Skill-based Assignment Report

**Q1) Most important determinants of credit risk application: -**

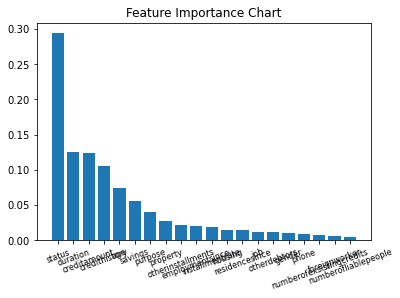
**a) Present convincing arguments based on insights from the dataset**

Based on observation of the Customer dataset alone, I believe features like Credit History, Age, Savings, Employment Since will be vital for effectively train a model and to predict if a credit card applicant is worthy or not.

Besides, studying the dataset, I have also ranked the independent variables (or selected features) based on *Feature Importance* technique, which provided a further insight into the dataset and helped me finalize the relevant features for training our model.

The whole set of predictor variables were fed into a decision tree model and based on the *feature\_importances property, a feature ranking was shown.*

The following features had better score in comparison to the other ones - *Status, Duration, Credit Amount, Credit History, Savings, Purpose.*



Based on the feature importance ranking, status of the credit applicant is the one of most significant predictor variables in this case. Just like other columns it’s a categorical data, that reveals if the credit applicant holds a checking account, and if yes if it’s above 200 Euro.

This feature along with Duration, Credit Amount, Credit History of the applicants could be the most important feature vectors in our case.

**(b) What are the limitations of your argument?**

Since we don’t have much data sample (988 rows) for training models, potentially the trained models can’t be deployed into production or in a live application. Techniques like Cross-validation might have to considered as well, rather than using the conventional Test Train Split method, as we are working on a limited data sample.

**Q2) The management team wants to use your analysis to propose a system. Can you give an estimate of how accurate your algorithm is likely to be?**

Since we are attempting to predict a categorical result (i.e., if an applicant is credit worthy or not), we will be implementing a supervised classification algorithm.

In this case, I have decided to use the following algorithms and do a comparison between these techniques

* *k-nearest Neighbours*

*As it’s one of the simplest, and faster classification algorithms that requires less training.*

* *Linear Support Vector Machine*

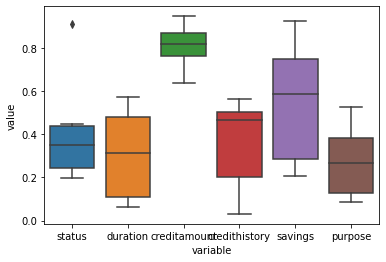
*Though it’s usually used in high dimension problem scenarios, I wanted to analyse how well it could fare in comparison to other techniques. SVM models also tend to perform better on sparse data as well.*

* *Random Decision Forest*

*Decision trees might be the default approach for a classification problem like this. Implementing random decision forest, as it’s an ensemble of decision trees and it’s likely to be less prone to overfitting and more accurate.*

The samples were found to be unique enough and there weren’t any NULL values in the records that had to be cleansed. However, since most of the features and the target variable (the credit worthy column) were textual categories, I had to use LabelEncoders to transform the data into integers – which were later used for training our models.

The data outliers in features were found by creating a chart using Box-plot, and these were removed from the DataFrame using Z-score function.



Once we performed pre-processing and cleaning, the dataframe will be fed into our model. Based on my experiments, Random Forest had a better prediction score (~77% at times) in comparison to the other two classification algorithms.

**Q3: What according to you are some of the benefits and risks of adopting such analytics in assessing credit applications.**

Introducing machine learning into such applications could reduce human intervention and also, as it could improve and evolve after deploying, predictive analytics are well suited for such applications. Human made errors could also be bit less, as a trained model will be evaluating such applications.

But at the same time, such applications will also require large datasets in order to be properly trained and tasks like understanding business requirements, shortlisting parameters, hyper parameter tuning, pre-processing and transforming data, training model and testing could all be bit time consuming.