Individual Assignment

Statistical & Machine Learning for Marketing

Student

Mohammad Hadi Alipour Motlagh

MBD IESEG 2021-2022

Contents

Introduction

In this research on banking data in the field of lending, I will introduce and briefly describe how the machine learning models are applied to analyze and predict this data set.

In the following, I will explain how each of these models works, how to implement them on the data, the basic steps for preparing the database, and how to create the initial base table and in the end, the achievements and final results are presented.

It should be noted that machine learning models include a wide range from simple to complex models, which sometimes make it difficult to explain how decisions have been made in complex models. And as we all know, in addition to looking for successful results, businesses still expect the results to be clear to them and they look for interpretable models.

This becomes even more important when dealing with banks and financial businesses. Transparency is one of the tasks of financial services and banks should be able to explain how their models work and defend their performance at the customer's request. Therefore, the models studied and used here have been selected taking into account the conditions of the financial industry.

Machine Learning Models

In machine learning, the subject is the design of machines that learn from the examples given to them and their own experiences. In fact, in this science, an attempt is made to design a machine using algorithms in such a way that it can learn and operate without explicitly planning and dictating each action.

In machine learning, instead of programming everything, the data is given to a general algorithm, and it is this algorithm that builds its logic based on the data given to it. Machine learning has a variety of methods, including supervised, unsupervised, and reinforcement learning.

Machine learning is the subject of studies that have emerged from artificial intelligence. Humans use artificial intelligence to build better and smarter machines. But researchers were initially unable to program machines to perform more complex tasks that are constantly challenging, except for a few simple tasks, such as finding the shortest path between points A and B.

Accordingly, the perception was formed that the only possible way to achieve this was to design machines that could learn from themselves. In this approach, the machine is like a child learning from itself. Thus, machine learning was introduced as a new capability for computers. Today, this science is used in various fields of technology, and its use has become so widespread that people are often unaware of its existence in their daily tools and accessories.

Supervised learning

Most machine learning methods use supervised learning. In supervised machine learning, the model tries to learn from the previous examples provided to it. In other words, in this type of Machine learning, the Model tries to learn the patterns based on the given examples.

Unsupervised learning

In unsupervised learning, the algorithm by itself must look for structures in the data. In other words, unsupervised learning refers to when there are only input variables *X* in the data set and no output data

variables. Therefore the name is unsupervised because, unlike supervised learning, there is no correct answer given, and the machine itself must look for the good answer.

Reinforcement learning

A computer program that interacts with a dynamic environment must achieve a specific action. The program provides feedback based on gain and loss and directs the issue accordingly. Using reinforcement learning, the machine learns to make specific decisions in an environment that is constantly subject to trial and error.

Supervised learning

As mentioned, supervised learning methods are more popular than other algorithms. In general, these algorithms can be divided into two main categories according to the type of objective variable, which includes regression and classification.

In regression, we seek to predict a range of data, such as forecasting sales, or revenue and expenditure. While in classification issues the target variable includes two or more limited options such as predicting customer churn or non-churn, profit or loss, default or not most of the classification problems are binary however there are multiclass classification problems as well.

If we want to study these problems mathematically, we can say that in the regression problems, we seek to optimize the loss function by minimizing the cost in this function, in other words, we want to calculate line or hyperplane which goes through the data and have minimum distance possible with all data points.

In classification problems, however, we seek to maximize the likelihood function so that we can find a function that optimally separates the two groups.

Considering the type of target Variable in our database used in this project, we are looking to find a function to predict credit card fraud. Therefore, this problem falls into the category of classification algorithms that use supervised learning. Therefore, in the following, I will review the 5 algorithms used in this field and then I will explain how to implement and measure the performance of each of these algorithms and on the data.

Logistic regression

One of the methods of "classification" in the topic of "supervised machine learning" is logistic regression. In this regression method, the concept and method of calculating the "odds ratio" are used. Therefore, it is better to get acquainted with this concept first.

To get acquainted with the concept of odd, let us give an example. Suppose in a family with six children, the ratio of boys to girls is equal to 2/4, This ratio shows that the number of girls in this family is twice the number of boys. On the other hand, we know that in such a family, the probability of choosing a son from among the children is equal to 2/6 And such a probability for girls is equal to 4/6 Now, if event A is considered the choice of one of the boys among the children, the chance for such an event is equal to:

$$Odd(A) = \frac{P(A)}{P(A')} = \frac{\frac{2}{6}}{\frac{4}{6}} = \frac{2}{4} = \frac{1}{2}$$

And if event B is the choice of one daughter the Odd will be equal to 2. This number shows that girls are twice as likely to choose as boys. As can be seen, the amount of luck is different from the probability because here, for example, the amount of luck is greater than 1.

