Statistical learning VS Machine learning

2021-2022

Realized by : GEFFLOT Claire, BENKIRAN Yasmine, PENICHON Romain

Directed by : DE PERETTI Philippe

Programming in : SAS

UNIVERSITÉ PARIS 1
PANTHÉON SORBONNE

Abstract

Scientific, technical and computer advances in recent years have greatly enhanced the capacity to measure, store and process data. However, dealing with high-dimensional data leads to other issues such as scattered data, increase dissimilarities between individuals...

Then to overcome this issues, we need to find the most efficient algorithm to select only the relevant explanatory variables of a model among a large number of variables. There are several variable selection approaches. These approaches are classified into two main categories: *Statistical Learning* and *Machine Learning*.

In this paper, we will investigate whether *Statistical Learning* and *Machine Learning* algorithms perform efficiently after presenting them. To do so, we will develop four distinct Data Generating Processes and test each algorithm with different information criteria. Finally, we will figure out which algorithm is the most efficient and under what conditions.

Contents

1 Introduction

In various contexts, empirical studies are frequently devoted to the analysis of the relationship between some dependent variable and a multitude of explanatory variables that may influence the dependent variable. Mostly, information on a large number of explanatory variables is gathered and during the data analysis phase these are 'statistically screened' to find the most important ones for an appropriate description of the relationship. But, when faced with a large number of variables, how to choose a set of relevant variables ? High dimensional data sets actually disperse data. This has a tendency to skew traditional data analysis methods. Indeed, when dealing with a large number of variables, selecting the best subset of predictors that are not redundant, not particularly explanatory, or non-significant information for the model becomes nearly impossible.

In this situation, variable selection techniques are commonly used as part of statistical modeling of the observed data to limit the number of explanatory variables and to find the best-fitting subsets of variables. In other words and more precisely, to obtain an appropriate model that (i) leads to stable parameter estimates, (ii) accurately predicts future values, and (iii) allows for a comprehensible interpretation because only a few important explanatory variables are chosen [1]. There are numerous approaches for selecting variables. These methods can be divided into two main categories : Statistical Learning and Machine Learning. Statistical learning is more concerned with interpretability and description, whereas Machine Learning is more concerned with prediction.

The purpose of this paper is to describe and test the reliability of different Statistical Learning and Machine Learning methods. To do so, we will apply each selection process to our 1000 models of 100 observations each (including 50 explanatory variables with 5 that explain well and 45 that explain poorly), previously built by 4 different Data Generating Processes (DGP). Data Generating Processes follow certain hypotheses such as whether or not there is a correlation, whether or not there are outliers. This gives us the opportunity to put up an evaluation procedure. Indeed, we will know which variables must have been selected and which should not have been when performing variable selection algorithms on each data set. In other words, the purpose of this evaluation technique is to assess the subset's performance and relevance. Eventually, thanks to this amount of knowledge on the data, we will be able to choose the best model for variable selection.

This paper's overall structure has been divided into six sections : the models, the method for extracting the results, and the results themselves are divided into two subsections, *Statistical Learning* and *Machine Learning*, respectively. The models that will be used in the paper are described in detail in Section 2. Section 3 is focused on the methods applied on SAS software to simulate data and extract results. In Section 4, these results are detailed for each model. In Section 5, we will make an application on the diabetes data. Finally, Section 6 will conclude with a discussion of the paper's overall results.

2 Models

2.1 Statistical learning

"Statistical modeling is the use of mathematical models and statistical assumptions to generate sample data and make predictions about the real world" [2]. Since statistics is the mathematical study of data, statistical studies cannot be performed out in the lack of data. Statistical modeling is generally carried out according to three objectives [3]. The first one is description, it consists in describing the relationship between Y and a set of X variables. If a variable X is added or dropped to the model, the coefficients and their interpretation will change. The second objective of a statistical analysis is to explain the data through the correlation relationships, in other words how X variables are causally related to Y. For instance, in our application we can relate the determination of risk factors for diabetes based on demographic variables (age, sex, and so on). Another example consists in estimating the amount of glucose in the blood of a diabetic patient, from the infrared absorption spectrum of that patient's blood. A specific change in X predicts a specific response in Y. Finally, the third objective, known as prediction allows us to model the possibility that an individual from this same population will develop breast cancer. However, statistical learning differs from machine learning, especially because its primary goal is the explanation and knowledge of a dataset, in giving a fairly weaker prediction power [4].

Both statistical and machine learning rely on data : statistical learning is formalized as a relationship between variables, whereas machine learning provides systems the ability to automatically learn and improve from experience without explicitly programmed instructions [5]. Statistical learning therefore refers to a set of tools for modeling and understanding complex datasets. It relies on rule-based programming, this implies that it is formalized in the way variables relate to one another. In contrast to machine learning, which focuses on the most accurate predictions, statistical learning is primarily about inference, most of the idea is generated from the sample, population and hypothesis. This hypothesis may involve making specific assumptions which you will validate after creating the models. Inference consists in, for example, asking ourselves what a variable's mean in the population would be based on the mean of the same variable in a smaller sample.

2.1.1 Criteria for selecting variables

When attempting to describe or understand a phenomenon, it is desirable to define the most efficient model. As a result, it is critical to compare information criteria that we have at our disposal to choose the best explanatory model. We are now going to present the diversity of these criteria.

- F test

An F-test is any statistical test in which the test statistic has an F-distribution under the null hypothesis : therefore it will follow in law $F(q, (N - (p + 1)))$. It is most often used when comparing statistical models adjusted on a data set, in order to identify the most suitable model for the initial population from which the data were sampled. Exact "F-tests" mainly arise when the models have been fitted to the data using least squares. Mathematically, the F statistic measures the Residual Sum of Square (RSS) variation for each additional parameter in the model. Consider q explanatory variables, X a matrix with p columns where $p = q + 1$, and the model $Y = X\beta + \epsilon$ such as :

$$RSS(\beta) \ = \ \sum_{i=0}^{N}(y_i \ - \ \hat{y}_i)^2 = \ \sum_{i=0}^{N}(y_i \ - \ X\beta_i)^2 \tag{1}$$

The F test formula, according to the above equation, is :

$$F \ = \ \frac{RSS_0 \ - \ RSS_1}{q} \ \cdot \ \frac{N \ - \ (p \ + \ 1)}{RSS_1} \tag{2}$$

- Likelihood Ratio Test (LRT)

The likelihood ratio test (LRT) is a statistical test of the goodness-of-fit between two statistical models. The LRT compares two hierarchically nested models to determine whether or not adding complexity to the model (i.e. adding one or more parameters) makes the model significantly more accurate. In summary, the LRT tells us whether adding parameters to our model is useful or whether we should continue with our simpler model.
Suppose that we have a statistical model with parameter space θ. If we consider the maximum likelihood estimator of θ denoted $\hat{\theta}$, than under H0 hypothesis the same estimator will be denoted $\hat{\theta}_0$ with $\theta \in \Theta_0$. The likelihood ratio test statistic for the null hypothesis H0 is given by :

$$\lambda \ = \ \frac{-2log(L(\hat{\theta}_0)}{L(\hat{\theta})} \tag{3}$$

These q constraints under null hypothesis imply $\lambda(x_1, \cdots, x_p) \rightarrow X^2(q)$. In comparison the best model keeps being the one with the highest likelihood.

- Akaike Information Criterion (AIC)

The Akaike Information Criterion (AIC) is a prediction error estimator and give us as a result, the relative quality of statistical models for a given set of data. The AIC estimates the quality of each model, relative to other models. And the best-fit model according to AIC is the one that explains the greatest amount of variation using the fewest possible independent variables [6]. Whether, for the use of the Likelihood Ratio Test (LRT) stated above, nested models are needed, Akaike suggests the following generalized measure :

$$AIC \ = \ -2log(L(\hat{\beta})) \ + \ 2p \tag{4}$$

with p explanatory variables. The best model is given by the one which minimizes the AIC.

- Akaike Information Criterion (AICc)

The model's information score (the lower-case "c" indicates that the value has been calculated from the AIC test corrected for small sample sizes). The smaller the AIC value, the better the model fit.

$$AICc \ = \ AIC \ + \ \frac{2k^2 \ + \ 2k}{n \ - \ k \ - \ 1} \tag{5}$$

- Bayes Information Criterion (BIC)

The Bayesian Information Criterion (BIC) is based, in part, on the likelihood function, and is closely related to Akaike Information Criterion (AIC). This criterion penalizes

more strictly complex models. It is possible to increase the likelihood of fitting a model by adding parameters, however this may result in overfitting. The BIC resolves this problem by introducing a penalty term for the number of parameters in the model. The penalty term is larger in BIC than in AIC [6]. Let N be available observations number, so we shall have :

$$BIC \ = \ -2log(L(\hat{\beta})) \ + \ p \ \cdot \ log(N) \tag{6}$$

As it can be seen in its definition, this criterion is dependent on N and the relative size of N and p. However, it penalizes free parameters more severely than AIC does. And, like for the AIC, the best model is the one with the minimal BIC.

Nevertheless when the number of explanatory factors is considerable, it is difficult to explore their possible combinations (i.e. models) to compare the information criteria mentioned above. Automatic variable selection algorithms have been presented as a solution to this problem, however their functioning and outcomes differ. Let us describe these solutions :

2.1.2 Forward method

Forward selection is a variable selection method which begins with an empty model (called the null model) that contains no variables (only an intercept) [7]. On the first step, Y, the explained variable, is regressed on the most significant variable and then one explanatory variable is added at a time to the model. This explanatory variable is the one that best optimizes the information criterion such as maximizes the F-test and the LRT or minimizes the AIC and the BIC. Indeed, you have to test the various variables that may be relevant, and the 'best' variable - the most significant variable according to the tests and criterion - is added to the model. In each forward step, we add the one variable that gives the single best improvement to the model. As the model continues to improve, we continue the process, adding the most significant variables one by one and testing at each step until there are no more significant variables to add that optimizes the criterion. Indeed, once the model no longer improves with adding more variables, the process stops.

For instance (Appendix section 7), if we have a model with five explanatory variables, we start the regression with the constant only, then add each variable one by one, regressing each time the obtained part-model. As a result, we generate five models, choosing the variable with the best explanatory power, which optimizes the specified information criterion. Once we have chosen a variable, we repeat the process with the other four, testing each one separately. And the process comes to a halt either when we opted for all variables, or when information criterion are not optimized anymore.

2.1.3 Backward method

As it name indicates, this method opposes itself from the previous one. Indeed, unlike forward stepwise selection, the backward elimination technique starts from the full least squares model containing all k predictors, and then iteratively eliminate the extraneous variables one by one until a stopping condition is satisfied [1] (Appendix section 7). Indeed, once the least statistically significant variable (or the variable that minimize the less the criterion) is spotted, the algorithm follows the same pattern regressing the same model but with $k-1$ variables. The backward selection stops only when all variables have been determined as significant (or when the criterion is optimized).

