Predicting Credit Card Approval of Customers using Machine Learning Algorithms

CIND 820 – Big Data Analytics Project

Supervisor: Dr. Ashok Bhowmick
Date of Submission: March 28, 2022

Prabhkiran Kang
Student Number: 501068149

Ryerson University

Table of Contents

Abstract

Banking industry contains large volume of data on customer's information and receives a large number of applications on the daily basis for a credit card request. As the number of applications to obtain a credit card increases, it becomes hectic to approve a credit card request through a manual process as it takes more time and effort. It is also easy to make errors with a manual process. Thus, it is important to automate a process to increase the efficiency and decrease the response time. By automating a process, banks would be able to classify and divide the applicants into categories of 'Good Client' and 'Bad Client'. This way bank can decide whom to approve for a credit card and whom to reject. This also lowers down the risk of any future credit defaulters, saving financial institutions lot of money. Process of making a decision can be automated using machine learning algorithms.

In this project my goal is to use the historical data and machine learning algorithms to predict if a customer's credit card application would be accepted or rejected. I would be applying six different classification algorithms to the dataset to find out which model gives the most accurate prediction results. The theme of my choice is Classification. I would be using a dataset from Kaggle Inc. website which contains two csv files.

GitHub Link: https://github.com/prabhkang1/CIND820-2022

Introduction

In current times, all aspects of daily life are transitioning to digital world which includes cashless transaction activities. The rise of internet has increased the usage of credit cards. Credit cards have become one of the most popular modes of payment for electronic transactions. However, as the number of credit card users are increasing exponentially, so are the credit card frauds and defaulters. Financial Institutions evaluate credit risk based on a customer's credit history. This historical information is analysed to avoid any financial losses to the institution.

The correct assessment for credit card approval is very important for financial institutions who provide credit card to the customers. Along with this, an automatic process is required to fasten the approval or rejection decision of the banks. A wide range of machine learning techniques have been developed to solve credit card related problems.

This capstone project would inspect the dataset taken from Kaggle website which merges the personal information data from the customer and the personal behavior information data. The objective of this project is to identify the most efficient and best performing models that can be used to predict the approval of credit cards based on the attributes of the credit card application. The focus will be on evaluating and comparing machine learning classification models such as Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machines, Decision Tree, Random Forest and XGBoost classifier. The performance of each of these classification models would be evaluated based on several performance evaluation measures such as Confusion Matrix, Precision, Recall, F-1 Score, Accuracy and AUC & ROC Score.

Literature Review

In the past, work has been done on similar research problem and dataset, various machine learning models have been proposed to determine and evaluate the credit scoring criteria. In this project, I collected and analyzed number of research papers published. Authors of these research papers applied various approaches to their research problems. They have applied different machine learning algorithms and compared the accuracy of the models to identify the most effective one.

K.S. Naik (2021) in the research paper builds a credit scoring model to forecast credit defaults for unsecured lending (credit cards), by employing machine learning algorithms. They have applied Synthetic Minority Oversampling Technique (SMOTE), to stabilize the imbalanced data which could cause a challenge in their predictive models. They have applied 7 different classifiers including Logistic Regression, SVM, KNN, Decision Tree, Random Forest, XGBoost and LGBM to the processed data set, they found out that Light Gradient Boosting Machine (LGBM) classifier model is efficient to manage larger data volumes.

Ji-Hui MUN, Sang Woo JUNG (2021) analyzed and predicted the delinquency and delinquency periods of credit loans according to gender, own car, property, number of children, education level, marital status and employment status. They have applied the Linear Regression analysis and enhanced decision tree algorithm in this paper. Their research predicted that the Boosted Decision Tree Algorithm made more accurate prediction.

D.Jayanthi (2018) analysed credit card approval data set from UCI machine learning repository. This is a smaller data set than the data set I have selected for this project. They have proposed a credit scoring model of consumer loans based on various analytical models. They have applied and compared Logistic regression and Classification and Regression tree models. Their research concluded with CART model to be more effective than Logistic regression model.

Siddhi Bansal, Tushar Punjabi (2021) has compared different Supervised Machine learning models in their research paper in order to predict how likely a credit card request would be approved on the basis of the parameters like Precision, Recall, Time, Accuracy, F1-Score. The result of their research indicated that the Random Forest Classifier is the best-suited model according to the F1-Score.

Arokiaraj Christian St Hubert, R. Vimalesh, M. Ranjith, S. Aravind Raj (2020) in their work has attempted to improve the available technology using decision tree algorithm, K Nearest Neighbor algorithm and Logistic Regression algorithm. Authors performed data collection, data cleaning, data analysis and visualization, and data splitting tasks before applying machine learning models to the processed dataset. They processed the trained and tested data through the above-mentioned algorithms to get the best accuracy result. They concluded that both Decision Tree and KNN algorithms provided good results after continuous training of different sets of collected data.

