Theoretical Guidelines for High Dimentional Data Analysis

Colin de Verdière Matthieu and Houzé de l'Aulnoit Arnaud

February 2020

1 Introduction

In this document, we will study the paper : **False Discoveries occur early on the Lasso path**. This paper presents a study of the characteristics of Lasso (Least Absolute Shrinkage and Selection Operator) regression, and in particular its tendency to predict false discoveries early in the Lasso path. First, we will recall the principle of Lasso regression, then we will see that Lasso has sometimes difficulties to select the appropriate variables to build a good model and we will explain the origin and the consequences of this important problem. After that, we will take a critical look at the results of the paper by studying their scope and limitations and then we will compare Lasso to other regression methods (least-square, ridge,...) in order to determine which are the best conditions to use Lasso. Finally, we will implement some interesting parts of the paper through a numerical approach.

2 Presentation of the background and key findings of the paper

2.1 Reminders on the use of Lasso in a regression model

We first recall the principle of regression with Lasso : Lasso is widely used to generate regression models. Let us consider a regression problem a problem of the type :

$$y = X\beta + z \qquad (1)$$

where X is a $n \times m$ matrix with entries which are assumed to be i.i.d Gaussian, β is a vector of size p containing the coefficients to estimate in the Lasso regression problem. Thus, the Lasso is defined as the solution to the problem :

$$\hat{\beta}(\lambda) = \underset{b \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2}\|y - Xb\|^2 + \lambda\|b\|_1 \qquad (2)$$

As an input to our problem, we have a matrix X composed of p features (columns) and n rows (samples), and the data from this matrix will be used to predict the output y. However, some of the information in this matrix corresponds to noise (z in formula *(1)*), some features in the matrix may be highly correlated or may have no impact on the output, and these elements constitute an obstacle for the prediction. Lasso will therefore aim to select the features X_j that contribute to the prediction of the output value. To do this, we use Lasso to update the $\hat{\beta}$ coefficients which will act as a filter for the variables (features) X_j (see Figure 1 below). It means that for a given column j of the matrix, we must have : $\beta_j = 0$ if we choose not to select the variable X_j and $\beta_j \neq 0$ if we keep the variable X_j.

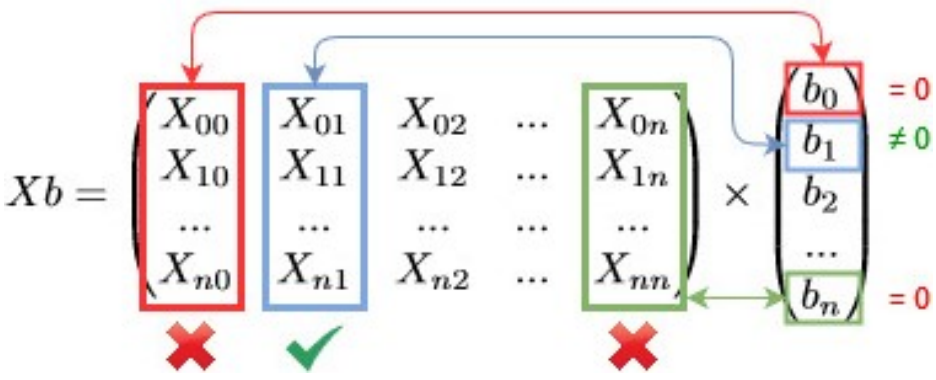


FIGURE 1 – Relationship between the value of coefficients and the selection of variables

The goal is therefore to achieve a perfect support recovery, that's to say to find the value of the beta coefficients such that only the variables important to the prediction are selected. In addition, the paper makes reference to the lasso path which allows to calculate different sequences of estimated $\hat{\beta}$ coefficients by varying λ between 0 and ∞. It will then be enough to choose the λ value associated with the coefficients giving the best prediction results.
In order to clarify the points of the next paragraphs we define now some important variables :

- $k = |j : \beta_j \neq 0|$ is the number of true signals
- p is the number of features (columns)
- $\epsilon = \frac{k}{p}$ is the sparsity ratio (number of true signal over the total number of features)
- $\delta = \frac{n}{p}$ is the subsampling rate (the number of samples over the number of features)

2.2 Constraints and conditions to use Lasso regression

Lasso is mostly used in a situation where the vector of regression coefficients β is sparse. What's more, this method is often used in large problems, and especially when p (number of columns in the X matrix) is greater than n (number of samples) : in these conditions, Lasso allows to select the true variables among a large number of features through the update of the β_j coefficients which act as variable filters.
The Lasso method produces good results if certain conditions are met : for example, the true signals (coefficients) must be stronger than the noise level in order to distinguish between the two. Furthermore, the variables must be weakly correlated. In this case, choosing the appropriate λ value will result in the selection of most of the variables important for prediction and very few variables associated with noise.

2.3 Study of the quality of predictor variables ordering

In order to study the quality of predictive variables, the notion of true/false discoveries is used : A feature (regressor) of index j is considered to be a false discovery if the model predict that the latter is dependant from the output, when that's not actually the case : from a mathematical point of view, if we name $V(\lambda)$ the number of false dicoveries, $\hat{\beta}$ our coefficient estimator and β the true coefficient, we have :$V(\lambda) = |j : \hat{\beta}_j \neq 0$ and $\beta_j = 0|$
Conversely, a feature (regressor) of index j is considered to be a true discovery if the model predict that the latter is dependant from the output, and that is indeed the case : from a mathematical point of view, if we name $T(\lambda)$ the number of true dicoveries, we have :$T(\lambda) = |j : \hat{\beta}_j \neq 0$ and $\beta_j \neq 0|$
From these definitions, we can define the ratios TPP (True Positive Proportion) and FDP (False Discovery Proportion) which will make it possible to evaluate the quality of the beta estimators. These two ratios are defined as follows :

$$FDP(\lambda) = \frac{V(\lambda)}{|\{j : \hat{\beta}_j(\lambda) \neq 0\}| \vee 1} = \frac{\text{Number of false discoveries}}{\text{Number of predicted discoveries}} \quad \text{(type I error)}$$

$$TPP(\lambda) = \frac{T(\lambda)}{k \vee 1} = \frac{\text{Number of true discoveries}}{\text{Number of true signals}} \quad \text{(type II error)}$$

where k is the number of true signals, that's to say non zero β_j coefficients.

