High-dimensional statistics (The LASSO)

Pierre Gaillard

January 2020

1 Introduction

In statistics or machine learning, we often want to explain some output $Y \in \mathcal{Y}$ from input $X \in \mathcal{X} \subset \mathbb{R}^p$ by observing a data set $D_n = \{(X_i, Y_i)\}_{1 \leq i \leq n}$ of i.i.d. observations. In this lecture, we would like to deal with high-dimensional input spaces, i.e., large p (possibly $p \gg n$). We will have two motivations in mind:

– *prediction, accuracy*: when $p \gg n$ classical models fail. Is it possible to have strong theoretical guarantees on the risk (i.e., generalization error)?

– *model interpretability*: by removing irrelevant features X_i (i.e, by setting the corresponding coefficients estimates to zero), the model will easier to understand.

Good references on this topic are Giraud [2014] and Friedman et al. [2001].

Why high-dimensional data? The volume of available data is growing exponentially fast nowadays. According to IBM two years ago, 10^{18} bytes of data were created every day in the world and 90% of data is less than two years old. Many modern data record simultaneously thousands up to millions of features on each objects or individuals. In many applications, data is high-dimensional such as with DNA, images, video, cookies (data about consumer preferences) or in astrophysics.

The curse of dimensionality

– High-dimensional spaces are vast and data points are isolated in their immensity.

– The accumulation of small errors in many different directions can produce a large global error.

– An event that is an accumulation of rare events may be not rare in high-dimensional space.

Example 1.1. In high-dimensional spaces, no point in you data set will be close from a new input you want to predict. Assume that your input space is $\mathcal{X} = [0,1]^p$. The number of points needed to cover the space at a radius ε in L2 norm is of order $1/\varepsilon^p$ which increases exponentially with the dimension. Therefore, in high dimension, it is unlikely to have a point in you data set that will be close to any new input.

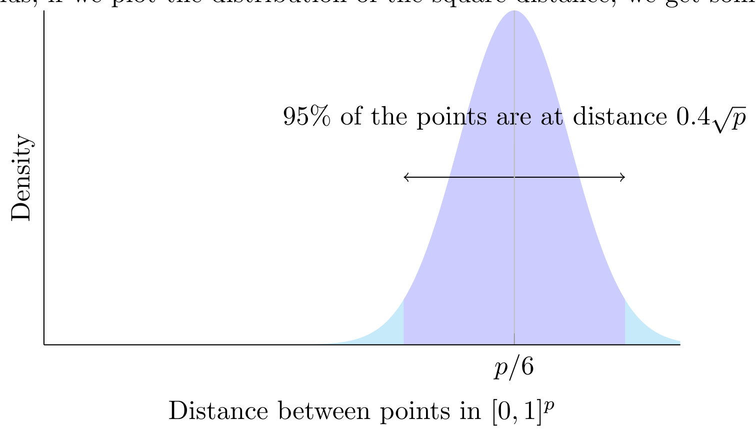
Example 1.2. In high-dimensional spaces classical distances are often meaningless: all the points tends to be at similar distance from one another. Consider the following example to convince ourselves. Assume that X, X' follow uniform distribution on $[0,1]^p$. Then, the expected distance in square L2-norm between X and X' is

$$\mathbb{E}\big[\|X - X'\|^2\big] = \sum_{i=1}^p \mathbb{E}\big[(X_i - X_i')^2\big] = p\mathbb{E}\big[(X_1 - X_1')^2\big] = p \int_0^1 \int_0^1 (x - x') dx dx' = \frac{p}{6}$$

Therefore, the average distance between the points increases with the dimension. Furthermore, the standard deviation of this square distance is

$$\sqrt{\mathrm{Var}\big(\|X - X'\|^2\big)} = \sqrt{\sum_{i=1}^p \mathrm{Var}\big((Xi - X_i')^2\big)} = \sqrt{p\mathrm{Var}\big((X_1 - X_1')^2\big)} = \frac{\sqrt{7p}}{6\sqrt{5}} \simeq 0.2\sqrt{p}\,.$$

Thus, if we plot the distribution of the square distance, we get something like:



Therefore, relatively to their distance, all points seem to be at similar distance from one another. The notion of nearest point distance vanishes. As a consequence, K-Nearest Neighbors gets poor performance in large dimension.

