# Project Report

# Fidelity International - Data Analytics Training Group C

CIDAP 22-04-25

Name: Scottie Yu
Email: scottieyu@gmail.com

GitHub URL: https://github.com/scottieyukc/UCDPA_SCOTTIEYU

## Abstract

Credit cards are being used in our daily life. Understanding how machine learning can be used to predict if an applicant is considered a good or bad customer is an interesting project. This project is about predicting credit card approval based on various different features from an applicant, and thus can be used as a smart approval process.

The project applies data analytics in data loading, data cleaning, data manipulation, data merging, machine learning and performs hyperparameter tuning to increase accuracy of the model.

## Introduction (Explain why you chose this project use case)

Commercial banks receive a lot of applications for credit cards. These credit card approval processes typically use credit scores to determine whether or not to approve the application. It's a widely used risk management process in credit card approval. It uses applicant's personal information (gender, age, income, household income etc) and data provided by credit card applicants to estimate if applicants will or will not default on their credit balance (equivalent of loan). From the machine learning technique learned from UCD courses, we can build a machine learning model to predict the approval or rejection for credit card applicants.

Using learnings from the UCD Data Analytics Training course and other online training (data camp) allowed the processing of data using Anaconda/Python. Python has an extensive library of add-on features which makes data processing and manipulation of large data sets much easier. Also python libraries include charting tools like matplotlib which helps to produce meaningful data visualisation of data analytics/ results.

## Dataset (Provide a description of your dataset and source. Also justify why you chose this source)

The project used two data sources. application_record.csv and credit_record.csv that can be downloaded from https://www.kaggle.com/datasets/rikdifos/credit-card-approval-prediction

Application record contains applicant's personal information, we can use these as features for predicting the result. Credit records, on the other hand, provide information on these applicants' repayment history. These two CSV files are connected using 'ID' column.

One of the main reasons why I choose this dataset is because this contains all the necessary data for the analysis. The update frequency is annually, although less frequent but is sufficient to train the logic. The usability score from kaggle for this data set is 10 meaning it's easy to understand and includes essential metadata. This data set are constantly being downloaded on a daily basis, with a total of 34k download, is a CSV file downloaded from kaggle.

The project also required to demonstrate our understanding to import data via API, I have put some code towards the end of the project to import stock price data from alphavantage to get stock price from Vodafone every hour.

## Implementation Process (Describe your entire process in detail)

This project was implemented using python and the following libraries

1. Pandas as pd - pandas dataframe to be used to manipulate large dataset
2. Matplotlib.pyplot as plt
3. Sklearn for machine learning
4. SVM.SVC
5. MinMaxScaler
6. SMOTE
7. Classification report
8. GridSearchCV

First step - loading the data

The two csv files are stored locally and read into respective dataframe using pandas read_csv functions

Second step - explore and clean the data

For application records, it was observed there are duplicate records in the application data set (438557 total count vs 438510 unique ID). When I check for na/null value for all the columns, Occupation_type has 134203 records of null/NA record, this column will not contain meaningful insight towards our result, so dropping the column from our dataset. When data fields are explored in detail, a lot of the columns are not in integers, for example, Male and Female are shown as 'M' and 'F', for our project, it is changed to 1 and 0. Similar conversion applies to the flag whether the applicant owned a car or not, owned a realty/property. There are other columns containing characters/words rather than a flag, for those columns (Income type, Education type, family status and housing type), I build a simple common function to print all all possible values from a dataframe called printallvalues which takes a numpy array and print all possible values. I called this function for those columns and translated their values to integers.

For credit record, there are only 2 columns (months_balance and status) other than the ID which is used to join the application record. For status columns, C means paid off and X means no loan for the month, in our project, it has the same value as 0 (no days past due), the columns need to be converted to 0 and for the other values (1,2,3,4 and 5) they will be

converted to 1 for the result. And the column type will also be required to be updated to integer. For the column “Months_Balance” column, since this represent the month where the data are extracted and it contain time series detail, -1 means previous month. For the purpose for this analysis, this will be required to be converted into positive value by multiplying by -1 and sort it in ascending order.