2.4 The problem of Lasso

In the many examples of the paper, we can see that the Lasso method tends to select null variables (i.e. variables that have no impact on the output) very early on the Lasso path. In this context, the main objective of the paper is to provide a mathematical and quantitative explanation of this phenomenon by proving the existence of a trade-off between the FDP and TPP ratios in a regime of linear sparsity, that's to say a regime where the regression vector only has a small number of non-zeros coefficients. Then, it shows that it is impossible to obtain both a high rate of true discoveries and a low rate of false discoveries.
To better visualize, we can represent a curve which separates the achievable pairs (TPP, FDP) from the pairs which are impossible to obtain regardless of the value of the signal-to-noise ratio. This makes it possible to show that there is an ideal region on the graph which is inaccessible, and in which the FDP is close to 0 and the TPP is close to 1. An example of a graph is shown below :

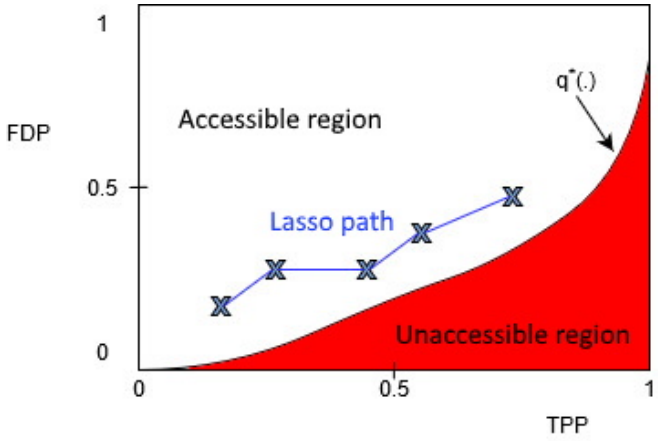


FIGURE 2 – An example of a lasso trade-off diagram

On the figure, q^* represents the lower envelope of all the instance-specific curves q^π with $P(\pi \neq 0) = \epsilon$, where Π is a random variable and the regression coefficients β_1, β_2, ..., β_p are independant copies of Π. In addition, the shape of the q^* curve will depend on the $\delta = \frac{n}{p}$ and $\epsilon = \frac{k}{p}$ parameters which characterizes the structure of the X matrix. Moreover, we can see in the article that to have the lowest possible trade-off curve, it is preferable to have the highest possible δ (a large number of samples compared to the number of features) and the lowest possible ϵ (less non-zero coefficients than columns). The trade-off curve is defined by a mathematical formula set out in the following section.

2.5 Theorem 1 and main results (trade-off formula)

The theorem number 1 of the article gives the mathematical formula representing the tradeoff between the key figures TPP and FDP. The latter states that in the noiseless $(\sigma = 0)$ or noisy case $(\sigma \neq 0)$, we have :

$$\boxed{\cap_{\lambda \geq \lambda_0} \{FDP(\lambda) \geq q^*(TPP(\lambda)) - \eta\}} \qquad \text{holds with probability tending to one}$$

with

$$q^* = \frac{2(1-\epsilon)\Phi(-t^*(u))}{2(1-\epsilon)\Phi(-t^*(u) + \epsilon u}$$

As said before, we can see that the quantity q^* depends on the sparsity (ϵ) and the dimensionality (δ) of the input matrix X. We can clearly see a linear relationship between the 2 ratios, and that if we want to increase the TPP, the FDP will also necessarily increase, whatever the value of the regularizing parameter λ.We will see in a next section that the origin of the imposed tradeoff comes from a pseudo noise introduced by shrinkage. Now that we have described the existence of the tradeoff, it is interesting to see how to approach any point on this curve, and we will see that this can be done by tuning the sharpness of the signals.
The best results for value pairs $(TPP(\lambda), FDP(\lambda)$ tend towards $(u, q^*(u))$ are obtained when the β coefficients are copies of the random variable Π along a certain distribution :

$$\Pi = \begin{cases} M & \text{with probability} \quad \epsilon.\epsilon' \\ M^{-1} & \text{with probability} \quad \epsilon.(1-\epsilon') \\ 0 & \text{with probability} \quad 1-\epsilon \end{cases}$$

We fix M with a very large value so that all signals are either very large or very small (which makes it easier to detect). In this context, the article shows that for any u (TPP) between 0 and 1, we can find a value of ϵ' such that :

$$\lim_{M \to \infty} \lim_{n,p \to \infty} (TPP(\lambda), FDP(\lambda)) \to (u, q^*(u)) \qquad \text{with } \lambda \to \infty$$

We can therefore say that in order to obtain the best possible pairs on the Lasso path, with each of them close to pairs $(u, q^*(u))$, the complete signal must be a combination of very strong and very weak signals (coefficients β). In this way, weak signals representing null variables will not be selected by Lasso, resulting in a reduction of the FDP. On a graph representing the average of the couples (TPP,FDP) on different Lasso paths, this will result in rebounds of the average Lasso path on the trade-off curve.