Md. Golam Kibria and Mehmet Sevkli (2021) have built a deep learning model which could support the credit card approval decision. Then, they compared the performance of their model with other two traditional machine learning algorithms, Logistic Regression, and Support Vector Machine. They have pre-processed and analysed the data and applied grid search technique to find the best parameters. Their result show that the overall performance of the deep learning model was slightly better than the other two models.

Therefore, after reading the papers, I observe that solving this problem has many approaches, and every model applied would lead to prediction with different accuracies.

Data Description

Data Source: The data set I have obtained is from Kaggle Inc. website. There are two csv files: application_record.csv and credit_record.csv.

Data Description: To predict the credit card approval decision, I would be using and combining two sets of data, one would contain the customer's personal information and other contains the customer's behaviour patterns.

Application_record.csv consists of personal information of the customers. It contains total of 18 features including ID, gender, car, number of children, total income, education level, family status, housing type, birth date, employment, phone, email, occupation, and family size. There are total of 18 columns and 438,557 rows.

Feature Name	**Feature Content**
ID	Customer number
CODE_GENDER	Gender
FLAG_OWN_CAR	Car ownership
FLAG_OWN_REALTY	Property ownership
CNT_CHILDREN	Number of children
AMT_INCOME_TOTAL	Total Income
NAME_INCOME_TYPE	Income type
NAME_EDUCATION_TYPE	Education level
NAME_FAMILY_STATUS	Marital status
NAME_HOUSING_TYPE	Housing status
DAYS_BIRTH	Birth date
DAYS_EMPLOYED	Employment start date
FLAG_MOBIL	Mobile phone
FLAG_WORK_PHONE	Work phone
FLAG_PHONE	Phone
FLAG_EMAIL	Email
OCCUPATION_TYPE	Occupation type
CNT_FAM_MEMBERS	Family size

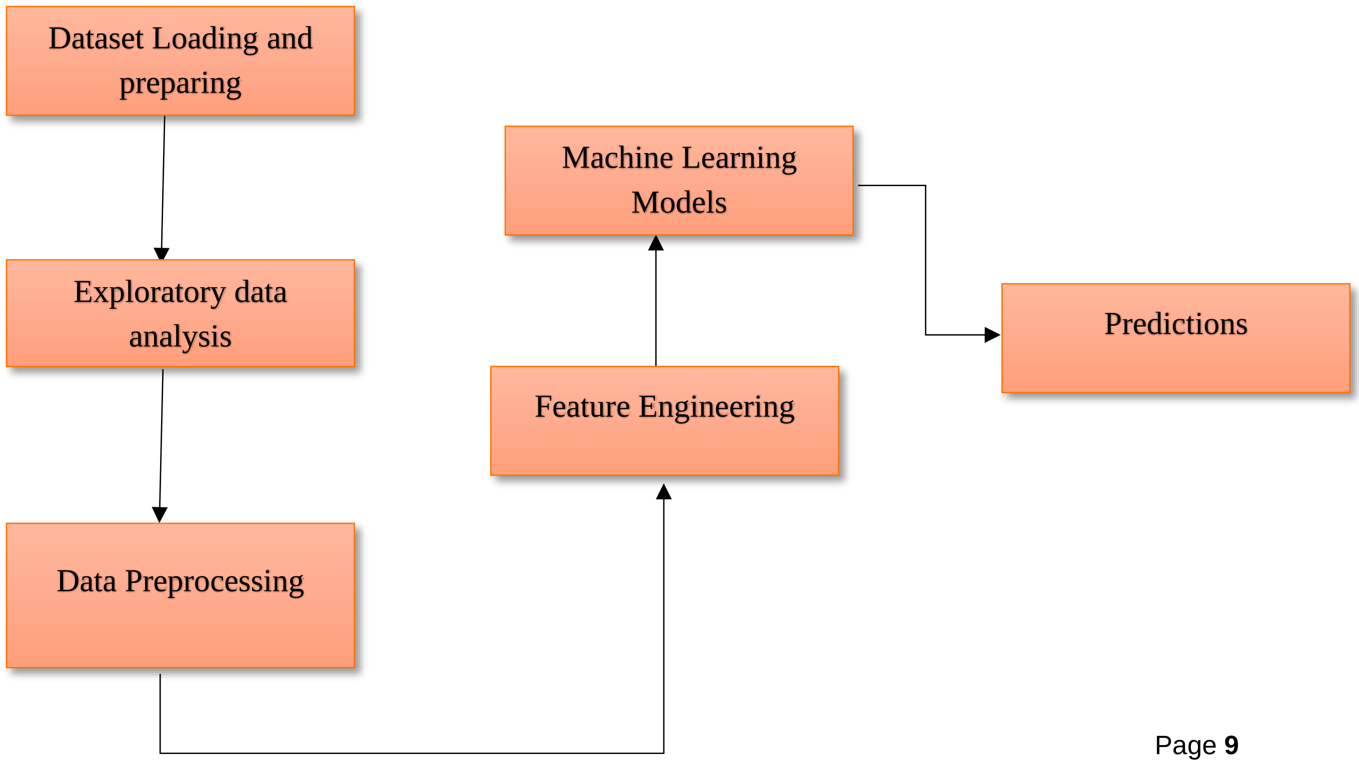
Table 1: Application_record.csv data description