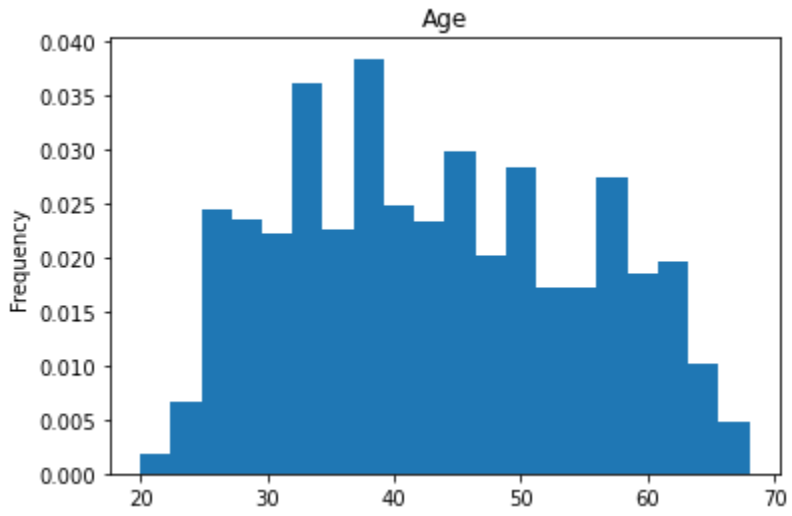
The two datasets are now ready to be merged. We split the test data with 30% and make sure the random state is set, so it can be reproducible. The next step involves scaling the data, and fixing issues with oversampling.

Since this is a binary classification project, I decided to use SVC in SVM from sklearn to perform machine learning. From the classification report, it achieved a 67% accuracy on average. Next I tried to perform hyper-parameter tuning using GridSearch but It is taking so long that I haven’t been able for the result to be generated. I commented the code out.

# Results (Include the charts and describe them)

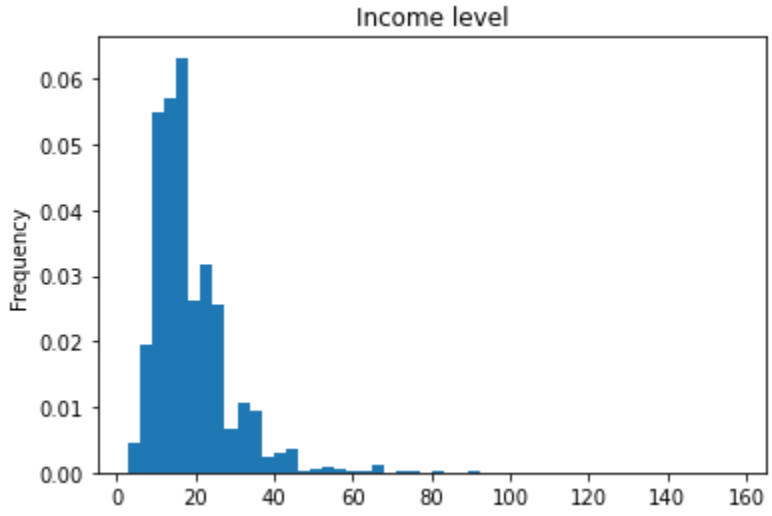
Observation 1

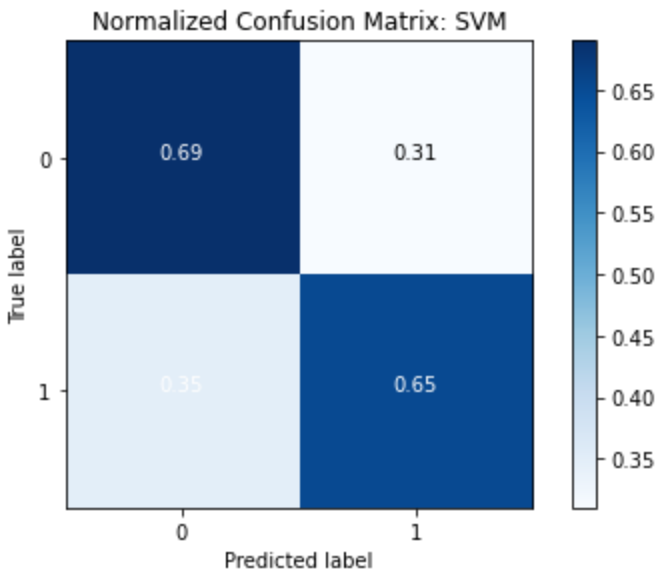
The applicant's age group are mainly between mid 20s and mid 60s, which are the majority of the workforce, there isn't a skew in particular age groups.




Observation 2

Majority of applicant has income level between 15 to 25k






## Insights (Point out at least 5 insights in bullet points)

- Credit card applicants in our datasets are mainly between mid 20s to mid 60s
- Majority of applicant in our datasets has income between 15 to 25k
- 38% of the applicant in our datasets owned a car
- 33% of the applicant in our datasets are male
- 67% of the applicant in our datasets own a property

## References (Include any references if required)