COLOMBIER Rémi

Professor: Minh Trung Hoai PHAN

Date: 3/31/2022

Individual assignment

Statistical Machine Learning

Table of contents

A) Machine learning predictive algorithms

1. Random Forest

Finally, we will present the Random Forest algorithm which is a supervised machine learning algorithm. This algorithm has many advantages over other machine learning models which we will explain below. This algorithm can be used for classification and regression and performs very well in general in terms of accuracy. As the name suggests, this forest algorithm is based on the decision trees we explained earlier. The Random Forest algorithm combines the results of the many decision trees that make up the random forest to get the most accurate result possible. The aim is to maximize the information gain from each tree. In fact, you must imagine yourself with an algorithm that is composed of hundreds of decision trees. Each tree will vote yes or no for the default according to the predictor and most of the votes will win. The bagging consists of randomly slicing (hence the name "Random Forest") the database into several small databases. Then the idea is to train one model per sample and then combine all the results of the models to get a result. Combining all these results makes the result more reliable and robust to noise. The strong point of random forest is that the default hypers parameter produces excellent results in most cases and these are easy to understand and few. Moreover, the algorithm is versatile because it deals with classification as well as regression in a very qualitative way. The random forest algorithm has much less variance than the single decision tree. The random forest does not require standard scaling of all the variables which is a time saver in the preprocessing steps. But this algorithm is very complex, and the construction of the trees is difficult. Moreover, this algorithm is very expensive in terms of computing power.

2. Logistic regression

The test and well on the training set, then we will have an overfitting model and conversely an underfitting model for much more accurate predictions on the test set than on the trainset. These predictions will be made by optimizing the regression coefficients of the variables we put in our model. The only condition for a logistic regression is that the dependent variable must be dichotomous, i.e., it must be coded in a binary way with only two values 0 and 1." This is the case in our database, since we have a dependent variable which tells us whether the customer has a default 1 or no default 0. Then the binomial logistic regression will calculate a regression coefficient for each continuous or categorical independent variable. This coefficient gives us an indication of the correlation of these independent variables with the customer's default. The sample size must be large enough to conduct a logistic regression, here we have 20,000 observations so it is largely sufficient and significant. It is important that there is no multi-collinearity between the independent variables. The correlations between these variables should not be too high. It is also important to remove the outliers from the variables to make the model as qualitative as possible. Thus, with this model we will predict the probability of the default event occurring or not. Then when the probability of the prediction is greater than 0.5 then the customer will default (1) and when the probability is less than 0.5 then the customer will not default (0). The steps for the logistic regression are as follows:

- Download the data
- Split the data into a training set 80% of the data to train the model and a testing set 20% to test the model.
- Then we use the general linear model (glm) function, remembering to specify family = "binomial" to run a logistic regression.
- Then we use the model to make predictions about whether a customer will default based on the predictors in the database.
- The last step is to check how our model will perform on our test set. So, when the probability of the prediction is greater than 0.5 then the customer will default (1) and when the probability is less than 0.5 then the customer will not default (0). But we can also choose the optimal cutoff function to find the optimal probability to use to have the most perfect accuracy for the model.
- We can then calculate the confusion matrix which compares the predictions obtained by the model and the true values. This confusion matrix then allows us to calculate the AUC metric and the higher the AUC, the better the model will predict the results.

One of the main advantages of logistic regression is that it is an easy model to set up, the algorithm is also easy to interpret and gives very good results. Logistic regression also allows us to interpret the coefficients as indicators of the importance of each variable. The coefficient tells us the direction of the importance, positive or negative. However, logistic regression has limitations, especially if the number of observations n is less than the number of predictors p, as this leads to an over-fitted model. In addition, logistic regression imposes a weak multi-collinearity between all independent variables. Many more powerful algorithms exist on the market such as neural networks which can obtain complex relationships between variables.

3. Decision tree

The decision tree is another supervised machine learning algorithm, it will allow us to predict a category 0 or 1. In fact, the decision tree is very well known in the world of datasience and has given birth to much more powerful algorithms today like RandomForest or XG boost. This method is easy to explain to businesspeople because it is very visual and logical to understand. This algorithm is used when we put more importance on the interpretability of the results than on the performance of the results. This algorithm considers all possible decisions and excludes none in order to conduct the classification. Moreover, this algorithm is not disturbed by the presence of missing values. In fact, this algorithm is intuitive for us humans and we are necessarily attracted by algorithms that we understand easily. The algorithm does not take long to run and therefore does not cost much in computer power. However, the performance of a decision tree is quite low and is easily overtaken by other machine learning algorithms like the Random Forest for example. This algorithm has a high risk of overfitting. That is to say that the model performs very well on the train set and in fact performs very poorly on non-exploited data (test set)

4. K-Nearest neighbours

The nearest neighbor algorithm is a supervised machine learning algorithm. This algorithm is used to answer classification problems. The algorithm works on the principle that similar things can have an identical class, basically similar things are next to each other. For example, the fish in a school of fish are close to each other because they belong to the same fish family. This algorithm competes with other algorithms because it gives us predictions with a very high accuracy. Contrary to the decision tree, this algorithm will be used when we are looking for performance rather than interpretability of the model. Moreover, the quality of the model depends on the distance of measurements (K).