Credit_record.csv is a dataset that contains credit card user's behavior pattern. Data consists of ID, Monthly balance, and status of the credit card user. MONTH_BALANCE variable proceeds in reverse order, the month of extracted data is the starting point. O is the current month, -1 is the previous month, and so on. There are total of 3 columns and 1,048,575 rows

Feature Name	**Feature Content**
ID	Customer number
MONTHS_BALANCE	Record month
STATUS	Status of monthly payment

Table 2: Credit_record.csv data description

Approach Overview



Data Loading and Data Preparation

From the different programming languages available for data analysis, language of my choice is Python as it has number of packages that can be used to accomplish big data analysis project.

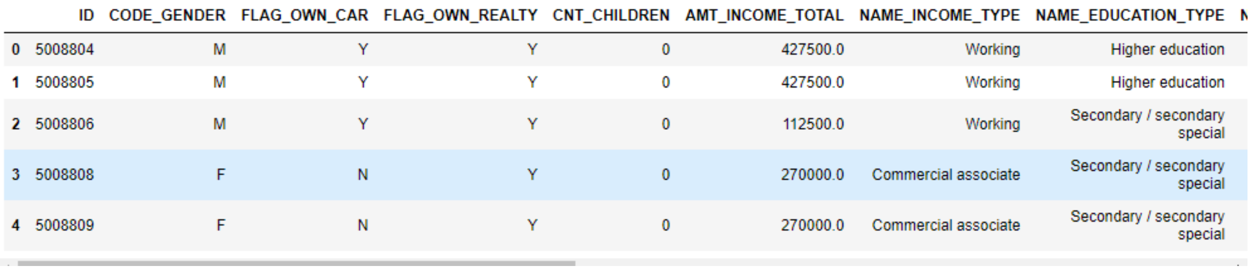
First step is to import the number of useful packages, including *Numpy*, *Pandas*, *Sklearn*, *Matplotlib*, *Seaborn* to prepare, clean, build and plot the dataset.

In the next step, I loaded the dataset that includes 2 csv files using *Pandas* package in python. I named the application record as df_application and credit record as df_credit.

Data Exploration

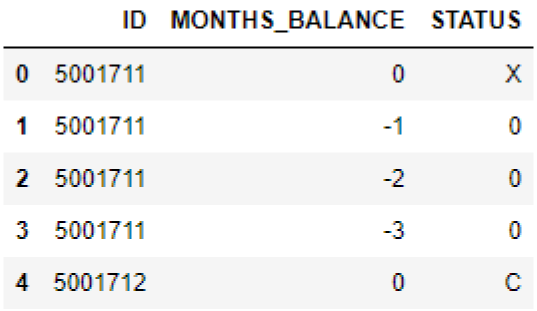
I started the data exploration process by exploring each csv file individually.

- I applied df_application.head() to get a snapshot of the Application record dataset.



	ID	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	NAME_INCOME_TYPE	NAME_EDUCATION_TYPE	N
0	5008804	M	Y	Y	0	427500.0	Working	Higher education	
1	5008805	M	Y	Y	0	427500.0	Working	Higher education	
2	5008806	M	Y	Y	0	112500.0	Working	Secondary / secondary special	
3	5008808	F	N	Y	0	270000.0	Commercial associate	Secondary / secondary special	
4	5008809	F	N	Y	0	270000.0	Commercial associate	Secondary / secondary special	

- I applied df_credit.head() to get a snapshot of the Credit record dataset.



