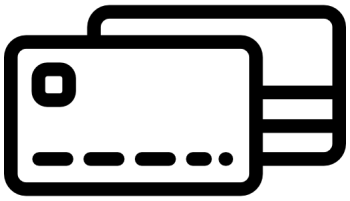
A Report on

CREDIT CARD APPROVAL PREDICTION PROJECT



Submitted by

Group 18

Aiwen Peng, Jianghui Li, Jinchao Zhao, Xintian Zhang

Submitted to

Dr. Emory Creel

December 9th, 2022

1. ABSTRACT

1.1 Problem Statement

Trust is the basis of human interaction, and it is also a moral purpose and means. Credit is a part of people's financial strength and has a significant role in their lives. Good credit may be the detail that makes or breaks your ability to get a credit card, car loan, or student loan. At the same time, bad credit will make it challenging to apply for a credit card or get a low-interest loan. Even if some people don't need a loan, good credit can have a significant impact. For example, landlords, insurance companies, and employers try to use credit information as a basis.

Credit scoring is a standard risk control tool for banks or companies. Personal information and historical data about the applicant are used to predict whether the applicant will default or have bad debts in the future. Banks can decide whether to offer applicants a loan or credit card by credit scoring. This can help the bank to reduce the risk.

1.2 Dataset

This project will use two datasets, application records and credit records. The application record dataset contains personal information about the customer, while the credit record dataset contains information about the customer's monthly credit status. The data comes from Kaggle, for the application record dataset, it has 438,557 observations and 18 features, including numerical and categorical data. For the credit record dataset, it has 1048575 rows and 3 columns.The dataset contains the information of over 400,000 customers, so we expect the customers' gender would proportion equally, but only about one third of the customers are male, and the rest are female.

1.3 Proposed techniques and data science methodology

We will use different solutions for different stages of the problem. These phases include Business Understanding, Data Preprocessing, Exploratory Data Analysis, Feature Engineering, Data Modeling, Model Evaluation, Recommendations and Insights. Combine the two data and clean the NA values in the data, and convert the data into meaningful variables, such as converting 'DAY_BRITH' and 'DAY_EMPLOYEE' in the data into understandable values. Feature engineering involves selecting the essential features from the data. Data modeling consists in dividing the data into two datasets for training and validation and using different algorithms such as Naive Bayes, Random Forest, Logistics Regression, Decision Tree, and GBT to build the best model. Use metrics for model evaluation like accuracy, precision, recall, F1 score, and AUC/ROC curve to select the best model. Get recommendations and insights from exploratory analysis and build models to solve business problems.

1.4 Business Questions intended to be analyzed

We can provide suggestions for banks or financial companies through the analysis results to help them avoid risks.

- Do applicants older than 25 have better chances of being ‘good’ clients?
- Do applicants with high education and high income have better chances of being 'good' clients?
- Does ‘Housing Type’ affect predicting an applicant is ‘good’ or ‘bad’?
- Does ‘the number of children’ and ‘Income’ together affect predicting an applicant is ‘good’ or ‘bad’?
- Does ‘Income Type’ have any impact on predicting an applicant is ‘good’ or ‘bad’?

By understanding the factors most likely to default, banks or financial companies can take measures to reduce credit card applications for these people or reduce the applicant's usage limit to avoid unnecessary risks.

2. DATA DESCRIPTION

2.1 Overview/Description

Two datasets will be used for this project: the application record and the credit record. The application record dataset includes customers' personal information, and the credit record dataset contains customers' monthly credit status information.

2.2 Number of rows and columns

For the application record dataset, it has 438557 rows and 18 columns. The columns are ['ID','CODE_GENDER','FLAG_OWN_CAR','FLAG_OWN_REALTY','CNT_CHILDREN','AMT_INCOME_TOTAL','NAME_INCOME_TYPE','NAME_EDUCATION_TYPE','NAME_FAMILY_STATUS','NAME_HOUSING_TYPE','DAYS_BIRTH','DAYS_EMPLOYED','FLAG_MOBIL','FLAG_WORK_PHONE','FLAG_PHONE','FLAG_EMAIL','OCCUPATION_TYPE','CNT_FAM_MEMBERS']. For the credit record dataset, it has 1048575 rows and 3 columns. The columns are ['ID','MONTHS_BALANCE','STATUS'].

Application record file:

- ID: client number
- CODE_GENDER: gender
- FLAG_OWN_CAR: if users have a car
- FLAG_OWN_REALTY: is there a property
- CNT_CHILDREN: number of children
- AMT_INCOME_TOTAL: annual income
- NAME_INCOME_TYPE: income category
- NAME_EDUCATION_TYPE: education level
- NAME_FAMILY_STATUS: warital status
- NAME_HOUSING_TYPE: way of living
- DAYS_BIRTH: birthday(Count backwards from current day (0), -1 means yesterday)
- DAYS_EMPLOYED: start date of employment(Count backwards from current day(0). If positive, it means the person currently unemployed)
- FLAG_MOBIL: is there a mobile phone
- FLAG_WORK_PHONE: is there a work phone
- FLAG_PHONE: is there a phone
- FLAG_EMAIL: is there an email
- OCCUPATION_TYPE: occupation
- CNT_FAM_MEMBERS: family size

Credit record file:

- ID: client number
- MONTHS_BALANCE: The month of the extracted data is the starting point, backwards, 0 is the current month, -1 is the previous month, and so on.
- STATUS: 0: 1 month past due; 1: 2 month past due; 2: 3 month overdue; 3: 4 month days overdue; 4: 5 month overdue; 5: Overdue or bad debts, write-offs for more than 150 days C: paid off that month X: No loan for the month

2.3 Sample predictors

The application's personal information (we will determine which are useful columns that can be used to make predictions as we further explore the dataset) and credit history is used to predict whether the applicant is a 'good' or 'bad client.

