**Predictive Analysis – Credit Card approval**

Akila Selvaraj

Masters of Data Science,

Bellevue University

DSC 630 T302

September 09,2022

Predictive Analysis – Credit Card approval

**Project Proposal:**

Credit risk as the board in banks basically centers around deciding the probability of a customer's default or credit decay and how expensive it will end up being assumed it happens. Banks receive lot of credit applications every year. These applications had to be manually analyzed and decisions will be taken based on various factors. If the customer of these applications has low income, high loan balances or bad credit report, their applications will get rejected. It is a manual and time-consuming work to analyze the applications and take decisions on them. With the power of machine learning, this process can be automated. In this project, I am going to build a machine learning model to make this task automated with the best accuracy possible using python. I am planning to build a model that can predict if an individual’s application for a credit card can be accepted or not.

**Data Selection:**

For this project, the datasets I am planning to use are Credit Card applications dataset and credit record datasets. Attributes in the dataset are gender, flags for if the customers own car, realty, family details, way of living, income, type of income, and education. Credit record dataset has attributes ID which can be used to join with application dataset, and the months balance. Application dataset has a combination of both numerical as well as non-numerical elements, specifically data that are of float64, int64 and object types, and this will be fixed in preprocessing step. From the features that have numerical values, we can get statistical information like min, max, mean. It also contains values from various reaches, in addition to that it contains a few missing values as well.

**Preprocessing the data:**

Preprocessing the dataset is required so that the model I pick can meet the expectations. Before building this credit card approval predictor model, I will tackle some of the most widely-known pre-processing steps such as scaling, label encoding, and missing value imputation.  Model may miss out useful information needed for training if we ignore the missing values in the dataset. These missing values will be analyzed and imputed to avoid this issue. Python library scikit-learn expects the data to be in numerical format. With the use of label encoding, I will convert all non-numeric columns into numeric format which also speeds up the computation. To ensure the convergence of cost function minimization in machine learning models, all feature variables will be scaled using [feature scaling techniques.](https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/) Scaling technique should be applied before implementing the model. Scaling technique I am planning to apply on this dataset is MinMaxScaler method. This converts features by rescaling them to a given range. This scalar technique scales and deciphers each feature separately to such an extent that it is in the given range on the training set.

**Machine learning Model:**

The task of predicting whether a credit card application will be approved or rejected based on values of feature variables is essentially a[supervised machine learning classification task](https://dl.acm.org/doi/10.5555/1566770.1566773).For this project, I can build the model from one of these five different types of classification models namely Logistic Regression, Decision Tree, Gradient Boost, XGBoost, and K-Nearest Neighbors (KNN). These are the most popular models used for solving classification problems. All these models can be conveniently implemented using python’s scikit-learn package except for the XGBoost model, which can be implemented using the [XGBoost package](https://xgboost.readthedocs.io/en/stable/" \t "_blank). A decent machine learning model accurately predicts the status of the applications with respect to the statistics. To begin with, the features that affect the credit card approval decision process will be correlated with each other. Because of this correlation, I will take advantage of the fact that generalized linear models perform well in these cases. So, I will begin machine learning modeling with a Logistic Regression model.

**Results evaluation:**

Using test dataset, performance of the model will be calculated. One of the metrics that will be used to evaluate the results is classification accuracy, which is defined as the fraction of times model prediction matches the value of the target variable. For a detailed evaluation of the model, confusion matrix will also be created. The values in the diagonal of the confusion matrix will denote the fraction of correct rejection or correct approval predictions by classification model.

**Improving model performance:**

After evaluating the model, it is better to analyze if the performance of the model can further be improved. To improve the model’s ability to make better predictions, grid search of the model parameters can be performed. GridSearchCV is used in performing hyperparameter tuning in order to determine the optimal values for a given model. It finds the optimal value at which the model gives the best possible accuracy. Grid search will be implemented using over tol and max\_iter of scikit learn model\_selection library. Tol tells the scikit library to stop searching for a minimum (or maximum) once some tolerance is accomplished, and max\_iter is maximum number of iterations taken for the solvers to converge.

# **Week 6**

**Will the data answer the question?**

The data present in application and credit score dataset has the important features that will be considered during the credit card approval process. Especially employment and income related details present in the data are some of the factors which would help in the prediction of credit card approval. So, to answer the question of if the credit card application can be approved or not could be answered with the data present in the files chosen.

**Data Visualization**

A visual analysis will be conducted on the dataset to have an idea of the possible relationships between the variables and observe any visible effect of each factor on whether an application is ultimately approved or not. To explain the data, visualizations that might be helpful are frequency distribution of variables like income type, family status, housing type, age, annual income, and working years. Another visualization that might be useful will be pair plot which will help to understand the relationship between the variables.

**Adjusting the data:**

Chosen data seems good for the prediction of credit card approval. However, some data transformation needs to be done on some of the variables for the model to perform better. In the data, dependent variable which says if the credit card is approved or not is not in the binary format. As this is a classification problem, variable we are going to predict should be in binary format. So, this variable will be transformed to binary values 0 and 1. There are some more attributes in the dataset like code gender, flag own car, flag own realty which are flag fields but the values are not in integer datatype. In data preparation step, these fields will be converted to binary format for better prediction.

**Model Adjustment:**

A common evaluation technique for classification models is to use a confusion matrix which gives false positive predictions and false negative predictions.

Based on the model performance, few adjustments can be done in the model. One of the ways that adjustments will be done is by modifying the independent variables we used to build the model.

Another way that the model we built can be tuned is by adjusting the model hyperparameters. Hyperparameters are the input parameters that are configured before the model starts the learning process. These are parameters whose values control the learning process and determine the values of model parameters that a learning algorithm ends up learning.

Logistic regression does not really have any critical hyperparameters to tune. However, the main hyperparameters we may tune in logistic regression are solver, penalty, and regularization strength. For KNN algorithm, the most important hyperparameter is the number of neighbors whereas the most important parameter in random forest is the number of random features to sample at each split point which is max\_features. Similarly, for gradient boosting algorithm, the parameters that can be tuned are learning rate and number of trees in the model.

**Expectations**

The exact judgement of person to be approved for credit cards allows the organizations to minimize losses and the same time make suitable credit arrangements as per requirement. The data of credit card defaults is always unbalanced, since few clients default in real world, which brings challenges to default model construction. To solve the problem of unbalanced data, we further build corresponding weighted models so that it can improve the prediction accuracy of default class with slightly higher prediction error of non-default class.