We know that linear regression means creating a parametric linear relationship to represent the relationship between an independent and a dependent variable. The form of a simple linear regression model is as follows:

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

As can be seen, this relation is the equation of a line, which is the error sentence or the same ε Added to it. The parameters of this linear model are the width of the origin β_0 and line slope β_1 are. In this case if Y The estimated value for the dependent variable can be considered as the average of the observations for the dependent variable for the constant value of the independent variable. So if we replace the mean with mathematical expectation, assuming that the mean of the error sentence is also zero, we will have:

$$\hat{y} = E(Y|X = x) = P(Y = 1|X = x) = p(x)$$

In this way if possible for the function p(x) A pattern is considered, then the regression model for the Bernoulli dependent variable is specified. According to this definition, we determine the estimation of logistic regression parameters.

Since in the previous section the prediction value for the dependent variable was made with a probability of p (x) ، to determine the relationship model between the dependent and independent variables instead of the linear relationship, we need a function that varies from about 0 to 1. In the logistic regression method, a function called "Logistic Function" is used. For this reason, this regression method is called logistic regression. In the continuation of this function, the introduction and its related diagram based on the parameters $b_0 = 0$ and $b_1 = 1$ can be seen in the following:

$$F(x) = \frac{e^{b_0+b_1 x}}{1 + e^{b_0+b_1 x}}$$

As can be seen by increasing the value of x The logistics function will be close to 1. Also by reducing the value of x The value of the function tends to zero. Now suppose this function is used for logistic regression to express the probability of a dependent variable. So we will have:

$$p(x) = \hat{Y} = E(Y = 1|X = x) = \frac{e^{b_0+b_1 x}}{1 + e^{b_0+b_1 x}}$$

In order to estimate the parameters of this model, "Logit Transformation" can be used. This conversion is on odd that provide earlier, we execute. In this case, the relationship can be written as follows:

$$g(x) = ln(\frac{P(x)}{1 - P(x)}) = \frac{\frac{e^{b_0+b_1 x}}{1 + e^{b_0+b_1 x}}}{1 - \frac{e^{b_0+b_1 x}}{1 + e^{b_0+b_1 x}}} = ln(e^{b_0+b_1 x}) = b_0 + b_1 x$$

Pros and cons of Logistic Regression

- advantages

    - It does not require a lot of computational resources,
    - it is very interpretable,
    - Another advantage of logistic regression is that it is very easy to implement and training is very efficient.
- Disadvantages
    - If the train set is small the overfitting probability of Logistic regression will be high.
    - It just can be used for binary classification.

K nearest neighbors algorithm

K-nearest neighbors algorithm is a non-parametric statistical method used for statistical classification and regression. In both cases, K contains the closest instructional example in the data space, and its output varies depending on the type used in the classification and regression. In the classification mode, according to the value specified for K, it calculates the distance of the point that we want to label with the nearest points, and according to the maximum number of votes of these neighboring points, decides on the label of the desired point. Various methods can be used to calculate this distance, one of the most important of which is the Euclidean distance.

In the K neighbors algorithm, we want to determine data with K group consisting of records from the set of training records that are closest to the experimental record and decide on the category of the experimental record based on the superiority of the category or the label associated with them.

In simpler terms, this method selects the categories that have the highest number of records in that selected neighborhood. Therefore, the category with the highest number of nearest neighbors among all K categories is considered as the new record category. The basic idea of the KNN method is that if a creature walks like a duck and quacks like a duck, then it must be a duck.

Using the KNN algorithm requires determining three issues:

- We must have a set of records.
- We must also have a criterion for calculating similarity.
- The value of K must also be specified so that it can be acted upon.

How it works

In KNN classifications, the following is done to categorize a record with an unspecified category:

- The distance of the new record is calculated from all training records.

- The nearest neighbors are identified.

- The nearest neighbor category K tag is used to predict the new record category.

In this way, between the voted K record and the category that has the highest number of views among this K record, will be considered as the new record category.

How to select K value

For binary classification problems, it is usually better to consider the odd number for K. Because it increases the possibility of winning one of the two categories.

For multi-category classification problems, the number K should be considered larger than the number of categories and also different from the number of categories in terms of even or odd. That is, if the number of categories is even, the final K must be considered odd, and vice versa.

Selecting the value of K in this classification method is a very important and key factor. If the value of K is selected too small, the algorithm becomes sensitive to noise. Noise close to that record may be misleading. If the value of K is selected too large, records from other categories may also be placed among the nearest neighbors.

When K is selected as a large number, it will result in a classifier error in classifying the input record. One of the ideas presented to solve this problem is to define the weight factor. This factor assumes a weight of (1/d2), which is the value of d representing the distance between each record and the input record. In this way, distances become important to the algorithm, and this weighting causes the records that are closer to the input record to be given more importance.

Calculation of distance

Based on the value of the variables there are two main kinds of measurement strategy, if the values are categorical we can use Hamming distance (1) and if they are continues we can choose between Minkowski distance (2), Euclidean distance (3), and Manhattan distance (4).