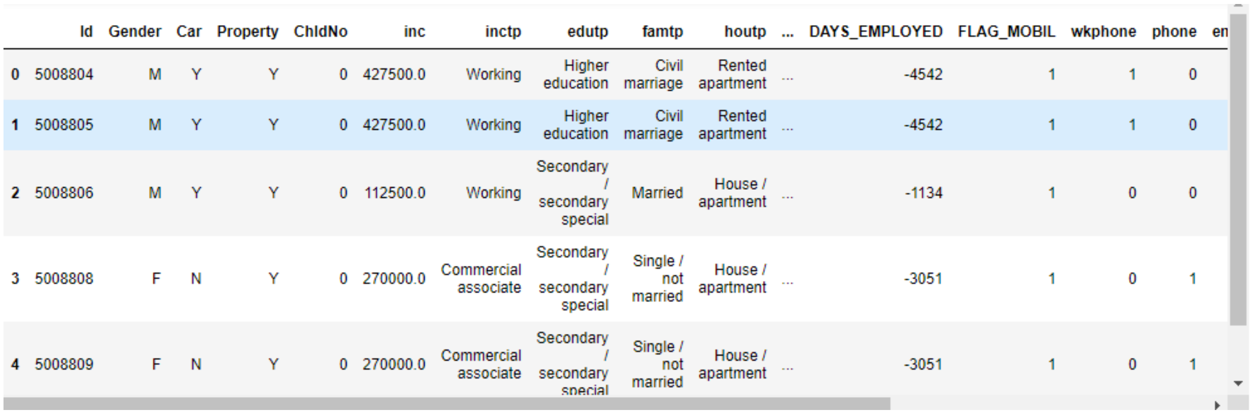
	ID	MONTHS_BALANCE	STATUS
0	5001711	0	X
1	5001711	-1	0
2	5001711	-2	0
3	5001711	-3	0
4	5001712	0	C

Data Preprocessing

In this step, I analyzed, cleaned, and transformed all of the features to use them in the Prediction Models.

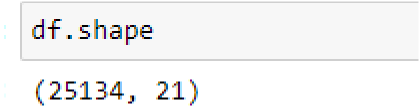
I. **Creating a target variable:** Target variable is the variable that needs to be predicted. I chose the users who overdue for more than 60 days as target risk users. I created a new variable 'dep_value' in credit record dataset to calculate the overdue time of customers so that we can create our target variable based on that. I merged the resultant data frame with main dataset to get a target column/variable.

II. **Renaming columns:** I renamed columns of the resultant data frame to simpler language.



	Id	Gender	Car	Property	ChldNo	inc	inctp	edutp	famtp	houtp	...	DAYS_EMPLOYED	FLAG_MOBIL	wkphone	phone	en
0	5008804	M	Y	Y	0	427500.0	Working	Higher education	Civil marriage	Rented apartment	...	-4542	1	1	0	
1	5008805	M	Y	Y	0	427500.0	Working	Higher education	Civil marriage	Rented apartment	...	-4542	1	1	0	
2	5008806	M	Y	Y	0	112500.0	Working	Secondary / secondary special	Married	House / apartment	...	-1134	1	0	0	
3	5008808	F	N	Y	0	270000.0	Commercial associate	Secondary / secondary special	Single / not married	House / apartment	...	-3051	1	0	1	
4	5008809	F	N	Y	0	270000.0	Commercial associate	Secondary / secondary special	Single / not married	House / apartment	...	-3051	1	0	1	

III. **Removing Null Values:** In the next step of preprocessing the data set, I dropped the Null Values. The result was:



```
df.shape
```

```
(25134, 21)
```

Feature Engineering

I. **WoE (Weight of Evidence) Value** (Bhalla, 2015)**:**

The weight of evidence tells the predictive power of an independent variable in relation to the dependent variable. Since it evolved from credit scoring world, it is generally described as a measure of the separation of good and bad customers. **"Bad Customers"** refers to the customers who defaulted on a loan. and **"Good Customers"** refers to the customers who paid back loan.

$$\text{WOE} = \ln\left(\frac{\text{Distribution of Goods}}{\text{Distribution of Bads}}\right)$$

Distribution of Goods *- % of Good Customers in a particular group*

Distribution of Bads *- % of Bad Customers in a particular group*

ln *- Natural Log*

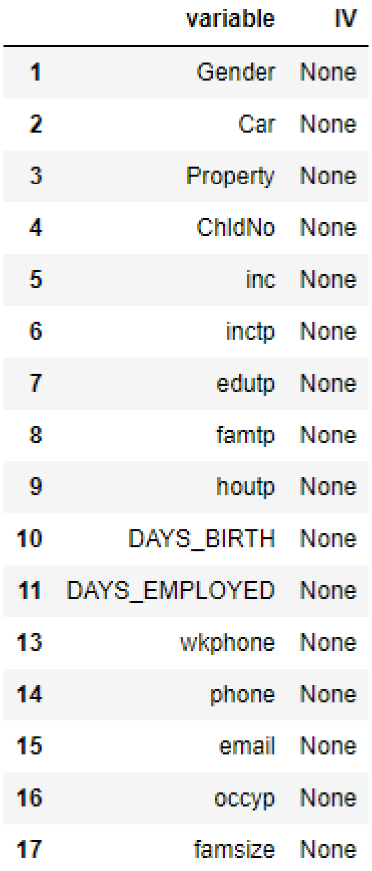
- Positive WOE means Distribution of Goods > Distribution of Bads
- Negative WOE means Distribution of Goods < Distribution of Bads

II. **Information Value (IV)** (Bhalla, 2015)**:**

Information value is one of the most useful techniques to select important variables in a predictive model. It helps to rank variables on the basis of their importance. The IV is calculated using the following formula:

$$\text{IV} = \sum (\text{\% of non-events - \% of events}) * \text{WOE}$$

Feature Selection: In the process of selecting important features, I created a new data frame ‘ivtable’ with just the main data set columns and IV Value, where IV was none. I then, dropped the unwanted columns from the data frame, 'FLAG_MOBIL' 'begin_month', ‘dep_value', 'target', 'Id'.



