UNIVERSIDAD POLITÉCNICA DE MADRID

Escuela Técnica Superior de
Ingenieros Informáticos

Máster Universitario en Ciencia de Datos

Practical Work 1 – Machine Learning

ANALYSIS OF NON-PROBABILISTIC SUPERVISED CLASSIFICATION METHODS

Author: Paul Eyzaguirre Barreda

Professors: Pedro Larrañaga and Concha Bielza

Madrid, October 2023

COMPARASION OF SUPERVISED CLASSIFICATION METHODS TO DETERMINE THE RISK OF BANKROUPTCY

1. Introduction

Supervised classification methods aim to transform a data set of labeled instances into a computational model able to make classification prediction for new observations taking into account their feature values. The learning process can be done applying different supervised classification algorithm for predicting a class variable.

In this work, the algorithms covered are five. Namely, k-nearest neighbors, rule induction (RIPPER), Decision Trees (DT), Support Vector Machine (SVM) and Artificial Neural Networks (MLP). All of them will be applied to a real case scenario to determine whether a company is prone to go bankruptcy or not. Firstly, and very importantly, we account for three features subset selection methods, which are methods for detecting and removing variables which are irrelevant or redundant. They are filter univariate, filter multivariate and wrapper. Further works can include embedded methods as well (Larrañaga, 2021). Advantages of filter techniques are that they easily scale to scale to very high-dimensional data sets, they are computationally simple and fast, they avoid overfitting problems, and are independent of the classification algorithm. As a result, filter feature selection needs to be performed only once.

Secondly, this work aims to show and adequately determine a performance measure and performance estimation method. Performance measures are used as figures of merit of supervised classifiers. The aim is to find the supervised classification model with the optimum value of a given performance measure. The algorithm that induces the model from the data set of cases is responsible for searching for this optimum value. To show this, it has been used the confusion matrix to denote the number of true positive (TP) classes and true negative cases. (FN). For our data set in this work, we have only two categorical values for the class C. Respectively the matrix would be as:

$$\frac{\Phi(x)}{\begin{pmatrix} TP & FN \\ FP & TN \end{pmatrix} C}$$

The main performance measures used from the confusion matrix are the accuracy, positive predictive values and negative predictive values. Namely:

$$Acc(\Phi) = \frac{TP + TN}{TP + FN + FP + TN}$$

$$PPV(\Phi) = \frac{TP}{TP + FP}$$

$$NPV(\Phi) = \frac{TN}{TN + FN}$$

Thirdly, is needed to make an honest estimation over the unseen data, thus, several methods can be used to split the data and evaluate the performance. However, to estimate the performance measure based on instances that have not been seen in the learning phase, an honest estimation method called K-fold cross validation has been chosen. In essence this method randomly partitions the original labeled data set into k-subsets of roughly equal sizes, called folds. Of the K-folds a single fold is retained as the test data set for testing model, and the remaining k -1 folds are used as training data. The cross validation the is repeated k times and the k results of the performance will be averaged to produce an estimation of the performance of the model induced from the whole original set. An accuracy measure is averaged as follows.

$$ACC\ cross\ validation\ (\Phi) = \frac{Acc_1 + Acc_2 + .. + Acc_n}{n}$$

At the end of this work is presented a comparison table of the supervised classification methods, each with the accuracy metric depending of the features subsets selected, and showing independently the performance measure in order to know which ones were good enough.

2. Problem description:

In the finance industry there are a lot of data, specifically balance sheets of commercial companies, specially of those that are in the stock market. The problem of this works states that based on a given data set with information of the performance of the company, is possible to classify wheatear or not a company is in bankruptcy.

The data were collected from the Taiwan Economic official Journal for the years 1999 to 2009. (Taiwanese Bankruptcy Prediction UCI Machine Learning Repository, 2020) Company bankruptcy was defined based on the business regulations of the Taiwan Stock Exchange. Having around 95 features and more than 3819 instances. In this scenario we want to predict through classification if a company is in bankruptcy taking as input the balance sheet statement of a given company. In figure 1, it is showed with value 1 all the companies in bankruptcy and with value 0 the companies not in bankruptcy.

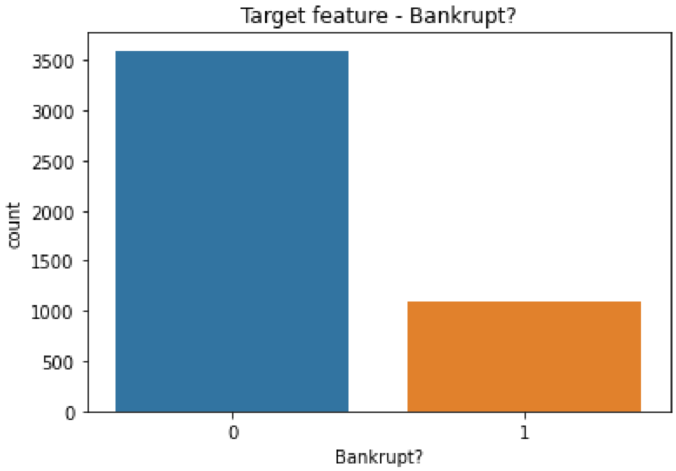
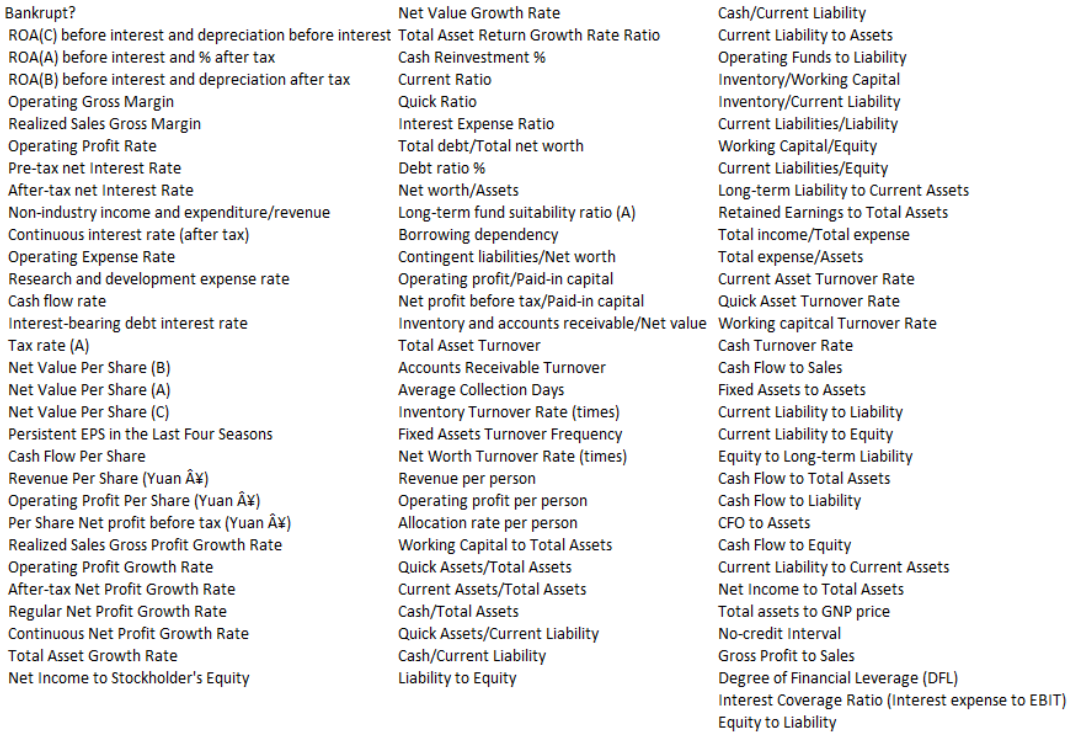


