Credit Card Application Approval - Case Study

PS: There are two notebooks attached with this document.

1. **Notebook run in Google Collab**
2. **Notebook run in EC2**

Overview

Initial Setup

1. Import necessary modules
2. Start Spark Session with required configurations

Data Ingestion

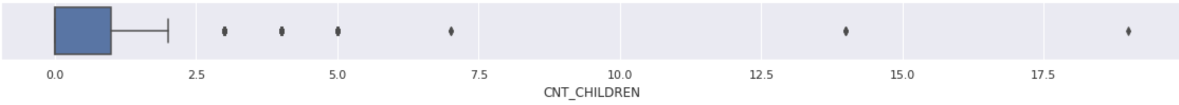
1. Imported credit record and application record dataset from S3.
2. Created Spark dataframes for each of the above datasets.
3. Checked the data structure.

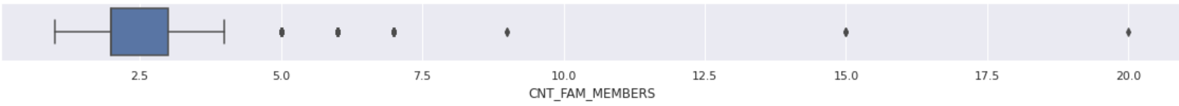
Feature Engineering

4. Added a `MONTHS_ON_RECORD` column signifying the number of months the customer has on record.
5. Added a `GOOD_BAD_DEBT` column signifying if a month's status is considered as Good or Bad Debt.
6. Dropping redundant columns like `MONTHS_BALANCE`, `STATUS`, `GOOD_BAD_DEBT` from credit record df.
7. Creating a new column on the application_record_df and marking all delinquent IDs from the credit_record_df. Any IDs that are not present in the credit_record_df are considered as non-delinquent customers.
8. Merge credit and application records. For all customers not present in the credit record, mark `MONTHS_ON_RECORD` as 0.
9. Creating a new column, `DELINQUENT`, on the application_record_df and marking all delinquent IDs from the credit_record_df. Any IDs that are not present in the credit_record_df are considered as non-delinquent customers.
10. Splitting the entire application to existing and new customers. - Since there are a lot of applicants without a credit history record, it would be better to split applications based on credit history. We will model our data based on the subset of customers with credit history and if needed later we can use the same model to predict application approval status for new customers.

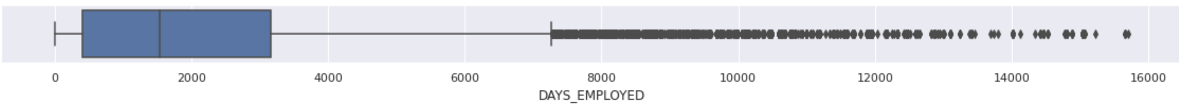
Exploratory Data Analysis

1. Dealing with missing values.
2. Only column OCCUPATION_TYPE has null values. Imputing the null values in this column to be "Not identified".
3. Creating 'Pensioner' occupation type for all pensioners, substantially reducing 'Not identified' group
4. Dealing with outliers for CNT_CHILDREN & CNT_FAM_MEMBERS

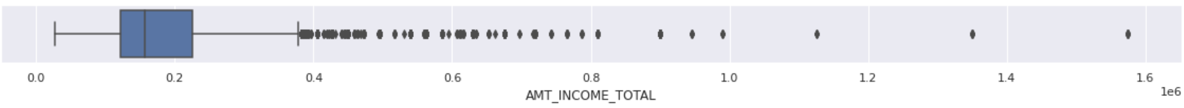




5. Dealing with outliers in NAME_INCOME_TYPE, NAME_EDUCATION_TYPE
6. Converting DAYS_EMPLOYED to be absolute
7. Dealing with outliers in DAYS_EMPLOYED



8. Creating buckets for DAYS_EMPLOYED
9. Convert negative values to absolute and bin the AGE column.
10. Creating buckets for MONTHS_ON_RECORD
11. Outlier clean up for AMT_INCOME_TOTAL

