**Credit Card Fraud Detection**

THE POLICE

Cem Akagündüz

Hasan Can Buruk

Göktuğ Övren

Mehmet Akif Tür

**Abstract**

In this project, we have tried to detect fraudulent transactions in a credit card transaction data. We used t-SNE method to see whether our data can be clustered, and after this process we used Decision Tree Classification, Logistic Regression, SVM and Random Forest Classification methods and compared the results afterwards. In the data-analysis part we analyzed our data which only three non-anonymized variables of 31 variables; time, amount and class. Since class is the result we are looking for, we made statistical analysis on time and amount. In the pre-processing part we standardized our non-anonymized features, we split our data into test and train sets with a ratio of 1/9 and we created a balanced dataset in other to train the algorithm to better recognize the fraudulent transactions. We removed outliers with IQR score method.

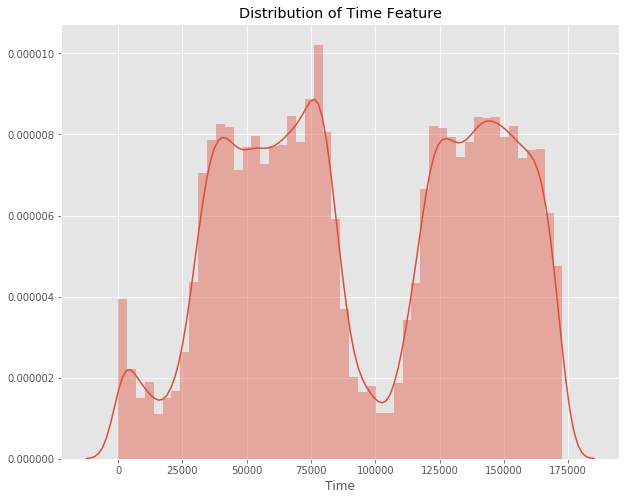
**INTRODUCTION**

The Concise Oxford Dictionary defines fraud as “criminal deception; the use of false representations to gain an unjust advantage.” Fraud is as old as humanity itself and can take an unlimited variety of different forms. However, in recent years, the development of new technologies (which have made it easier for us to communicate and helped increase our spending power) has also provided yet further ways in which criminals may commit fraud. Traditional forms of fraudulent behavior such as money laundering have become easier to perpetrate and have been joined by new kinds of fraud such as mobile telecommunications fraud and computer intrusion.

We begin by distinguishing between fraud prevention and fraud detection. Fraud prevention describes measures to stop fraud from occurring in the first place. In contrast, fraud detection involves identifying fraud as quickly as possible once it has been perpetrated. Fraud detection comes into play once fraud prevention has failed. In practice, of course fraud detection must be used continuously, as one will typically be unaware that fraud prevention has failed. [[1]](#footnote-1)

Our purpose is firstly to understand the relationship between variables in data and fraud, then, to predict fraudulences successfully using these correlations under favor of machine learning algorithms and statistical data analysis. In order to understand correlations, it is obligatory to understand data by analyising it, then, organize it to get a functional form if needed.

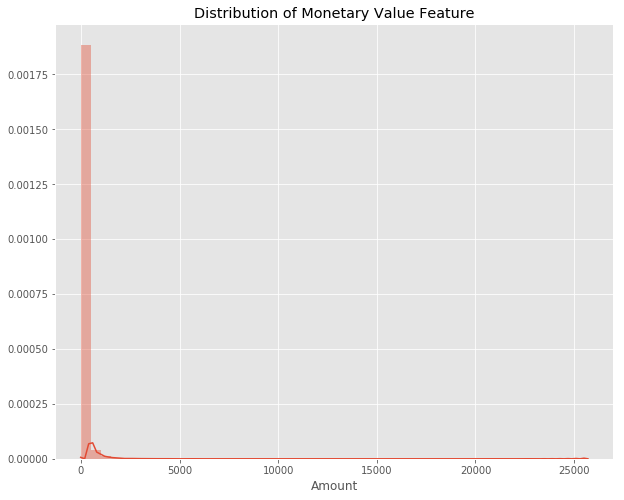
**DATA ANALYSIS**

Data set contains 31 variables, totally. A great portion of these variables are anonymous. So, focus on non-anonymized variables makes more sense firstly. The known predictors in our data set are class, time, and amount. Class is a categorical variable that determines whether transaction is fraudulent or not. It means it is not very useful to get a statistical meaning in the first step except its proportion regarding total transactions.