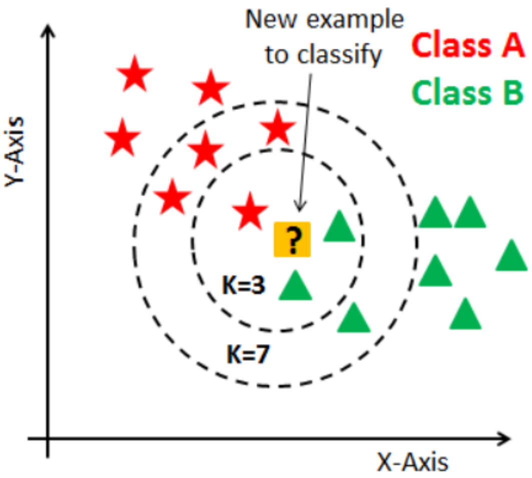

Figure 1: K-nearest neighbours' mechanism

Let us take as an example red stars and green triangle to be classified. The KNN algorithm is used to determine the class of the unclassified point. The first step of the model is to select the number of K neighbors, for example 5 on the picture on the left. Then we count the number of neighbors next to the unclassified point and take the majority of the class present and assign it to the unclassified point. The advantages of KNN are that it is an easy algorithm to implement and can be used for classifications and regressions. It is a very interesting algorithm when one has several classes to predict. Also, this algorithm allows the user to choose the method to calculate the distance: Euclidean distance, Hamming distance, Minkowski distance, Manhattan distance, etc... However, this algorithm has many limitations. Firstly, the larger the database becomes, the less efficient the algorithm becomes. Furthermore, when we add more independent variables then output becomes much more challenging. Of course, the main problem in this algorithm is to find the right number of neighbors to fix. Also, this model cannot consider missing values because it must calculate the distance between the points so there must be a value. The outliers have a very strong impact on the model because it must calculate very large distances with outliers

5. Support Vector Machine

SVM is a supervised machine learning algorithm that addresses classification and regression problems. SVM is a very flexible and reliable algorithm with complex and robust models. This algorithm was developed in the 1990's and aims to separate the data into different groups of data through a "boundary". Let's dive into how this algorithm works. We give the algorithm data where we already know the two classes so that it can train and learn the data. The SVM will come and determine this boundary that will separate these data into two distinct groups. In fact, the algorithm will learn after several trainings where the boundary that best separates the data is. The objective is that the distance between the data and the boundary that separates them is maximum. The points where the distance is minimal are called "support vectors". These vectors are very important because the boundary will depend only on these support vectors. These vectors come to bring their support, hence the name of the vectors. The support vectors are essential to find the boundary which does not separate all the data correctly, but they do separate them optimally. The data are therefore linearly separated by this boundary, but it is also likely that the relationship is not linear. This is where the kernels come in. These are mathematical functions that will project the data into several dimensions. This algorithm has a considerable advantage, it is that there are very few parameters and the default parameters that there are work in most cases. This allows many people who are not necessarily data experts to use these algorithms. The performance of this model is like neural networks but unlike neural networks it does not require us to understand how the model works to apply it. Moreover, to be effective this algorithm needs a large volume of data to be trained. This algorithm is unfortunately very expensive to run on the machine because it needs a lot of computing power.

	Advantages	**Disadvantages**
Logistic regression	○ Easy to set up, ○ Easy to interpret ○ Good results.	○ Zero multicollinearity between variables required ○ Not the most efficient prediction algorithm ○ Impossible when n < p
Decision tree	○ Not disturbed by missing values ○ Intuitive ○ Easy to understand ○ Not expensive in energy to run.	○ Low performance ○ Important risk of overfitting
K-Nearest Neighbor's	○ Easy to implement ○ Choice in distance calculation method (Euclidean, etc..)	○ Choosing the right number of k neighbors is complicated ○ The larger the database the less efficient the algorithm becomes ○ Outliers distort the model enormously
Support Vector Machine	○ Very few parameters and very powerful default parameter, optimal results.	○ Very long to run ○ Need a large database to get good results
Random Forest	○ Very powerful default parameter, versatile algorithm (classification/regression), no need to scale all variables to the same level.	○ Complex algorithm, trees construction is difficult, expensive in computing power.