2.4 Data Source

The dataset was downloaded from Kaggle, https://www.kaggle.com/datasets/rikdifos/credit-card-approval-prediction?select=application_record.csv.

2.5 Interesting or surprising about the data

The dataset contains the information of over 400,000 customers, so we expect the customers' gender would proportion equally, but only about one third of the customers are male, and the rest are female. Also, by looking at the min and max values, we found one not married female customer has 19 children.

3. DATA PREPROCESSING

3.1 Data Cleaning

Firstly, we checked if there are any missing values in both dataframes. For the application record, there are 134203 null values in the OCCUPATION_TYPE column. The credit record doesn't have any missing value. In the later process, we merged two datasets and counted null values for each column, and there is no null value because null values were dropped during the merging process. Hence, we don't need to further interpolate or drop any data.

3.2 Data Merging

We merged the 'application record dataset' and the 'credit record dataset.' In this dataset, we do not have a target column to tell whether the application is approved or not. So, we need to define what applicants are 'good' and what applicants are 'bad.' In this project, we suppose a customer whose 'STATUS' equals '0', '1', 'C', and 'X' is a 'Good' customer; ('STATUS' C means loan paid off; 'STATUS' X means No loan; 'STATUS' 0 means one month past due; 'STATUS' 1 means two month past due.) A customer who had other 'STATUS' is a 'Bad' customer.

After merging, there may be multiple rows of records for each customer. For example, the figure below shows 4 records for a customer whose ID is 5001711; and 16 records for a customer whose ID is 5001712. In this situation, a customer may have many 'Good' and 'Bad' labels.

Then, we defined a ratio that equals the number of 'Bad' labels for one customer divided by the number of total labels for this customer. If one has been a customer for a long period of time, he or she tends to have a higher chance to have 'Bad' record, so we decided that when the bad ratio (count(bad)/count(bad+good)) was bigger than 0.03, we set this customer's label to 'Bad' and other customers as 'Good', which means if one has been a customer for more than 33 months, they need to have 2 'Bad' records to be labeled 'Bad'.

ID	MONTHS_BALANCE	STATUS
5001711	0	X
5001711	-1	0
5001711	-2	0
5001711	-3	0
5001712	0	C
5001712	-1	C
5001712	-2	C
5001712	-3	C
5001712	-4	C
5001712	-5	C
5001712	-6	C
5001712	-7	C
5001712	-8	C
5001712	-9	0
5001712	-10	0
5001712	-11	0
5001712	-12	0
5001712	-13	0
5001712	-14	0
5001712	-15	0

Figure 3.1

In the original dataset, the customer's age is expressed in days, which is very indirect. We added a new column named 'month_old' to represent age. We also added a new column named 'work_month' to indicate how many months a customer worked instead of the 'DAYS_EMPLOYED' column.

4. DESCRIPTIVE STATISTICS

We performed Exploratory Data Analysis and used those visualizations to find insights about the credit card application data. We found that many features do not demonstrate traits of "good" and "bad" applicants. All the figures below are the visualizations of selected features that can show us the difference between "good" and "bad" applicants.

- The distribution of family status by customer type

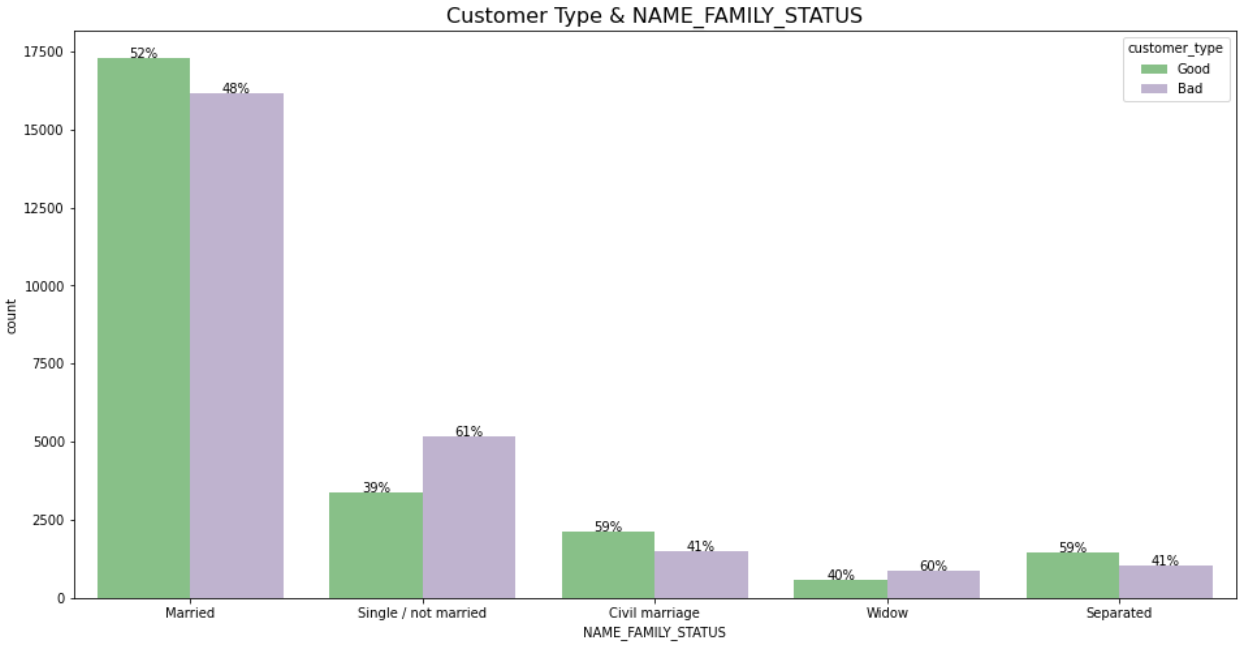


Figure 4.1

Figure 4.1 tells us that single/not married people have a higher chance of being "bad". Civil marriage people have a higher chance of being "good".