2.6 The shrinkage with the Lasso

The shrinkage is the main concept of the Lasso and of the l_1 regularization. In this process the purpose consists in reducing the size of the coefficients estimates, in other words shrinking them toward zero. What's more if a coefficient is shrunk to exactly zero, then the input variable associated to this coefficient is removed from the model. But what is the real goal of shrinkage ? Its main objective is to improve the simple linear regression in two different ways :

- First, reducing the number of coefficients estimates reduces the variance, but increases the bias in return.

  Below is a representation to understand the influence of bias and variance on our estimators [3]. The blue dots represent the estimators and their position on the target represents their distance from the true coefficients (represented by the red area). We can see that the bias corresponds to the accuracy of the estimator while the variance corresponds to their spacing. Concretely, a high bias means that the regression line fit the training data very well and a high variance means that the line did not fit the testing data very well. The objective is therefore to obtain a set of estimators with a low bias and a low variance.

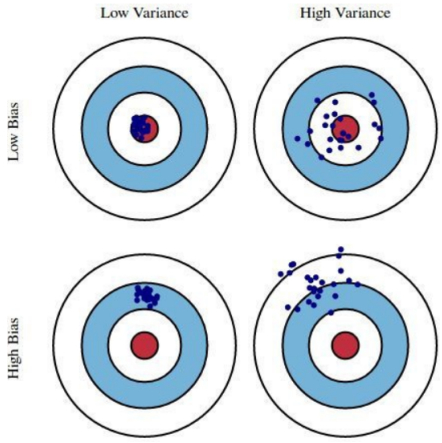


FIGURE 3 – Tradeoff schematization of Variance-Bias [3]

  Thus, in order to improve the performance of the model, it is necessary to find a compromise to minimize the total error defined as the sum of the error associated with the variance of the coefficients and that associated with the bias.

- Moreover, shrinking ensures the interpretability of the model : indeed, if we keep all the coefficients, it may be difficult to establish a relationship between each of the variables, and it may be wise to simplify the model by keeping only the variables that have the most impact on the prediction in order to obtain better performance.

In order to obtain a suitable shrinkage, the lasso is first tested for different lambda values. This then allows us to obtain a graph [6] showing the evolution of the shrinkage of each coefficient according to the lambda value. An example of this graph is represented below :

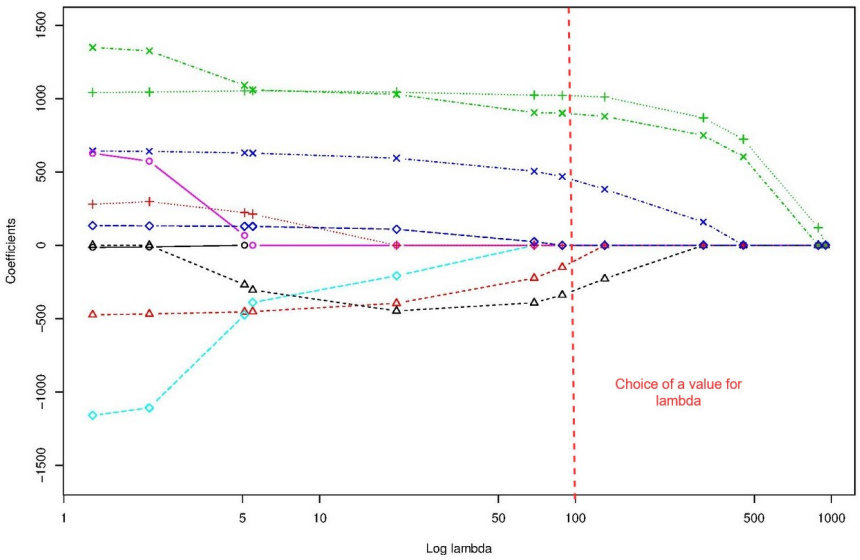


FIGURE 4 – Example of the evolution of coefficients estimated by the lasso as a function of λ [6]

We can see on the graph that the more the λ value increases, the more the coefficients are shrinked (and then disappear). Once this path graph is obtained, we still have to choose the value of lambda which will determine a certain level of shrinkage for the coefficients. The choice of lambda can be done in different ways such as cross validation (or with the methods AIC, BIC, RIC,...)

To sum up, the goal of the shrinkage to zero of the β_j is to remove from the model parameters with no incidence on the prediction of y and to give importance to parameters that have a dependency with the output y. However, despite its usefulness, the shrinkage sometimes leads to prediction errors and the algorithm may consider what is in reality a false discovery (the parameter has no explanatory power) for a true signal and vice versa. This can cause a vanishing effect regarding some features that could have been useful or on the contrary it could give importance to features that are not interesting for the understanding of y. In the next part, we describe the shrinkage noise that is at the origin of this problem.

2.7 The shrinkage noise

The shrinkage is efficient in appropriate conditions detailed at the beginning of this paper. To explain the mistakes of the Lasso algorithm, the reference paper describes a phenomenon of "pseudo noise" that may occur. There are two ways to explain this noise. Firstly the characteristics of the l_1 norm implies that if λ is disproportionately large it will have a huge impact what will cause a huge bias and distort the estimation of $\hat{\beta}_j$. Secondly for a fixed value of λ on the Lasso path and if we assume that the conditions for a good behavior of Lasso are met then we may also consider the residuals from $\|y - Xb\|^2$ as a noise due to the shrinkage. This problem of the Lasso and more generally of algorithms using the l_1-penalty is the apparition of the noise during the shrinkage process.