1. $D_H = \sum_{i=1}^{k} |X_i - Y_i|$
2. $\left(\sum_{i=1}^{k} (|X_i - Y_i|)^q\right)^{1/q}$
3. $\sqrt{\sum_{i=1}^{k} (X_i - Y_i)^2}$
4. $\sum_{i=1}^{k} |X_i - Y_i|$

Pros and cons of KNN

- advantages
  - Simple algorithm
  - Relatively high accuracy
  - Multipurpose - for classification and regression
- Disadvantages
  - Expensive calculators
  - High memory requirement - because the algorithm stores all previous data.
  - The prediction phase may be slow (with large N)
  - Sensitive to inappropriate features and data scale

Decision Trees

In general, decision tree analysis is a predictive modeling tool that can be used in many fields. Decision trees can be created using an algorithmic solution that can differentiate data sets based on different conditions in different ways. Decision trees are one of the most powerful algorithms that are considered a subset of supervised algorithms.

They can be used for both classification and regression tasks. The two main elements of a tree are the decision nodes where the data is distributed and the leaves from which we get the output. The following is an example of a binary tree that provides a variety of information.

The tree will be created based on the three main components:

The root: The main root that can consist of all variables we selected for training the model

The Branche: the Variables that model selected based on their importance and contribution in defining the target variable

The leaf: the condition (status of selected variable and its index.

There are different applications to define the decision index like CART, ID3, C4.5, CHID, in the Tree Models here we will discuss the Gini Index and entropy. I will present it in the following lines.

Methods

Gini Index and CART

This algorithm is used to establish regression and classification trees. Presented in 1984 by Leo Breiman et al. The important point is that this algorithm creates binary trees so that two edges come out of each inner node and the resulting trees are pruned by the cost-effectiveness method.

One of the features of this algorithm is the ability to generate regression trees. In this type of tree, the leaves predict the actual value instead of the class. The algorithm searches for the separators, the minimum square error. In each leaf, the prediction value is based on the average error of the nodes.

It is the name of a cost function used to evaluate the binary separation of a data set, and works with the definite objective variables success and failure.

The higher the Gini index, means the higher the homogeneity. The ideal value for the Gini index is 0, and the worst value for a problem with two classes is 0.5. Using the following steps, the Gini index can be calculated for a breakdown.

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

The classification and regression tree (CART) algorithm generate binary segregation using the Gini method. $\hat{p}_{mk}$ is the number of observations in the selected region of the model.

Entropy and ID3

ID3 is one of the very simple decision tree algorithms developed by Ross Quinlan in 1986. The information obtained is used as a criterion for differentiation. This algorithm does not use any pruning process and does not take into account nominal and missing values.

If we choose ID3 algorithm then the entropy formula will used to make the decision tree as follow:

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \ \log \hat{p}_{mk}$$

Building a tree

We can define the creation of a model based on these steps:

Step 1: Calculate index Score,

Step 2: Separation of a data set, It can be defined as separating a data set into two lists of rows that have an index of a property, and a split value of that property. After taking the two groups of right and left from the data set, we can use the index(like Gini score) score, which was calculated in the first step, to calculate the separation value. The value of the separation will decide which group the attribute will remain in.

Step 3: Evaluation of segregations, The next step after finding the index score and splitting the data set is to evaluate all the groups. To this end, we must first consider all the values associated with each attribute as a separation candidate. Then, we need to estimate the cost of separation to find the best possible separation point. The best separation will be used as a node in the decision tree.

Pros and cons of Decision tree

- advantages
  - Self-explanatory, easy for interpretation
  - No need for variable selection, it will be done by the model
  - We can evaluate the performance of the model using other statistical methods
  - It will run even in the raw data (data that not processed)
- Disadvantages
  - Bad performance for regression models
  - If the number of classes is high the probability of failure will be high.
  - It is expensive in respect of resources and time
  - If the tree is large, errors can accumulate from surface to surface (accumulation of layer errors and the impact on each other).
  - Designing an optimal decision tree is difficult, and the performance of a classification tree depends on how well it is designed.
  - If the nodes overlap, the number of end nodes increases.

Random Forest

Random forest is basically like a bag containing n decision trees that have different sets and are taught under different data sets. Suppose we decide to have 100 trees that were created randomly in a random forest. As I said, these decision trees have a different set of parameters and a subset of different training data, so the decision or prediction provided by these trees can be very different.

Let's say we train all 100 of these trees with their data sets. All 100 existing trees are now being examined for their predictions about test data. Now we just have to decide for an example or a test data, we do it by a simple voting. We accept as a result what most trees have predicted for that example.

The Random Forest algorithm is essentially a supervised learning algorithm. This can be used for both regression and classification tasks. But in this article, we will discuss its use for random Forrest classification because it is more intuitive and easier to understand and as mentioned earlier our problem is classification. Random forest is one of the most widely used algorithms due to its simplicity and stability.