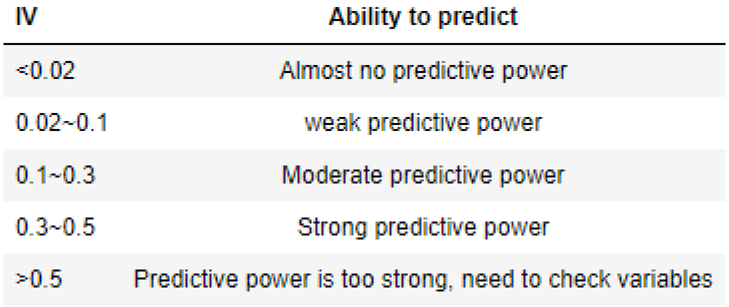
	variable	IV
1	Gender	None
2	Car	None
3	Property	None
4	ChldNo	None
5	inc	None
6	inctp	None
7	edutp	None
8	famtp	None
9	houtp	None
10	DAYS_BIRTH	None
11	DAYS_EMPLOYED	None
13	wkphone	None
14	phone	None
15	email	None
16	occyp	None
17	famsize	None

I defined the following functions:

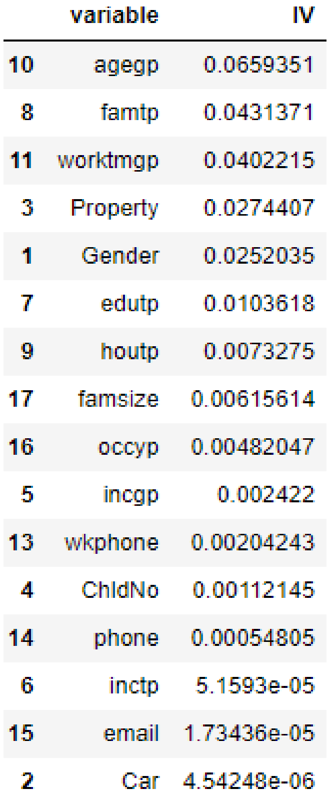
- Function to calculate information value
- Function to encode the numerical features with more than two unique values
- Function to make the categories of required numerical features
- Function to plot the confusion matrix for the machine learning models

In my next step, I calculated the Information Value (IV) of the features, did encoding and made the categories for the features: Gender, Car, Property, Phone, Email, Work Phone, Number of children, Annual Income, Age, Working years, Family size, Income type, Occupation type, House type, Education, Marital status.

Relationship between IV value and predictive power



IV	Ability to predict
<0.02	Almost no predictive power
0.02~0.1	weak predictive power
0.1~0.3	Moderate predictive power
0.3~0.5	Strong predictive power
>0.5	Predictive power is too strong, need to check variables



	variable	IV
10	agegp	0.0659351
8	famtp	0.0431371
11	worktmgp	0.0402215
3	Property	0.0274407
1	Gender	0.0252035
7	edutp	0.0103618
9	houtp	0.0073275
17	famsize	0.00615614
16	occyp	0.00482047
5	incgp	0.002422
13	wkphone	0.00204243
4	ChldNo	0.00112145
14	phone	0.00054805
6	inctp	5.1593e-05
15	email	1.73436e-05
2	Car	4.54248e-06

Machine Learning Algorithms

As mentioned in the literature review section, there are a different option when it comes to applying machine learning models to predict if a customer's credit card application would be accepted or rejected. I have applied six Classifiers to the dataset to see which machine learning model is the most accurate.

Split the data into Train and Test Sets

I started with creating Y data frame for target variable and X data frame for selected features. I then applied SMOTE technique to overcome the sample imbalance problem.

Synthetic Minority Oversampling Technique (Satpathy, 2021)

SMOTE is an oversampling technique where the synthetic samples are generated for the minority class. This algorithm helps to overcome the overfitting problem posed by random oversampling. It focuses on the feature space to generate new instances with the help of interpolation between the positive instances that lie together.

I proceeded with splitting the dataset into Test & Test using *train_test_split()*.

Model Description

I. **Logistic Regression:** A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

II. **Decision Tree:** A decision tree is a predictive model based on a branching series of Boolean tests that use specific facts to make more generalized conclusions.