B. Models benchmark and application to the credit card default dataset.

When we run the algorithms without feature selection or cross validation, we have unreliable models with too large gap between train accuracy and test accuracy and especially low AUC. We can see that apart from logistic regression and support vector machine, the gap is too large which shows that the model is overfitting and is therefore biased.

	Accuracy		**AUC**	
	Train	**Test**	**Train**	**Test**
Logistic	0.78	0.79	0.50	0.50
Decision tree	1.00	0.72	1.00	0.61
Random Forest	1.00	0.83	1.00	0.67
Knn	0.84	0.73	0.71	0.55
Svm	0.78	0.79	0.50	0.50

To have efficient, reliable and less overfitting models, the ideal is to select the most important independent variables to run our model, which capture the most information about the dependent variable. This overfitting often comes from some variables that have absolutely nothing to do with the independent variable we want to predict. This bias is very often seen when we want to over-interpret data. Many variables can be correlated with each other when there is absolutely no link between them. In fact, overfitting is a phenomenon that will degrade the performance of the machine learning model, to avoid it we can use cross validation and feature selections

Features Selection

The purpose of feature selection is to identify the variables that have the greatest influence on the dependent variable. In fact, the aim is to eliminate all the columns called "noise" that do not add any value to our model. All the variables that do not make sense in predicting Y, then we do not take them into account in our model. We must be careful because the variables that are very important to predict Y for the random forest will not necessarily be the same important variables for another algorithm (decision tree for example). In fact, the combinations of variables can be completely different according to the models. So, it is imperative to make very precise feature selection for each model. Then, once the features are selected, we can do a cross validation and obtain much more reliable and higher metrics than without features selection. Finally, reducing the number of features is extremely important when we have a large database. This will allow the model to be faster and above all to avoid overfitting

Cross validation

Once we have these variables, we run the cross-validation algorithm specific to each model which will allow us to find the best hyper parameters by trying many combinations. In addition, the cross validation will allow us to test the performance of the machine learning model we are applying. Once the model is built, then this model will be used to predict on untrained data. And so, we need to measure the accuracy of the predictions. There are several cross-validation techniques. Firstly, the simplest method, K-fold, allows us to ensure that all observations in the database can be represented equally in the training test and the test set. It is up to us to choose the K fold number, which will represent the number of times the sample is split. The more we increase the K, the more the model will be biased and vice versa, so it is important to choose a K greater than 5 and less than 10. Then each fold will have its metrics (accuracy, AUC, etc...) and the algorithm will average these metrics to measure the overall performance of the model. Most of the time the accuracy gap is greatly reduced between the train and the test set and we now have a model that is reliable and accurate and no longer overfitted.

1. RandomForest

We use step forward feature selection to choose the best variables for our models. Without doing any feature selection or cross validation our model has a train accuracy of 0.99 and a test accuracy of 0.82. This difference between these two accuracies explains why the model is overfitting. Therefore, we will work to reduce this gap by selecting only a few significant variables. We choose to section only the most important feature with the step forward features selection algorithm. This is a method that starts with 0 variables and then adds the variables that capture the most information about the target variables one after the other. The most significant variables are chosen when the p-value is the lowest or when the R2 is the highest. In our case we select the 5 most significant variables which turn out to be variables 3,4,6,7,10. Then we look for the best parameters for this model using the RandomizedSearchCV algorithm. Finally, we run the model with the 5 significant variables and the best parameters. The results of this algorithm are encouraging, especially since the gap between the train and test set is really reduced with a train accuracy of 0.81 and a test accuracy of 0.83, so the model is reliable, efficient, and no longer overfitting. And we have an AUC of 0.67 which is not bad at all for this model.

2. Logistic regression

For the logistic regression, when we look at the results of the model without feature selection or cross validation, we have scores that are not overfitting with a train test of 0.78 and a test set accuracy of 0.79. When we plot the feature importance of the logistic regression, we do not observe a clear pattern between the important and less important variables. The graph shows us how the variables impact whether a customer will default or not. When the coefficient is positive, it affects whether the customer will default. And when the coefficient is negative, it means that the variable impacts on the fact that a customer does not default. Here the coefficient never exceeds 0.0001 so it is not significant. Therefore, we will not perform a feature selection but only a cross validation to select the best parameters to run the logistic regression. With the new optimized parameters, we have a train accuracy of 0.80 and a test accuracy of 0.82.

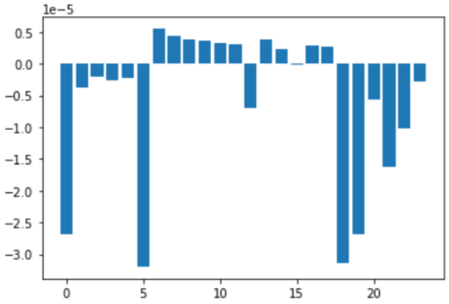

Figure 2: Features importance for logistic regression

Moreover, when we calculate the AUC of our model with the cross validation it jumps from 0.5 to 0.61 for the test set and when we look at the train test it is the same, so it is consistent, and the model is not overfitting anymore. We can therefore conclude that cross validation and feature selection has enabled our model to be much more efficient and accurate.