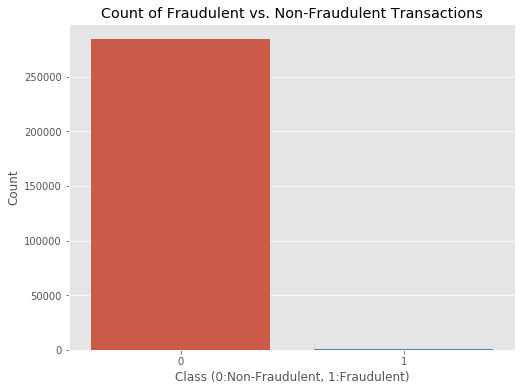
Our first focus is time value and we are going to examine how the number of transactions differs as regards intervals of increasing time. Our time value is given as seconds format which start from 0 and goes up to 172792.

The time between our two local minimum of frequency of transactions according to time is approximately ((100,000-15,000)/3600) 24 hours. So, we can assume that the volume of transactions drops significantly during the night.

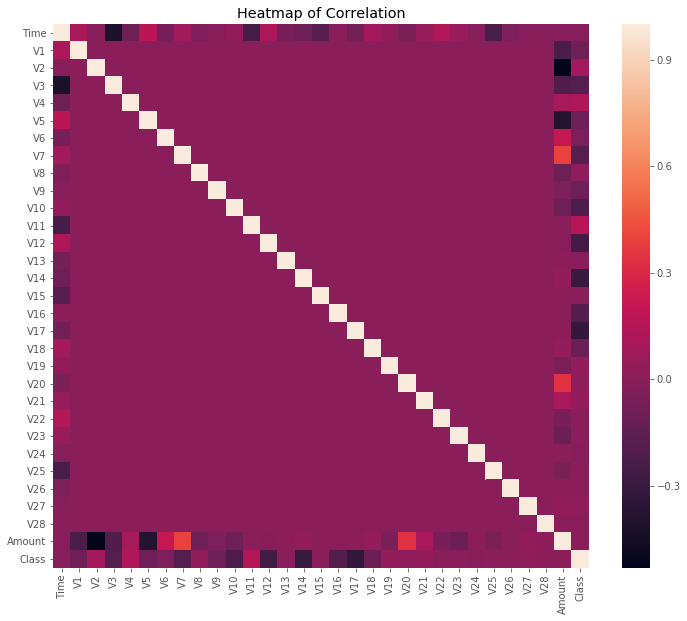
Our second focus is amount value by getting the describe of it. The mean value of our transactions is 88.350 but maximum value of all transactions is 25691.160. Therefore, we predicted by considering the mean and maximum, the graphical distribution of our amount value of all transactions is right-skewed. A great portion of the transactions are very small as against maximum amount value of transactions and it also means the number of high value transactions are very low.



Our third focus is a categorical value, class. Class means whether this transaction is fraudulent or not. If class equals to one, this transaction is fraudulent and if class equals zero, this transaction is not fraudulent. As we can expect, a very great ratio of transactions is non-fraudulent. As you can see in the following graph, there were 284315 non-fraudulent transactions (99.827%) and 492 fraudulent transactions (0.173%).



Finally, we come to the most interesting part. In this part, we will check “Is there any important correlations between anonymous variables and specifically our class variable?” In order to do that, firstly, we calculated correlations then, formed its visual graph using heatmap. As you can see in the following graph, some variables seem to be correlated with our class variables. Still, there is no very significant correlation for this big data. Most probably, this is because the great class imbalance deflects the significance of some correlations regarding class variable.



**Data Preparation**

**Standardizing Time and Amount columns**

Before the analysis we are going to do, we should take into account that our anonymized features have been scaled and centered around zero; on the other hand, “time” and “amount” features have not. Not scaling these features result in some of the machine learning algorithms that give weights to features (logistic regression) or rely on a distance measure (KNN) performing much worse. To prevent this, we standardized both the time and amount column. Luckily, there are no missing values and we, therefore, do not need to worry about missing value imputation.

**Copying standardized dataset before other processes**

In the following processes, we are going to remove more than 99% of our data to get better results for curtain machine learning algorithms. However, we are also going to implement Artificial Neural Network (ANN) in our model, and ANN is a deep learning method, which gives better results with more data. Therefore, we copy our data before other processes to use in machine learning methods.