The word "random" is displayed when constructing a subset of data for trees. A subset of data is created by randomly selecting x the number of attributes (columns) and the number of y samples (rows) from the original set of n attributes and m examples.

Random forests are one of the most well-known ensemble methods, they are more stable and reliable than a decision tree.

Steps to create a random forest model:

Step 1: Select data points from the tutorial set.

Step 2: Create decision trees based on selected data.

Step 3: create N decisionTrees.

Step 4: Repeat steps last two steps.

Step 5: group data and take the average of the results and aggregate the results.

As I mentioned above, we are creating a bag of trees, and therefore the other name for the random forest that is used in the books is bagging, if we have a different function for each decision tree and we know that each of these functions (F) has applied on a different set of the data B the function for the random forest can drive from an average of all functions and will be as follow:

$$\hat{f}_{avg}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}^{b}(x)$$

Since we need to have one final dataset that applies the model, the final model based on the bootstrapping of all sets and choosing the final dataset will be the final function and it is as follow:

$$\hat{f}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}\,*^{b}\,(x)$$

Pros and cons of Random Forest

- advantages
  - having different Hyperparameters to try fit better Model
  - Can give user variable importance
  - Mostly will have better result than decision tree
  - Good for multiclass problems
- Disadvantages
  - The probability of Overfitting in Random Forests are high
  - Costly in respect of Time and resources.
  - The number of trees must choose wisely

Support Vector Machines

The original SVM algorithm was invented in 1963 by Vladimir Vapnik and generalized in 1995 by Vapnik and Corinna Cortes for the nonlinear model.

Support vector machines is supervised learning used in classification and regression. This method is one of the new methods that has good performance compared to older methods for classification. The basis of the SVM classifier is a linear classification for data, and in linear segmentation of data, we try to choose the line that has the maximum margin between two or more datagroup.

The SVMs are distinguishing classifiers defined by a separator hyperplane. In other words, by receiving labeled training data (supervised training), the algorithm will create an optimal separator hyperplane that separates new samples. the objective function in this model is the Line or hyperplane that can separate the groups of data in a way that:

1. All data which belong to the same group stand on the same side of the hyperplane
2. Maximize the distance between the nearest data point to the hyperplane

Then the Objective function that we want to solve will be as follow:

Maximizing the distances $\beta_0, \beta_{11}, \beta_{12}, \ldots, \beta_{p1}, \beta_{p2} \ldots, \epsilon_n, M^M$ Subject to

$$y_i\left(\beta_0 + \sum_{j=1}^{p} \beta_{j1} X_{ij} + \sum_{j=1}^{p} \beta_{j2} X_{ij}^2\right) \geq M(1 - \epsilon_i),$$

$$\sum_{i=1}^{n} \epsilon_i \leq C\ , \epsilon_i \geq 0, \sum_{j=1}^{p} \sum_{k=1}^{2} \beta_{jk}^2 = 1$$

The Optimal Separator hyperplane was first introduced by Vapnik in 1963 as a linear separator. In 1992, Bernhard Boser, Isabelle Guyon, and Vapnik proposed a way to create a nonlinear classifier, using the kernel to find the most marginal hyperplane. The proposed algorithm is similar, except that all point multiplications by A nonlinear core function have been replaced. This feature allows the algorithm to be suitable for the hyperplane with the most margins in a transformed multidimensional space. The deformation may be nonlinear and the multidimensional space may have higher dimensions. The categorizer, however, is a hyperplane in the multidimensional space with large dimensions, which may also be nonlinear in the early space.

Pros and cons of SVMs

- advantages
  - Can be very efficient, especially for binary classifications
- Disadvantages
  - The mos Expensive model Between these five models That I select
  - No good performance for multiclass problems
  - We cant use this method with categorical and binomial variables

Application of Models

Here I will briefly explain the steps I did to prepare the data, create the base table, do some primary analysis. How I select the Variables using different methods and how I applied Models, what are the model's parameters and results like accuracy. And in the final step, I will discuss the conclusions.