- The distribution of housing type by customer type

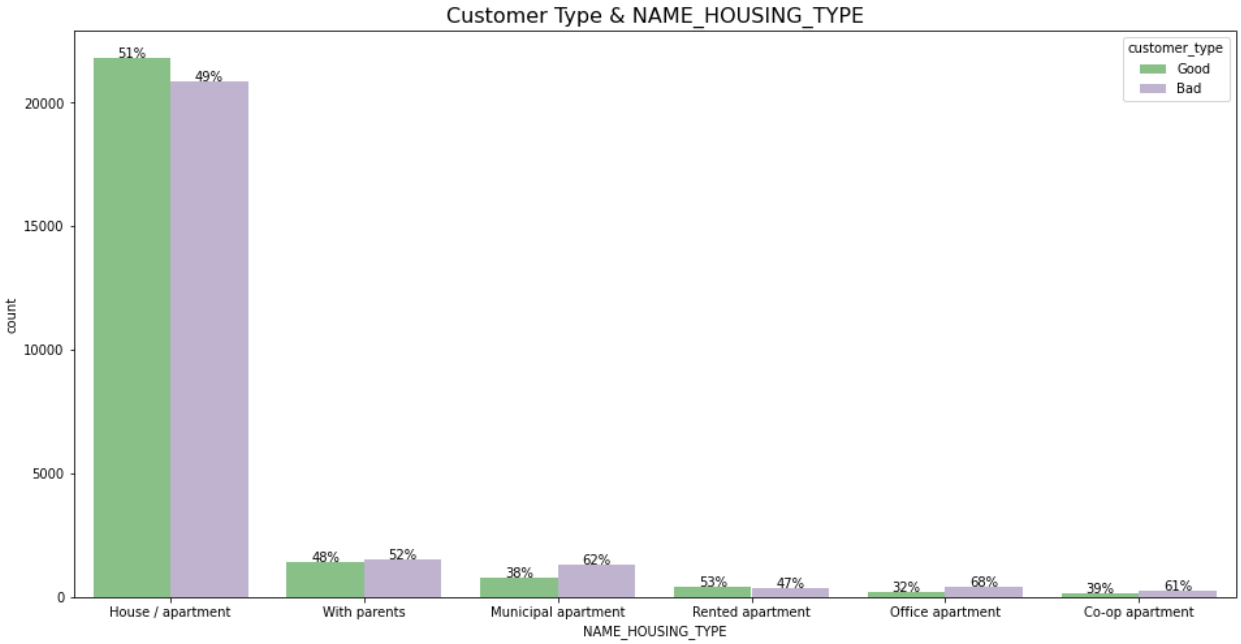


Figure 4.2

Around ⅔ of people who live in municipal apartments, office apartments, and co-op apartments tend to be “bad” applicants.

- The distribution of occupation type by customer type

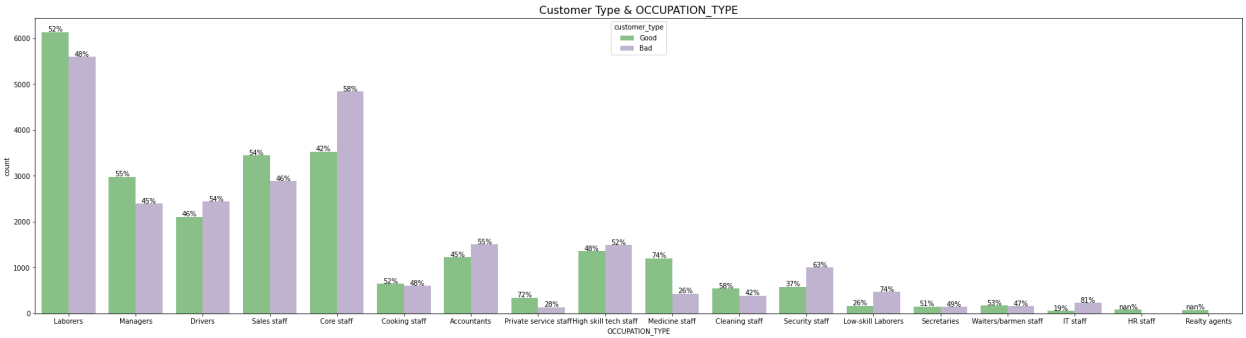


Figure 4.3

The figure 4.3 tells us that laborers and sales staff have a better chance to be “good” customers, while core staff have a larger portion of “bad” customers.