**Getting Train and Test Dataset**

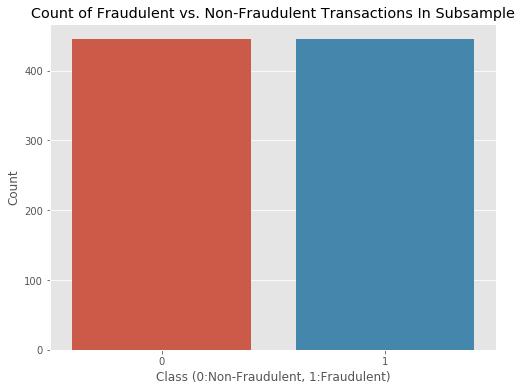
We manually separated our data into train and test datasets using NumPy’s random.rand, with the ratio of 1/9 (test/train). Now we have 256,334 data in train dataset and 28,473 data in test dataset. There are 449 fraudulent transactions out of 256,334 transactions, which gives the percentage of 0.173.

*Note: We arrange* np.random.seed(0) to get consistent result.

**Creating Balanced Dataset**

As it can be seen from the analysis made, only 0.173% of transactions are fraudulent meaning frauded transactions is seen extremely rarely. Using the original dataset in the model is not a smart idea because it is very likely that our model will predict the transaction as not fraudulent which would give us a good accuracy anyways. However, what we want is not predicting non-fraudulent transactions but predicting fraudulent transactions with higher accuracy. In other words, decreasing False Negative predictions as much as possible.

For this purpose, we created a training dataset with a balanced class distribution that will force the algorithms to detect fraudulent transactions as such to achieve high performance. This method is called random under-sampling. After random under sampling we chose a dataset that has same amount of fraudulent transactions with the non-fraudulent transactions.



**Removing the outliers**

Before removing outliers, we look how outliers located thanks to the boxplot plotting in Seaborn library. However, looking all the features might be fallacious, so we plotted the features which has high negative and high positive correlation with the class column.