Data preprocessing

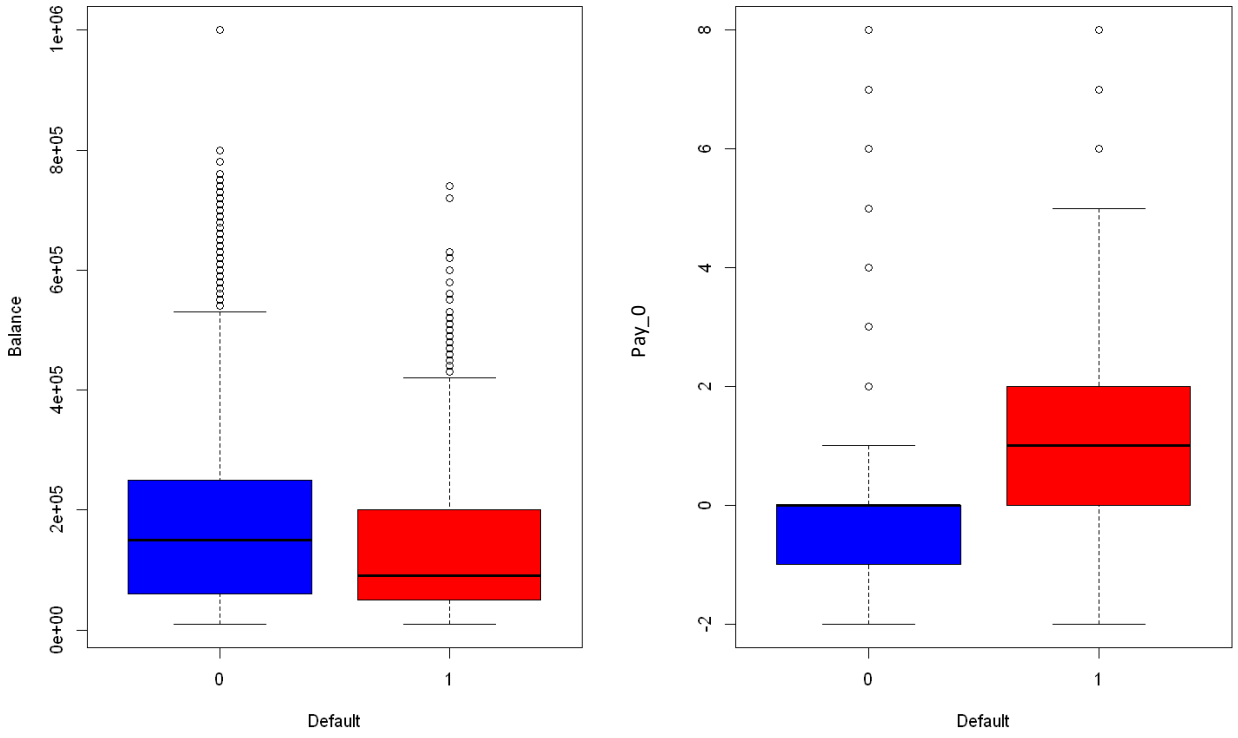
In this stage I did these main tasks for preparing the data:

1. Dealing with missing data: I replace the missing data in categorical and binary columns with the mode of that variables and for the rest, I replaced them with the median
2. One hot encoding: For binary and categorical variables I did one-hot encoding and I dropped the first ones.

I save the table and load the new file for my next step.

Data description

I did some data description using pair plots that you can see in the notebook, here I just add two boxplots to explain the result of the description, since this is not the main part of the project I just explain these two plots.



As we can see in the above chart there is a significant difference between the two groups of default and non-default in respect of average balance amount and their payment average in the first month.

We can see that the average balance of the default group is significantly less than the non-default group and also the default group used to spend more money in the period of the first payment.

We can assume that these two variables can have a significant coefficient and high contribution to the final prediction and main model. However, this is just an assumption in this stage and we need more exploration in this respect. Therefore the next step is feature selection.

Feature Selection

In this step, I use all possible feature selection algorithms like backward and forward, Lasso and Ridge, and dimensionality reduction like PCR and PLS. I combine the final results of all models after applying all necessary steps for each one. The next chart shows the features selected by each method.

At the End based on the all feature selection models my final features are:

1. 'LIMIT_BAL'
2. 'BILL_AMT2'
3. 'PAY_AMT4'
4. 'PAY_0_2'
5. 'PAY_AMT6'
6. 'PAY_3_3'
7. 'PAY_AMT1'
8. 'PAY_6_2'
9. 'PAY_6_3'

You can see in the notebook that models suggested categorical features but I decied to use numerical features as well like ballace and Pay_0, this have two main reason:

1. I applied suggested variables and get bad results and the variables coefficient were not significant
2. Having toomany binary variables is not good for models like Knn and it cannot work.

We can see that all variables I choosed have significance coefficient in models.

Performance of Models

Here I provide my result for each model seperately

Logistic Regression

For logistic regression my accuracy for train set was 0.81 and I choosed three thresholds (0.25,0.5,0.75) and the Acuracy and AUC for each of them are as follow:

Threshold	Acuracy	AUC
0.25	0.8	0.69
0.50	0.81	0.74
0.75	0.8	0.76

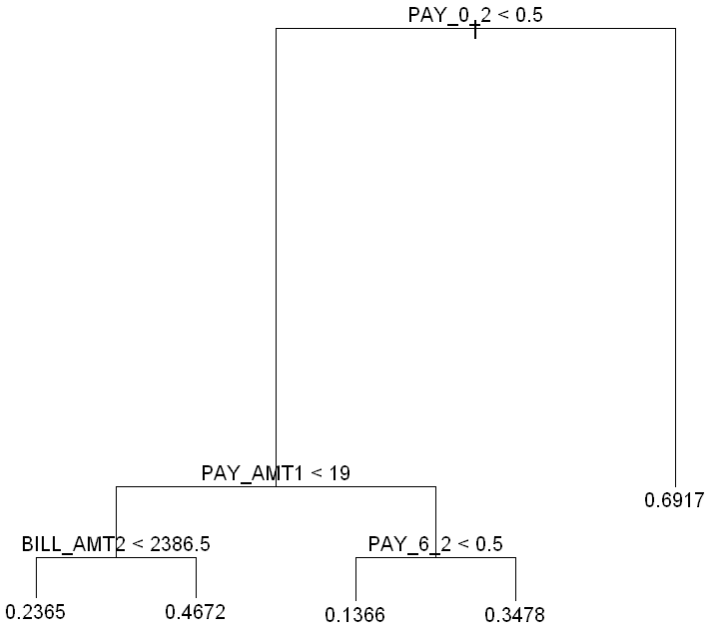
I applied the cross validation on train dataset and the results of average auce was 0.72 and the accuracy is 0.81.

KNN

For KNN my train set accuracy is equal to 0.79 and the AUC is 0.70 and the result for test set is Acuracy equal to 0.61 and AUC equal to 0.61. for the cross validation my accuracy is 0.81 and AUC is 0.73

Decision Tree

In the decision tree I got 3 categorical and one numerical variables that contribute to creation of the model. The accuracy of train set for decision tree is 0.81 and the AUC is 0.75. The accuracy of test set is equal to 0.81 and the AUC is 0.74 and my threshold for this results is 0.50.



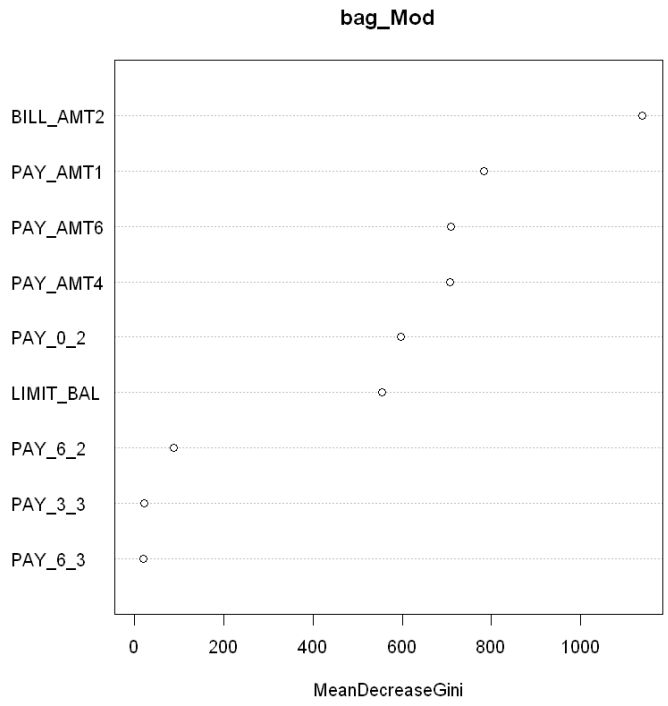
The AUC of Cross validation of train set for decision tree is equal to 0.61 and the accuracu is 0.81.

Random Forest

Based on the variables I select below you can find the Gini index and importance of the variables for RandomForest model. My accuracy and AUC for train set are respectively 0.99 and 0.98 and for the test set are 0.8 and 0.7 and this is the exact example that what I discuss earlier about the drawbacks of the Random forest.

This model learned the train set by heart and there is obviously overfitting for random forest.

The AUC od cross validation of train set for random forest is 0.71 and the accuracy is 0.80.



SVM

My accuracy and AUC for train set are respectively 0.81 and 0.75 and for the test set are 0.81 and 0.74 The AUC of cross validation of train set for SVM is 0.68 and the accuracy is 0.81.

Conclusion

We can see the AUC and Acuracy of the all models for train,test and cross validation in below table.

	Train ACC	Train AUC	Test ACC	Test AUC	CV ACC	CV AUC
Logistic Regression	0.81	0.75	0.81	0.74	0.81	0.72
KNN	0.79	0.70	0.79	0.61	0.81	0.73
Decision Tree	0.81	0.75	0.81	0.74	0.81	0.61
Random Forest	0.99	0.98	0.8	0.7	0.80	0.71
SVM	0.81	0.75	0.81	0.74	0.81	0.68

Looking to this table I can say that the most stable and model is the SVM and it is also the most time cosuming model. The decision tree and and logistic regresson is my second and third models.