Example 1.3. Let us consider another example in high-dimensional linear regression. We consider the ordinary least square estimator (OLS) for the linear model

$$\widehat{\beta} \in \underset{\beta \in \mathbb{R}^p}{\arg\min} \left\|Y - X\beta\right\|^2 \qquad \text{where} \quad Y_i = x_i^\top \beta^* + \varepsilon_i, \quad X = (x_1, \dots, x_n)^\top \in \mathbb{R}^{n \times p} \quad \text{and} \quad \varepsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)\,.$$

If $rg(X) = p$ (i.e., $p \leq n$) then $\widehat{\beta} = (X \top X)^{-1} X^\top Y$ and as we saw in previous lecture the estimator satisfies

$$\mathbb{E}\big[\|\widehat{\beta} - \beta^*\|^2\big] = \mathrm{Tr}\big((X^\top X)^{-1}\big)\sigma^2\,.$$

In particular, in the very gentle case of an orthogonal design, we get $\mathbb{E}\big[\|\widehat{\beta} - \beta^*\|^2\big] = p\sigma^2$. Therefore, the variance of the estimator increases linearly with the dimension and the later gets unstable for high-dimensional data. Furthermore, OLS only works for $p \gg n$ because otherwise the matrix $X^\top X$ is not invertible and using pseudo-inverse would lead to highly unstable estimator and over-fitting. One needs to regularize.

The previous examples seem to show that the curse of dimensionality is unavoidable and we are doomed to poor estimators in large dimension. Hopefully, in many cases, data has an intrinsic low complexity (sparsity, low dimensional structure,...). This is the case of the data (for instance with images) or of the machine learning methods which is used (for instance Kernel regression).

What can we do with high-dimensional data? There are three classes of methods to deal with large dimensional input spaces:

- *Model selection*: we identify a subset of $s \ll p$ predictors that we believe to be related to the response. We then fit a model (for instance OLS) on the s variables only.
- *Regularization*: Lasso, Group Lasso, Elastic net,...
- *Dimension reduction*: the objective is to find a low-dimensional representation of the data. If we consider linear transformation, we may project the p predictors into a s-dimensional space with $s \ll p$. This is achieved by computing s different linear combination or projections of the variables. Then these projections are used as new features to fit a simple model for instance by least squares. Examples of such methods are PCA, PLS,...

In this lesson, we will focus on the regularization approach.

2 The Lasso

The high-level idea of the Lasso is to transform the penalize the empirical risk minimizer with a sharp penalty that will favorize sparse solutions. We define the LASSO estimator

$$\widehat{\beta}_\lambda \in \mathop{\arg\min}_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{2} \|Y - X\beta\|^2 + \lambda \|\beta\|_1 \right\}. \tag{LASSO}$$

The solution $\widehat{\beta}_\lambda$ may not be unique but the prediction $X\widehat{\beta}_\lambda$ is.

2.1 Geometric insight

By convex duality, the Lasso is also the solution of

$$\widehat{\beta}_\lambda \in \mathop{\arg\min}_{\beta \in \mathbb{R}^p : \|\beta\|_1 \leq R_\lambda} \left\{ \|Y - X\beta\|^2 \right\},$$

for some radius $R_\lambda > 0$ (that may depend on the data (X_i, Y_i)). The non-smoothness of the ℓ_1-norm puts some coefficients to zero. In Figure 1, we can see that because of the corners of the ℓ_1-ball, the solution $\widehat{\beta}_\lambda$ gets zero coefficients which is not the case when regularizing with the ℓ_2-norm (on the right).

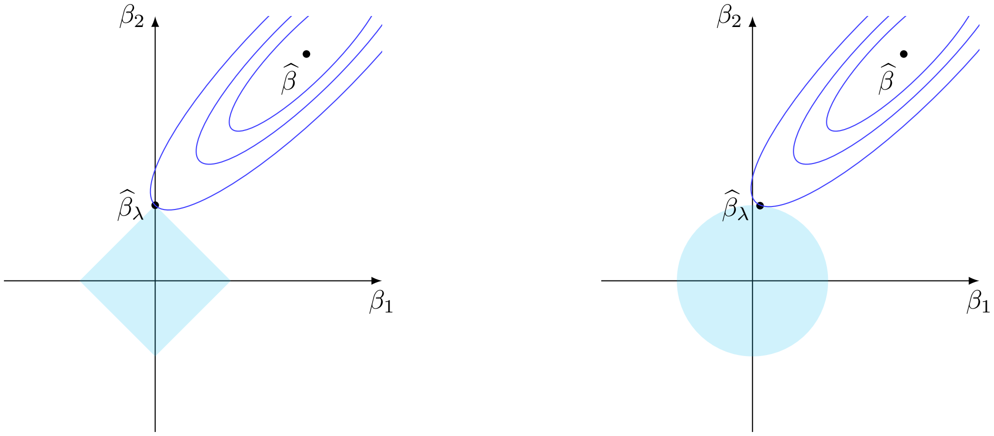


Figure 1: $\widehat{\beta}$ denotes the minimizer of the empirical risk and the blue lines denote level lines of the empirical risk [left] Regularization with a ℓ_1-ball [right] Regularization with a ℓ_2-ball.