2.1.4 Stepwise method

"Stepwise regression is a combination of the forward and backward selection techniques" [8]. This approach follows the same pattern as the forward selection, with some modifications : after each step in which a variable was added, all candidate variables in the model are checked to see if their significance has been reduced below the specified tolerance level. If a nonsignificant variable is found, it is removed from the model. The goal of this continuous updating is to prevent non-significant variables from appearing : when we add a new variable, a previously significant one can become non-significant. It is a phenomenon that neither forward nor backward techniques are able to detect. If there is a correlation between two variables, this could be the case. When this happens, the stepwise method selects the more significant variable and deletes the other one, as the backward approach would do. When the algorithm is unable to add new significant variables, the process comes to a halt. We will notice that the stepwise approach does not work well with a large number of explanatory variables, which is why, other algorithms such as Stagewise, will be discussed in the following section.

2.2 Machine learning

Machine learning (ML), is defined as "the study of computer algorithms that allow computer programs to automatically improve through experience"[9] offers some appealing answers to various problems, and it has been shown to be effective in developing predictive models based on vast sets of data. When applied to real-world data, supervised ML approaches may capture complicated interactions and non-linear relationships among explanatory variables, resulting in strong model performance. Machine Learning is at the crossroads of Computer Science and Statistics.

In this section we will focus on supervised machine learning and more specifically on penalized regressions. Penalized regression methods estimate the regression coefficients by minimizing the residual sum of squares (RSS) and also add a constraint on the size of the regression coefficients. This penalty on the size of the regression creates biased coefficient estimates, but it improves the model's overall prediction error by lowering the variance of the coefficient estimates. The aim of penalized regression is to identify the parameter β that minimizes an objective function:

$$\hat{\beta} \;=\; argmin_{\beta} \left\{ \frac{1}{n} \left\| Y - X\beta \right\|_2^2 \;+\; \lambda \left\| \beta \right\|_2^2 \right\} \tag{7}$$

λ seeks to strike a balance between the data's exposure and the model's complexity. A general model can result in information loss, whereas an extremely complex model can result in poor prediction due to overfitting. The model is complex if the λ is high. Now, we will discuss various approaches for penalized regression.

2.2.1 Incremental Forward Stagewise Regression (FS)

According to Ryan J.Tibshirani[10], FS follows a very simple strategy: it begins with all coefficients equal to zero, and iteratively updates the coefficient (by a small amount of the learning rate ϵ) of the most correlated variable to the residuals. The learning rate is initially quite low, but gradually, the coefficient's value goes towards the value of the *Ordinary Linear Square* (OLS), until another variable becomes more correlated with the recalculated residuals, then we modify its coefficient and so on. In this end, we will find OLS's results. The algorithm will end until there is no longer any correlation between explanatory variables and residuals. For clarity, we will go over each step of the algorithm:

1. Find residuals $r_i = y$ - $\bar{y}$ with $\beta_i = 0$
2. Determine the explanatory variable X_n most correlated with r_i
3. Update the coefficient β_i as follow : $\beta_{i+1} = \beta_i + \delta_i$ where $\delta_i = \epsilon \cdot sign(corr(r_i, X_n)$
4. Update the residuals as : $r_{i+1} = r_i$ - $\delta_i \mathrm{X}_n$
5. Steps 2 and 4 should be repeated until no correlation is detected between the residuals and any other explanatory variable in the set.

2.2.2 Least-angle regressions (LARS)

LARS algorithm is used to fit linear regression models to high-dimensional data, developed by Bradley Efron and al. [11] in 2004. It generates a series of regression models, one parameter at a time, ending with the full least squares solution after all parameters have been entered into the model. *LARS* algorithm [12] is:

1. Normalise all values to have a mean of 0 and a variance of 1.

2. Find residuals $r_i = y$ - $\bar{y}$ with $\beta_i = 0$
3. Find the explanatory variable X_i the most correlated with r_i, then move the regression line in that direction until we reach another variable that has the same or higher correlation.
4. Gradually move β_i towards its correlation coefficient (with r obtained using OLS), until a variable β_j has a higher correlation with the observed residuals. Increase the information set.
5. When we have two variables that have the same correlation, move the regression line to an angle that lies between the two (i.e. the least angle between the two variables).
6. Gradually move (β_i;β_j) towards their OLS estimators with the residuals, until a variable β_c has a stronger correlation with the observed residuals. Increase the information set.
7. The loop continues until all the coefficients are in the model.

This method has two major drawbacks: firstly, the method is very sensitive to noise, and secondly, the algorithm is less efficient when the model presents multi-collinearity.

Finally, *LARS* can recreate *Forward Stagewise* and *LASSO* regressions under specific conditions.

LARS to *LASSO*:
Simply eliminate the explanatory variables if a non-zero coefficient reaches zero to obtain the LASSO regression using the LARS method. Then, recalculate the OLS and continue the method to produce the LASSO estimator.

2.2.3 Ridge

Ridge regression uses an *L2 norm* penalty (Appendix section 7), which reduces the value of the regression coefficients (shrinkage), but not to 0. It stabilises the estimation of the coefficients in the case of strong correlations between the variables. The estimator is obtained by minimising the following criterion:

$$\min_\beta \left\{ \frac{1}{n} \|Y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \right\} \text{ with } \lambda \geq 0 \quad (8)$$

The estimator will be :

$$_{(\lambda)} = \left(X^T X + \lambda I \right)^{-1} X^T Y \quad (9)$$

This estimator was introduced by Hoerl and Kennard in 1970 [13]. Although this estimator is biased, the $\hat{\beta}_R$ estimator has a lower variance than the least squares estimator, so it is a matter of finding the right bias-variance compromise. Adding a term on the diagonal of $X^T X$ (with $X = (X_1, ..., X_n)^T$) stabilises the calculations of the inverse, making this estimator more stable to the multi-colinearity problem.

→ Highest coefficients decrease the most.

→ No non-zero coefficients are reduced to zero, so all features stay in the model.

→ Ridge allows to lessen the influence of some variables by lowering their coefficients, but it does not allow to delete the poor ones.

2.2.4 Least Absolute Shrinkage and Selection Operator (LASSO)

Introduced by Tibshirani in 1996 [14], *LASSO* uses as a penalty function the *L*1 *norm* (Appendix section 7) of the parameter vector β which can combine the capabilities of a classical variable elimination by automatically removing small coefficients, and the capabilities of an *L*2 *norm* to stabilize the estimates. *LASSO* estimator is a solution to the following optimization problem:

$$\min_\beta \left\{ \frac{1}{n} \|Y - X\beta\|_2^2 \; + \; \lambda \; \sum_i |\beta_i| \right\} \text{ with } \lambda \geq 0 \quad (10)$$

λ denotes the significance of the penalty. Specifically, for $\lambda = 0$ no penalty is applied, the least squares estimator is found and for $\lambda \to \infty$ all variables are associated with a zero estimated coefficient. The use of the *L*1 *norm* reduces the small coefficients to exactly zero and thus simplifies the model. This norm has the advantage of being convex and guarantees the uniqueness of the solution (if n > p and the correlation between the variables is not severe ; n = sample p = number of explanatory variables). Beware that LASSO can significantly bias the estimates by reducing the coefficients too much.

→ In comparison with Ridge, LASSO removes some variables from the model.

→ All coefficients reduced equally.

2.2.5 Elastic Net

Elastic Net Regression proposed by Zou and Hastie [15] in 2005 combines *LASSO* (*L*1 *norm*) and *Ridge* (*L*2 *norm*), which is beneficial when the number of variables is large that we don't know their usefulness on the explained variable, nor if they are correlated with each other.

$$\min_\beta \left\{ \frac{1}{n} \|Y - X\beta\|_2^2 \; + \; \lambda_1 \; \sum_i |\beta_i| \; + \; \lambda_2 \; \|\beta\|_2^2 \right\} \quad (11)$$

When two relevant variables are correlated, *LASSO* will tend to select only one of them. Adding *Ridge* penalty will favour the selection of the whole. However, *Elastic Net* regression may have some limitations due to the need to set two penalty parameters (λ_1 = *LASSO* regression and λ_1 = *Ridge* regression) which can lead to poor performance in prediction.

3 Method

This paper aims to test the performance of the variable selection algorithms presented in the previous sections. To do so, we need to generate data sets with a known and precise correlation structure (manually parameterized). Indeed, in order to determine whether or not an algorithm performs effectively, it is necessary to ensure that the data we work on follows a precise correlation structure. Thus, we created 1,000 databases from 5 DGPs (Data Generating Processes), for a total of 4,000 databases. Each database contains a set of 51 variables (1 explained variable Y and 50 explanatory variables $X1$ - $X50$) and 100 observations. Finally, we test each algorithm on the 4,000 datasets and conclude on their results.

3.1 Multivariate normal distribution

For each Data Generating Process, 1,000 samples of size $N = 100$ were generated from a multivariate normal distribution. So, each vector of a sample follows a univariate normal distribution and the relationship between two vectors in a sample is determined by a specified variance covariance matrix.

3.2 Toeplitz matrix

Simulating a specific correlation structure between variables involves creating a correlation matrix. Correlation matrices are positive symmetric semi-defined. In order to fulfill these conditions, we chose to use Toeplitz matrix. The Toeplitz matrix is a diagonal constant matrix, which means that the entries of the matrix depend only on the differences of the indices, if it is symmetric, it has all the properties of a correlation matrix. In the symmetrical case, we obtain a correlation matrix with a diagonal of 1 and covariances between the variables of different values according to the DGP. In absolute terms, two strongly correlated variables are considered to have a correlation coefficient between 1 and 0.6 ; two moderately correlated variables have a correlation coefficient between 0.6 and 0.3 ; two weakly correlated variables have a correlation coefficient between 0.3 and 0.

3.3 Data Generating Processes (DGP)

Each DGP is made up of 1,000 databases, themselves composed of 100 rows (sample size) and 51 columns (number of variables), in which the first column is column Y and is followed by columns $X1$ to $X50$.

3.3.1 First Data Generating Process (DGP1)

First of all, we generated a classical dataset representing the perfect model, i.e., without the presence of correlation between variables, and without extreme values. For the creation of a database, we simulated a dataset (the X matrix) of 50 variables using the multivariate normal distribution and the identity matrix as the variance-covariance matrix:

$$\begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix}$$

Then, we generated Y based on the following equation :

$$\mathrm{Y} = 1.5 \cdot X1 \ + \ 0.9 \ \cdot \ X2 \ + \ 1 \ \cdot \ X3 \ + \ 0.1 \ \cdot \ X4 \ - \ 0.5 \ \cdot \ X5 (12)$$

Finally, we joined the column Y with the columns from $X1$ to $X50$, to create a database with 51 columns. The value of Y is dependent on the randomly obtained values of the variables $X1$ to $X5$. We could interpret the relationship between Y and the variables $X1$ to $X5$ as positive and negative correlations, and the relationship between Y and the variables $X6$ to $X50$ as zero correlations.