Refrences

1. James, G, et al (2017), An Introduction to Statistical Learning. Springer.

2. https://en.wikipedia.org/wiki/Support-vector_machine
3. https://en.wikipedia.org/wiki/Logistic_regression
4. https://en.wikipedia.org/wiki/Decision_tree
5. https://en.wikipedia.org/wiki/Random_forest
6. https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm
7. https://blog.faradars.org/support-vector-machine-simplified/
8. https://blog.faradars.org/understaing-support-vector-machine-example-code/
9. http://cafetadris.com/blog/%D9%85%D8%A7%D8%B4%DB%8C%D9%86-%D8%A8%D8%B1%D8%AF%D8%A7%D8%B1-%D9%BE%D8%B4%D8%AA%DB%8C%D8%A8%D8%A7%D9%86-support-vector-machine/
10. https://virgool.io/@pbarjoueian/svm-tutorial-overview-eaxqctpf4bvu
11. https://www.matlab1.ir/%D8%A2%D8%B4%D9%86%D8%A7%DB%8C%DB%8C-%D8%A8%D8%A7-%D9%85%D8%A7%D8%B4%DB%8C%D9%86-%D8%A8%D8%B1%D8%AF%D8%A7%D8%B1-%D9%BE%D8%B4%D8%AA%DB%8C%D8%A8%D8%A7%D9%86-support-vector-machine-%DB%8C%D8%A7-svm/
12. https://mbainfinity.com/support-vector-machin/
13. https://www.bigdata.ir/1394/05/svm-%DB%8C%D8%A7-%D9%85%D8%A7%D8%B4%DB%8C%D9%86-%D8%A8%D8%B1%D8%AF%D8%A7%D8%B1-%D9%BE%D8%B4%D8%AA%DB%8C%D8%A8%D8%A7%D9%86-%D8%A8%D9%87-%D8%B2%D8%A8%D8%A7%D9%86-%D8%B3%D8%A7%D8%AF%D9%87/
14. https://shahaab-co.com/mag/edu/support-vector-machine-part1-theory/
15. http://www.farabar.net/wp-content/uploads/2016/11/@Farabar_BI-Data-Mining_Concepts.pdf
16. https://faradars.org/courses/fvdm9406s08-decision-trees-in-machine-learning
17. https://www.bigdata.ir/1396/09/%DB%8C%D8%A7%D8%AF%DA%AF%DB%8C%D8%B1%DB%8C-%D9%85%D8%A7%D8%B4%DB%8C%D9%86-%D9%85%D9%81%D8%A7%D9%87%DB%8C%D9%85-%D9%BE%D8%A7%DB%8C%D9%87-%D8%AF%D8%B1%D8%AE%D8%AA-%D8%AA%D8%B5%D9%85%DB%8C%D9%85/
18. http://www.tahlildadeh.com/ArticleDetails/Decision-tree
19. https://dataio.ir/%D8%A7%D9%84%DA%AF%D9%88%D8%B1%DB%8C%D8%AA%D9%85-%D8%AF%D8%B1%D8%AE%D8%AA-%D8%AA%D8%B5%D9%85%DB%8C%D9%85-cv8f1qghlrgw
20. https://programstore.ir/%D8%A7%D9%84%DA%AF%D9%88%D8%B1%DB%8C%D8%AA%D9%85-%DB%8C%D8%A7%D8%AF%DA%AF%DB%8C%D8%B1%DB%8C-%D9%85%D8%A7%D8%B4%DB%8C%D9%86-%D8%AF%D8%B1%D8%AE%D8%AA-%D8%AA%D8%B5%D9%85%DB%8C%D9%85/
21. https://fa.wikipedia.org/wiki/%D8%AF%D8%B1%D8%AE%D8%AA_%D8%AA%D8%B5%D9%85%DB%8C%D9%85
22. https://pmpiran.com/decision-tree/
23. https://sokanacademy.com/blog/decision-tree-%D8%AF%D8%B1%D8%AE%D8%AA-%D8%AA%D8%B5%D9%85%DB%8C%D9%85-%DA%86%DB%8C%D8%B3%D8%AA
24. https://fa.wikipedia.org/wiki/%D9%85%D8%A7%D8%B4%DB%8C%D9%86_%D8%A8%D8%B1%D8%AF%D8%A7%D8%B1_%D9%BE%D8%B4%D8%AA%DB%8C%D8%A8%D8%A7%D9%86%DB%8C
25. https://fa.wikipedia.org/wiki/%D8%B1%DA%AF%D8%B1%D8%B3%DB%8C%D9%88%D9%86_%D9%84%D8%AC%D8%B3%D8%AA%DB%8C%DA%A9