Figure 1. Distribution of the classification variable.

The variables used are as in figure 2.



Bankrupt?
ROA(C) before interest and depreciation before interest
ROA(A) before interest and % after tax
ROA(B) before interest and depreciation after tax
Operating Gross Margin
Realized Sales Gross Margin
Operating Profit Rate
Pre-tax net Interest Rate
After-tax net Interest Rate
Non-industry income and expenditure/revenue
Continuous interest rate (after tax)
Operating Expense Rate
Research and development expense rate
Cash flow rate
Interest-bearing debt interest rate
Tax rate (A)
Net Value Per Share (B)
Net Value Per Share (A)
Net Value Per Share (C)
Persistent EPS in the Last Four Seasons
Cash Flow Per Share
Revenue Per Share (Yuan Â¥)
Operating Profit Per Share (Yuan Â¥)
Per Share Net profit before tax (Yuan Â¥)
Realized Sales Gross Profit Growth Rate
Operating Profit Growth Rate
After-tax Net Profit Growth Rate
Regular Net Profit Growth Rate
Continuous Net Profit Growth Rate
Total Asset Growth Rate
Net Income to Stockholder's Equity

Net Value Growth Rate
Total Asset Return Growth Rate Ratio
Cash Reinvestment %
Current Ratio
Quick Ratio
Interest Expense Ratio
Total debt/Total net worth
Debt ratio %
Net worth/Assets
Long-term fund suitability ratio (A)
Borrowing dependency
Contingent liabilities/Net worth
Operating profit/Paid-in capital
Net profit before tax/Paid-in capital
Inventory and accounts receivable/Net value
Total Asset Turnover
Accounts Receivable Turnover
Average Collection Days
Inventory Turnover Rate (times)
Fixed Assets Turnover Frequency
Net Worth Turnover Rate (times)
Revenue per person
Operating profit per person
Allocation rate per person
Working Capital to Total Assets
Quick Assets/Total Assets
Current Assets/Total Assets
Cash/Total Assets
Quick Assets/Current Liability
Cash/Current Liability
Liability to Equity

Cash/Current Liability
Current Liability to Assets
Operating Funds to Liability
Inventory/Working Capital
Inventory/Current Liability
Current Liabilities/Liability
Working Capital/Equity
Current Liabilities/Equity
Long-term Liability to Current Assets
Retained Earnings to Total Assets
Total income/Total expense
Total expense/Assets
Current Asset Turnover Rate
Quick Asset Turnover Rate
Working capitcal Turnover Rate
Cash Turnover Rate
Cash Flow to Sales
Fixed Assets to Assets
Current Liability to Liability
Current Liability to Equity
Equity to Long-term Liability
Cash Flow to Total Assets
Cash Flow to Liability
CFO to Assets
Cash Flow to Equity
Current Liability to Current Assets
Net Income to Total Assets
Total assets to GNP price
No-credit Interval
Gross Profit to Sales
Degree of Financial Leverage (DFL)
Interest Coverage Ratio (Interest expense to EBIT)
Equity to Liability

Figure 2. Data-set features.

3. Methodology

In this work we used the software Python and the following libraires:

Skit-learn
Pandas
Numpy
Matplotlib
Wittgenstein
Mlxtend

As first step, the data has been extracted with free license from the official web page from the Taiwan Economic official Journal. The data was transformed to a data frame in Pandas and cleaned of null values and empty rows-columns. The dimension of the data frame ended up with 93 columns (features) and 4659 rows. The target value is a qualitative value transformed to a quantitative value, being 1 as bankruptcy and 0 as no bankruptcy. It has considered convenient not to scale the data because we are going to use several methods and some of them are sensitive to scaled data whereas others not, which will make an unfair comparison at the end.

For each algorithm we have performed 4 important analysis which are described as following:

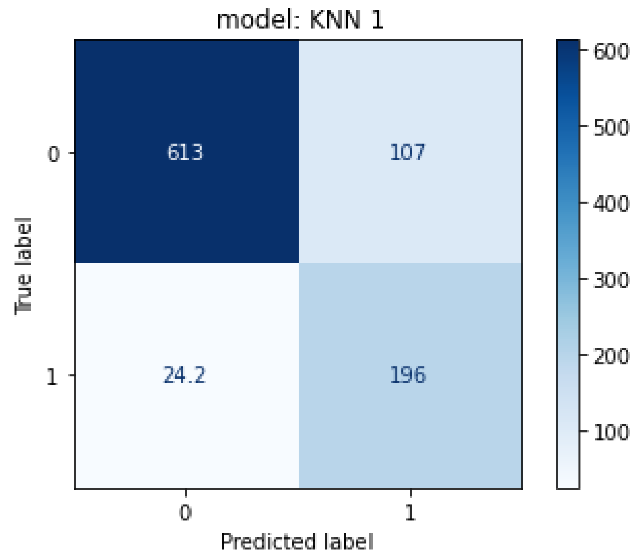
1. First analysis with all the original features.
2. Second analysis using a univariate feature subset selection.
3. Third analysis using a multivariate feature subset selection.
4. And a fourth analysis using a wrapper method.