3.3.2 Second Data Generating Process (DGP2)

For the second DGP, we generated a dataset in which only the variables $X1$ to $X5$ are positively correlated with each other, whereas the variables $X6$ to $X50$ are not. To create this database, we simulated a first data set (the Xa matrix) of 5 variables ($X1$ to $X5$) following a multivariate normal distribution (mean = 0), and with positive and symmetric correlation between the first 5 variables. The correlation matrix for the variables $X1$ to $X5$ was done using the Toeplitz matrix. We obtain a variable $X1$ which is weakly correlated to $X2$, moderately correlated to $X3$ and $X4$, and strongly correlated to $X5$ (COV($X1$,$X5$) = 0.714).

cov				
1	0.2857143	0.4285714	0.5714286	0.7142857
0.2857143	1	0.2857143	0.4285714	0.5714286
0.4285714	0.2857143	1	0.2857143	0.4285714
0.5714286	0.4285714	0.2857143	1	0.2857143
0.7142857	0.5714286	0.4285714	0.2857143	1

Figure 1 : Correlation between the 5 variables ($X1$ to $X5$)

Then, we simulated a second data set (the Xb matrix) of 45 variables ($X6$ to $X50$) following a multivariate normal distribution (mean = 0) and zero correlation between these variables. For the correlation matrix, we used the identity matrix as in DGP1. After simulating the two datasets, we created a matrix X that joins the two datasets, in order to have only one data table. Then, we generated Y with the same equation in DGP1.

Finally, we joined the Y column to the X matrix, in order to create a database with 51 columns in which only Y is correlated to the first 5 variables, which are themselves correlated with each other. The variable Y is not correlated to the variables $X6$ to $X50$, and they are themselves not correlated with each other.

3.3.3 Third Data Generating Process (DGP3)

For the third DGP, we generated a dataset in which the variables $X1$ to $X50$ are positively correlated with each other. To create the database, we simulated a dataset (X) of 50 variables ($X1$ to $X50$) following a multivariate normal distribution with mean 0, and positive and symmetric correlation between the 50 variables. The correlation matrix for the variables $X1$ to $X50$ was done using the Toeplitz matrix. Each variable is highly

correlated with the closest variables in the matrix. The further away a variable is from another, the less correlated it is. For example, we obtain a variable $X1$ that is highly correlated with the variables $X2$ to $X16$, moderately correlated with the variables $X17$ to $X36$, and weakly correlated with the variables $X37$ to $X50$.

	COL1	COL2	COL3	COL4	COL5	COL6	COL7	COL8	COL9	COL10	COL11	COL12	COL13	COL14	COL15	COL16
ROW1	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84	0.82	0.8	0.78	0.76	0.74	0.72	0.7
ROW2	0.98	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84	0.82	0.8	0.78	0.76	0.74	0.72
ROW3	0.96	0.98	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84	0.82	0.8	0.78	0.76	0.74
ROW4	0.94	0.96	0.98	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84	0.82	0.8	0.78	0.76
ROW5	0.92	0.94	0.96	0.98	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84	0.82	0.8	0.78
ROW6	0.9	0.92	0.94	0.96	0.98	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84	0.82	0.8
ROW7	0.88	0.9	0.92	0.94	0.96	0.98	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84	0.82
ROW8	0.86	0.88	0.9	0.92	0.94	0.96	0.98	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84
ROW9	0.84	0.86	0.88	0.9	0.92	0.94	0.96	0.98	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86
ROW10	0.82	0.84	0.86	0.88	0.9	0.92	0.94	0.96	0.98	1	0.98	0.96	0.94	0.92	0.9	0.88
ROW11	0.8	0.82	0.84	0.86	0.88	0.9	0.92	0.94	0.96	0.98	1	0.98	0.96	0.94	0.92	0.9
ROW12	0.78	0.8	0.82	0.84	0.86	0.88	0.9	0.92	0.94	0.96	0.98	1	0.98	0.96	0.94	0.92
ROW13	0.76	0.78	0.8	0.82	0.84	0.86	0.88	0.9	0.92	0.94	0.96	0.98	1	0.98	0.96	0.94
ROW14	0.74	0.76	0.78	0.8	0.82	0.84	0.86	0.88	0.9	0.92	0.94	0.96	0.98	1	0.98	0.96
ROW15	0.72	0.74	0.76	0.78	0.8	0.82	0.84	0.86	0.88	0.9	0.92	0.94	0.96	0.98	1	0.98
ROW16	0.7	0.72	0.74	0.76	0.78	0.8	0.82	0.84	0.86	0.88	0.9	0.92	0.94	0.96	0.98	1

Figure 2 : Correlation between the first 16 variables ($X1$ to $X16$)

Then we generate Y according to the the same equation in DGP1. Finally, we joined the column Y to the matrix X, in order to create a database with 51 columns in which Y is the result of the random generation of the first 5 variables. The variables $X1$ to $X50$ are correlated with each other in a positive and symmetric way.

3.3.4 Fourth Data Generating Process (DGP4)

For the DGP4, we generated a data set in which the variables $X1$ to $X50$ are not correlated but have extreme values. For the creation of the database, we simulated a dataset (the X matrix) of 50 variables using the multivariate normal distribution with extreme values and used the identity matrix as the variance covariance matrix. Then, we generate Y via the same equation in DGP1. Finally, we joined the column Y to the matrix X, in order to create a database with 51 columns in which Y is the result of the random generation of the first 5 variables. The variables $X1$ to $X50$ have extreme values without correlation.

Variable	N	Moyenne	Médiane	Ec-type	Minimum	Maximum	Skewness
Y	100	1.4357788	1.2666021	2.7432659	-3.5052689	11.6194136	1.0218561
X1	100	0.5749834	0.3695833	1.2705384	-1.8841834	4.1277627	0.8081053
X2	100	0.4279570	0.2155005	1.1723277	-1.4596133	3.9475644	1.1410578
X3	100	0.4440444	0.2189416	1.0448656	-1.3768446	3.5905765	0.9260893
X4	100	0.5854832	0.3172623	1.4435285	-1.6992922	4.9062037	1.0307473
X5	100	0.6569685	0.4345195	1.4035370	-1.7549820	5.1623703	1.0766524
X6	100	0.7432935	0.3530501	1.3984719	-1.5521474	6.3305478	1.3630765
X7	100	0.6333892	0.4746197	1.4066915	-1.8230100	6.5342292	0.9609556
X8	100	0.4653106	0.3047884	1.0447579	-1.6616973	3.3051792	0.6659390
X9	100	0.5737427	0.2851955	1.3423104	-1.8445882	6.9105051	1.4074521
X10	100	0.4538237	0.2725889	1.0845913	-1.5619963	3.4377514	0.5362435

Figure 3 : Maximum and minimum value per random variable

3.4 Results' extraction

To evaluate the performance of each method, we calculate the probability that the selection procedure used selects only the 5 variables of interest (i.e., variables $X1$ to $X5$). When the selection procedure is run for a "generated data table", the selected variables are assigned a correlation coefficient. By extension, the variables without an assigned coefficient are the "unselected" variables, they appear as a missing value in the coefficient results, these variables are automatically removed. Then we find a new table that we will call "selection table" with only the columns of the selected variables. For example, if in a generated data table, the method used selects only the variables $X1$ to $X10$, then the selection table will be composed of the columns from $X1$ to $X10$.

In order to compare the different selection methods[16], we first compare the individual selection frequencies of each variable according to the different data sets. And in a second step, we compare the selection frequencies of the 5 variables of interest only according to the different data sets.

For the individual frequency, we have created a table that we will call the "selection table in 1", allowing to gather all the selection tables. Then, we replaced each coefficient of the "selection table in 1" by 1 and each missing value by a 0. The resulting matrix will be used to build a bar chart that gives the frequency of each variable that has been chosen individually. The frequency of selection of each variable obtained is divided by 1,000 to obtain the frequency of selection in percentages, so we can use them to express the likelihood of variable selection on a scale ranging from 0 to 100.

For the frequency of selection of the 5 variables of interest only, we build a new database consisting of 0 and 1, which assigns 1 when these 5 variables are selected from the same dataset ; 0 if at least one variable is not selected with the others. For example, we define $stepwise12345$, as the joint selection of $X1$, $X2$, $X3$, $X4$, and $X5$ with the *Stepwise* method : if this condition is fulfilled, the value 1 is written in the column, otherwise 0. Once this is done, we just need to count the number of 1's among the 1,000 databases and divide the frequency by 1,000 to get the results in percentages. If the joint frequency is 0/100, this means that $X1$, $X2$, $X3$, $X4$, and $X5$ are never selected together, whereas if the joint frequency is 100/100, then these variables of interest are joined each time.

We will have for each stated procedure (*Stepwise*, *Backward*, *Forward*, *LARS*, *LASSO*, *Elastic Net*) two histograms, one represents the probability that a variable is selected among the 1,000 regressions, and the other represents the probability that the 5 variables of interest are uniquely selected among the 1,000 regressions.

4 Results

Now that we have the graphs for each model's probability of selecting a variable, as well as the joint probabilities of selecting multiple variables at the same time, we will summarize them. Indeed, we studied several selection criteria for each approach, including *SBC*, *AIC*, and *AICc*. However, we will only provide the most interesting criteria and method for each DGP in this part. All the criteria can be found in our code *SAS*. Finally, we will compare the most relevant method between *Statistical Learning* and *Machine Learning*.

4.1 First Data Generating Process (DGP1)

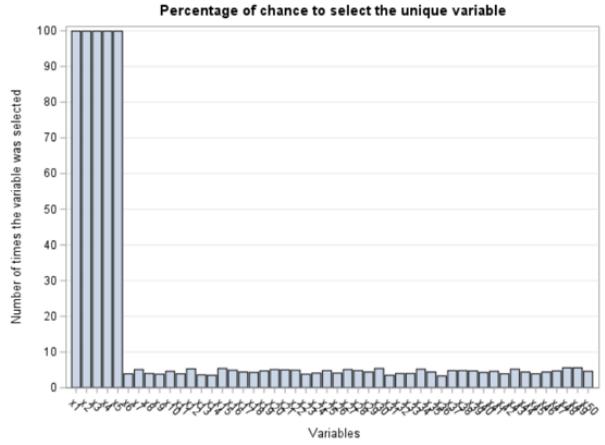
4.1.1 Statistical Learning

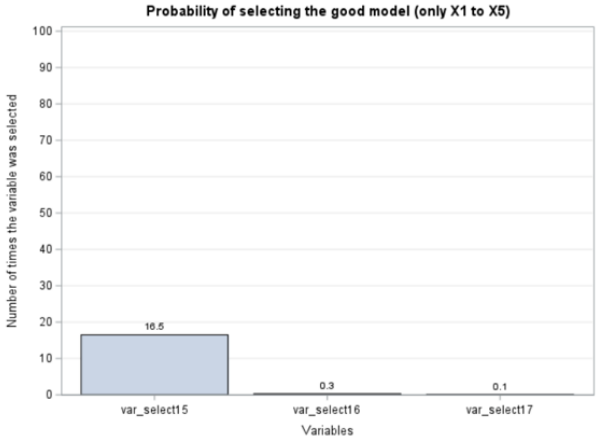
First, we will compare the different methods of *Statistical Learning* according to the imposed selection criteria. The selection methods tested in this section are *Stepwise*, *Forward*, and *Backward*.

Two graphs are shown for each model that was tested. The individual selection probabilities of the variables are shown on the left. The joint selection probabilities (i.e. the selection of several variables at the same time) are shown on the right. We will mostly rely on the graph of joint selection probabilities since it allows us to assess the efficacy of the method.