- The distribution of gender by customer type

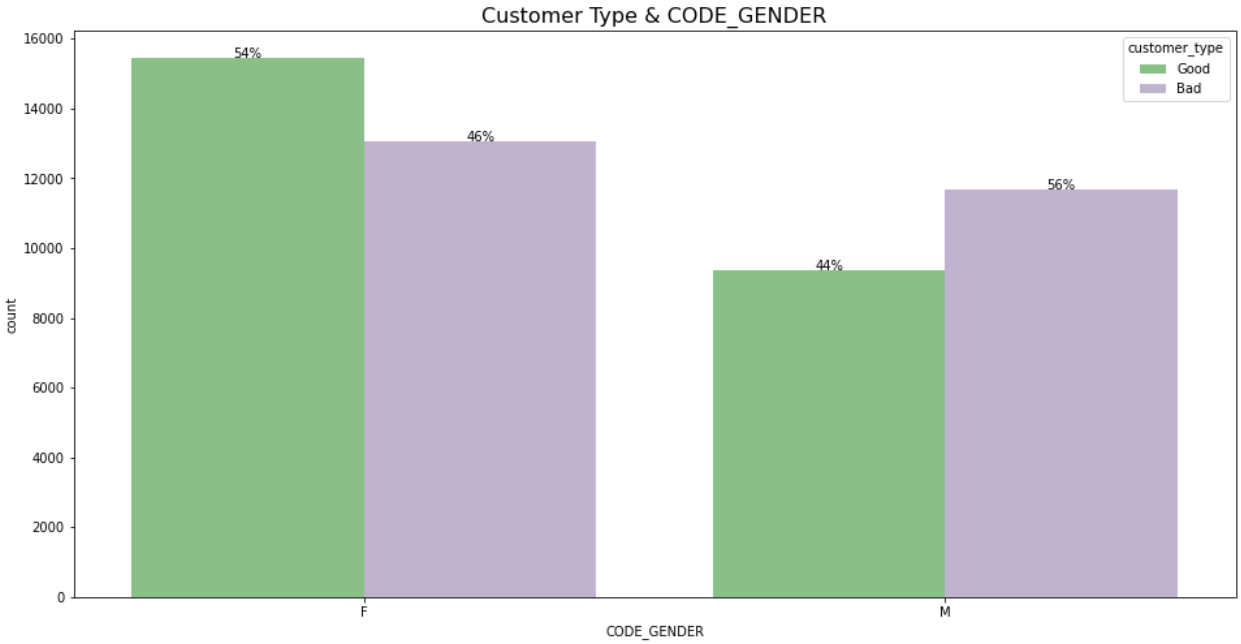


Figure 4.4

The above chart shows that female applicants are more likely to be “good” customers.

- The distribution of work_month by customer type

Figure 4.5

Figure 4.5 shows that the longer the number of work months, the better chance to be “bad” customers.

- The distribution of begin month by customer type

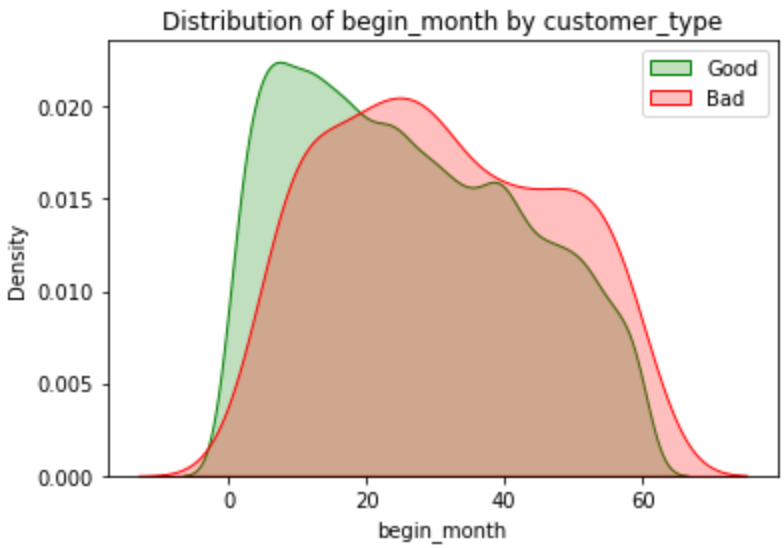


Figure 4.6

Figure 4.6 shows that the start point of the customer has an effect on prediction. From the above chart, it's easy to say that the longer the customer exists, the better chance of him/she to be a "bad" applicant. Also, we need to note that the credit record dataset does not contain applicants who are already lost or bad because their latest record is month 0, so we can use their credit record as a feature later.

- The distribution of month_old by customer type

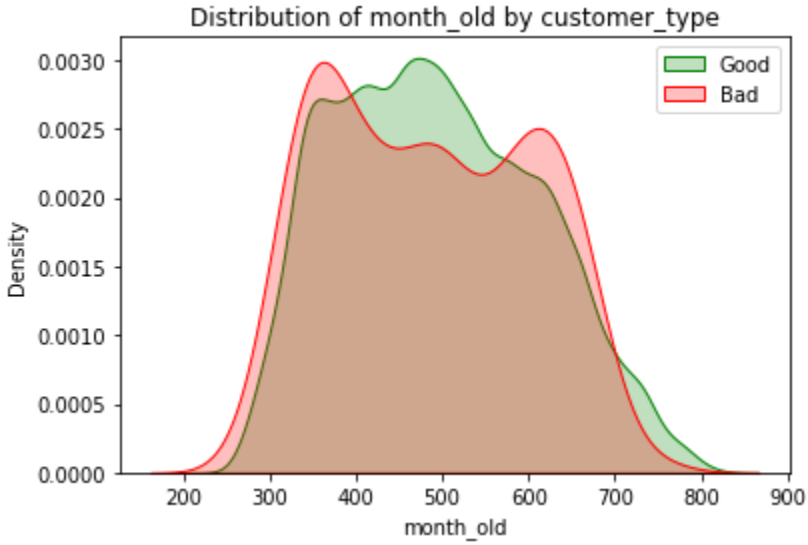


Figure 4.7

In figure 4.7, the count of months shows that applicants between 35-45 years old are more likely to be "good". Young adults seem to be more likely to be "bad".