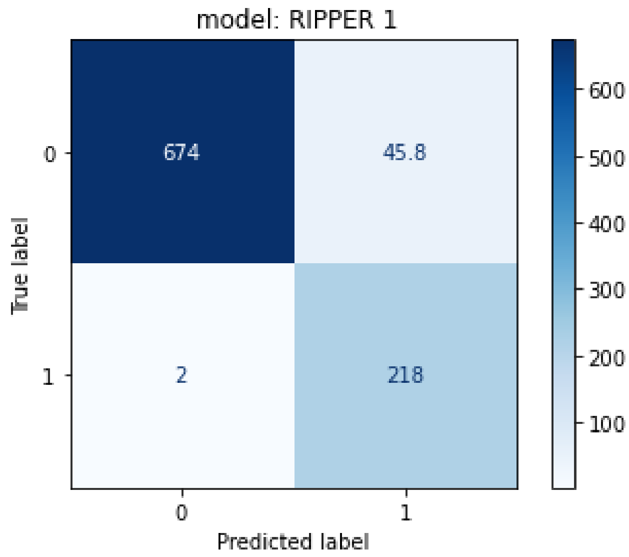
Each analysis was performed using a performance measure and performance estimation method. Namely a confusion matrix with the average accuracy and the k-fold cross validation method.

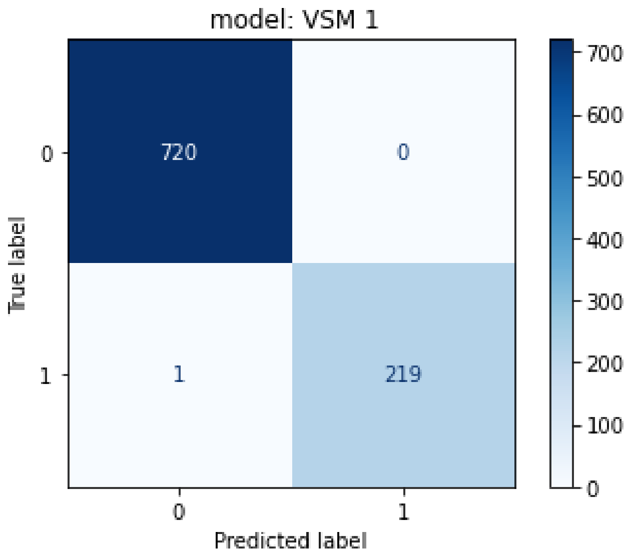
4. Results and discussion

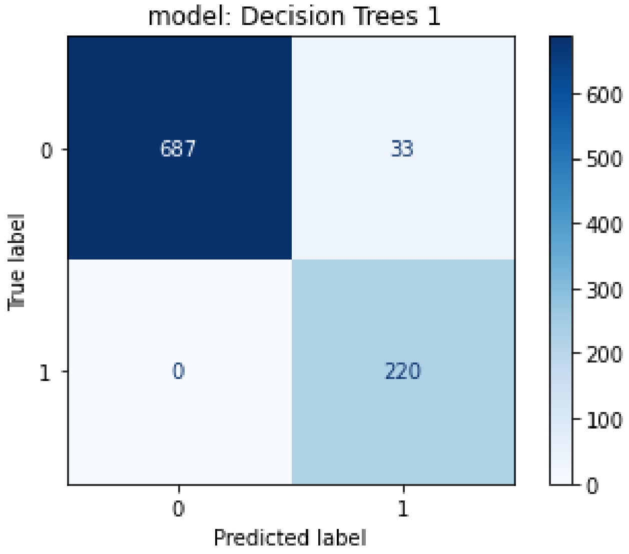
This section will show the analysis performed for the feature subset selection and at the end it will be displayed a comparison table with the accuracy of all the models. Each confusion matrix represents the average of the k-fold times that the algorithm has been executed. The 1 predicted value means that a company is in bankruptcy and the 0 predicted value refers the opposite.

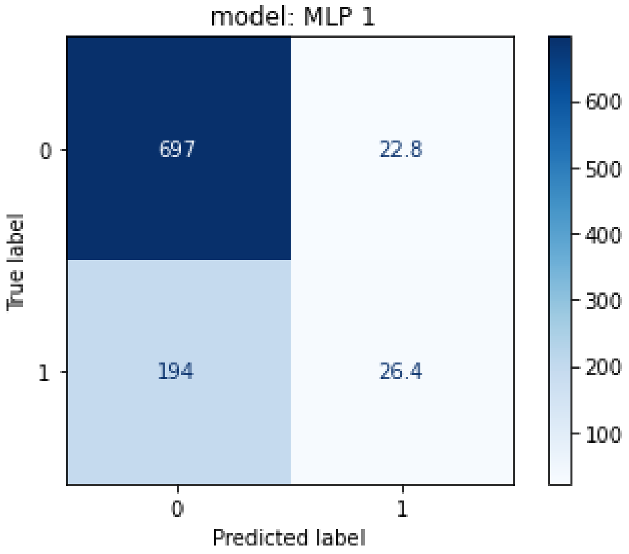
4.1 First analysis with all the original features.


model: KNN 1
True label
Predicted label
0
1
0
1
613
107
24.2
196
600
500
400
300
200
100


model: RIPPER 1
True label
Predicted label
0
1
0
1
674
45.8
2
218
600
500
400
300
200
100


model: VSM 1
True label
Predicted label
0
1
0
1
720
0
1
219
700
600
500
400
300
200
100
0


model: Decision Trees 1
True label
Predicted label
0
1
0
1
687
33
0
220
600
500
400
300
200
100
0


model: MLP 1
True label
Predicted label
0
1
0
1
697
22.8
194
26.4
600
500
400
300
200
100

4.2 Second analysis using a univariate feature subset selection.

In univariate filtering, a feature relevance score is calculated, and low-scoring features are removed. Afterwards, the subset of the top “x” best features scores are selected and that subset is used as an input of the algorithm to get any improvement. In this analysis the top “x” best features it was limited by the author to the number of 11 features. This was for computational convenience and ease.

Notice that for the data set we have the need to use a continuous predictor. Thus, it was chosen to Compute the ANOVA F-value (Jafari and Azuaje 2006).