III. **Random Forest:** Random Forest is a robust machine learning algorithm. It is an ensemble method, meaning that a random forest model is made up of many small decision trees, called estimators, which each produce their own predictions. The random forest model combines the predictions of the estimators to produce a more accurate prediction.

IV. **SVM:** Support vector machine (SVM) is machine learning algorithm that analyzes data for classification and regression analysis. SVM is a supervised learning method that looks at data and sorts it into one of two categories. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible. SVMs are used in text categorization, image classification, handwriting recognition and in the sciences.

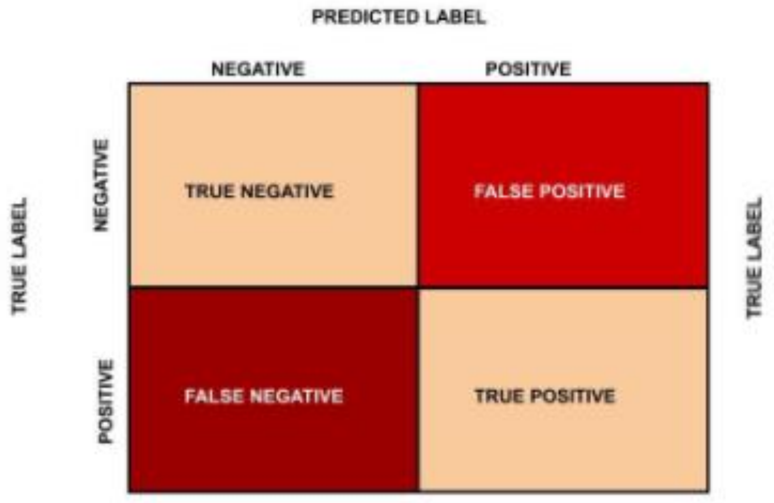
V. **XGBOOST:** It is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. Execution speed and high performance are the main reasons to use XGBoost.

VI. **K-Nearest Neighbor (KNN):** A k-nearest-neighbor algorithm, often abbreviated k-nn, is an approach to data classification that estimates how likely a data point is to be a member of one group or the other depending on what group the data points nearest to it are in.

Evaluation

I evaluated the above-mentioned machine learning models and validated how good or bad it is using the performance metrics.

I. **Confusion Matrix:** Confusion Matrix as the name suggests gives us a matrix as output and describes the complete performance of the model.



		PREDICTED LABEL	
		NEGATIVE	POSITIVE
TRUE LABEL	NEGATIVE	TRUE NEGATIVE	FALSE POSITIVE
TRUE LABEL	POSITIVE	FALSE NEGATIVE	TRUE POSITIVE

- **True Positives**: The cases in which we predicted YES and the actual output was also YES.

- **True Negatives**: The cases in which we predicted NO and the actual output was NO.

- **False Positives**: The cases in which we predicted YES and the actual output was NO.

- **False Negatives**: The cases in which we predicted NO and the actual output was YES.

II. **Precision:** Precision is the ratio of true positives and total positives predicted.

$$P = \frac{TP}{TP+FP} \quad 0<P<1$$

III. **Recall:** A Recall is the ratio of true positives to all the positives in ground truth.

$$R = \frac{TP}{TP+FN} \quad 0<R<1$$

IV. **F-1 Score:** The F1-score metric uses a combination of precision and recall. F1 score is the harmonic mean of the two. The formula of the two essentially is:

$$F_1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}}$$

The range for F1 Score is [0, 1]. It tells you how precise your classifier is, as well as how robust it is.

V. **Accuracy:** Classification Accuracy is what we usually mean when we use the term accuracy. It is the ratio of number of correct predictions to the total number of input samples.

$$Accuracy = \frac{Number\ of\ Correct\ predictions}{Total\ number\ of\ predictions\ made}$$

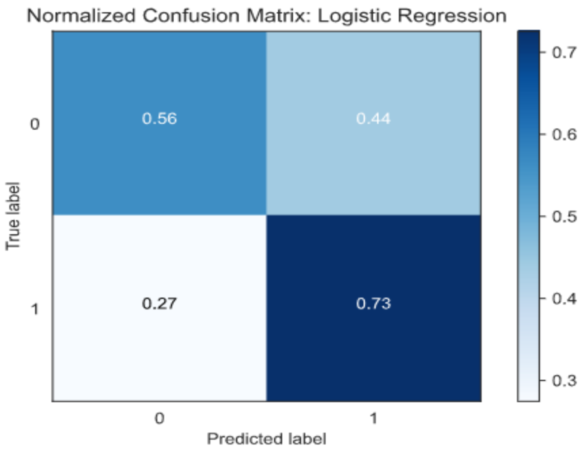
VI. **AUC & ROC Score:** It makes use of true positive rates (TPR) and false positive rates (FPR).