5. METHODOLOGY

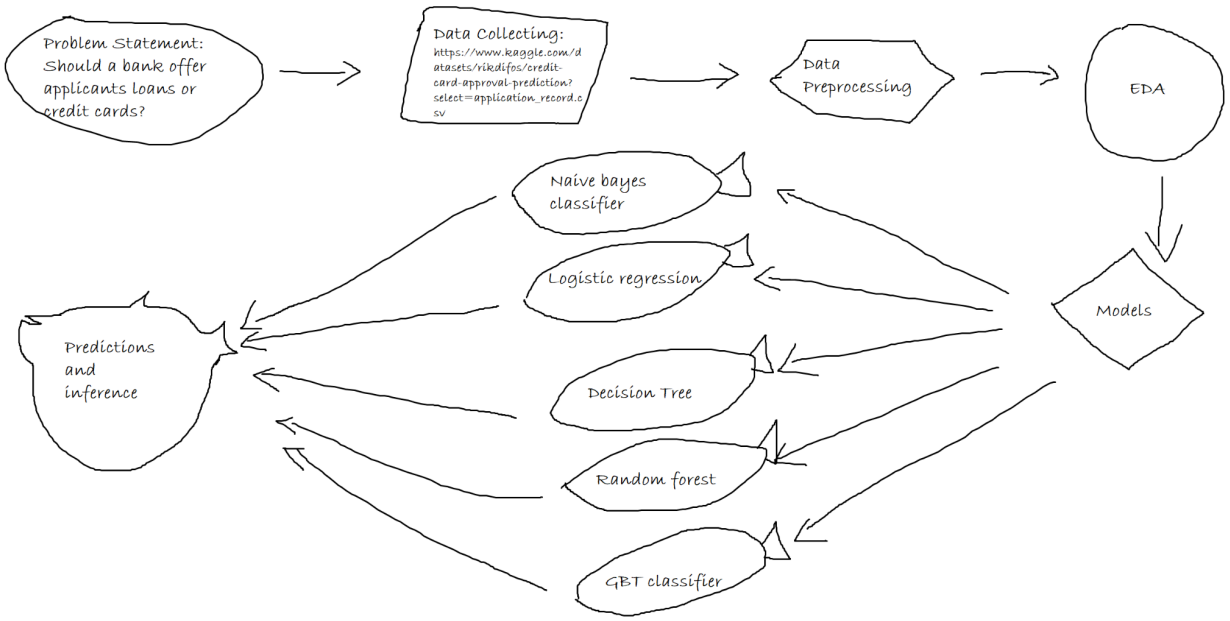


Figure 5.1

5.1 Feature Engineering

There are many columns in the merged dataset that are categorical. Hence, we need to select and transform raw data into features that can be used in supervised machine learning models. The index variable “ID” will not be used, so we can drop it. We dropped the “FLAG_MOBIL” column since all of the values in this feature are “1”, so this feature will not be helpful for training our model later. For other categorical features, we indexed them so that they are converted to numerical values.

5.2 Data Balancing

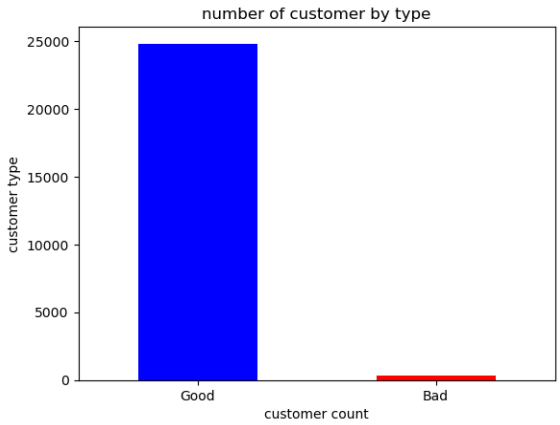


Figure 5.2

By looking at the figure 5.2 above, we can clearly tell that the data is not balanced for training. Hence, we decided to use the undersampling technique so that we keep all the "bad" applicants and sample "good" applicants to be the similar total count of "bad" applicants.

6. MODELS & INFERENCE

6.1 Naive Bayes

First, credit card applicants were classified by the Naive Bayes model. This model applies Bayes' theorem with strong independence assumptions between attributes. It is fast and able to get the prediction output quickly.

Table 6.1

	AUC-ROC	AUC-PR	Accuracy	Precision	Recall	F1
Scores	0.490	0.452	0.526	0.500	0.508	0.504

The first model evaluation indicator we want to look at is the accuracy, and the accuracy for Naive Bayes model is 52.6%, which is slightly better than random prediction. Precision is the total true positive over sum of the true positive plus false positive, and recall is the total true positive over sum of the true positive plus false negative. Higher precision normally means false positive rate is low while higher recall commonly represents less false negative predictions. F1 is the average of precision and recall rate. Most evaluation indicators for the Naive Bayes model is around 50%. The ROC curve is the trade of between true positive rate and false positive rate, and it is 49%. As for the PR (precision vs recall) curve, it is only 45.2% because it does not take true negative into account, and Naive Bayes model produced more true negative values than that of the true positive.

6.2 Logistic Regression

We employed the Logistic Regression method to classify whether applicants are good or bad. It works by modeling the relationship between a dependent variable (the customer types) and one or more independent variables (all features that we use to make predictions). Therefore, Logistic regression can be used to predict likelihood that a loan applicant will default on their loan.

Table 6.2

	AUC-ROC	AUC-PR	Accuracy	Precision	Recall	F1
Scores	0.643	0.608	0.609	0.573	0.683	0.623

The overall performance of our Logistic Regression model is better than that of the Naive Bayes model. The accuracy is 60.9%, and we are pleased to see that the recall rate is 68.3%. Having fewer false negative values here means the model predicted fewer “bad” applicants who are actually “good”. Thus, the model will have a lower chance to assume a “good” applicant “bad”. In our prediction result, applicants who were predicted “good” would normally have a small portion of actually “bad” applicants. However, the predicted “bad” applicants might be mixed with a big portion of actually “good” applicants. At least, we know the model is good at filtering good people, so it can assist banks to narrow down the selection range.