2.8 l_0-penalty and shrinkage

To bypass the shrinkage noise due to the particularity of the l_1 norm with the absolute value other penalty are interesting to consider. In the reference paper the authors study the qualities of the l_0-penalty which is quite fascinating because thanks to the l_0 norm we can face the problem of the shrinkage noise. With this norm the equation that we want to minimize is :

$$\hat{\beta}_0 = \underset{b \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2}\|y - Xb\|^2 + \lambda \|b\|_0$$

Regarding the norm l_0 it is also remarkable that it is possible to have better results in detecting the true signal than with the norm l_1 while the l_1-penalty is particularly appropriated to sparse problem. We explain that because the l_0-penalty is also appropriate to deal with problem with few parameters and penalises complex models with too much parameters.

With respect to the same hypothesis used before with a small sparsity ratio compared to the subsampling rate : $\epsilon < \delta$ and Π a prior such that :

$$\Pi = \begin{cases} M, & w.p. \quad \epsilon \\ 0, & w.p. \quad 1-\epsilon \end{cases}$$

It follows that we can find $\lambda(M)$ such that in probability, the discoveries of the l_0 estimator respects :

$$\lim_{M\to\infty} \lim_{n,p\to\infty} \text{FDP} = 0 \quad and \quad \lim_{M\to\infty} \lim_{n,p\to\infty} \text{TPP} = 1$$

It turns out that the l_0-penalty allow to find the best case in probability with all true signals picked up and no false discovery what is unreachable with the l_1-penalty because of the trade off between the error of type I and II.

3 critical analysis of paper

3.1 Scope and Limitations

In this section, we will first discuss the scope and limitations of the results proposed in the paper and then perform a Lasso comparison with other regression methods.
Let's first talk about the results of Lasso : In the paper, we see that in order to obtain a perfect support recovery, the following points must be taken into account : First of all, we observe that for the trade-off of theorem 1 to be valid, the conditions below must be met :

- The input matrix contains i.i.d Gaussian entries (with a variance of $\frac{1}{n}$)
- The noise terms (errors) z_i must be i.i.d Gaussian entries (with a variance of σ^2)
- Coefficients β_j must be copies of a random variable Π with $\mathbb{E}\Pi^2 < \infty$
- Parameters n and p of the input matrix tend towards infinity.
- We are in a regime of (sub)linear sparsity, i.e. the number of non-zero signals is low.

We can therefore notice that the trade-off is only highlighted in specific cases where all these conditions are met : for example, one of the conditions stipulates that the input matrix X must have i.i.d Gaussian entries to facilitate the variable selection by the model. However, it is questionable whether perfect support recovery or at least the setting up of a trade-off could be achieved in the case of having a different distribution (such as a random design) for these entries.
Moreover, one can also wonder how the article chooses certain parameters such as the prior distribution Π. In part 2.5 on sharpness, the paper advises to take a very high value of M to obtain a combination of strong and weak signals, but it does not specify precise values or methods to choose the most suitable value of M. Similarly, and still concerning the sharpness formula, the paper does not specify how it chose its ϵ' values.

What's more, the article states that it is possible to obtain a perfect support recovery if we have (in addition to the above conditions) :

- The nonzero coefficients must have a magnitude of at least $c\sigma\sqrt{2\log p}$ with c a constant
- A sample size $n \geq (2 + o(1))k \log p$

However, it doesn't work in the case where n and p are relatively weak. Indeed, in the example of the article, we can see that for a matrix of dimensions 250x1000 with k=18, we obtain an FDP of 21.7% when we reach full power. We can then see that these results are very close to the (TPP,FDP) boundary defined in theorem 1. We can therefore deduce that the trade-off formula of theorem 1 can be applied in the case of different range of sparsity levels.

In addition, new questions could be asked in connection with the results of the paper. For example, one could predict the range at which the first False discovery would appear. Indeed, when the number of true signals increases above a certain threshold ($k > \frac{n}{2logp}$), we see that the rank of the first false discovery decreases significantly, which means that we have more and more early false discoveries, unlike a low sparsity regime where we get full power before the first variable noise is selected. It would then be interesting to be able to justify this from a mathematical point of view and to find a formula (as for the trade-off of theorem 1) to predict the rank of apparition of the first False discovery from the input parameters.

To sum up, we can say that the main advantages of Lasso are :

- Lasso is adapted to judiciously select a restricted set of variables having an impact on the output according to the regularization parameter lambda .
- If we have the β vector which is sparse with kip, under certain conditions, we will be able to achieve a perfect support recovery, i.e. to select all true signals before the null variables.

However, the article proves that Lasso also has some drawbacks :

- When the true variables are highly correlated, Lasso will privilege only one of them and the other will be removed from the model, which calls into question the coherence of the selection of this method.
- As mentioned in the article, in the case where some conditions on the input variables (n, p and k) are not respected, Lasso presents the appearance of False discoveries very early on the Lasso path that prevent to have a good variable selection score.