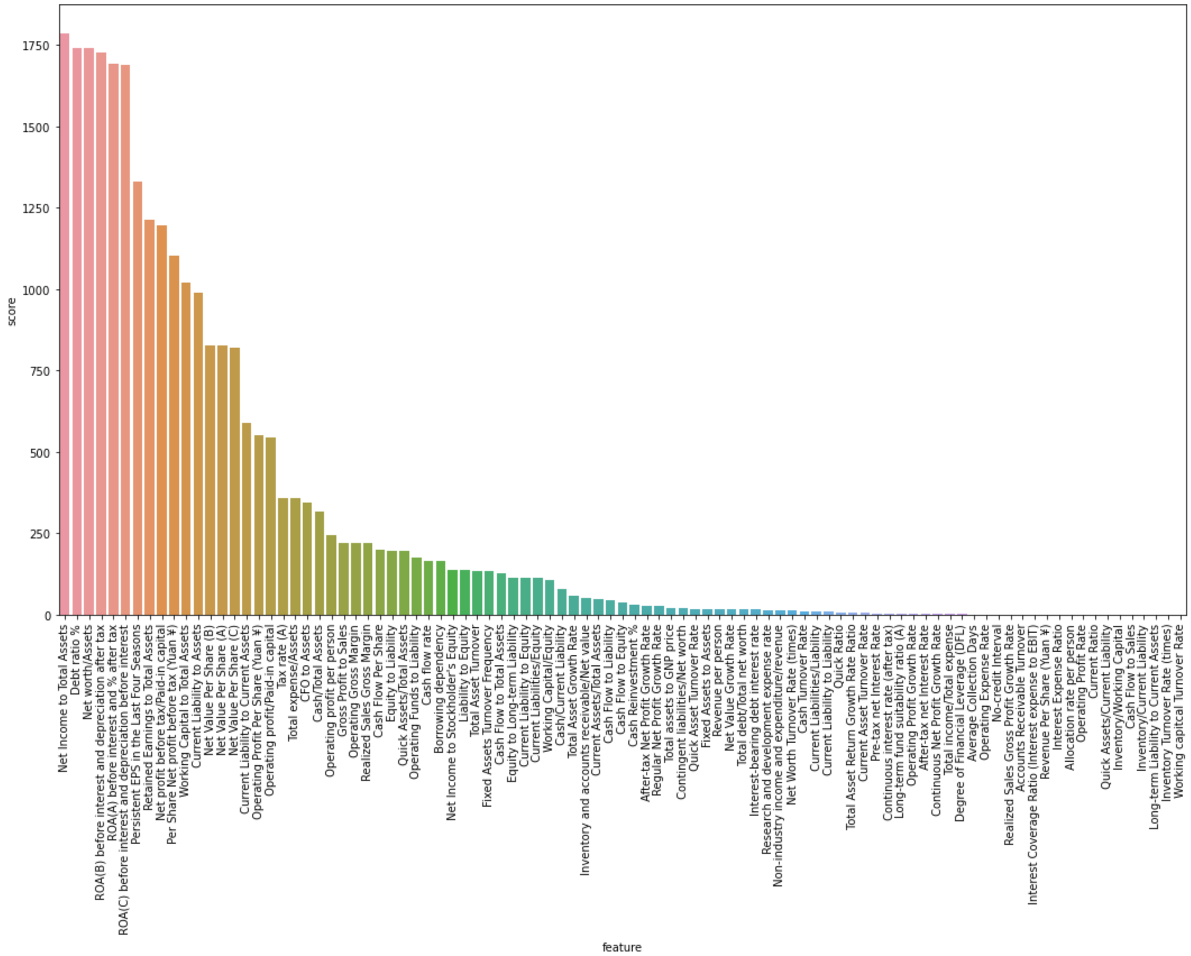
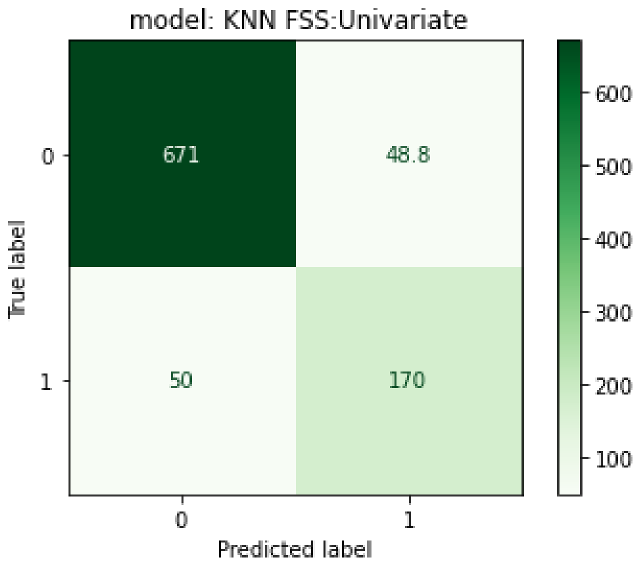
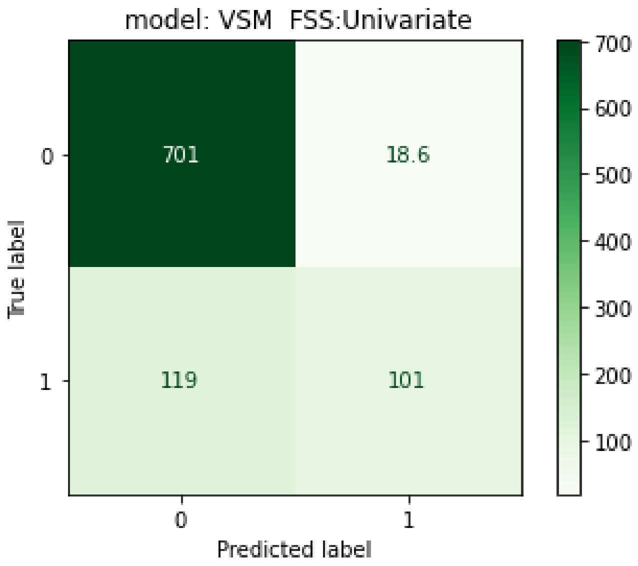
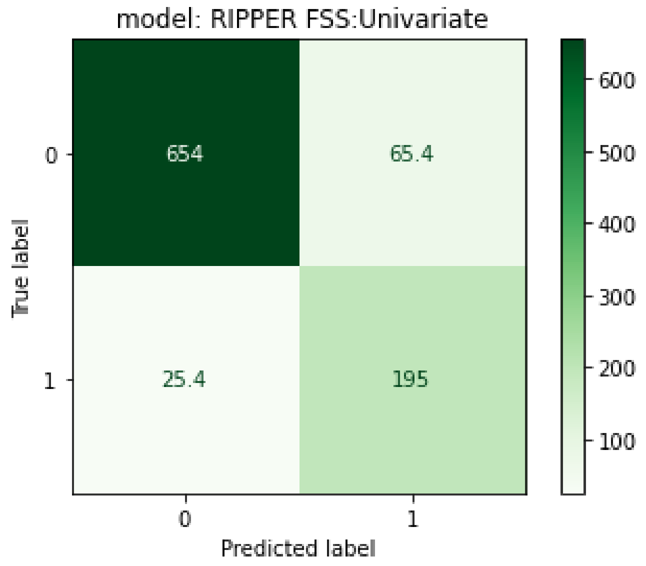
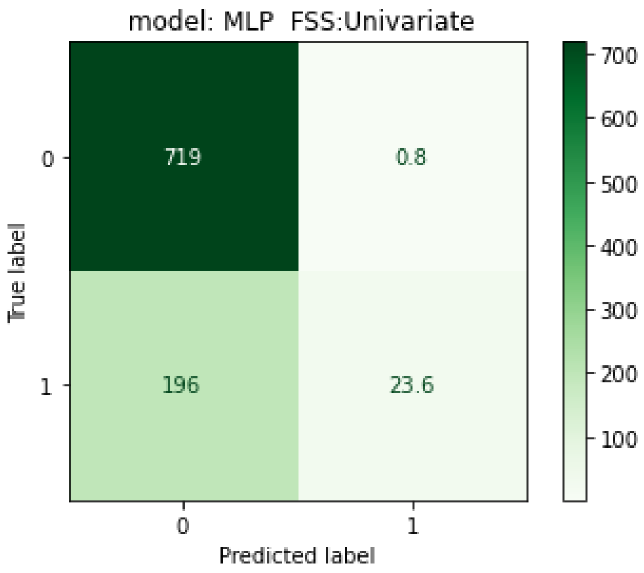