6.3 Decision Tree

The first tree based model we used was the Decision Tree. Decision Tree is a type of supervised machine learning algorithm that can be used for both classification and regression tasks. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

Table 6.3

	AUC-ROC	AUC-PR	Accuracy	Precision	Recall	F1
Scores	0.615	0.564	0.609	0.596	0.540	0.567

Looking at the accuracy score for the Decision Tree model, it is just as good as that of the Logistic Regression model. However, if we compare recall to that of the Logistic Regression model, the Decision Tree only has 54%. Naturally, this model would produce more false negative outcomes. As we discussed in the Logistic Regression model inference, having a good recall rate can help accurately classify “good” applicants thereby people can further determine “good” and “bad” among predicted “bad” applicants.

6.4 Random Forest

The Random Forest model is a type of ensemble machine learning model that is composed of many decision trees. Similar to gradient boosted trees, random forests are often used for classification tasks. However, while gradient boosted trees train each tree sequentially, random forests train each tree independently, using a random subset of the data. This makes random forests less sensitive to the specific choice of training data and can improve the model's performance. Additionally, because the individual trees in a random forest are trained independently, the model can be easily parallelized, making it efficient to train on large datasets.

Table 6.4

	AUC-ROC	AUC-PR	Accuracy	Precision	Recall	F1
Scores	0.763	0.749	0.684	0.662	0.683	0.672

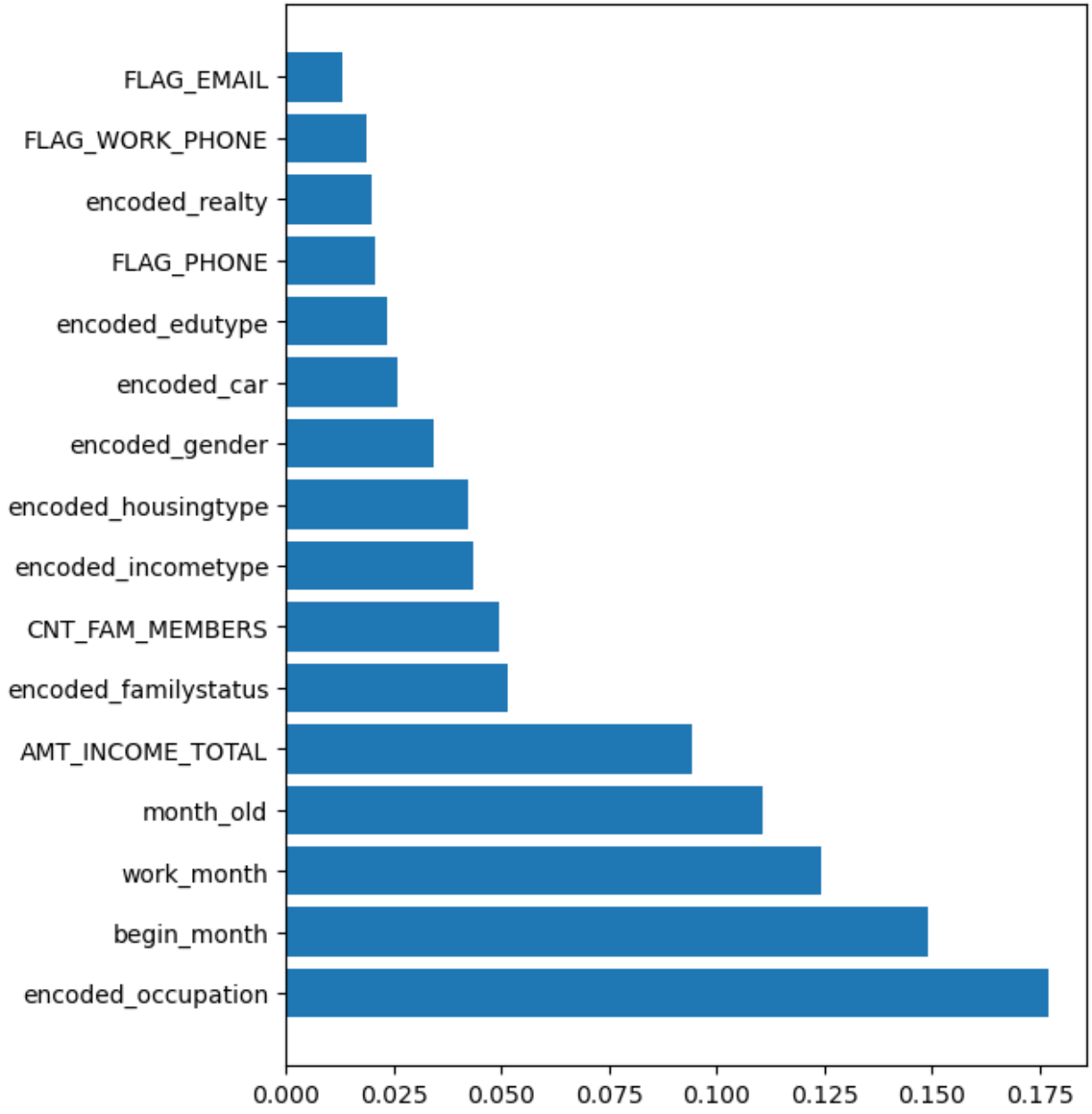


Figure 6.1

The performance of the Random Forest model outshined that of all the models. The accuracy is raised to 68.4%, and the recall rate is 68.3%. The feature importance scores show that occupation, years of being a customer, age, employment years, and income are the top five most important features. Earlier in our exploratory analysis, we found that these features do have an impact on classifying whether the applicants are “good” or “bad”. Note that even the top five most important features’ scores lie between 0.1 to 0.15, which would explain why the accuracy is not very high.

6.5 Gradient Boosted Trees

Gradient Boosted Trees model is another type of tree-based machine learning model that can be used for our classification task. The model works by combining the predictions of multiple "weak" learners, typically decision trees, into a single "strong" model. The individual trees are trained sequentially, with each tree being trained to correct the errors made by the previous trees. This sequential training process allows gradient boosted trees to capture complex patterns in the data that may be missed by individual trees.