2.2 What does the solution of the Lasso looks like?

To solve the problem of Lasso, if the objective function $\mathcal{L} : \beta \mapsto \frac{1}{2}\|Y - X\beta\|^2 + \lambda\|\beta\|_1$ was differentiable, one would cancel the gradient. However, because of the ℓ_1-norm the later is not differentiable and one needs to generalize the notion of gradient to convex functions which are not necessarily differentiable. This is done with the following definition.

Definition 1 (Subdifferential)**.** *A subgradient of a convex function $f : \mathbb{R}^p \to \mathbb{R}$ at a point $\beta_0 \in \mathbb{R}^p$ is a vector $z \in \mathbb{R}^p$ such that for any $\beta \in \mathbb{R}^p$ the convex inequality holds*

$$f(\beta) - f(\beta_0) \geq z^\top (\beta - \beta_0).$$

The set of all subgradients of f at β_0 is denoted $\partial f(\beta_0)$ and is called the subdifferential of f at β_0.

The subdifferential of the ℓ_1-norm is

$$\partial\|\beta\|_1 = \Big\{ z \in [-1,1]^p \text{ s.t. for all } 1 \le j \le p \quad z_j = \mathrm{sign}(\beta_j) \text{ if } \beta_j \neq 0 \Big\}$$

and the subdifferential of the objective funtion of the Lasso is

$$\partial\mathcal{L}(\beta) = \big\{ -X^\top(Y - X\beta) + \lambda z : \; z \in \partial\|\beta\|_1 \big\} .$$

Any solution of the Lasso should cancel the subdifferential. Therefore, if $\widehat\beta_\lambda$ is a solution of the Lasso, it exists $\widehat z \in \partial\|\widehat\beta_\lambda\|_1$ (i.e., $\widehat z_j = \mathrm{sign}(\widehat\beta_\lambda(j))$ if $\widehat\beta_\lambda(j) \neq 0$ and $\widehat z_j \in [-1,1]$ otherwise) such that

$$-X^\top(Y - X\beta) + \lambda\widehat z = 0 \quad \Rightarrow \quad X^\top X\widehat\beta_\lambda = X^\top Y - \lambda\widehat z . \tag{1}$$

If the gram matrix $X^\top X$ is general, it is not possible to solve the later in close form. To get some insights about the solution of the Lasso, let us assume the orthonormal setting $X^\top X = I_p$. Then, from (1), we get for all $j \in \{1,\dots,p\}$ such that $\widehat\beta_\lambda(j) \neq 0$

$$\widehat\beta_\lambda(j) = X_j^\top Y - \lambda\mathrm{sign}(\widehat\beta_\lambda(j)) .$$

Therefore, $X_j^\top Y = \widehat\beta_\lambda(j) + \mathrm{sign}(\widehat\beta_\lambda(j))$ and $\widehat\beta_\lambda(j)$ have same sign and we obtain for all $1 \le j \le p$

$$\begin{aligned} \widehat\beta_\lambda(j) &= \begin{cases} X_j^\top Y - \lambda\mathrm{sign}(X_j^\top Y) & \text{if } |X_j^\top Y| \ge \lambda \\ 0 & \text{if } |X_j^\top Y| \le \lambda \end{cases} \\ &= S_\lambda(X_j^\top Y) , \end{aligned}$$

where S_λ is the soft-threshold function:

$$S_\lambda(x) = \begin{cases} 0 & \text{if} \quad |x| \le \lambda \\ x - \lambda\mathrm{sign}(x) & \text{otherwise} \end{cases} .$$

In the orthonormal setting, the Lasso performs thus a soft threshold of the coordinates of the OLS.