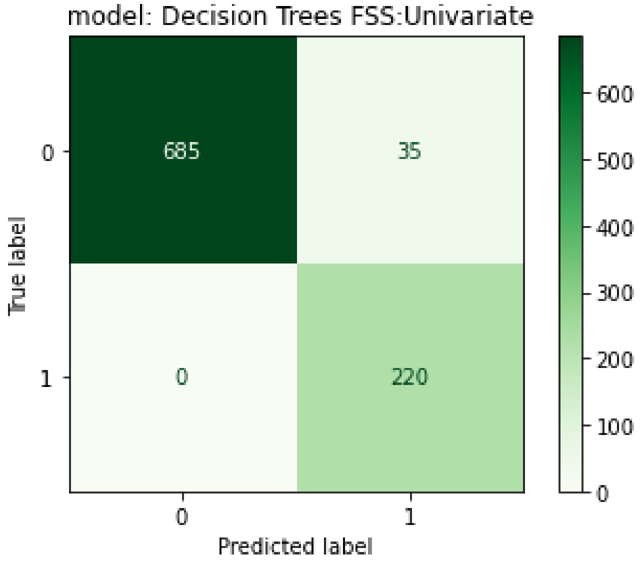
Figure 3. Features ordered by score relevance.


model: KNN FSS:Univariate
True label
Predicted label
0
1
0
1
671
48.8
50
170
600
500
400
300
200
100


model: VSM FSS:Univariate
True label
Predicted label
0
1
0
1
701
18.6
119
101
700
600
500
400
300
200
100


model: RIPPER FSS:Univariate
True label
Predicted label
0
1
0
1
654
65.4
25.4
195
600
500
400
300
200
100

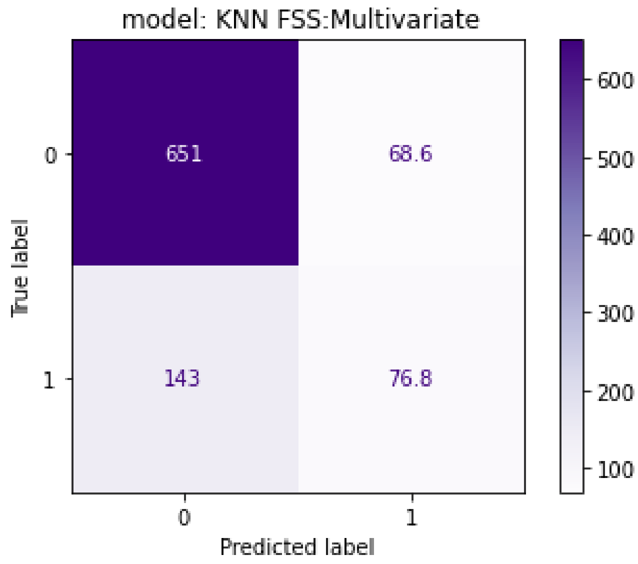

model: MLP FSS:Univariate
True label
Predicted label
0
1
0
1
719
0.8
196
23.6
700
600
500
400
300
200
100

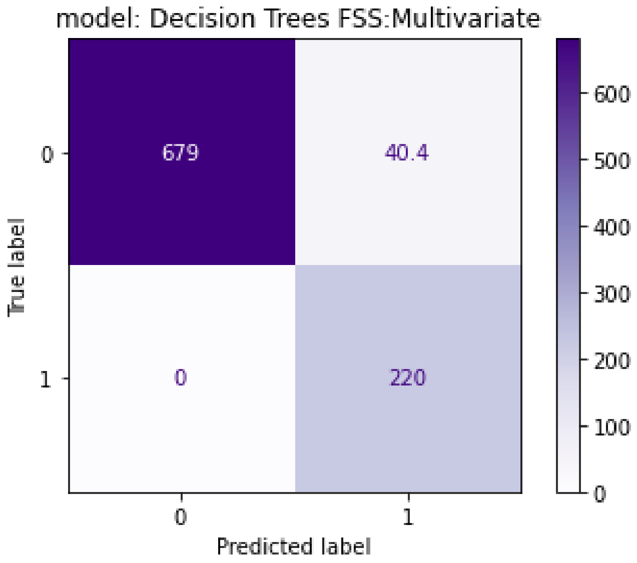

model: Decision Trees FSS:Univariate
True label
Predicted label
0
1
0
1
685
35
0
220
600
500
400
300
200
100
0

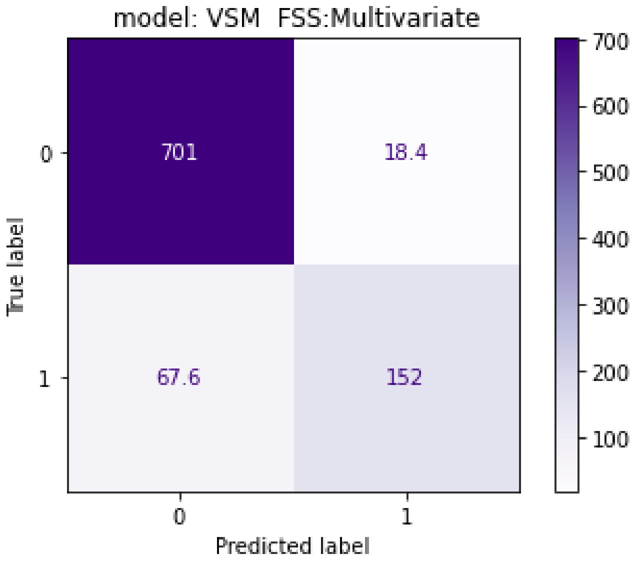
4.3 Third analysis using a multivariate feature subset selection.