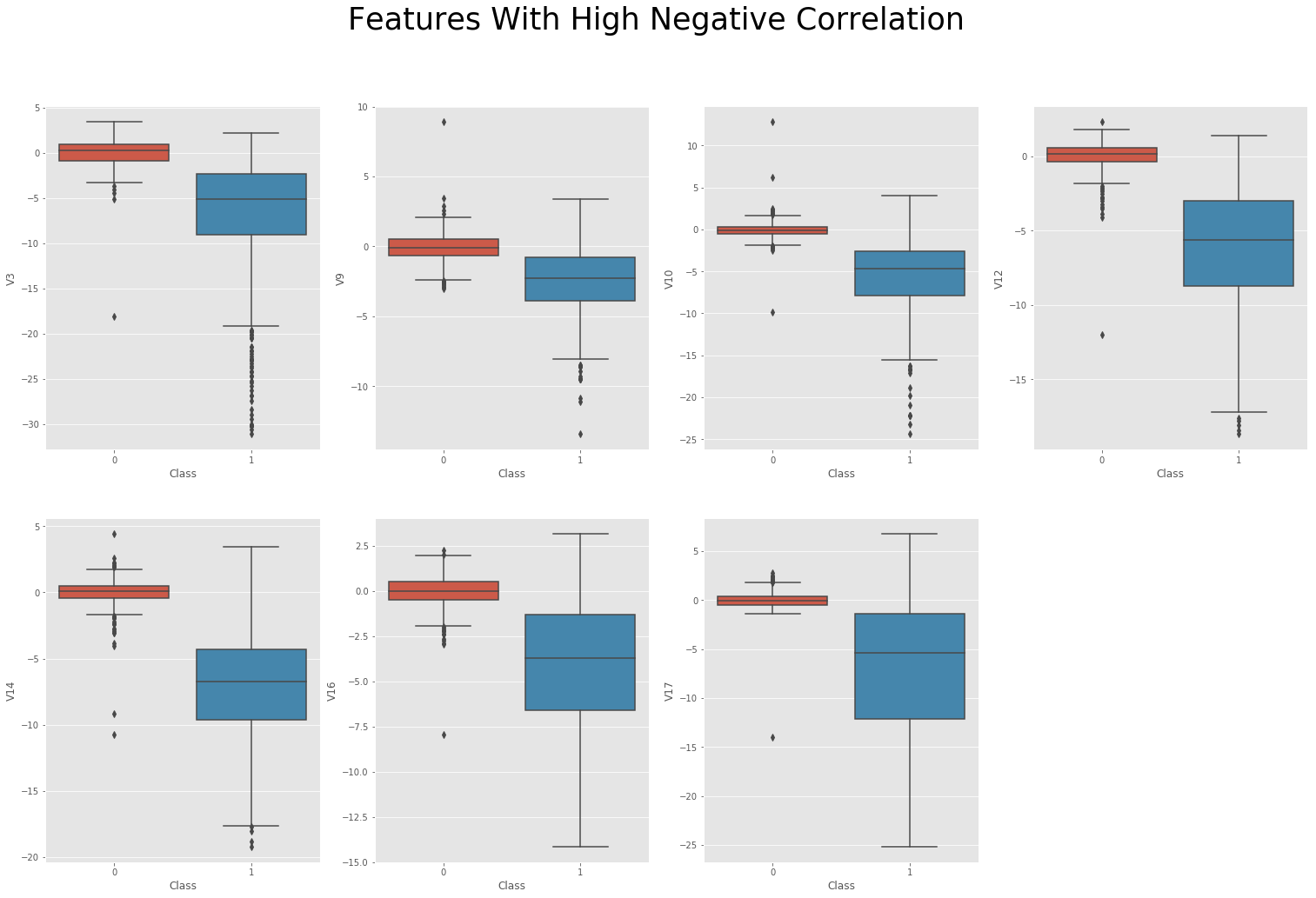
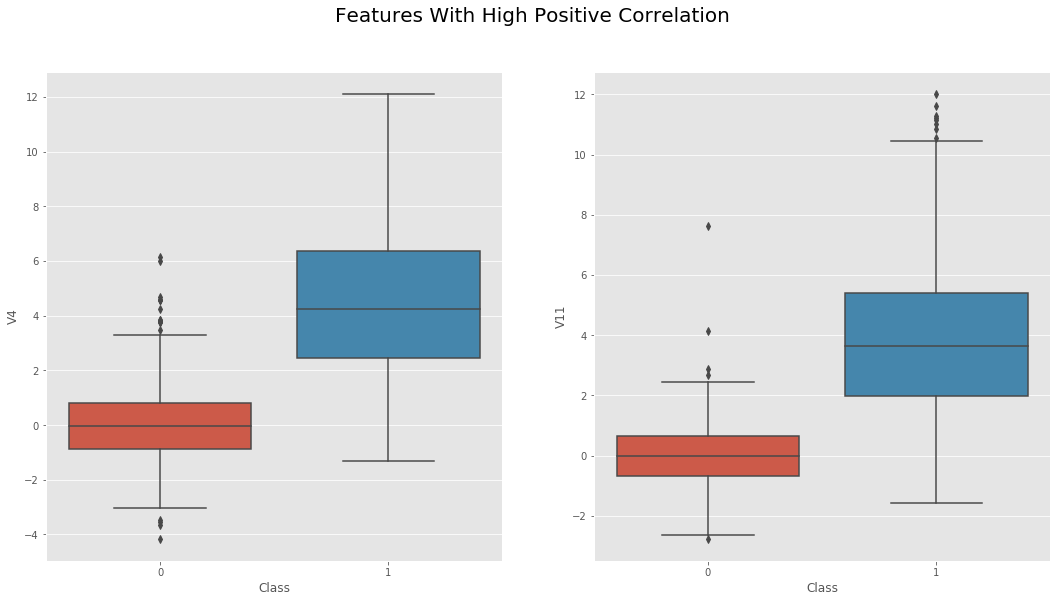
Features with high negative correlation Features with high positive correlation (corr < -0.5) (corr > 0.5)

|  |  |  |
| --- | --- | --- |
| **V3** |  | -0.569 |
| **V9** |  | -0.545 |
| **V10** |  | -0.623 |
| **V12** |  | -0.681 |
| **V14** |  | -0.745 |
| **V16** |  | -0.591 |
| **V17** |  | -0.560 |

|  | **Class** |
| --- | --- |
| **V4** | 0.708 |
| **V11** | 0.686 |
| **Class** | 1.000 |

|  | **Class** |
| --- | --- |
| **V4** | 0.708 |
| **V11** | 0.686 |
| **Class** | 1.000 |