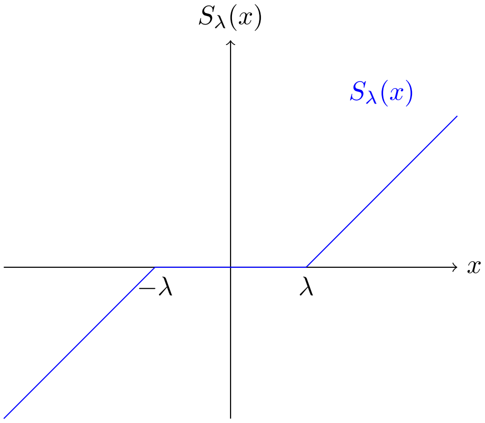


Figure 2: Soft-threshold function

2.3 Statistical property of the Lasso estimator

Consider the sparse linear model

$$Y = X\beta^* + \varepsilon\,, \quad Y = (y_1, \ldots, y_n) \in \mathbb{R}^n, \quad X \in \mathbb{R}^{n\times p}, \varepsilon \sim \mathcal{N}(0, \sigma^2 I_n)\,. \tag{2}$$

We consider $p \gg n$ but we assume that β^* has only $s < p$ non-zero coordinates. We denote by $m^* \subset \{1, \ldots, p\}$ the set of non-zero coordinates of β^*.

For λ large enough $\lambda \simeq \sigma\sqrt{\log p}$, under some additional condition on the design (relaxed version of orthonormal design), it is possible to show (see Giraud [2014]) that the Lasso does not assign any weight to coefficients that are not in m^*. If λ is properly chosen, it recovers exactly the coefficients of β^* and its risk is controlled with high probability as

$$R(\widehat{\beta}_\lambda) = \left\|X\beta^* - X\widehat{\beta}_\lambda\right\|^2 \le \inf_{\beta\in\mathbb{R}^p\setminus\{0\}} \Big\{ \|X\beta - X\beta^*\|^2 + \square_X \lambda^2 \|\beta\|_0 \Big\}\,,$$

where $\lambda^2 \simeq \sigma^2 \log p$. The constant $\square_X$ is the compatibility constant. It depends on the design X and can be arbitrarily bad for non-orthogonal design. It can be shown that it is not possible to avoid it for efficient (polynomial time) procedures.

2.4 Computing the Lasso estimator

Since this is the solution of a convex optimization problem, the solution of the Lasso can be obtained efficiently. There are three main algorithms used by the community.

Coordinate descent (cf. practical session) the idea is to repeatedly minimize the objective function $\mathcal{L}(\beta)$ with respect to each coordinate. It converges thanks to the convexity of $\mathcal{L}$. As we saw in Equation (1), the solution of the Lasso satisfies

$$X^\top X\widehat{\beta}_\lambda = X^\top Y - \lambda\widehat{z}$$

where $\widehat{z} \in \partial\|\beta\|_1$. We saw that the solution equals $\widehat{\beta}_\lambda(j) = S_\lambda(X_j^\top Y)$ when $X^\top X = I_p$. This equation has however no closed-form solution in general. The idea of coordinate descent is to solve this equation only for one coordinate, fixing all the other coordinates.
Let $1 \le i \le n$ and fix coordinates $\beta_j \in \mathbb{R}$ for $j \neq i$. Solving the i-th coordinate optimisation problem given by

$$\min_{\beta_i} \mathcal{L}(\beta) = \min_{\beta_i\in\mathbb{R}} \left\{ \frac{1}{2}\left\|Y - X\beta\right\|^2 + \lambda\|\beta\|_1 \right\}\,,$$

we get that the i-th partial sub-derivative of $\mathcal{L}$ should cancel, which gives similarly to previously

$$X_i^\top X\beta = X_i^\top Y - \lambda z_i\,,$$

where $z_i \in \partial|\beta_i|$. This can be rewritten as

$$X_i^\top X_i\beta_i + X_i^\top X_{-i}\beta_{-i} = X_i^\top Y - \lambda z_i\,,$$

where $X_{-i} \in \mathbb{R}^n \times (p-1)$ is the input matrix without column i and $\beta_{-i} \in \mathbb{R}^{p-1}$ is the fixed parameter vector without coordinate i.
Assume $\beta_i \neq 0$, then $z_i = \operatorname{sign}(\beta_i)$ and

$$X_i^\top X_i\beta_i + \lambda\operatorname{sign}(\beta_i) = X_i^\top(Y - X_{-i}\beta_{-i})\,,$$

Since $X_i^\top X_i > 0$, we have $z_i = \operatorname{sign}\big(X_i^\top(Y - X_i^\top X_{-i}\beta_{-i})\big)$ which implies

$$\beta_i = \frac{S_\lambda\big(X_i^\top(Y - X_i^\top X_{-i}\beta_{-i})\big)}{X_i^\top X_i}. \tag{3}$$