3. Decision tree

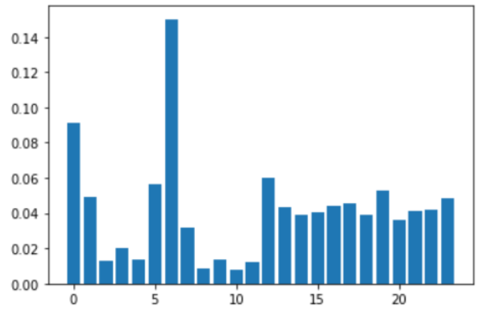

Figure 3: Features importance decision tree

The decision tree model is completely biased when we do not perform feature selection or cross validation because we have a training accuracy of 1 and a test accuracy of 0.72. We will therefore select the most significant variables to predict the number of default clients. Then with the selected variables we carry out a cross validation to obtain the best parameters for the decision tree model. The results are very encouraging since we now have a train accuracy of 0.81 and a test accuracy of 0.83. Moreover, we obtain a much higher AUC with the features selection and cross validation. Indeed, the AUC increases from 0.62 to 0.66 for the test set. Thus, the model is much more consistent and reliable than without features selection and cross validation.

4. K – nearest neighbors

For the K-nearest neighbor algorithm, we observe that the model is overfitting as it has a train accuracy of 0.82 and a test accuracy of 0.72. We therefore carry out a cross validation to optimize our results. And so, we now have a test accuracy that goes from 0.72 to 0.78 and there is no more overfitting. As for the AUC, we notice that it remains quite low below 0.55 which proves that for this database this model is not adapted and will not get a good score.

5. Support Vector Machine

For the support Vector Machine algorithm, we see that the model is not overfitting when we take all the variables, and the model is well represented with a train accuracy of 0.77 and a test accuracy of 0.78. When we run the algorithm with a 10-fold cross validation we get a training score of 0.78 and a test accuracy of 0.79 and we do not observe any significant increase. When we run the algorithm with a 10-fold cross validation we obtain a training score of 0.78 and a test accuracy of 0.79 and we do not observe a significant increase.

Below is a **summary table of the benchmark of machine learning algorithms** we studied:

	No features selection & no cross validation				**After Features selection & Cross validation**			
	Accuracy		**AUC**		**Accuracy**		**AUC**	
	train	**test**	**train**	**test**	**train**	**test**	**train**	**test**
Random Forest	1	0.83	1	0.67	0.83	0.82	0.67	0.67
Logistic regression	0.78	0.79	0.50	0.50	0.80	0.82	0.59	0.61
Decision tree	1	0.72	1	0.62	0.81	0.83	0.63	0.66
K-Nearest Neighbor’s	0.84	0.73	0.71	0.55	0.78	0.78	0.53	0.52
Support Vector Machine	0.78	0.79	0.5	0.5	0.78	0.79		

Conclusion

To conclude, after studying the different models, we can conclude that there are models that are more efficient than others. But that can change depending on the type of data analyzed. For our study, we seek to identify the customers who will have a payment default and those who will not, what interests us is high precision with a high AUC to properly identify these two classes. After having optimized the performance of the algorithms by choosing the best choice of parameters for each algorithm or the selection of the most important variables capturing the most variance, we deduce that the model which best identifies these two classes is the Random Forest model. To go further in this study, it could be interesting to analyze other algorithms such as XGboosting or even neural networks.

References

https://www.datatechnotes.com/2020/06/classification-example-with-svc-in-python.html#:~:text=Classification%20Example%20with%20Support%20Vector%20Classifier%20%28SVC%29%20in,and%20we%20can%20use%20it%20in%20classification%20problems

https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74

https://www.kdnuggets.com/2018/06/step-forward-feature-selection-python.html

https://machinelearningmastery.com/calculate-feature-importance-with-python/

https://www.geeksforgeeks.org/svm-hyperparameter-tuning-using-gridsearchcv-ml/

https://medium.com/analytics-vidhya/decisiontree-classifier-working-on-moons-dataset-using-gridsearchcv-to-find-best-hyperparameters-ede24a06b489

https://datascienceplus.com/k-nearest-neighbors-knn-with-python/

https://datascienceplus.com/k-nearest-neighbors-knn-with-python/

https://www.mygreatlearning.com/blog/knn-algorithm-introduction/

- Stackoverflow community
- Iesegonline - Statistical and machine learning 0754
- *An Introduction to Statistical Learning (with application in R),* Garet James, Daniela Witten, Trevor Hastie, Robert Tibshirani.