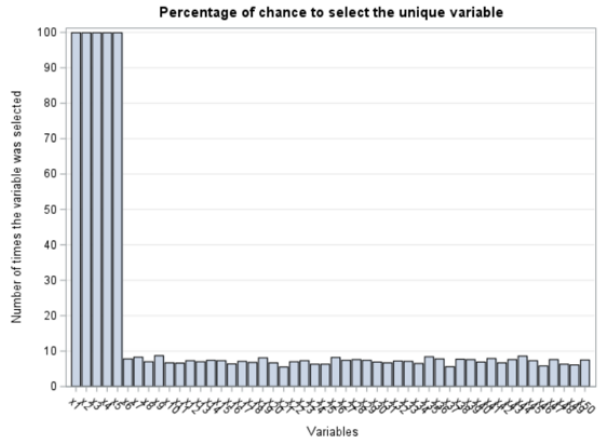
- SBC

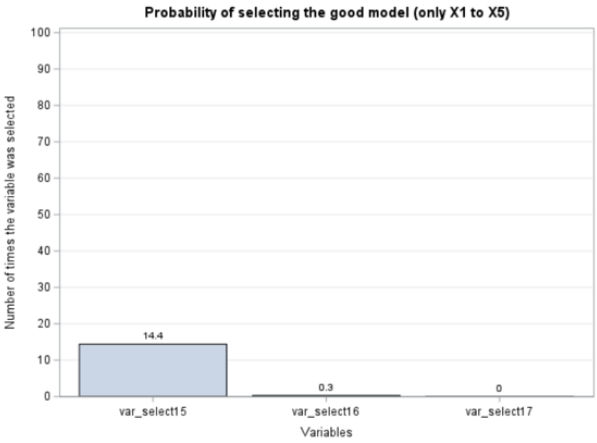
Forward





Backward

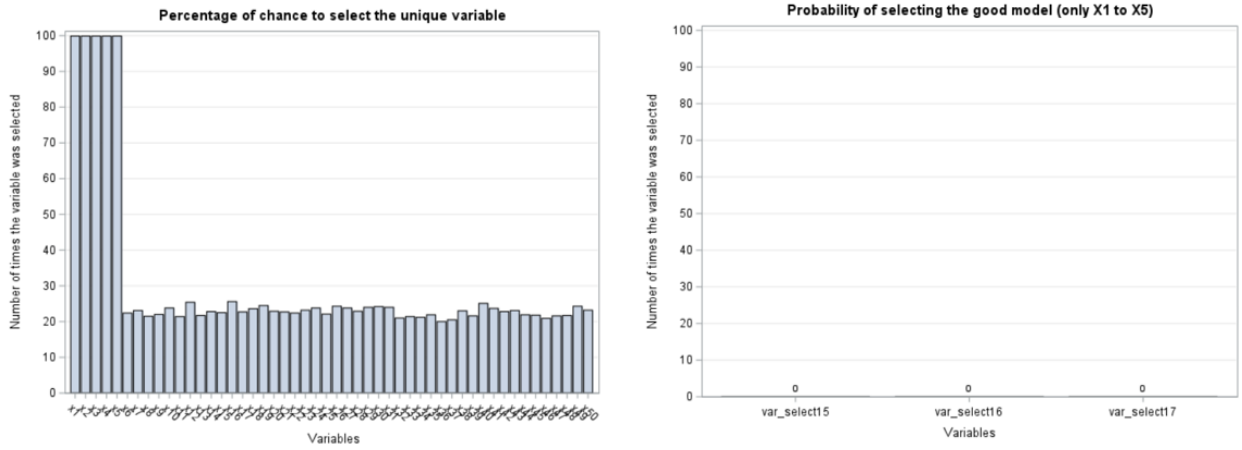




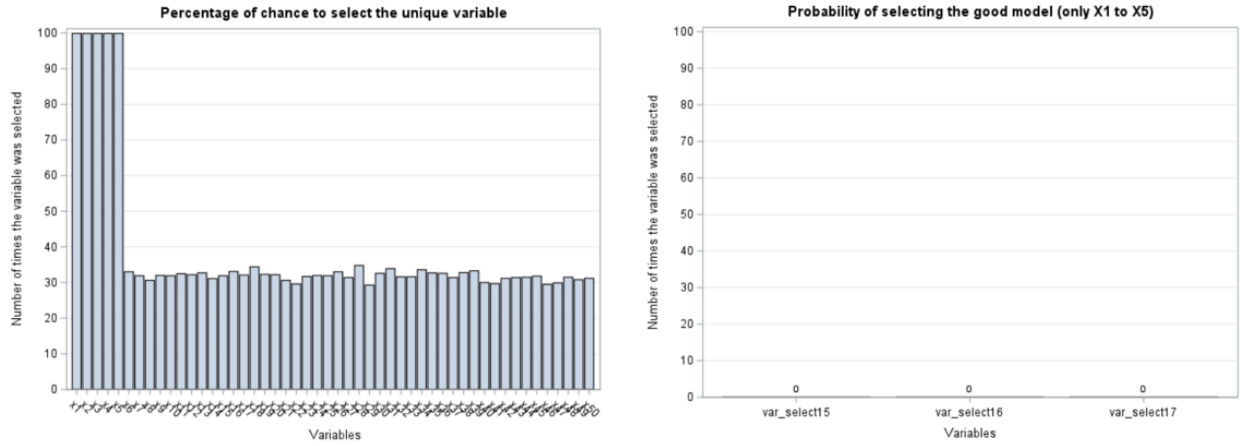
We can see that all of the variables of interest have been individually selected using the SBC selection criterion. However, we see that, on average, the selection methods only discover the variables of interest together in 15% of the cases, which is a low percentage. Even though this selection criterion does not yield the best results, we may conclude that it is suitable for DGP1 scenario. The *Forward* approach, which has a perfect selection probability of 16.5 percent, would perform best with this selection criterion.

- AIC

Forward



Backward



With the AIC selection criterion, we chose not to present the individual selection graph of the *Forward* method because this graph leads to the same result as the *Stepwise* method. Whatever the method, we can observe that all the variables of interest have been selected individually, but we can also see that the other variables ($X6$ to $X50$) have been strongly selected with respect to the previous criterion, we call this "overfitting". The latter is defined by the fact that a method selects many variables when it has to find the model justifying the value of Y, which tends to lead to bad results. On the side of the joint selection, we notice that the selection methods find the joint variables of interest in 0% of the cases. This result is explained by a high overfitting rate, which means that these selection models will tend to select more variables than expected. So it is unlikely that these selection procedures will find the right variables with this selection criterion. We can conclude that this selection criterion is not at all adapted to the conditions of DGP1 whatever the method.

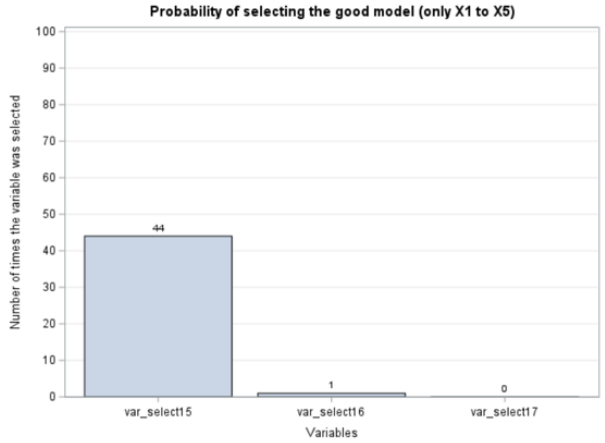
To conclude, for databases of the DGP1, the most efficient selection procedure method in *Statistical Learning* is the *Forward* method with the SBC selection criterion, proposing a 16.1% probability of selection of the variables of interest joined. The *Backward* method is not considered as efficient because this method tends to overfitting whatever the selection criterion used, thus giving unconvincing results.

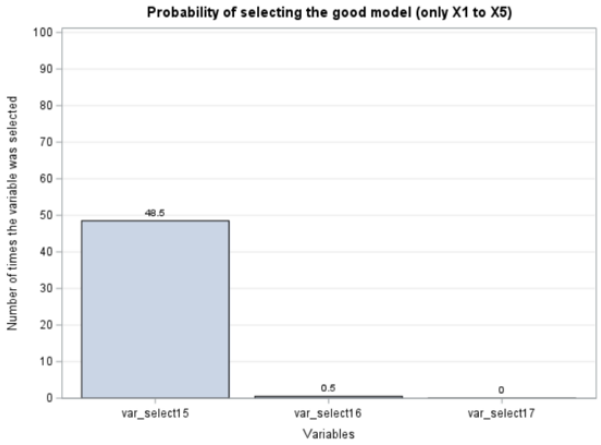
4.1.2 Machine Learning

In this part, we will compare the different methods of *Machine Learning* according to the imposed selection criteria. The selection methods evaluated in this section are *LARS*, *LASSO*, and *ElasticNet*.

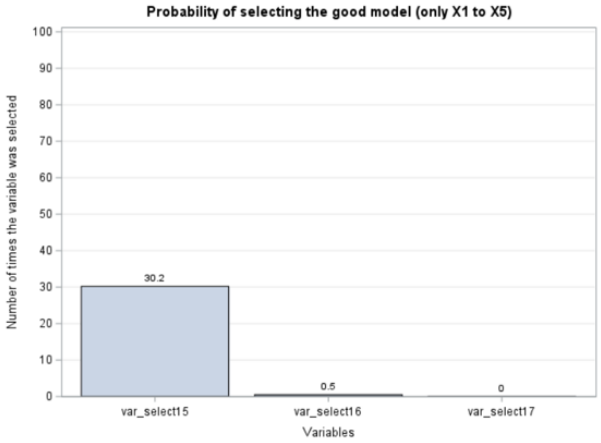
- AICc

LARS **LASSO**





ElasticNet



With the AICc selection criterion, we notice that the selection methods find the joined variables of interest in 42% of the cases on average, which corresponds to about the same probability as with the SBC and BIC selection criteria. *ElasticNet* with AICc has a probability of finding the joint variables of interest of 30%. The *Lasso* procedure has a probability of 48.5% and *LARS* a probability of 44%. The procedure that would work best with this selection criterion is the *LASSO* method which has a perfect selection probability of 48.5%.

To conclude, the most efficient selection procedure in *Machine Learning* in this DGP1 is the *LASSO* method with the AICc selection criterion, proposing a joint probability of selection of the variables of interest of 48.5%. We can also say that *LARS* can be a good selection tool in view of its selection probabilities very close to those of *LASSO*. *ElasticNet* has higher selection probabilities than the *Statistical Learning* procedures, but it is not the best selection tool in this context. If we compare the best procedure in *Statistical Learning* (*Forward*) and the best one in *Machine Learning* (*LASSO*), it is obvious that the *LASSO* procedure with the AICc selection criterion (48.5% success rate) is more successful than the *Forward* procedure with the SBC selection criterion (16.5% success rate). The *LASSO* procedure with the AICc selection criterion is therefore the most efficient method to find the variables of interest in a DGP1 type database.