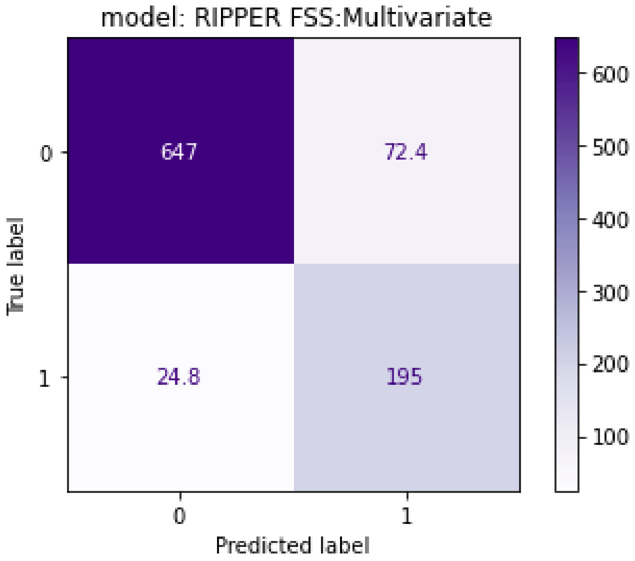
Multivariate filter methods choose the subset of features according to their relevance (with respect to the class) and redundancy. For this experiment, it was chosen the algorithm RELIEF (Kira and Rendell 1992). For example, two nearest neighbors are sought for each instance ***x* ∈ *D***, one is near-hit ***xh* ∈ *D***, and the other is near-miss ***xm* ∈ *D***. The distance should be minimum for ***x*** and ***xh*** and maximum for ***x*** and ***xm***. The instance of each feature for each randomly picked instance is accumulated in a weight vector ***W*** with the same dimension as the number of features. Thus, RELIEF focuses on sampling instances without explicitly searching for feature subsets.

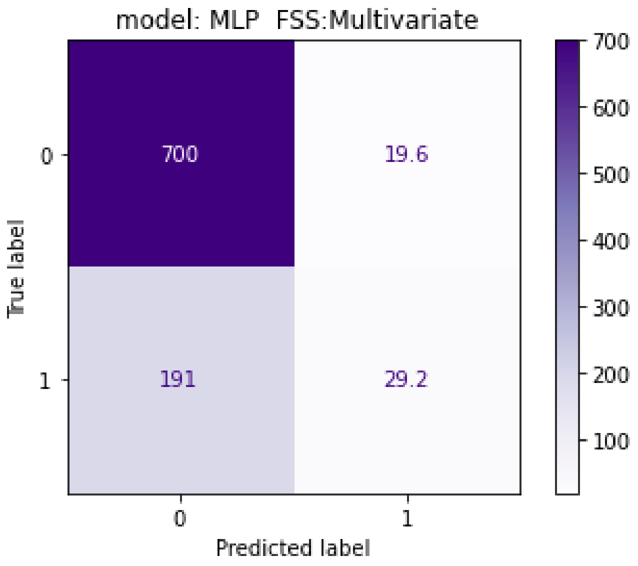
Notice that for the execution, no parameters were added to this model.









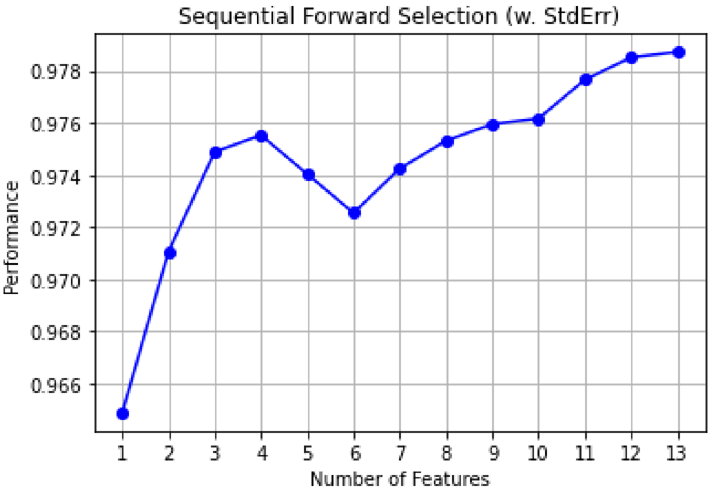


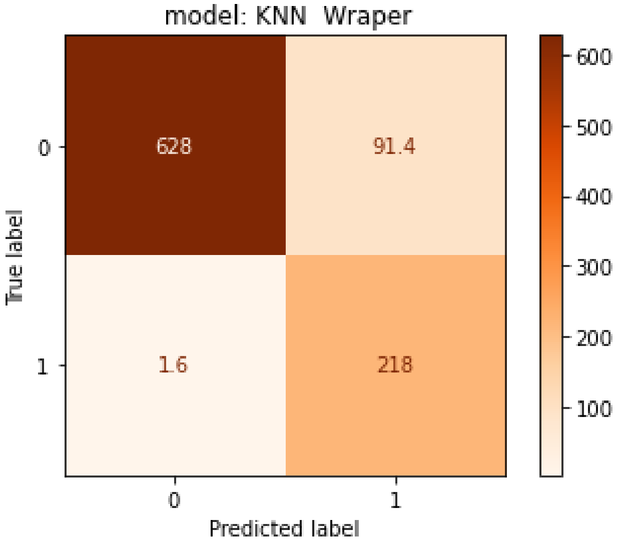
	ALL VARIABLES	FILTER FEATURE SUBSET	
	CONTINUOS	UNIVARIATE	MULTIVARIATE
		KBEST	RELIEFF
KNN	0.8606	0.8949	0.7746
SVM	0.9989	0.8536	0.9085
DT	0.9649	0.9628	0.9570
RIPPER	0.9491	0.9034	0.8966
MLP	0.7697	0.7902	0.7761
Number of Features	94	11	11