26. https://blog.faradars.org/logistic-regression/
27. http://cafetadris.com/blog/%D8%B1%DA%AF%D8%B1%D8%B3%DB%8C%D9%88%D9%86-%D9%84%D8%AC%D8%B3%D8%AA%DB%8C%DA%A9-logistic-regression/
28. https://raahbord.com/regression-logistic/
29. https://chistio.ir/%D8%B1%DA%AF%D8%B1%D8%B3%DB%8C%D9%88%D9%86-%D9%84%D8%AC%D8%B3%D8%AA%DB%8C%DA%A9-logistic-regression/
30. https://blog.faradars.org/random-forest-algorithm/
31. https://fa.wikipedia.org/wiki/%D8%AC%D9%86%DA%AF%D9%84_%D8%AA%D8%B5%D8%A7%D8%AF%D9%81%DB%8C
32. https://raahbord.com/random-forest/
33. https://mihanblockchain.com/what-is-random-forest-algorithm/
34. https://chistio.ir/%D8%A7%D9%84%DA%AF%D9%88%D8%B1%DB%8C%D8%AA%D9%85-%D8%AC%D9%86%DA%AF%D9%84-%D8%AA%D8%B5%D8%A7%D8%AF%D9%81%DB%8C-random-forest/
35. https://bigdata-ir.com/%D8%AC%D9%86%DA%AF%D9%84-%D8%AA%D8%B5%D8%A7%D8%AF%D9%81%D9%8A-%D9%8A%D8%A7-%D8%B1%D9%86%D8%AF%D9%85-%D9%81%D8%A7%D8%B1%D8%B3%D8%AA-%DA%86%D9%8A%D8%B3%D8%AA-%D9%88-%DA%86%DA%AF%D9%88%D9%86%D9%87-%D9%88/
36. https://www.statlearning.com/resources-second-edition
37. https://book.huihoo.com/introduction-to-statistical-learning/code.html
38. https://blog.faradars.org/introduction-to-machine-learning/
39. https://fa.wikipedia.org/wiki/%D9%85%D8%A7%D8%B4%DB%8C%D9%86_%D8%A8%D8%B1%D8%AF%D8%A7%D8%B1_%D9%BE%D8%B4%D8%AA%DB%8C%D8%A8%D8%A7%D9%86%DB%8C
40. https://www.matlab1.ir/%D8%A2%D8%B4%D9%86%D8%A7%DB%8C%DB%8C-%D8%A8%D8%A7-%D9%85%D8%A7%D8%B4%DB%8C%D9%86-%D8%A8%D8%B1%D8%AF%D8%A7%D8%B1-%D9%BE%D8%B4%D8%AA%DB%8C%D8%A8%D8%A7%D9%86-support-vector-machine-%DB%8C%D8%A7-svm/
41. https://fa.wikipedia.org/wiki/%D8%B1%DA%AF%D8%B1%D8%B3%DB%8C%D9%88%D9%86_%D8%AE%D8%B7%DB%8C
42. http://www.tahlildadeh.com/ArticleDetails/Decision-tree
43. https://danesh-sara.ir/%D8%A7%D9%84%DA%AF%D9%88%D8%B1%DB%8C%D8%AA%D9%85-k-%D9%86%D8%B2%D8%AF%DB%8C%DA%A9%D8%AA%D8%B1%DB%8C%D9%86-%D9%87%D9%85%D8%B3%D8%A7%DB%8C%D9%87/
44. https://fa.wikipedia.org/wiki/%D8%A7%D9%84%DA%AF%D9%88%D8%B1%DB%8C%D8%AA%D9%85_%DA%A9%DB%8C-%D9%86%D8%B2%D8%AF%DB%8C%DA%A9%E2%80%8C%D8%AA%D8%B1%DB%8C%D9%86_%D9%87%D9%85%D8%B3%D8%A7%DB%8C%D9%87
45. https://novinilya.com/%D8%A7%D9%84%DA%AF%D9%88%D8%B1%DB%8C%D8%AA%D9%85-%D8%AF%D8%B1%D8%AE%D8%AA-%D8%AA%D8%B5%D9%85%DB%8C%D9%85/
46. https://blog.faradars.org/random-forest-algorithm/
47. https://pmpiran.com/decision-tree/

48. https://dataio.ir/%D8%A7%D9%84%DA%AF%D9%88%D8%B1%DB%8C%D8%AA%D9%85-%D8%AF%D8%B1%D8%AE%D8%AA-%D8%AA%D8%B5%D9%85%DB%8C%D9%85-cv8f1qghlrgw
49. https://behsanandish.com/tag/%D9%85%D8%B2%D8%A7%DB%8C%D8%A7-%D9%88-%D9%85%D8%B9%D8%A7%DB%8C%D8%A8-%D8%AF%D8%B1%D8%AE%D8%AA-%D8%AA%D8%B5%D9%85%DB%8C%D9%85/
50. http://modir1.com/%D8%AA%D8%AD%D9%84%DB%8C%D9%84-%D8%AF%D8%A7%D8%AF%D9%87/%D8%AF%D8%A7%D8%AF%D9%87-%DA%A9%D8%A7%D9%88%DB%8C?id=339
51. http://www.tahlildadeh.com/ArticleDetails/Decision-tree
52.