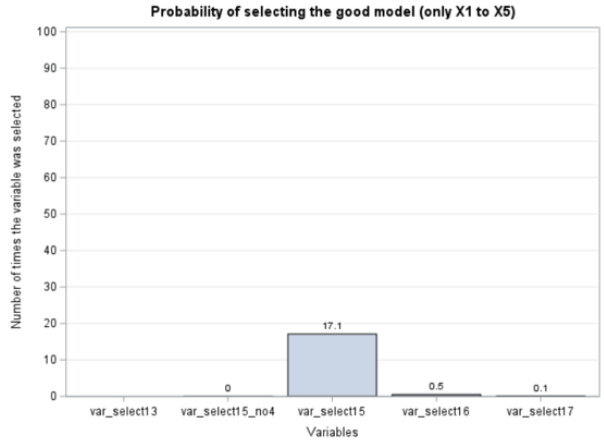
4.2 Second Data Generating Process (DGP2)

In this section, we have retained the approach used in DGP1 and adapted it to DGP2.

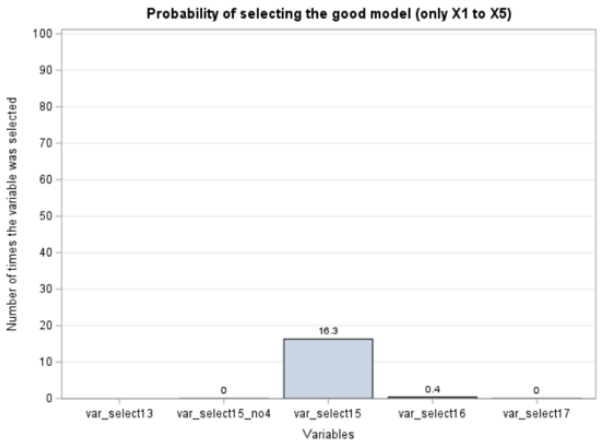
4.2.1 Statistical Learning

- SBC

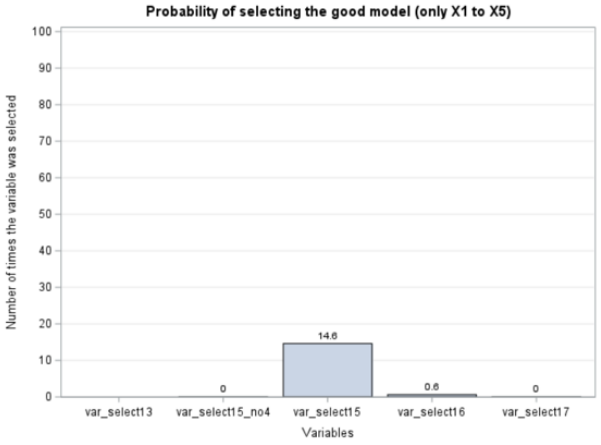
Stepwise



Forward



Backward



It can be seen that every time the SBC selection criterion was used, all variables of interest were selected individually. However, the selection methods only find the joint variables

of interest in an average of 16% of cases. It is also noted that there is a small difference compared to the average joint selection probability performed with DGP1 (=15%). The *Stepwise* approach, which has a perfect selection probability of 17.1%, would perform best with this selection criterion. We may deduce that *Forward* can operate well also with this selection criterion because *Stepwise* and *Forward* have similar selection procedures.

- AIC

We choose not to provide the individual selection graph of the Forward technique using the AIC selection criterion because it yields the same result as the Stepwise graph. Whatever method is used, we can see that in 100% of the cases, all of the variables of interest have been individually selected, but we can also see that the other variables (X6 to X50) have been strongly selected compared to previous analyses, indicating that we are dealing with a high overfitting rate. On the joint selection side, we can see that the approaches only locate the joint variables of interest in 0% of the situations. This finding is explained by a significant overfitting rate, which suggests that these selection models will tend to choose more variables than expected. With this selection criterion, no process can operate.

- AICc

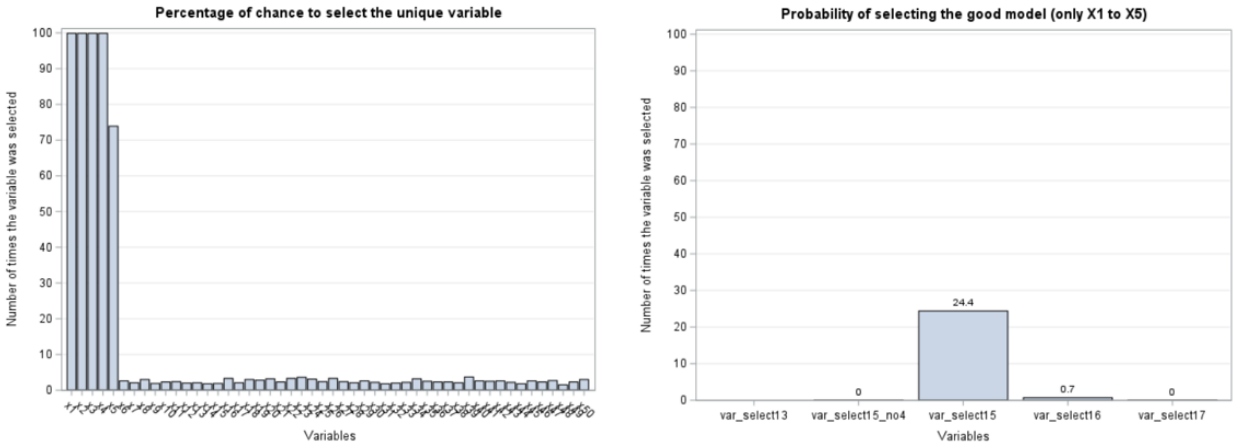
We get the same conclusion with the AICc selection criterion as we did with the prior selection criteria. With this selection criterion, it is doubtful that these selection processes will locate just the good variables.

To conclude, we know that SBC is the only relevant selection criterion for *Statistical Learning* processes based on the study of the produced outcomes. However, it has been noted that the results obtained are very similar, if not identical, to those found in the examination of the DGP1 selection technique results. This demonstrates that the DGP2's positive correlation between variables X1 to X5 has no effect on the likelihood of selecting the variables of interest for *Statistical Learning* techniques.The most efficient selection procedure approach in Statistical learning for databases of the DGP2 type is the Stepwise and Forward methods with the SBC selection criterion, which propose a joint probability of selection of variables of interest of 17.1 percent and 16.3 percent, respectively. Backward is not regarded efficient since it tends to overfit whatever selection criterion is used, resulting in unreliable results.

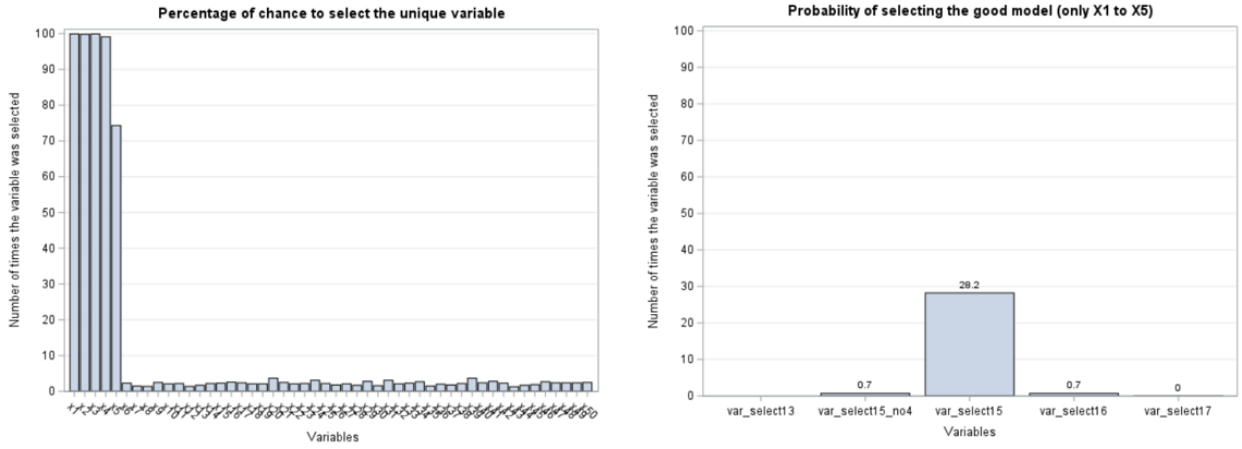
4.2.2 Machine Learning

- SBC

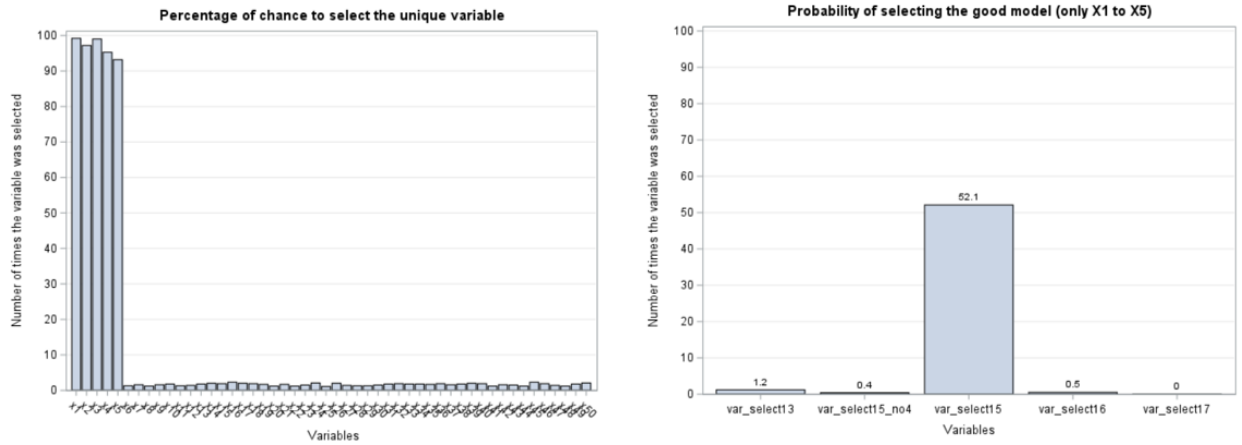
LARS



LASSO



ElasticNet



With the SBC selection criterion, we can see with *LARS* and *LASSO* procedures, that only the first four variables of interest were selected individually every time. We also see that the 5th variable of interest ($X5$) was selected in 72% of the cases. To begin with, this is owing to the strong link between $X5$ and $X1$ (programmed in DGP2). Secondly, when *LARS* or *LASSO* have to choose variables, they will go with the ones that are the least associated but also the most significant. As a result, when they come across two variables that are highly correlated, they must select between one of the two variables since one of the two variables is not the proper variable, according to them (variable of interest). They have to choose between $X1$ and $X5$ in this scenario, which explains why $X5$ isn't usually chosen. For *ElasticNet* method, the 5 variables of interest are selected individually in 90% of the cases. Whatever the procedure, we can see that the other variables ($X6$ to $X50$) have a low probability of being selected individually. Looking at the graphs on the right, we notice that *ElasticNet* procedure is the most efficient (with a joint selection probability equal to 52.1%) among the 3 procedures tested. This can be justified by the fact that *ElasticNet* with the SBC criterion is less sensitive to correlations, so it will have a greater tendency to select both the $X1$ and $X5$ variables, unlike *LARS* and *LASSO*. The procedure that would work best with this selection criterion is *ElasticNet* method which has a perfect selection probability of 52.1%.

