Case Study: Credit Worthiness - REPORT

* S. Vishakan, SSN College of Engineering, ML Using Python Online Course Batch – 1

## Problem Statement:

The provided Excel sheet CreditWorthiness.xlsx was used as input in this problem. The objective was to build a machine learning model that could help Cautious PLC to predict defaults on payments by their customers. It must be made in a manner that gives them early warnings on defaulters and to take credit decisions pertaining to an individual before granting loans.

## Analysis of Data:

The given data contained a mix of categorical and numerical variables. There seemed to be no NULL values. On analysis using correlation, it was found that **Camt** and **Cdur**, **NumCred** and **Chist** were highly correlated. There were no significant relationships between other variables. A few outliers seem to exist pertaining to the **age** column, but no action was taken as it was not too drastic, as revealed by the boxplot. The **Camt** column was also not normally distributed, so a log conversion was done to ensure normality. The **Age** column also showed a slight divergence from the normal plot, but no actions were taken.

It was decided that **Cbal**, **Chist**, **Sbal**, **Oparties** & **creditScore** were to be transformed into numerical data using logical encoding of numerical values, as indicated in the script file, for better understanding and working of the model.

It was also decided to discard the following columns: **foreign**, **telephone**, **Rdur**, **MSG** & **Prop**.

The column **foreign** had not much variation in its values, with only 37 records having different values out of 1000.

The column **telephone** was not relevant from a domain perspective.

The column **Rdur** could be removed as it did not affect the accuracy of the model.

The column **MSG** was rendered irrelevant due to the column **Ndepend** which has correlation to it from a domain perspective.

The column **Prop** was also removed as it did not affect the model’s accuracy significantly.

## Solution Approach:

* + Logistic Regression
  + Random Forest Classifier
  + Support Vector Machine
  + Multi Layer Perceptron

## Logistic Regression:

Logistic Regression gave an accuracy score of **73%** and **81** samples were misclassified. Therefore it was decided to implement another model which could better work with categorical as well as numerical data.

## Random Forest Classifier:

By understanding, it felt that the use of Random Forest classifier made the most sense, as it is an apt model in this scenario. Of all the tested models, Random Forest performed best. The number of estimators and maximum depth were iterated over a RandomizedSearchCV() to find the best parameters.

This particular model yielded **76.3%** accuracy with **71** misclassified samples.

## Support Vector Machine:

Since SVMs can also be used as a classification technique, it was decided to employ this model as well. Standard scaling of variables were performed.

Linear SVC gave an accuracy of **71.6%** on the test data with **85** misclassified samples.

SVC gave an accuracy of **71%** on the test data, with **87** misclassified samples.

## Multi-Layer Perceptron:

I decided to use a neural network approach as well, as it can also be used for classification purposes. Despite extensive testing with the parameters like activation, hidden layers, iterations and solver, it did not yield appreciable results.

The model only gave an accuracy of **67%** on the test data with **97** misclassified samples.

# Conclusion and Inferences:

Therefore, by the above presentation, it was decided that the **Random Forest** model is the most apt for the above problem statement. Other alternative approaches were rejected due to lower accuracies, despite the complexity of the model. MLP could not be improved despite exhaustive testing. SVM on the other hand could be used as well, but Random Forest has a slight edge over SVM with a higher accuracy score. Logistic Regression was too simple and could not be improved upon; therefore it was dropped as it had a comparatively low accuracy.

There were 5 attributes removed in total, leaving behind 15 attributes. Removal of any of those attributes cause a significant loss of accuracy in the model. Therefore it was decided to leave them behind.

PCA was not used here as the amount of data/no. of attributes weren’t that high to necessitate its usage.