Table 6.5

	AUC-ROC	AUC-PR	Accuracy	Precision	Recall	F1
Scores	0.664	0.597	0.647	0.621	0.651	0.636

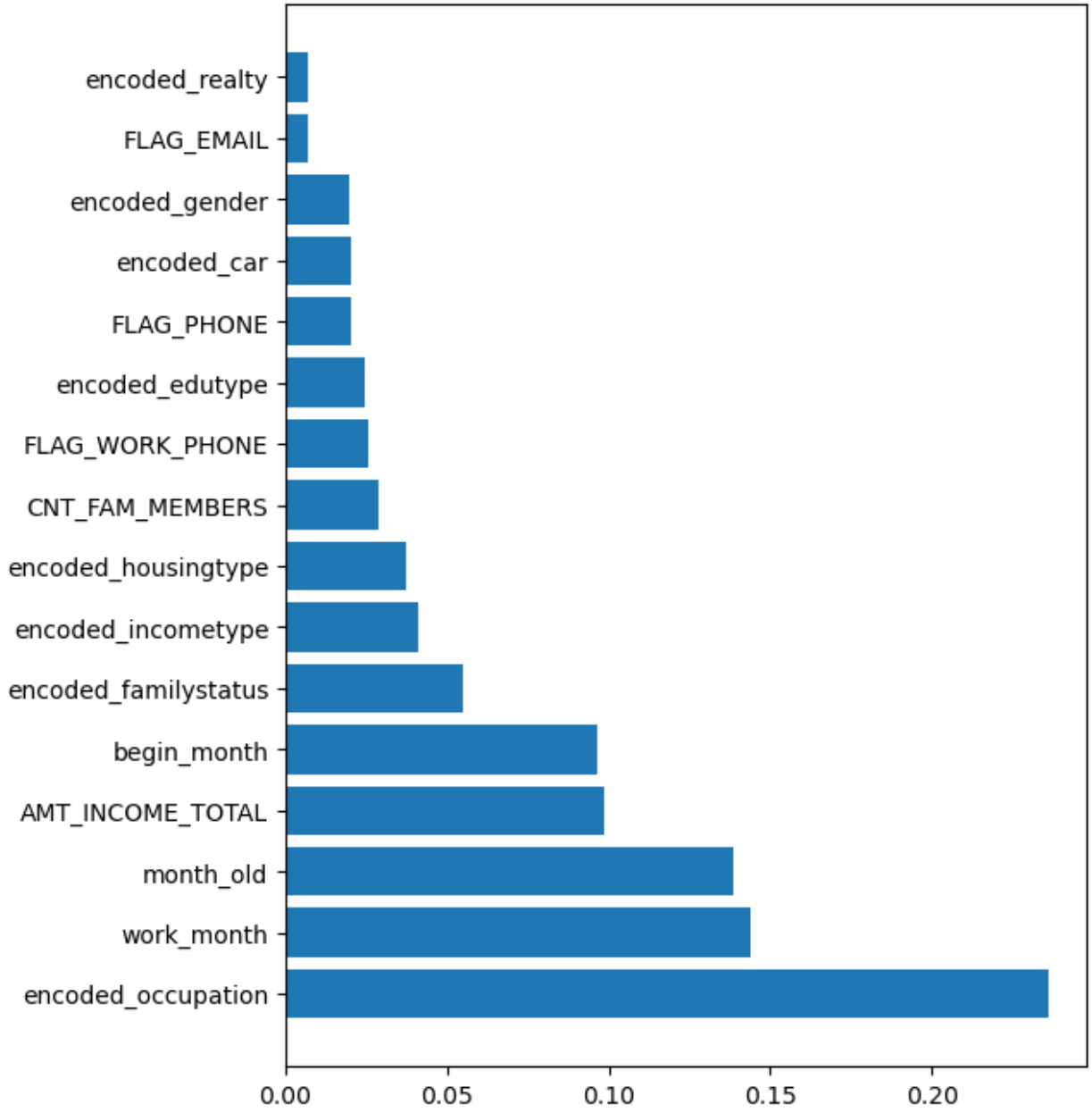


Figure 6.2

The overall performance of the Gradient Boosted Trees model is the second best among all of the models. One drawback of the GBT model is that training such a model is very time-consuming. If we have enough computing resources, we would expect the GBT model would have slightly better performance than that of the RF model. If we look at the feature importance graph, we found that the top six most important features for the GBT model remain the same compared to that of the RF model. The GBT model seems to be good at emphasizing more on the most important features and making the less relevant features less important.