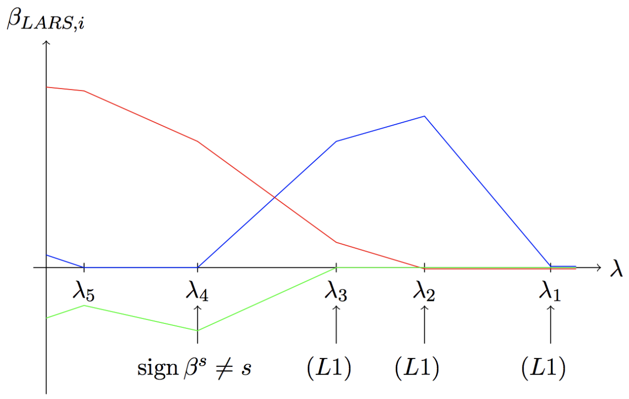


Figure 3: Lasso regularization path computed with LARS

where S_λ is the soft-threshold function:

$$S_\lambda(x) = \begin{cases} 0 & \text{if} \quad |x| \leq \lambda \\ x - \lambda \operatorname{sign}(x) & \text{otherwise} \end{cases} .$$

The algorithm of coordinate descent consists in sequentially repeating the update (3) for $i = 1, \ldots, p, 1 \ldots, p, \ldots$ minmizing the objective function with respect to each coordinate at a time.

Fista (fast iterative shrinkage thresholding algorithmn) It uses the explicit formula in the orthogonal design setting for computing recursively an approximation of the solution

LARS The insight of the algorithm comes from equation (1): $X^\top X \widehat{\beta}_\lambda = X^\top Y - \frac{\lambda}{2}\widehat{z}$. We then consider the function $\lambda \mapsto \widehat{\beta}_\lambda$. For non-zero coefficients, $\widehat{z}_j = \operatorname{sign}(\widehat{\beta}_\lambda(j))$ and is constant while $\lambda \mapsto \widehat{\beta}_\lambda(j)$ does not change sign. Therefore, the function $\lambda \mapsto \widehat{\beta}_\lambda$ is piecewise linear in λ with a change when for some coordinate $\widehat{\beta}_\lambda(j)$ changes sign. LARS computes the sequence $\{\widehat{\beta}_{\lambda_1}, \widehat{\beta}_{\lambda_2}, \ldots\}$ of the Lasso estimator corresponding to the break points of the path $\lambda \mapsto \widehat{\beta}_\lambda$. At each break point, the model $m_\lambda = \{i \in \{1, \ldots, p\} : \widehat{\beta}_\lambda(i) \neq 0\}$ is updated and we solve the linear equation

$$X_{m_\lambda}^\top X_{m_\lambda} \widehat{\beta}_\lambda(m_\lambda) = X_{m_\lambda}^\top Y - \frac{\lambda}{2} \operatorname{sign}(\widehat{\beta}_\lambda(m)) ,$$

until the next break point. This algorithm is slower than the other two algorithms but it provides the full regularization path $\lambda \mapsto \widehat{\beta}_\lambda$ (see Figure 3).

2.5 Final remarks and variants

Removing the bias of the Lasso The Lasso estimator $\widehat{\beta}_\lambda$ is biased. Often one might want to remove the bias for instance by first computing $\widehat{\beta}_\lambda$ to select to good model $\widehat{m}_\lambda$ and then solve the OLS or Ridge on the model $\widehat{m}_\lambda$ only.

No penalization of the intercept In practice, the intercept is often no penalized and the Lasso solves

$$\widehat{\beta}_\lambda \in \arg\min_{\beta \in \mathbb{R}^p} \left\{ \sum_{i=1}^n (Y_i - \beta_0 - \beta^\top X_i)^2 + \lambda \|\beta\|_1 \right\} .$$

Group Lasso It is an extension when coordinates are sparse by groups. In other words, we have some groups $G_k \subset \{1, \ldots, p\}$ and we assume that all coordinates β_i for $i \in G_k$ are either all zero or all non-zero.

Elastic net It is a mix of ℓ_1 and ℓ_2 regularization

$$\widehat{\beta} \in \operatorname*{arg\,min}_{\beta \in \mathbb{R}^p} \left\{ \|Y - X\beta\|^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2 \right\}.$$

It also selects variables thanks to sharp corners and it is heavily used in practice.

Calibration of λ It is a crucial point in practice. A common solution is to perform K-fold cross validation. There are a few other techniques such as the slopes heuristic.

References

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

Christophe Giraud. *Introduction to high-dimensional statistics.* Chapman and Hall/CRC, 2014.