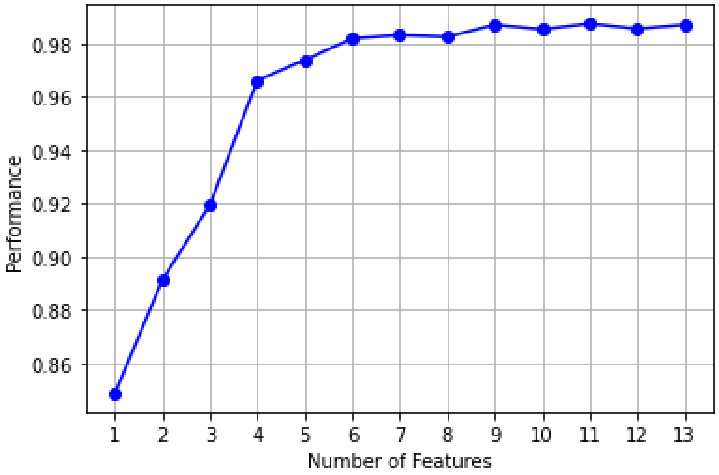
Table 1 Comparison for FSS

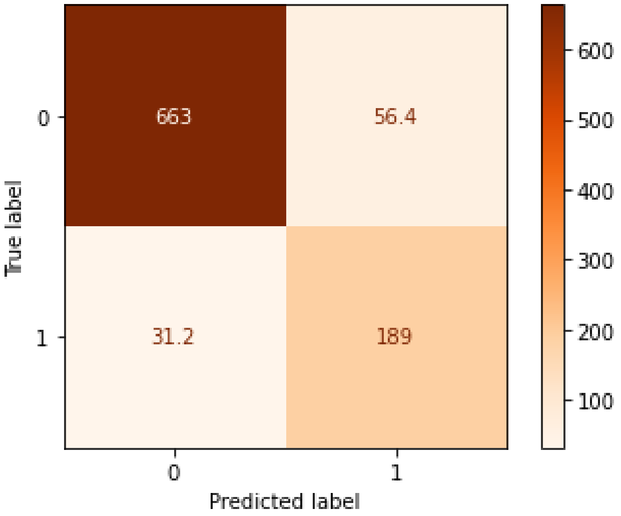
4.4 Fourth analysis using a wrapper method.

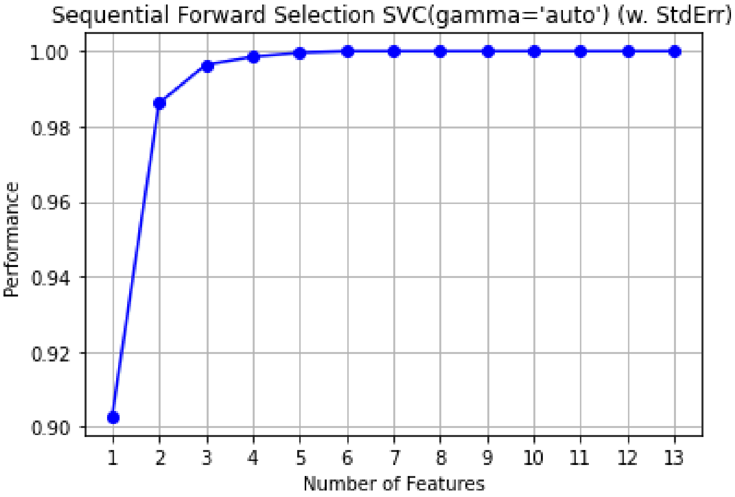
Wrapper methods evaluate each possible subset of features with a criterion consisting of the estimated performance of the classifier built with this subset of features. It is important to mention that wrapper methods are dependent on the classifier as they perform a search over the space of all possible subset features, repeatedly calling the algorithm as a subroutine to evaluate various subsets. It is very impractical for large-scale subsets. In relation to the heuristic, it was used the sequential forward feature selection. It starts with a empty an empty set of features and adds the feature yielding the highest value for the objective function at the first step. From the second steps and onwards, the remaining features are added to the current subset and the new subset is evaluated. Additionally, to the confusion matrix where we see the average accuracy and error, we have added a plot where we checked the number of features with which the model gets a better performance.