7. CONCLUSION

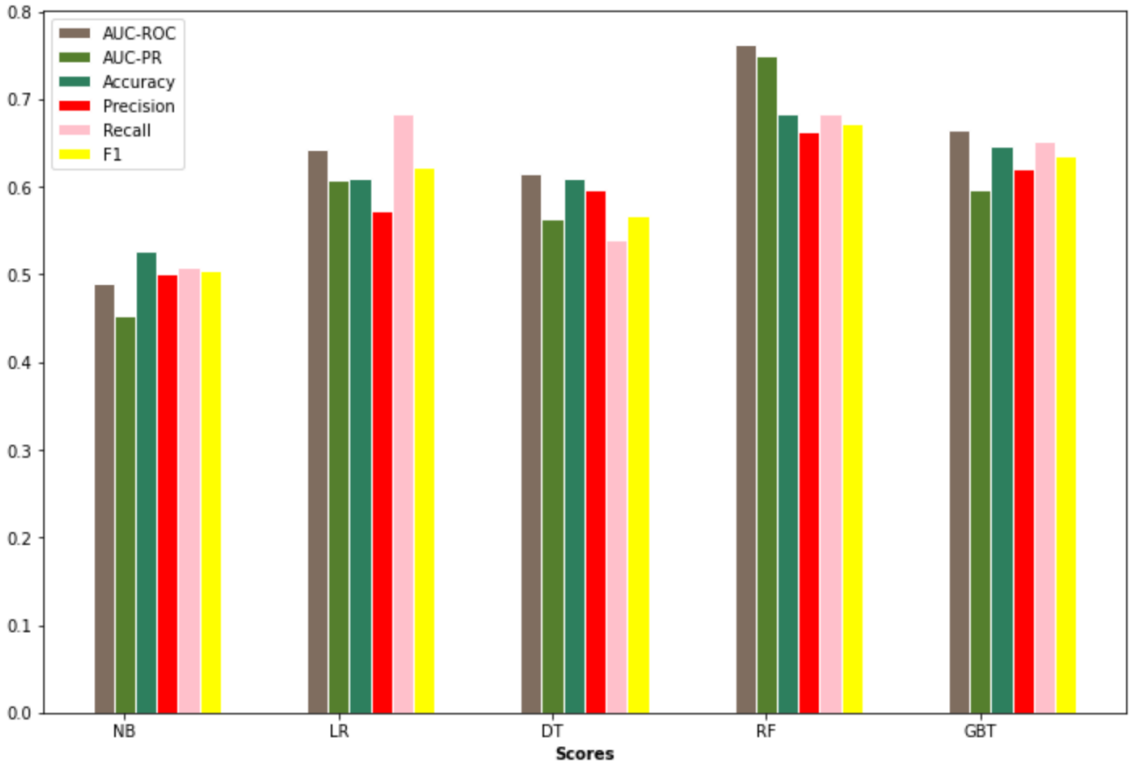


Figure 7.1

Using the combination of grid search and three-fold cross validation, figure 7.1 is a summary plot of our model outputs. Using only applicants' basic information is difficult to predict whether they are good or not. The Random Forest, GBT, Logistic Regression, and Decision Tree models manage to have greater than 60% accuracy. Since higher recall rate is preferred, most of our models did give us a decent recall rate. The GBT model can still be improved if we broaden the grid search range since it relies on building many weak learners to gradually correct errors, but training a GBT model would consume a very long period of time. The Random Forest model has the highest overall performance because it works with multiple decision trees so that it has a smaller chance to overfit.

`rf_feature_importance`

	features	importance
14	encoded_occupation	0.177128
15	begin_month	0.149239
10	work_month	0.124427
9	month_old	0.110862
3	AMT_INCOME_TOTAL	0.094414
6	encoded_familystatus	0.051497
8	CNT_FAM_MEMBERS	0.049625
4	encoded_incometype	0.043402
7	encoded_housingtype	0.042237
0	encoded_gender	0.034213
1	encoded_car	0.026134
5	encoded_edutype	0.023736
12	FLAG_PHONE	0.020864
2	encoded_realty	0.020010
11	FLAG_WORK_PHONE	0.018925
13	FLAG_EMAIL	0.013290

`dt_feature_importance`

	features	importance
3	AMT_INCOME_TOTAL	0.154662
14	encoded_occupation	0.135235
15	begin_month	0.129085
10	work_month	0.122388
9	month_old	0.096566
8	CNT_FAM_MEMBERS	0.057001
6	encoded_familystatus	0.054227
0	encoded_gender	0.049248
2	encoded_realty	0.040928
5	encoded_edutype	0.039791
4	encoded_incometype	0.035450
7	encoded_housingtype	0.025630
1	encoded_car	0.021030
13	FLAG_EMAIL	0.020332
12	FLAG_PHONE	0.010767
11	FLAG_WORK_PHONE	0.007659

`gbt_feature_importance`

	features	importance
14	encoded_occupation	0.236124
10	work_month	0.144095
9	month_old	0.138700
3	AMT_INCOME_TOTAL	0.098718
15	begin_month	0.096506
6	encoded_familystatus	0.054707
4	encoded_incometype	0.040852
7	encoded_housingtype	0.037330
8	CNT_FAM_MEMBERS	0.028929
11	FLAG_WORK_PHONE	0.025542
5	encoded_edutype	0.024370
12	FLAG_PHONE	0.020451
1	encoded_car	0.020366
0	encoded_gender	0.019704
13	FLAG_EMAIL	0.006960
2	encoded_realty	0.006645

Figure 7.2

For most part, tree-based models performed well in our classification task. Figure 7.2 above is the inference summary of our tree-based models. Some of the most important features for predicting customer types are occupation types, income, age, employment years, and how long one has been a customer. Even though these features are important among all of the features, their importance score isn't high. Hence, constructing a model with high accuracy is challenging.

To sum up, our goal is to build a model to assist a bank or finance company save time on classifying customers, and the Random Forest model and the Gradient Boosted Trees model are doing a decent job classifying credit card applicants. A bank or financial company can use its human resources more efficiently by using predictive models to predict which customers are more likely to be good customers. When performing customer screening, the model can predict the credit risk of customers based on their basic information, such as age, income, and credit history. Models with good recalls can help banks or financial companies quickly screen out the most credible customers. However, it isn't easy to achieve accurate predictions if you only rely on customers' basic information. We can improve our models by performing better hyperparameter tuning, dividing applicants into more classes, and better defining applicants types. For the time being, manual inspection is still required to ensure accuracy and reliability while using the model to screen customers. Still, current models can partially replace manual review, as they may miss some important information or have bias and error.

8. APPENDIX

Data Sources:

https://www.kaggle.com/datasets/rikdifos/credit-card-approval-prediction/code?select=application_record.csv

Models:

https://spark.apache.org/docs/latest/