = M_{i,j} \text{ for all } (i,j) \in \Omega. \tag{7}$$

This problem can be solved efficiently, since we have here to minimize a convex function over a convex set. However, it is a priori not clear if the solution of this new optimization problem will have a small rank.

We have seen in Section 2.2 that the ℓ_1-penalization encourages sparsity. In the same spirit, penalization by the nuclear norm $\|\cdot\|_*$ will tend to produce low-rank solutions. Indeed, the nuclear norm of a matrix X is by definition the ℓ_1-norm of the singular values of X:

$$\|X\|_* = \sum_{i=1}^{\min(n,m)} \sigma_i(X).$$

Heuristically, nuclear norm minimization will therefore encourage the vector of singular values $(\sigma_1(X), \ldots, \sigma_{\min(n,m)}(X))$ to be sparse. Recall now that the rank of X is equal to the number of non-zero singular values, we see that nuclear norm minimization tend to produce low-rank solutions.

Further reading

Chapter 3 of [3] is an excellent reference for linear regression. See [2, 1] for further developments on nuclear norm minimization for matrix completion.

References

[1] Emmanuel J Candes and Yaniv Plan. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936, 2010.

[2] Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717, 2009.

[3] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

[4] Ryan J Tibshirani et al. The lasso problem and uniqueness. *Electronic Journal of Statistics*, 7:1456–1490, 2013.