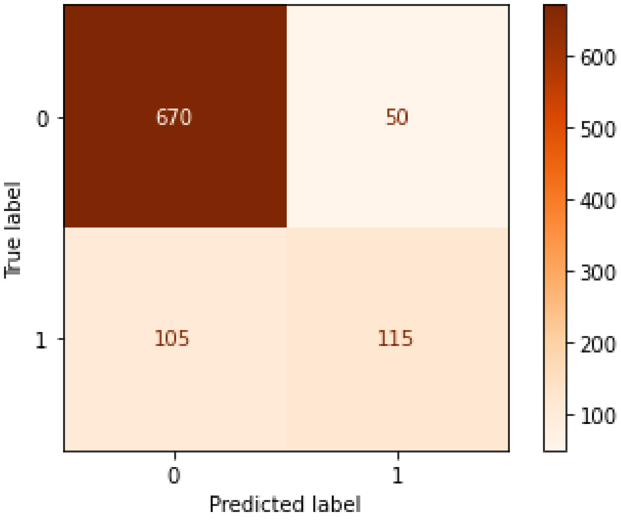


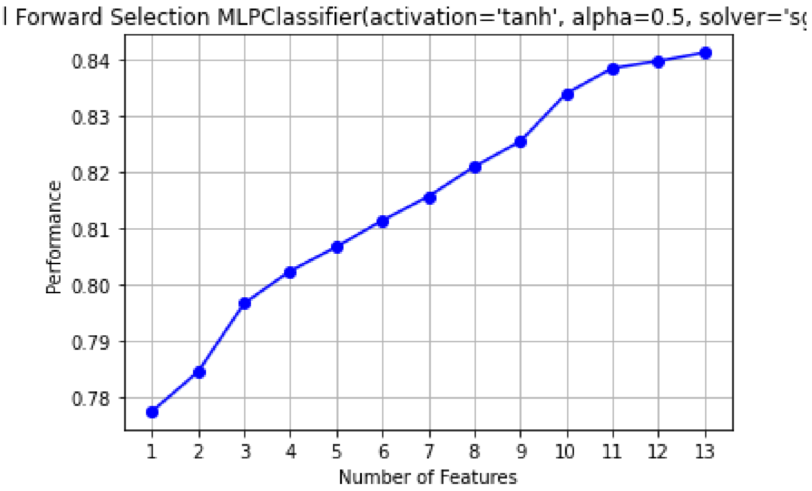


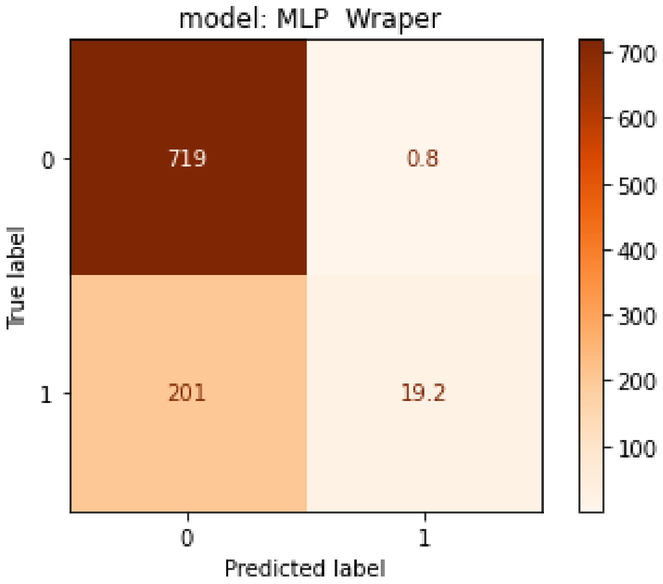












	KNN	SVM	DT	RIPPER	MLP
SEQUENTIAL FORWARD FEATURE SELECTION	0.9010	0.8354	0.9655	0.9068	0.7663
Number of Features	13	5	8	11	13

Table 2 Number of features used for Wrapper

	ALL VARIABLES	FILTER FEATURE SUBSET		WRAPPER
	CONTINUOS	UNIVARIATE	MULTIVARIATE	SEQUENTIAL FORWARD FEATURE SELECTION
		KBEST	RELIEFF	
KNN	0.8606	0.8949	0.7746	0.9010
SVM	0.9989	0.8536	0.9085	0.8354
DT	0.9649	0.9628	0.9570	0.9655
RIPPER	0.9491	0.9034	0.8966	0.9068
MLP	0.7697	0.7902	0.7761	0.7854

Table 3 Accuracy comparison among all the methods

Is important to mention that due to computational capacities and due to the large amount of data, it was not possible to try a larger features range for wrapper model. Each execution costed around 40 minutes of computational time. However, we didn't see an improvement more than for Decision trees algorithm. DT so far has been leading the accuracy and best performance.

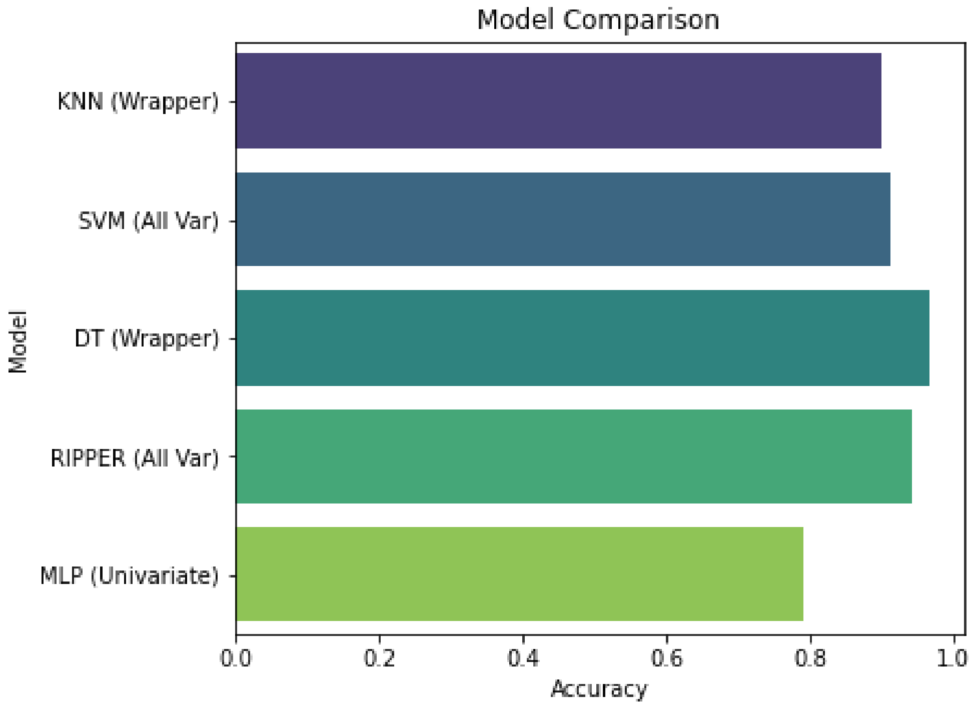


Figure 4 Accuracy performance of all algorithms

5. CONCLUSSION

It has been shown that different classification methods can predict with enough accuracy if a company is in bankruptcy or not. Also, it has been proved that simply methods of performance measure and performance estimation can be effective to determine the effectivity of a model, such as the honest estimation method K-fold cross validation with help of a confusion matrix. Moreover, and most importantly, it has been tested the real data in five different supervised classification algorithms. Among all of them we conclude stating that the best algorithm classifying the bankruptcy was Decision tree algorithm. Its accuracy has been the best among the others, in the last three analysis we performed. On average it has been 96% accurate predicting the class variable.

Lastly, it is worth mentioning that further work can be carried out testing the several different parameters for each classification method. Since that was not the main purpose of this work and involves deeper research, it has been skipped. However, in each software library used for building the algorithms, there are several parameters combinations that can greatly improve the accuracy of the classification problem.

6. REFERENCES

Code can be found at my GitHub repository: https://github.com/PaulAndree/ML-Classification-Methods/

Larrañaga, C. B. (2021). *DATA DRIVEN COMPUTATIONAL NEUROSCIENCE.* Cambridge Press.

Araujo, B. S. (2006). *Aprendizaje Automatico conceptos basicos y avanzados.* Pearson.

Deron Liang, C.-C. L.-F.-A. (2023). Financial ratios and corporate governance indicators in bankruptcy prediction : A comprehensive study. *sciencedirec*. Retrieved from https://www.sciencedirect.com/science/article/pii/S0377221716000412

Taiwanese Bankruptcy Prediction UCI Machine Learning Repository. (2020). Retrieved from https://doi.org/10.24432/C5004D