**Now let’s look at the boxplots the features that has high negative correlation with the class column:**

  
**And look for high positives:**

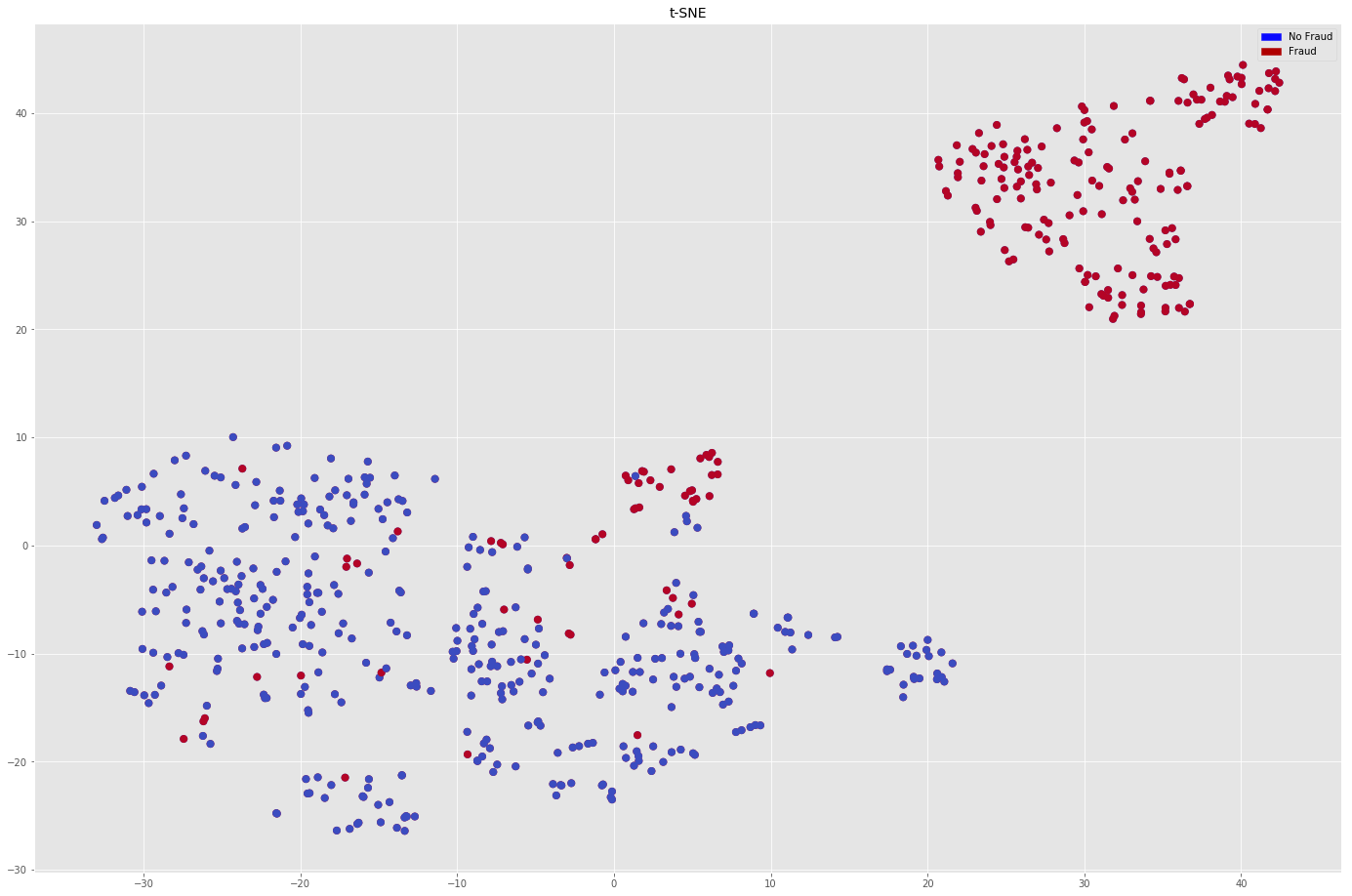
As it can be seen from the boxplots, we have big number of outliers.

Therefore, to remove the outliers, we used IQR score method. However, we manipulated the classical method little bit to not lose our data, which is already decreased to 809 transactions. In classical method the coefficient is chosen as 1.5, but in our data, we chose 2.5, which result in to remove “extreme” outliers. With the classical method, there remains only 486 transactions, however, we have 633 transactions thanks to the manipulation.

**MACHINE LEARNING PROCESSING**

In third part, we studied machine learning processing. We firstly used clustering algorithm, which is t-SNE algorithm. By means of this algorithm, we reduced our features to two, also learned how hard to handle our problem.