Moreover, to understand more deeply the scope, conditions of use and limitations of Lasso, we present you a comparison with other regression methods :

3.2 Comparison with least-square regression

Concerning the scope of the results, it may be interesting to make a comparison between the least-square (l_0) and Lasso (l_1) methods. The paper of M.Zhang [1] entitled *Article lower bounds on the performance of polynomial-time algorithms for sparse linear regression* takes up the same problem of the search for optimal methods for high-dimensional sparse linear regression and justifies the use of the l_1 standard and thus of Lasso in the following way :
If we take the same input parameters as those defined previously, i.e. a model characterized by the triplet of variables (n, p, k) where n is the number of samples, p the number of features and k the sparsity, then we could first think we could solve this regression problem $y = X \times \beta + z$ by finding the sequence of beta minimizing the least-square cost characterized by the formula : $\|y - X\beta\|_2^2$, then we obtain :

$$\hat{\beta_{l_0}}(\lambda) = \underset{\beta \in \mathbb{B}_0(k)}{\operatorname{argmin}} \|y - X\beta\|_2^2$$

However, such an estimator is not easy to compute because it is an NP-Hard problem involving considering k among p subsets of size k. This is why it is interesting to turn to the use of approximation methods and heuristics to find approximate estimators but less complex to compute : this is the reason why we can use Lasso instead of least-square. Thus by replacing the regularization l_0 by the constraint l_1, we obtain an estimator that can be computed in polynomial time since it is based on a simple convex optimization problem. Graphically, Lasso regression results in a line with a little bit more bias but less variance than least square.

Furthermore, despite the fact that the Lasso estimator is less complex to compute, its performances won't in some cases be as good as the least-square method : indeed, the problem of selection of variables by Lasso due to the shrinkage of regression coefficients occurs even when the signal-to-noise ratio tends towards infinity and in the case where each feature is stochastically independent, which is not the case for the l_0 penalty method. Indeed, the formula of the Theorem 1 (presented previously) shows that one cannot go below the trade-off curve (u, q*(u)) with Lasso, whereas on the contrary, theorem 2 indicates that one can tend to a FDP of 0 and a TPP of 1 with an l_0 penalty. To illustrate this, the beginning of the article presents (on page 3) a concrete simulation with a comparison of these 2 methods :
We can see that for a model characterized by n=1010, p=1000 and k=200 which means that only the first 200 beta coefficients are non-zero, the first 200 discoveries with the least-square method are true discoveries (i.e. the model has achieved full power without obtaining a single false discovery), whereas the Lasso model reaches a rate of 8% FDP when the FDP reaches only 20% : as explained before, these poorer performances are due to the phenomenon of shrinkage of the coefficients which leads to the appearance of a trade-off in terms of performances for the pairs (TPP, FDP).

3.3 Comparison with ridge regression

Lasso and ridge regressions work in a very similar way : just like Lasso, ridge regression adds a penalty for non-zero coefficients, but ridge regression penalizes the sum of squared coefficients unlike Lasso which penalizes the sum of absolute values. The consequence of this is that for Lasso, the coefficients can shrink to 0 whereas this is not the case for ridge.

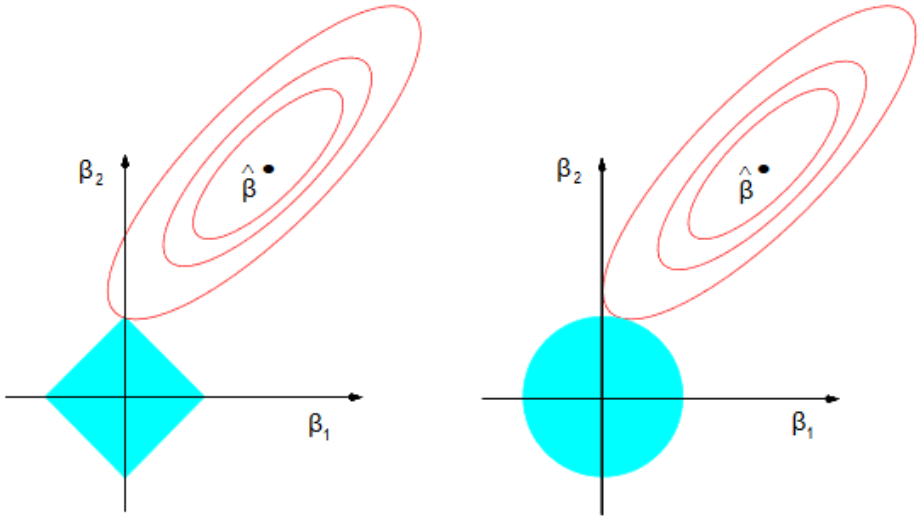


FIGURE 5 – Comparative graph between Lasso regression which can reduce dimension of feature space and ridge regression $[2, 4, 5]$

The figure below graphically explains the difference in the type of shrinkage between the 2 methods $[2, 4, 5]$. In the figure, the colored areas correspond to the regional constraints. We can see that the Lasso area is diamond shaped while the ridge area is circular. In addition, the ellipses represent the cost function of the linear regression. From a graphical point of view and for both methods, the goal is to find the non-zero coefficients by finding the first point of intersection of the ellipse with the constraint region. In the case of Lasso, we see that diamond has corners, and when the ellipse intersects one of these corners, this leads to the vanishing to zero of one of the coefficients, which is not the case for the circular constraint region associated with ridge.

Moreover, in the case where the input matrix is composed of many highly correlated features, these features will be associated with roughly similar coefficients for the ridge regression, while Lasso will remove one of the variables from the model and keep the other one, attributing to it all the impact of the prediction, which will lead to an increase of the bias.
We can therefore deduce that in the case where the number of true signals (variables that have a real impact on the output) is high, the False Discoveries occur early on the Lasso path, and thus it is better to use ridge regression. On the other hand, Lasso regression is better than ridge regression at reducing the variance in a model that contains a lot of useless variables, that's to say a model with a sparse vector β.
We can also evoke in opening the Elastic Net regression which is a mixture of the Lasso and ridge approaches and which combines in the formula of its loss function both penalties l_1 and l_2 : this allows to benefit from the advantages of both methods. To conclude, the major difference between lasso and ridge regressions is that Lasso can exclude useless variables, which makes the final equation simpler and easier to interpret.