With the AIC or AICc selection criterion, we can see that *ElasticNet* technique selects all of the variables of interest separately in more than 90% of the cases. By comparing the joint selection graphs of *ElasticNet* with AIC and *ElasticNet* with AICc, we can see that *ElasticNet* approach performs the best with the 2 criteria (with a combined

selection probability of around 40%). This can be explained by the fact that *ElasticNet*, unlike *LARS* and *LASSO*, is less sensitive to correlations when using the AIC (or AICc) criterion, hence it is more likely to choose both the $X1$ and $X5$ variables.

To conclude, *ElasticNet* method with the SBC selection criterion is the most efficient selection procedure in *Machine Learning* in this context of DGP2, with a selection probability of 52.1% for the variables of interest combined. *LARS* and *LASSO* have better selection probability than *Statistical Learning* processes, but they aren't the greatest selection strategies in this case.

We can see that *ElasticNet* process with the SBC selection criterion (52.1% of success) is more efficient than the *Stepwise* procedure with the SBC selection criterion when we compare the best *Statistical Learning* (*Stepwise*) and the best *Machine Learning* (*ElasticNet*) procedures (17.1% of success). The most effective technique for finding the variables of interest in a DGP2 type database is to use *ElasticNet* with the SBC selection criterion.

4.3 Third Data Generating Process (DGP3)

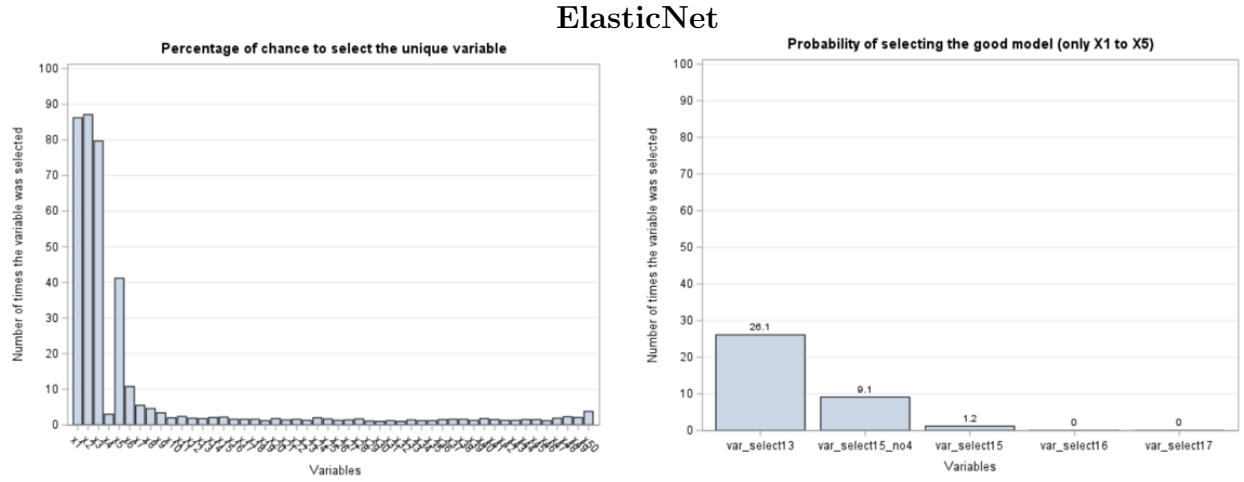
Whatever approach was used, we can see that all of the variables of interest $X1$, $X2$, $X3$, $X5$ were independently selected in each case. The presence of correlation between the variables, as well as the fact that its multiplier coefficient ($= 0.1$) is near to 0, explains why the variable of interest $X4$ is not selected in all circumstances. This suggests that the selection methods are less likely to choose this variable since the multiplier coefficient of $X4$ is insignificant to support the value of Y. The fact that the models choose the variable $X4$ in a poor way has consequences for the evaluation of the capabilities of the selection methods that will be studied after.

4.3.1 Statistical Learning

The *Stepwise* technique with the SBC selection criterion is the most efficient selection procedure approach in *Statistical Learning* for databases of the DGP3 type, with a probability of selection of the joint variables of interest of 18.2%. However, because the outcomes of *Stepwise* and *Forward* are so similar, we may conclude that *Forward* can also be an efficient approach when using the SBC criterion. *Backward* is not regarded efficient since it tends to overfit whatever selection criterion is used, resulting in unreliable results.

4.3.2 Machine Learning

- SBC



ElasticNet approach is the only method that finds the joint variables of interest with a probability of 1.2% using the SBC selection criterion. Due to under-fitting, *LARS* and *LASSO* do not uncover any joint variables of interest. When it comes to find the model that justifies the value of Y, the latter is determined by the fact that a technique selects the fewest variables possible. *LARS* and *LASSO* methods, when used with the SBC selection criteria, are ineffective. *ElasticNet* approach, which has a perfect selection probability of 1.2%, is the *Machine Learning* procedure that would function best with this criterion. Because this result is so low, we can conclude that *ElasticNet* approach using the SBC criterion is ineffective.

ElasticNet approach is the only method that finds joint variables of interest with a probability of 1% using the AIC and AICc selection criteria. *LARS* and *LASSO*, on the other hand, do not identify any joint variables of interest. The *LARS* and *LASSO* procedures, in combination with the AICc selection criteria, are ineffective. The *ElasticNet* approach, which has a perfect selection probability of 1%, is the *Machine Learning* procedure that would function best with this requirement. Given this poor result, we may conclude that *ElasticNet* approach using the AIC and AICc criterion is also ineffective.

To summarize, *ElasticNet* technique with the SBC selection criterion is the most efficient selection procedure method in *Machine Learning* for databases of the DGP3 type, with a selection probability of the variables of interest linked of 1.2%. *ElasticNet* is more efficient than *LASSO* because it "combines the regularization of both Lasso and Ridge." The advantage is that the high collinearity coefficient is difficult to eradicate. As a result, no *Machine Learning* approach can be regarded truly efficient because it tends to underfit whichever selection criterion is used, resulting in unconvincing results.

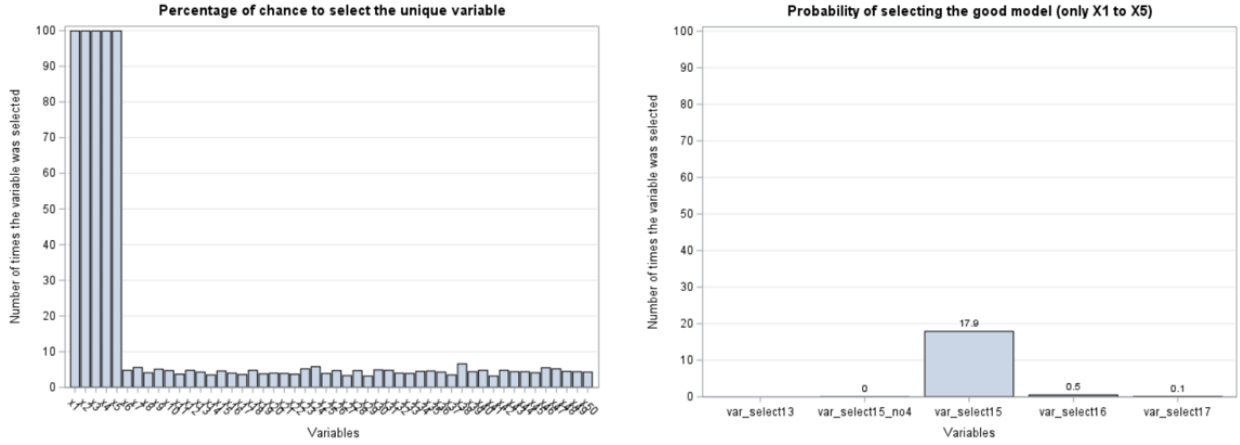
When we compare the best procedure in *Statistical Learning* (*Stepwise*) with the best procedure in *Machine Learning* (*ElasticNet*), we can clearly see that the *Stepwise* procedure with the SBC selection criterion (18.2% success rate) is more efficient than *ElasticNet* procedure with the SBC selection criterion (1,2% success rate), because when there is a correlation between the variables, the *Machine Learning* methods tend to underfit whatever the selection criterion is. Finding the variables of interest in a DGP3 type database stepwise with the SBC selection criterion is the most efficient way.

4.4 Fourth Data Generating Process (DGP4)

4.4.1 Statistical Learning

- SBC

Forward

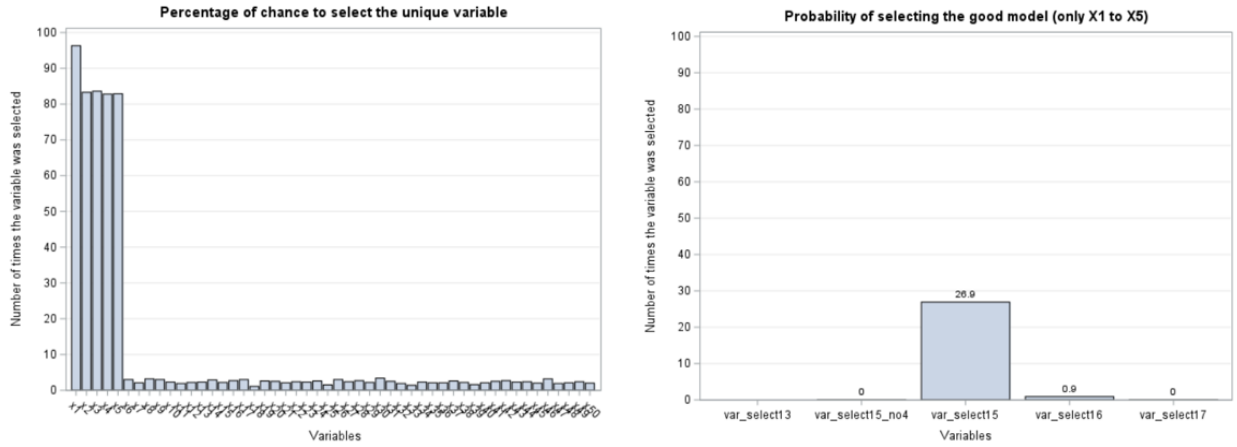


With the DGP1 and DGP4 datasets, statistical learning approaches produce nearly identical perfect selection results. As a result, the high values appear to have little effect on the *Statistical Learning* approaches. The *Forward* method with the SBC selection criterion is the most efficient selection procedure method in statistical learning, with a probability of selection of the combined variables of interest of 17.9%. *Backward* is not regarded efficient since it tends to overfit whatever selection criterion is used, resulting in unreliable results.

4.4.2 Machine Learning

- AICc

LASSO



To conclude, the *Machine Learning* methods show lower perfect selection results with DGP4 database compared with DGP1 database. The presence of extreme values in the databases has an impact on the *Machine Learning* techniques, unlike the *Statistical Learning* methods. The performances of *LARS* and *LASSO* are nearly identical. In this data generator framework (DGP4), we will use the *LASSO* technique with the AICc selection criterion as the most efficient selection procedure in *Machine Learning*. The latter suggests a 26.9% joint chance of selecting variables of interest. *LARS* could be a

good selection tool also because its selection probabilities are quite similar to *LASSO*. Although *ElasticNet* has higher selection probabilities than *Statistical Learning* procedures, it is not the best selection tool in this situation.