We looked at graph and grabed that new features are almost distinct from each other and reached our classification algorithms will have high performance.

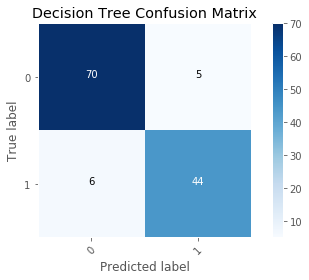


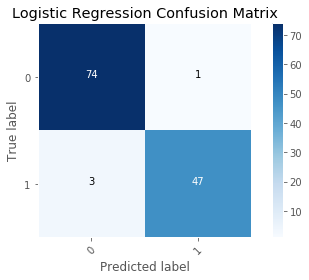
Then we passed to classification algorithms, but beforely we created some functions for using ongoing processes which are accuracy matrix, f1 and pickle functions.

Now, we are at the classification process. This process has 4 step. In the first process, we used basic classification algorithms, which are decision trees, various support vector machines and logistic regressions.

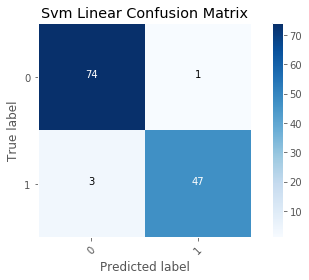
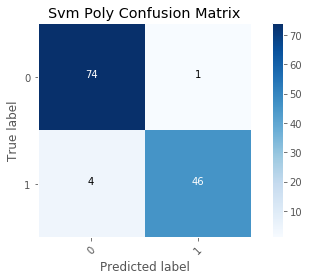
Below you see accuracy scores and f1 scores:

Decision Trees Logistic Regression

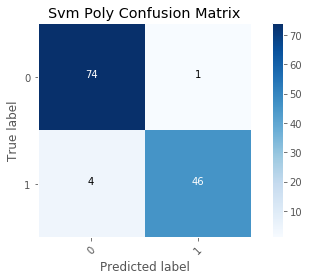




SVM Kernel SVM Kernel - Poly

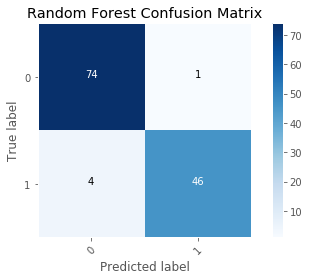


SVM Kernel – Sigmoid



Then, in the second step, we created pure ensemble techniques, which are random forest and ANN(Artificial Neural Network) . We also visualized Random Forest Algorithm since it has both its visualization advantages and high percentage accuracy.

Below you see accuracy scores and f1 scores of Rf Algorithm

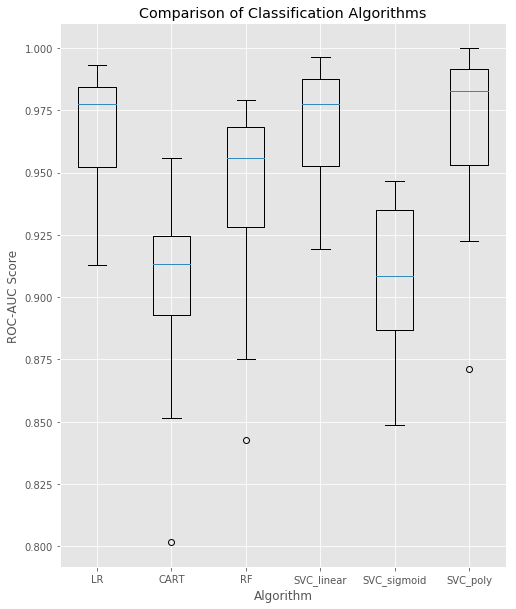


Also you see random forest visualizations:

We also developed an ANN, but we won’t give a detail in our project but people who want to reach project details can reach results by looking at project file.

In 3rd process, we developed our all algorithms with model selection techniques. We used k-cross fold validation for this progress, and significantly increased our algorithm’s effectivity. Also 2 since support vector machines’ algorithms performed poorly when compared to other classification algorithms we used grid search for increasing its accuracy performance by selecting best parameters for it.