4 Personal exploration of the paper

4.1 Numerical implementation : Introduction

In the realization of a numerical implementation on the strength of the reference paper we have decided to focus on Lasso and its derivative in order to go into detail in the behaviour of the l_1-norm.

To do that we were firstly interested in doing several tests regarding the implementation of the Lasso path in order to represent the FDP (false discovery proportion) and TPP (true positive discovery). The first step consists in computing the β for the selected lambda.

In order to compare our results with the article we placed ourselves in the same conditions as the authors i.e. with a number of samples $n = 1010$, a number of features $p = 1000$, $\beta_1 = ... = \beta_{200} = 4$, $\beta_{201} = ... = \beta_{1000} = 0$ and no noise. We define X with a Gaussin design of shape $n * p$. Thus we can compute the output y which represent the true labels that we want to find with ours $\hat{\beta}$ in the following equation :

$$y = X * \beta + z, \text{ with } z = 0 \text{ for the moment}$$

For a certain number of λ such that $0.001 < \lambda < 20$ that we defined using 2000 values equaly separated thanks to the numpy function linspace, we compute an approximation of $\beta(\lambda)$ such that :

$$\hat{\beta}(\lambda) = \underset{b \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2} \|y - Xb\|^2 + \lambda \|b\|_1$$

It is interesting to note that we have two possibilities to compute the Lasso path :

- using Lasso as a linear model and use the fit function, but we have to do that for each lambda
- using the pre-computed lasso-path function that directly returns all the $\hat{\beta}$

In those cases we obtained similar results than the one computed in the reference article (they are logically not exactly the same compared to the original one because the values of lambda are different) :

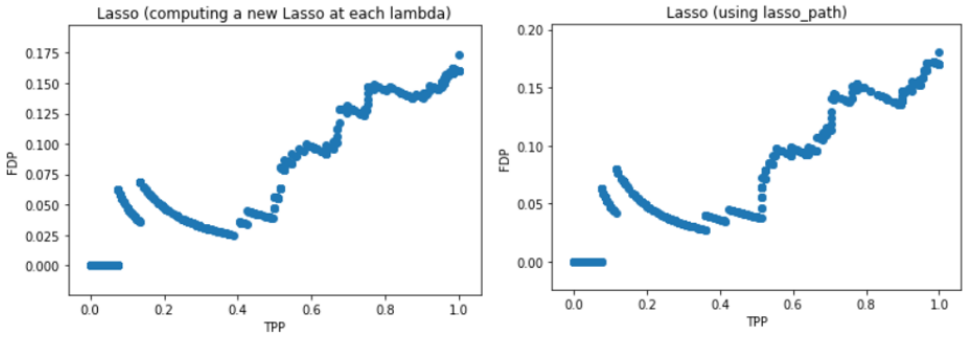


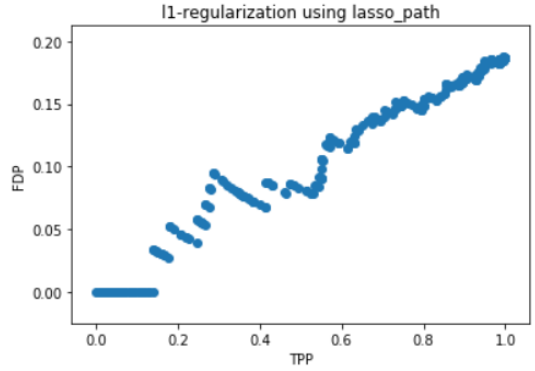
FIGURE 6 – Results of the proportion of FDP and TPP

In the continuation of this paper we will use the pre-computed lasso path for a reason of efficiency in the code. This function from Scikit Learn is moreover able to find strategic values of lambda what is interesting for us to have a more homogeneous graph. For instance when we defined our lambda they are equally distributed but it is more interesting to have a optimal scale to be sure that we will see a better diversity in the ratio FDP and TPP.

4.2 Differences in the l_1-norm implementations

During our research on the l_1 regularization and in the reading of the paper we notice that there are several different ways to compute the l_1-norm path and all of them have their own specificities. In this paper we have decided to compere three of them : the classic one which is the simple Lasso, the Lasso with the LARS (Least Angle Regression) algorithm and the elastic net.

Firstly using the regular Lasso that we studied previously and with the λ set automatically we obtain the following representation of FDP and TPP :

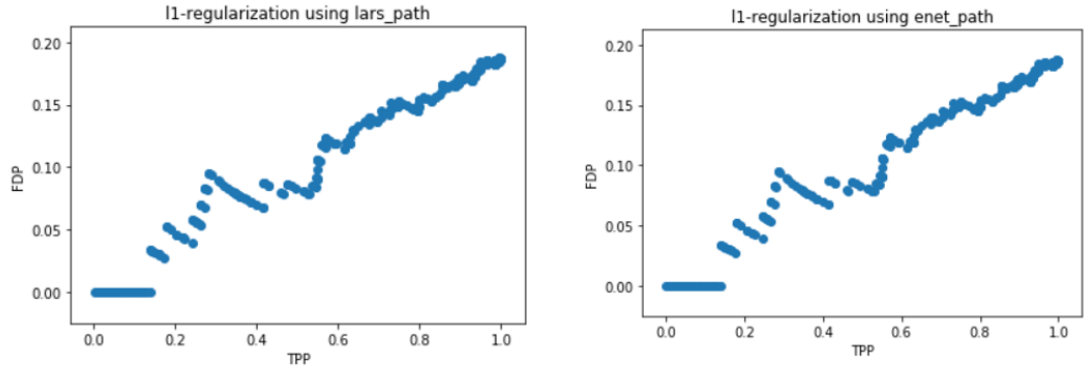



```
FDP mean :  0.16520047690425138
TPP mean :  0.880485
```

FIGURE 7 – Results of the proportion of FDP and TPP using lasso_path

This function selects automatically a certain number of λ defined in the code and in order to compare the results of the lasso_path with those of the lars_path and of the enet_path we will use the same values of λ that are returned by the lasso_path.