When we compare the best *Statistical Learning* (*Forward* SBC criteria) and the best *Machine Learning* (*LASSO* AICc criteria) methods, we can see that the *LASSO* procedure with the AICc selection criterion (26.9% success rate) is clearly superior to the *Forward* procedure with the SBC selection criterion (17.9 percent success rate). Despite the existence of extreme values and the decline in performance of *Machine Learning* approaches, *LASSO* with the AICc selection criterion remains the best method for retrieving variables of interest in a DGP4 type database.

After examining all of the selection methods in various conditions, it can be concluded that in DGP1, DGP2, and DGP4 type situations, *Machine Learning* techniques are more successful than *Statistical Learning* methods. *Statistical Learning* techniques are more successful than *Machine Learning* methods only in the scenario of DGP3. If we compare all of the outcomes, we can conclude that *LASSO* and *ElasticNet* are the best approaches.

We can see from these results that the *LASSO* and *ElasticNet* approaches are limited in some instances. First and foremost, we know that due to the nature of the coefficient restriction, *LASSO* will prefer to choose a subset of variables, but it will also tend to under-fit. While *LASSO* is more likely to avoid presenting variables that are of no interest, it is also more likely to overlook some factors that are of interest. Furthermore, if two variables are highly associated and critical for prediction, *LASSO* will favor one over the other, as in the case of DGP2. When the variables of interest are associated with other variables, correlations might become an issue. The consistency of the *LASSO* selection is no longer guaranteed in this scenario.

ElasticNet uses a weighted combination of L1 and L2 regularizations. As you can probably see, the same function is used for both *LASSO* and *Ridge* regression, only the $L1_{wt}$ argument changes. This argument determines how much weight goes to the L1 norm of the partial slopes. If the regularization is pure L2 (*Ridge*) and if $L1_{wt} = 1.0$ the regularization is pure L1 (*LASSO*). *ElasticNet* allows to add to *LASSO* a penalty *Ridge*, in order to reduce the selection bias of *LASSO*, *ElasticNet* thus finds better results compared to *LASSO* in DGP2 and DGP3 type situations.

5 Real-life application: diabetes database

Following a theoretical comparison of statistical learning VS machine learning approaches, it would be interesting to test our best method on real data. We chose a diabetic database. The latter is composed of 442 observations, one explained variable Y and 10 explanatory variables including *age*, *sex*, *body mass index* (BMI), *average blood pressure* (BP), and *six blood serum measurements* ($S1$ *to* $S6$).

Firstly, we analysed our *diabete* database to see if there were any outliers. The Skewness coefficient tells us whether or not the distribution is symmetrical (Skewness=0). Using the output below ($figure$ 4), we can observe that when the mean is greater than the median we get a positive Skewness coefficient. A positive Skewness means that the distribution is skewed to the right and that there are outliers. A positive Skewness is observed for the values $S3$ and $S4$ with 0.79 and 0.73 respectively. Conversely, a negative Skewness means that the distribution is asymmetric towards the left and that there are also outliers. As a result, there are some extreme values in our database. However, we have chosen not to remove them from our *diabete* table since the presence of extreme values is not always negative. Indeed, after having identified the outliers it was useful according to us to keep them since it is important for the algorithm to learn the notion of anomalous behaviour. Moreover, the outliers present in our database are quite low. Even if we should not forget that in another situation keeping important outliers can bias a model.

Variable	N	Mean	Median	Std Dev	Minimum	Maximum	Skewness	Kurtosis
AGE	442	48.5180995	50.0000000	13.1090278	19.0000000	79.0000000	-0.2313815	-0.6712237
SEX	442	1.4683258	1.0000000	0.4995612	1.0000000	2.0000000	0.1273845	-1.9928110
BMI	442	26.3757919	25.7000000	4.4181216	18.0000000	42.2000000	0.5981485	0.0950945
BP	442	94.6470136	93.0000000	13.8312834	62.0000000	133.0000000	0.2906584	-0.5327973
S1	442	189.1402715	186.0000000	34.6080517	97.0000000	301.0000000	0.3781082	0.2329479
S2	442	115.4391403	113.0000000	30.4130810	41.6000000	242.4000000	0.4365918	0.6013812
S3	442	49.7884615	48.0000000	12.9342022	22.0000000	99.0000000	0.7992551	0.9815075
S4	442	4.0702489	4.0000000	1.2904499	2.0000000	9.0900000	0.7353736	0.4444017
S5	442	4.6414109	4.6200500	0.5223906	3.2581000	6.1070000	0.2917537	-0.1343668
S6	442	91.2601810	91.0000000	11.4963347	58.0000000	124.0000000	0.2079166	0.2369167
Y	442	152.1334842	140.5000000	77.0930045	25.0000000	346.0000000	0.4405629	-0.8830573

Figure 4 : Data analysis for extreme values

Secondly, the Kurtosis coefficient ($figure$ 4, *above*), also known as the flattening coefficient, provides information on the distribution's tails. Indeed, the higher this coefficient is, the further the values are from the mean. For some variables, such as $S2$ or $S3$, the kurtosis is positive, indicating that the tails contain more observations than in a Gaussian distribution. On the other hand, *sex* and *age* for example have a negative kurtosis, which indicate that the tails have less observations.

Thirdly, we performed a count to ensure that no data was missing, as well as an analysis of the correlation between the variables. The correlation matrix between the variables in the model is depicted in $figure$ 5. Some variables, such as $S1$ and $S2$, show a high positive correlation (0.89). The variables $S3$ and $S4$, on the other hand, are negatively correlated. As a result, we have a situation that is a combination of DGP3 and DGP4.

Pearson Correlation Coefficients, N = 442 Prob > |r| under H0: Rho=0											
	AGE	SEX	BMI	BP	S1	S2	S3	S4	S5	S6	Y
AGE	1.00000	0.17374 0.0002	0.18508 <.0001	0.33543 <.0001	0.26006 <.0001	0.21924 <.0001	-0.07518 0.1145	0.20384 <.0001	0.27077 <.0001	0.30173 <.0001	0.18789 <.0001
SEX	0.17374 0.0002	1.00000	0.08816 0.0640	0.24101 <.0001	0.03528 0.4594	0.14264 0.0026	-0.37909 <.0001	0.33212 <.0001	0.14992 0.0016	0.20813 <.0001	0.04306 0.3664
BMI	0.18508 <.0001	0.08816 0.0640	1.00000	0.39541 <.0001	0.24978 <.0001	0.26117 <.0001	-0.36681 <.0001	0.41381 <.0001	0.44616 <.0001	0.38868 <.0001	0.58645 <.0001
BP	0.33543 <.0001	0.24101 <.0001	0.39541 <.0001	1.00000	0.24246 <.0001	0.18555 <.0001	-0.17876 0.0002	0.25765 <.0001	0.39348 <.0001	0.39043 <.0001	0.44148 <.0001
S1	0.26006 <.0001	0.03528 0.4594	0.24978 <.0001	0.24246 <.0001	1.00000	0.89666 <.0001	0.05152 0.2798	0.54221 <.0001	0.51550 <.0001	0.32572 <.0001	0.21202 <.0001
S2	0.21924 <.0001	0.14264 0.0026	0.26117 <.0001	0.18555 <.0001	0.89666 <.0001	1.00000	-0.19646 <.0001	0.65982 <.0001	0.31836 <.0001	0.29060 <.0001	0.17405 0.0002
S3	-0.07518 0.1145	-0.37909 <.0001	-0.36681 <.0001	-0.17876 0.0002	0.05152 0.2798	-0.19646 <.0001	1.00000	-0.73849 <.0001	-0.39858 <.0001	-0.27370 <.0001	-0.39479 <.0001
S4	0.20384 <.0001	0.33212 <.0001	0.41381 <.0001	0.25765 <.0001	0.54221 <.0001	0.65982 <.0001	-0.73849 <.0001	1.00000	0.61786 <.0001	0.41721 <.0001	0.43045 <.0001
S5	0.27077 <.0001	0.14992 0.0016	0.44616 <.0001	0.39348 <.0001	0.51550 <.0001	0.31836 <.0001	-0.39858 <.0001	0.61786 <.0001	1.00000	0.46467 <.0001	0.56588 <.0001
S6	0.30173 <.0001	0.20813 <.0001	0.38868 <.0001	0.39043 <.0001	0.32572 <.0001	0.29060 <.0001	-0.27370 <.0001	0.41721 <.0001	0.46467 <.0001	1.00000	0.38248 <.0001
Y	0.18789 <.0001	0.04306 0.3664	0.58645 <.0001	0.44148 <.0001	0.21202 <.0001	0.17405 0.0002	-0.39479 <.0001	0.43045 <.0001	0.56588 <.0001	0.38248 <.0001	1.00000

Figure 5 : Correlation analysis

Fourthly, we used $PROCREG$ to evaluate the significance of the model's explanatory variables. With a 1% significance level, the variables that better explain Y are SEX, BMI, BP, and $S5$ (*lamotrigine measure*).

Finally, after analyzing our data as well as the theoretical work done in parts 2 and 3, we found it relevant in our case to use the machine learning method *Elasticnet* to select the appropriate variables. The algorithm selects the following variables using the *Elasticnet* method: *intercept*, SEX, BMI, BP, $S3$ ($high - density\ lipoproteins$), $S5$, and $S6$ (*blood sugar level*). The algorithm chose $S3$ and $S6$, which are not statistically significant at the 1% level, yet the $R2$ of this model is 50.05%. Knowing that our previous regression had an $R2$ of 51.77%, the explanatory variables describe the model pretty well. Thus, despite the presence of correlation and minor outliers, the *Elasticnet* algorithm performs well in variable selection.