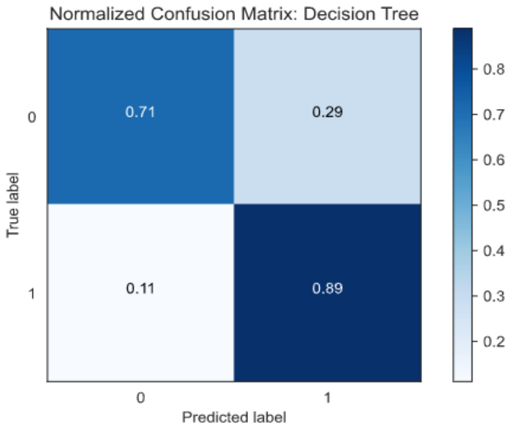
$$TPR = \frac{TP}{TP+FN} \qquad FPR = \frac{FP}{FP+TN}$$

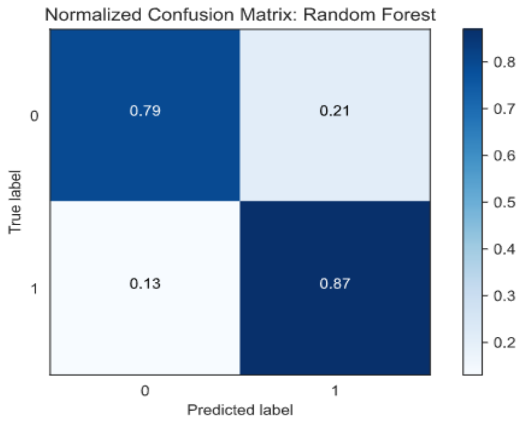
AUC has a range of [0, 1]. The greater the value, the better is the performance of our model.

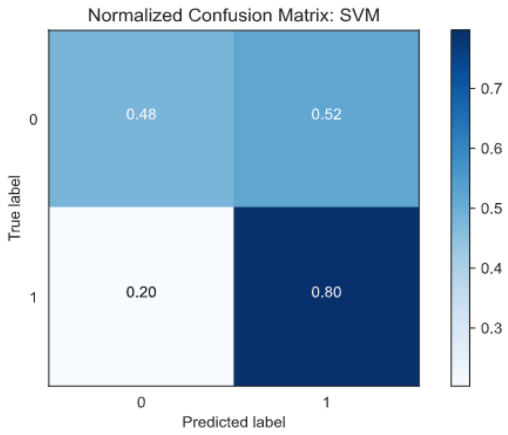
Results

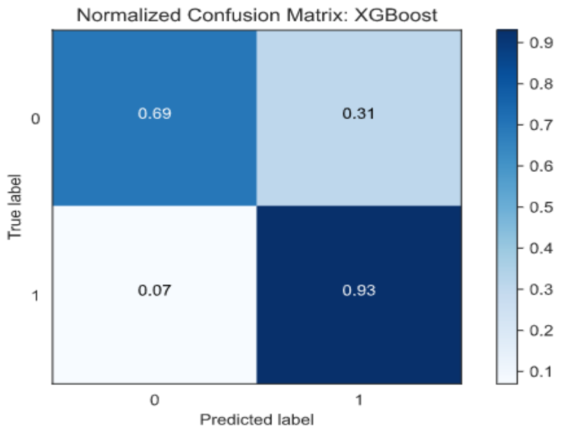
Confusion Matrix:

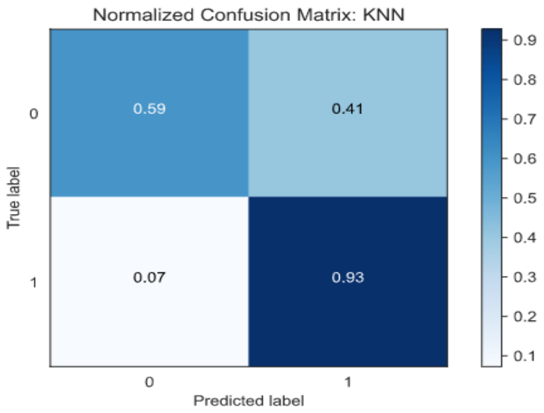

Normalized Confusion Matrix: Logistic Regression


Normalized Confusion Matrix: Decision Tree


Normalized Confusion Matrix: Random Forest


Normalized Confusion Matrix: SVM


Normalized Confusion Matrix: XGBoost


Normalized Confusion Matrix: KNN

Result Table:

MODELS	Evaluation Metrics				
	Accuracy	F-1 Score	Precision	Recall	AUC Score
Logistic Regression	0.64	0.67	0.62	0.71	0.70
Decision Tree	0.80	0.81	0.75	0.89	0.89
Random Forest	0.83	0.83	0.80	0.87	0.93
SVM	0.64	0.69	0.60	0.80	0.67
XGBoost	0.81	0.83	0.75	0.93	0.89
KNN	0.76	0.80	0.70	0.93	0.78

Findings:

The trained and tested data are processed through six different machine learning algorithms to achieve the best accuracy result. We used six different performance metrics to evaluate the performance of these models. From the result table, I noticed that all the models performed well. It shows that Random Forest model gives the best accuracy result of 83%. A very close competition was given by XGBoost classifier with same F-1 score. I selected Random Forest as best performing model on the basis of Accuracy and AUC Score. On the other hand, I noticed Logistic Regression and SVM algorithms performed poorly compared to other models with 64% accuracy.

Conclusion

In this project, I predicted if the credit card application of the customer would be accepted or rejected based on the customer information and behavioral data provided by Kaggle. The analytical process started with introduction of two datasets, data cleaning and processing. Important features were finalized after completion of feature engineering. Six machine learning models were implemented and compared, to identify the most efficient and best performing model. Six different performance metrics were used to evaluate and validate the performance of the models. The best accuracy score is provided by Random Forest model. Thus, applicants with poor credit history would get rejected because of probability of them to not pay back to the bank. Age group plays an important characteristic in deciding if the credit card application should be accepted or rejected.

Future Improvement

Future work on this dataset or any similar data set could include combination of two or more techniques to build a classification model with a higher degree of accuracy. If the improved model is used by the creditors, it can greatly reduce the risk of granting credit to potential defaulters. Therefore, it can be efficient technique to assess financial risks and make appropriate financial decisions.

References

1. *Credit Card Approval Prediction*. (2020, March 24). Kaggle. https://www.kaggle.com/rikdifos/credit-card-approval-prediction

2. Naik, K. S. (2021). Predicting Credit Risk for Unsecured Lending: A Machine Learning Approach. *arXiv preprint arXiv:2110.02206*. Available at: https://arxiv.org/pdf/2110.02206.pdf

3. Shin, W. -S., & Shin, D. -H. (2020). A Study on the Application of Artificial Intelligence in Elementary Science Education. *Journal of Korean Elementary Science Education, 39(1), 117-132.* Available at: https://www.koreascience.or.kr/article/JAKO202116758671173.pdf

4. Bansal, Siddhi, & Punjabi, Tushar. (2021). Comparison of Different Supervised Machine Learning Classifiers to Predict Credit Card Approvals. *International Research Journal of Engineering and Technology (IRJET), E-ISSN: 2395-0056, P-ISSN: 2395-0072, 8(3), 1339-1348, March 2021.* Available at: https://www.irjet.net/archives/V8/i3/IRJET-V8I3277.pdf

5. D.Jayanthi. (2018). Credit Approval Data Analysis Using Classification and Regression Models. *IJRAR-International Journal Of Research And Analytical Reviews (IJRAR), E-*

ISSN 2348-1269, P- ISSN 2349-5138, 5(3), 162-169, September 2018. Available at: https://www.ijrar.org/papers/IJRAR190B030.pdf

6. Arokiaraj Christian St Hubert, R.Vimalesh, M. Ranjith, & S. Aravind Raj. (2020). Predicting Credit Card Approval of Customers Through Customer Profiling using Machine Learning. *International Journal of Engineering and Advanced Technology (IJEAT), 9(4), 52-557.* Available at: https://www.ijeat.org/wp-content/uploads/papers/v9i4/D7293049420.pdf

7. Kibria, Md & Şevkli, Mehmet. (2021). Application of Deep Learning for Credit Card Approval: A Comparison with Two Machine Learning Techniques. *International Journal of Machine Learning and Computing, 11(4), 286-290, July 2021.* Available at: 10.18178/ijmlc.2021.11.4.1049.

8. Bhalla, D. (2015). *WEIGHT OF EVIDENCE (WOE) AND INFORMATION VALUE (IV) EXPLAINED*. Listen Data. https://www.listendata.com/2015/03/weight-of-evidence-woe-and-information.html

9. P. (2018, April 3). *IV + WoE Starter for Python*. Kaggle. https://www.kaggle.com/code/puremath86/iv-woe-starter-for-python/notebook

10. Satpathy, S. (2021, January 6). *SMOTE | Overcoming Class Imbalance Problem Using SMOTE*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/10/overcoming-class-imbalance-using-smote-techniques/

11. Bajaj, A. (2022, March 18). *Performance Metrics in Machine Learning [Complete Guide]*. Neptune.Ai. https://neptune.ai/blog/performance-metrics-in-machine-learning-complete-guide

12. Mishra, A. (2021, December 29). *Metrics to Evaluate your Machine Learning Algorithm*. Medium. https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234