Using the same initial conditions and the lambdas given by the lasso_path we obtain the following results for Lars and Elastic Net :




```
FDP mean :  0.16520047690425138
TPP mean :  0.88049
```

```
FDP mean :  0.16520047690425138
TPP mean :  0.880485
```

FIGURE 8 – Results of the proportion of FDP and TPP using lars_path and enet_path

As we expected that the results are almost exactly the same. Firstly it is logical to have exactly the same result between Lasso and Elastic Net because Elastic Net is a proportion between a l_1-norm and l_2-norm and in the computation we have set the l_2-norm to zero. Thus it acts like a Lasso. We decided this setting because otherwise the number of false discoveries increased drastically because the purpose of the l_2-norm is not to set the useless parameters to zero bu only to minimize their impacts. Thus with this regularization we cannot consider a true discovery and and a false discovery in the same way as the l_1-norm. Secondly the LARS regularization regarding the way we use it can act like a Lasso regularization. In our case we notice a slight difference between the TPP mean of the Lasso and the Lars because while doing the interpolation some values can be slightly rounded. Finally we note that there is different possibility to compute a l_1-norm and as it said in the article there are many other ones.

4.3 The influence of noise in the Lasso path

In this section we are going to add a noise obtaines thanks to a normal distribution centered on 0 and with a standard deviation between 0 and 1. To do so we use the lasso_path algorithm.

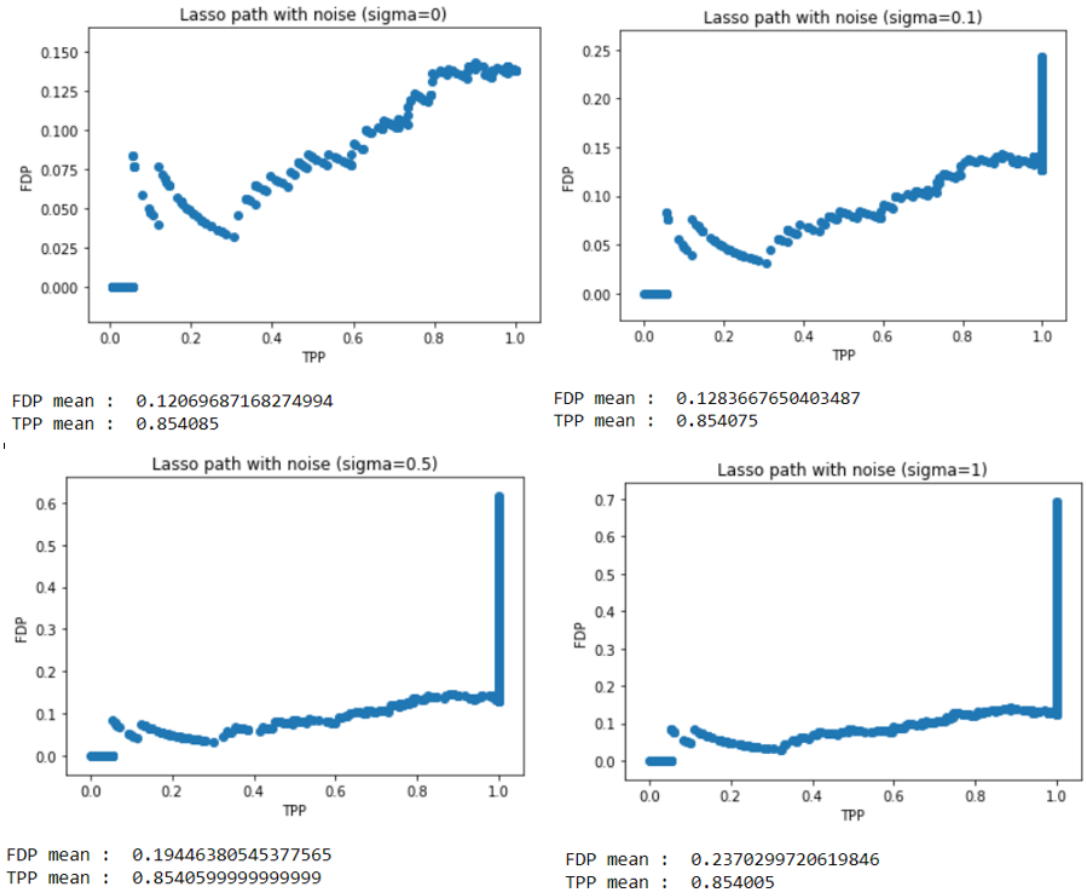


FIGURE 9 – Results of the proportion of FDP and TPP using lars_path with and without noise

We can see that, in the initial case with β_j equal to 4 for a true signal or 0 else, with a noise varying between 0.1 and 1 the Lasso the algorithm is still able to find with approximately the same ratio as TPP but the ratio of FDP increases. It is logical because the algorithm will consider some paramaters to have an importance that they doesn't because of the noise. Thus it will consider as true signals some that are not and vice versa.