6 Discussion

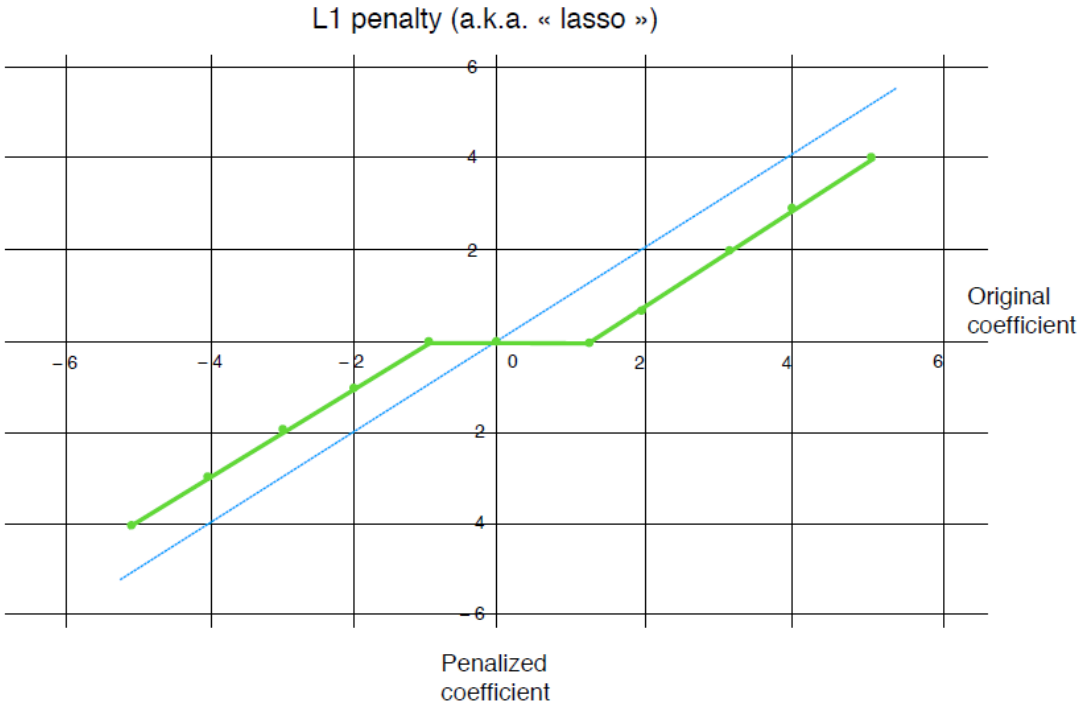
After explaining the differences that govern *Statistical Learning* and *Machine Learning* categories we acquired the necessary results to conclude our study. Nevertheless, our paper has several limitations. For example, other criteria may have been used to establish which model is the best. As for instance, the Mallows CP compares the accuracy and bias of full model against those models including only a subset of the predictors, ensuring that the amount of predictors in the model is balanced. The lower the Mallows CP the more accurate the model. If the Mallows CP is close to the number of variables, the model is unbiased in estimating real regression coefficients and forecasting future responses. But the Mallows CP is not displayed in the findings if a predictor is substantially connected with another predictor. As a result, we may conclude that it would have been useless in the initial data set simulation.

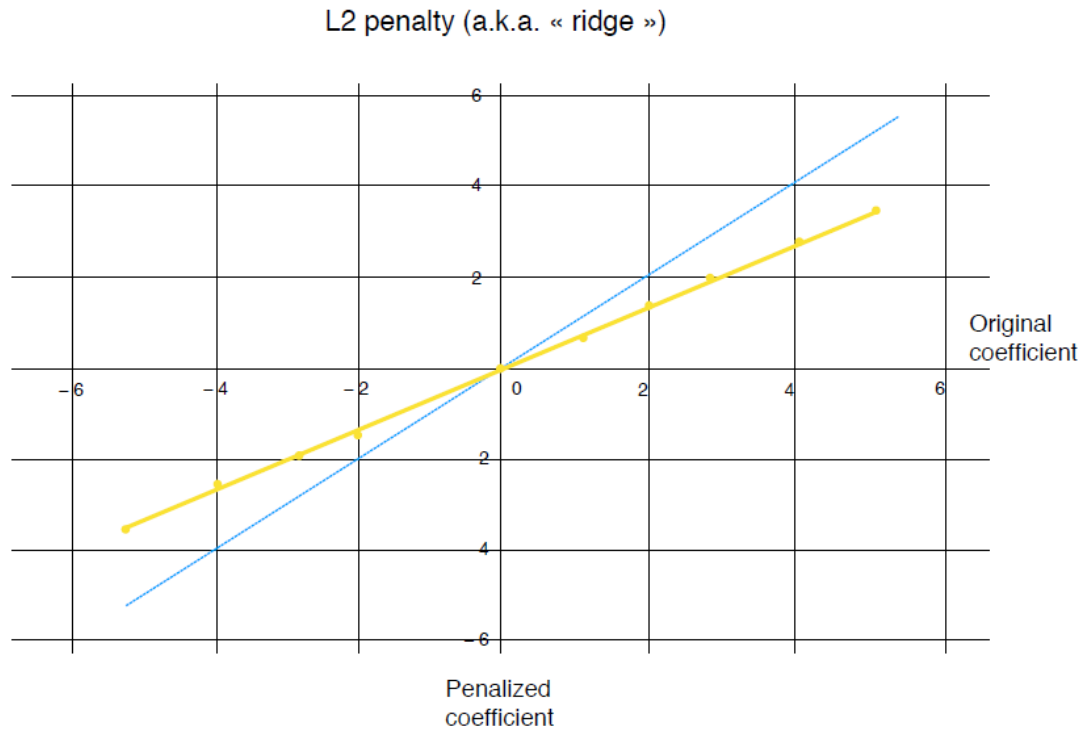
Other limitations throw into question the dissertation's central premise, which is to compare different selection methods. Indeed, comparing different selection methods does not necessarily lead to seeking an improvement of these methods. For a firm, conducting this type of dissertation toward an improvement goal rather than a comparison goal might be more interesting. However, we could suggest a way to improve *Machine Learning* methods only by setting some selection, choice or stopping criteria. For example, we may use *ElasticNet* to suggest a choice criterion C(p), which gives a selection probability of more than 50% for variables of interest in DGP1 and DGP2.

Finally, we could ask ourselves whether the DGPs developed for this theory are coherent with reality. It would be more interesting to design a DGP6 that takes into account both the existence of correlation between the variables and the presence of extreme values in order to improve our paper. As a result, we will be more likely to encounter databases of the DGP6 type in the actual world than databases of the DGP3 or DGP4 types, because it is rare to find databases with merely correlations and no extreme values, and vice versa.

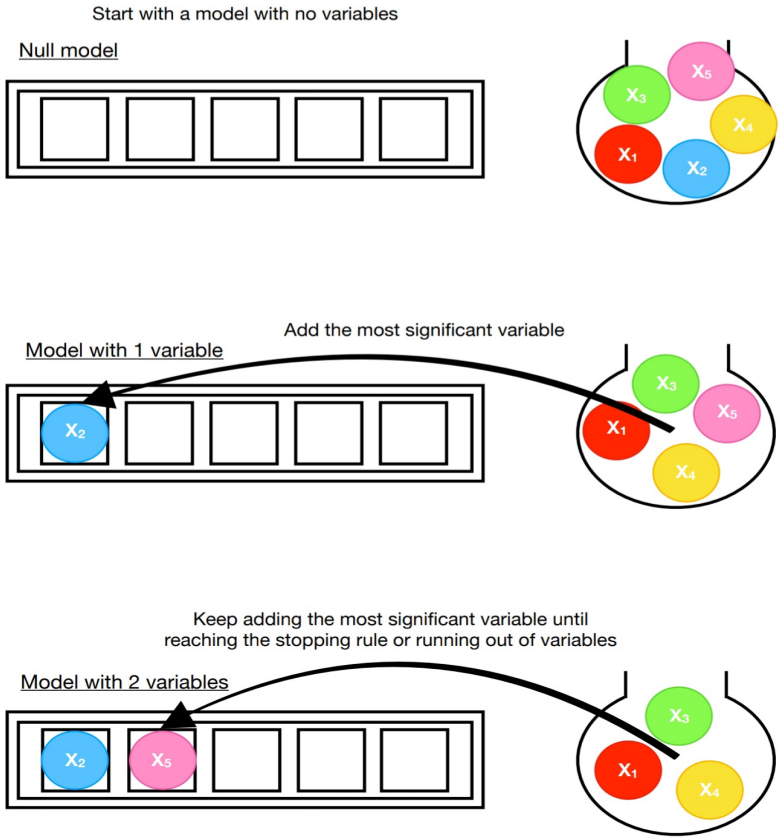
7 Appendix

Penalty norms [17]

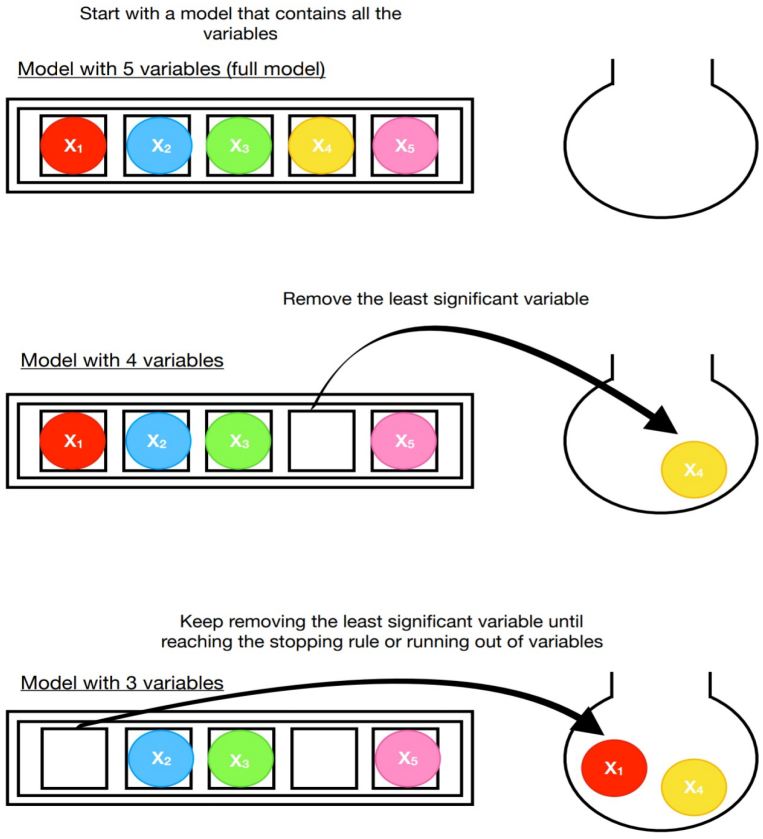




Forward stepwise selection[7]



Backward stepwise selection[7]



8 Bibliography

[1] Olaf Gefeller and Rainer Muche. Variable selection techniques implemented procedures of the sas software. *Proceedings of SEUGI*.

[2] Omni Sci. Statistical modeling. URL https://www.omnisci.com/technical-glossary/statistical-modeling.

[3] Jeffrey A. Walker. *Applied Statistics for Experimental Biology*. URL https://www.middleprofessor.com/files/applied-biostatistics_bookdown/_book/an-introductionto-statistical-modeling.html.

[4] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 1999.

[5] Ulrike Von Luxburg and Bernhard Schölkopf. Statistical learning theory : Models, concepts, and results. Elsevier, 2011.

[6] David Dalpiaz. Applied statistics with r. 2019.

[7] G Choueiry. Understand forward and backward stepwise regression. *Quantifying Health*, 2020.

[8] NCSS Statistical Software. Stepwise regression. URL https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Stepwise_Regression.pdf.

[9] Tom Michael Mitchell. *The discipline of machine learning*. Carnegie Mellon University, School of Computer Science, Machine Learning, 2006.

[10] Ryan J Tibshirani. A general framework for fast stagewise algorithms. *J. Mach. Learn. Res.*, 2015.

[11] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *The Annals of Statistics*, 2004. URL http://www.jstor.org/stable/3448465.

[12] Acervo Lima. Régression du moindre angle (lars). URL https://fr.acervolima.com/regression-du-moindre-angle-lars/.

[13] Arthur E Hoerl and Robert W Kennard. Ridge regression: applications to nonorthogonal problems. *Technometrics*, 1970.

[14] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 1996.

[15] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 2005.

[16] SAS. The glmselect procedure. URL https://support.sas.com/documentation/onlinedoc/stat/131/glmselect.pdf.

[17] Patrick Crutchley. Penalized regression, 2016. URL https://www.youtube.com/watch?v=nQ4G45AbHyU.