You see the results with ROC-AUC:



We also added SVM with Grid-Search:

In this algorithm, we obtain :

En iyi tahmin oranı: % 94.74

En iyi parametreler: {'C': 0.1, 'kernel': 'linear'}

In 4th process, we created basic hybrid machine learning algorithm which is majority voting classifier. This algorithm classifies by looking decisions of the other algorithm’s performances which are basic and pure ensemble classification algorithms in our cases. We couldn’t find proper library therefore we wrote a function to run it.

You see the results of Hybrid ML Model:

It is the result of Cnf\_2 model

f1\_score = 0.9202453987730062

recall = 0.9868421052631579

precision = 0.8620689655172413

It is the result of Cnf\_3 model

**Conclusion**

Since over 99% of our transactions are non-fraudulent, we would have got highly “accurate” results if we write an algorithm that predicts that all the transactions are non-fraudulent. However, what we want is not to get high accuracy but lowering the False-Negative errors as much as possible; in other words, we aimed to get high F1 score.

In first step, logistic regression is more succesfull.

In second step, random forest showed a good performance but remained behind logistic regression with a little differences.

In third step, logistic regression with K-CV and SVM with K-CV showed good performances and random forest with k-cv remained behind just a little bit.

In fourth step, we used majority classifier but it remained behind them.

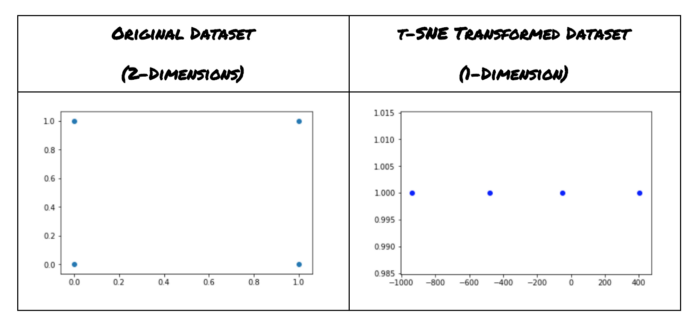
Normally, LR with K-CV or SVM with K-CV should be selected since RF with K-CV remained a little bit behind.

But RF algorithms can be visualized better and showed good cause and reason structure, we think that we should use it for a business strategy.

**Methods**

**t-SNE**

t-SNE, (t-distributed stochastic neighbor embedding) is a clustering technique that uses a simpler to understand, repel/attract approach to map points from high-dimensional space into lower dimensional space. The focus of many clustering algorithms is to identify similarity in a high-dimensional dataset in such a way that dimensionality can be reduced. The tSNE algorithm works to preserve the linear spatial relationships in the higher space.

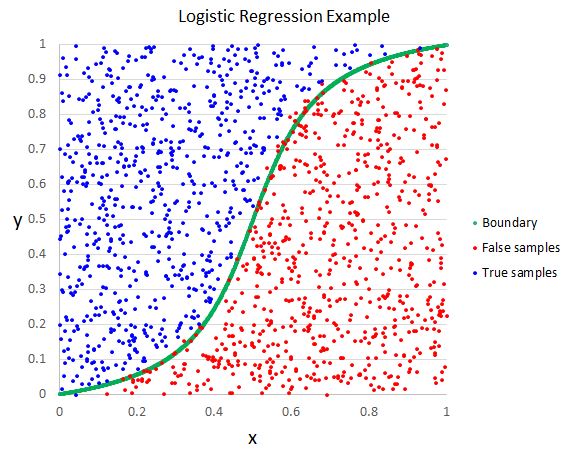


**Support Vector Machine**

[](https://www.analyticsvidhya.com/wp-content/uploads/2015/10/SVM_1.png)“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well (look at the below snapshot).

Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

**Logistic Regression**

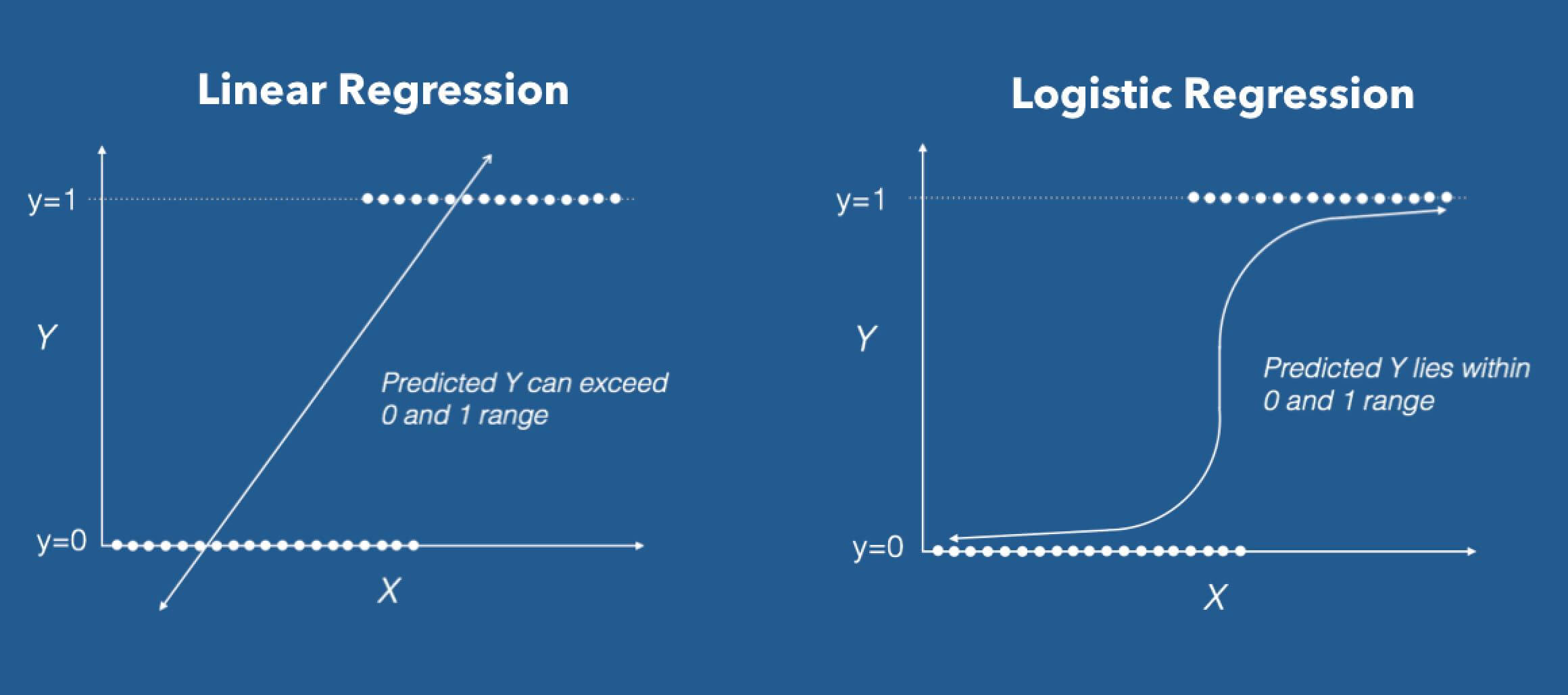
Despite its name, it is an algorithm for classification problems. Logistic Regression is the go-to method for binary classification. It’s outcome is discrete binary. To say it in simpler words, it’s outcome is either one thing or another.

**How it works**

Logistic Regression measures the relationship between the dependent variable (our label, what we want to predict) and the one or more independent variables (our features), by estimating probabilities using it’s underlying logistic function.

These probabilities must then be transformed into binary values in order to actually make a prediction. This is the task of the logistic function, also called the sigmoid function. The Sigmoid-Function is an S-shaped curve that can take any real-valued number and map it into a value between the range of 0 and 1, but never exactly at those limits. This values between 0 and 1 will then be transformed into either 0 or 1 using a threshold classifier.

 Difference between linear and logistic regression is logistic regression gives you a discrete outcome but linear regression gives a continuous outcome.



# **The Random Forest Classifier**

Random forest includes large number of decision trees all of which makes a prediction and the most predicted result becomes model’s prediction.

Random forest model works efficiently because of the wisdom of crowds.

***A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models.***

**References**

*Statistical Fraud Detection: A Review*, Richard J. Bolton and David J. Hand <https://projecteuclid.org/download/pdf_1/euclid.ss/1042727940>

<https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/>

<https://machinelearning-blog.com/2018/04/23/logistic-regression-101/>

<https://towardsdatascience.com/understanding-random-forest-58381e0602d2>

<https://towardsdatascience.com/tsne-clustering-for-map-data-507ba4e62b20>

1. *Statistical Fraud Detection: A Review*, Richard J. Bolton and David J. Hand <https://projecteuclid.org/download/pdf_1/euclid.ss/1042727940> [↑](#footnote-ref-1)