4.4 Implementation of the boundary q*

In the reference article, the authors underlines a repository containing some definitions computed in Matlab allowing us to compute the boundary q*. Thus we translated the code in python and realised some tests to see the influence of the parameters :

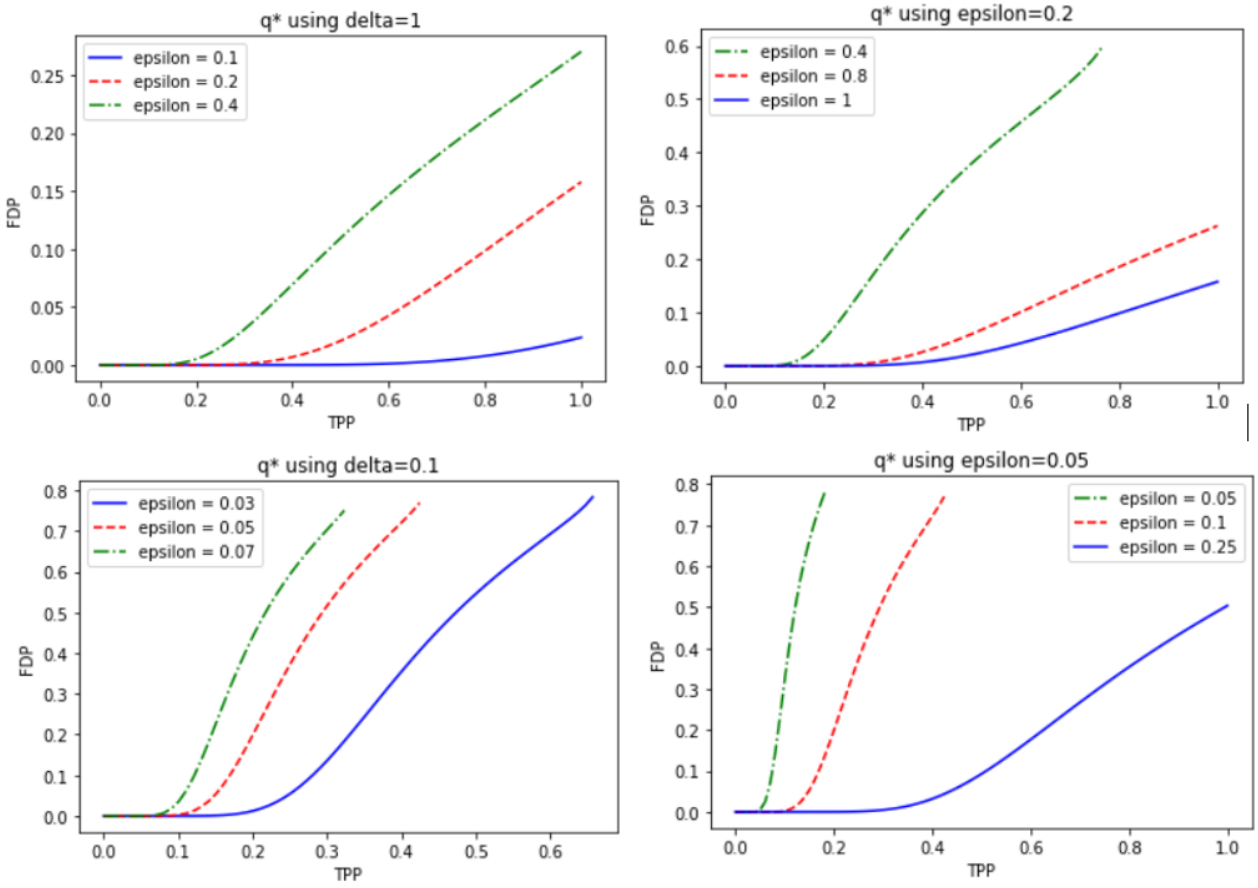


FIGURE 10 – Result of the q* boundary using different parameters delta and epsilon

We can see that we obtain the same results as the ones in the reference paper. It confirms that when ϵ increases the unreachable area is bigger and when δ increases this area is smaller. This confirms what we saw previously.

5 Conclusion

We can therefore conclude that Lasso is a linear regression method for performing feature selection in a linear sparsity regime. In spite of the efficiency of this method to filter the true variables, it turns out that Lasso can be limited in its performances : this is what the trade-off imposed by the phenomenon of shrinking coefficients shows and also what the article highlights. It is then judicious to apply parameter adjustment methods (sharpness) in order to get the Lasso path points as close as possible to the trade-off curve, in order to obtain the best possible feasible scenario. Finally, it is important to study the input parameters before deciding which regression method to use, and in the case where Lasso is the best solution, it is necessary to take into account the performance limits defined in the article and adjust the different parameters in order to select the most true features to obtain the best prediction in this linear regression problem.

6 Bibliography

- **Papers :**
  - [0] **Reference article :** Weijie Su, Ma lgorzata Bogdan and Emmanuel J. Candès. False Discoveries occur Early on the Lasso Path
  - [1] Yuchen Zhang, Martin J. Wainwright and Michael I. Jordan. Lower bounds on the performance of polynomial-time algorithms for sparse linear regression : https ://arxiv.org/abs/1402.1918
  - [2] Trevor Hastie, Robert Tibshirani, Jerome Friedman. The Elements of Statistical Learning - Data Mining, Inference, and Prediction : https ://web.stanford.edu/ hastie/ElemStatLearn/printings/ESLII_print12.pdf

- **Websites :**
  - [3] https ://towardsdatascience.com/a-comparison-of-shrinkage-and-selection-methods-for-linear-regression-ee4dd3a71f16
  - [4] Ridge and Lasso : https ://towardsdatascience.com/ridge-and-lasso-regression-a-complete-guide-with-python-scikit-learn-e20e34bcbf0b
  - [5] http ://wikistat.fr/pdf/st-m-hdstat-linselect.pdf
  - [6] https ://fr.wikipedia.org/wiki/Lasso_(statistiques)