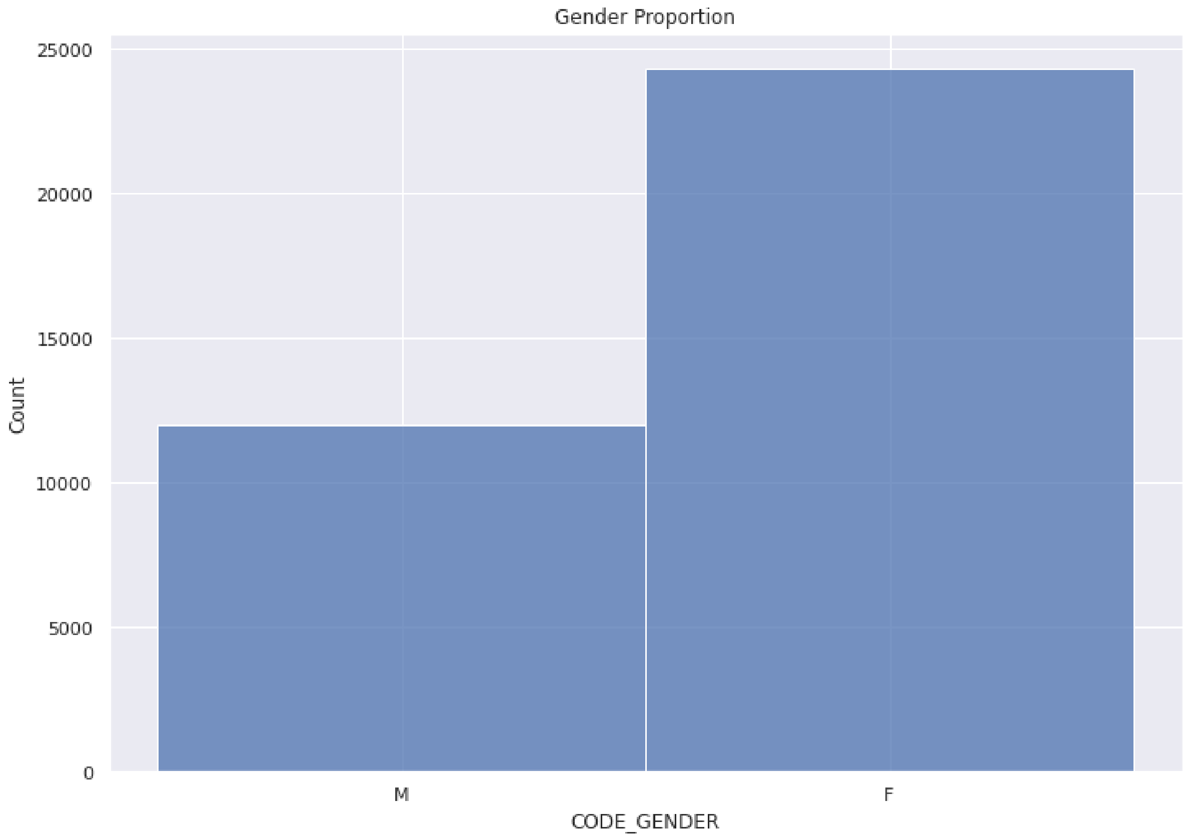


12. Creating buckets for AMT_INCOME_TOTAL
    a. Choosing the number of buckets to 7 such that all buckets have roughly the same number of elements and each bucket has at least more than 5% of the total number of data.
13. Dropping redundant columns
    a. DAYS_BIRTH - We have created an AGE column as a substitute.
    b. FLAG_MOBIL - There is only value across the data.

Questions

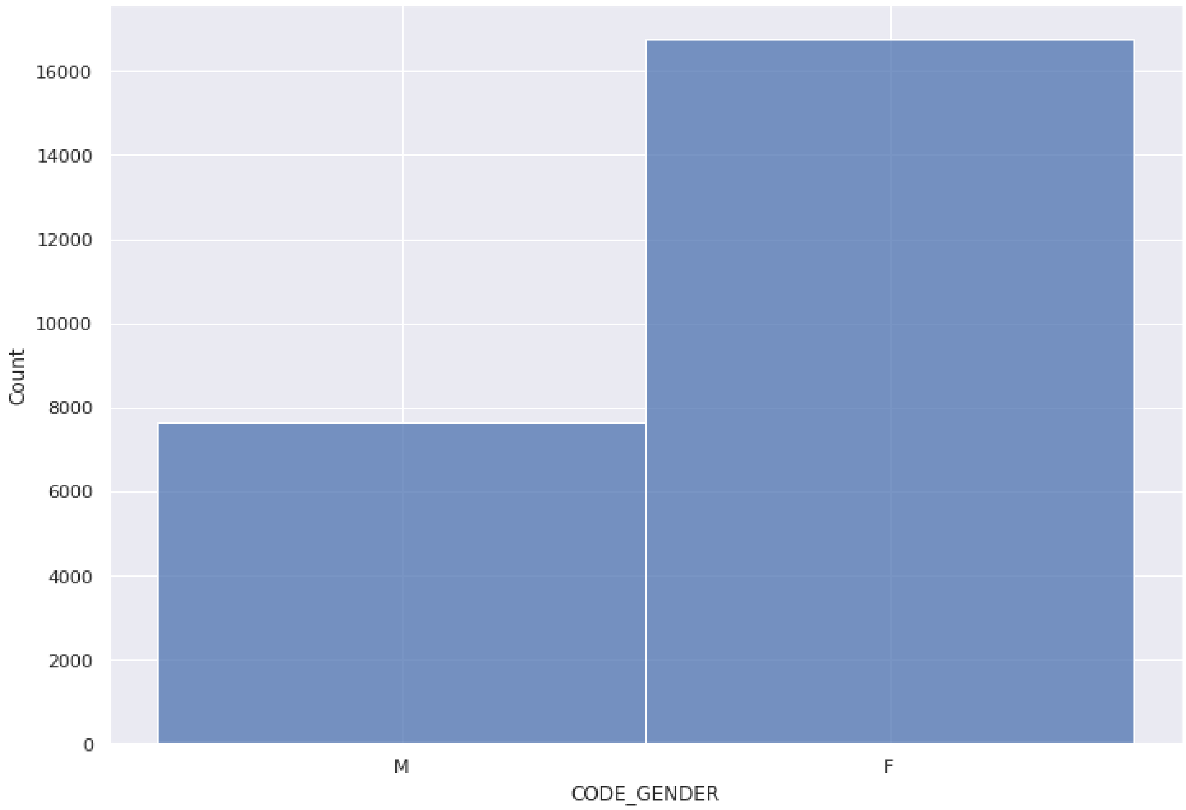
1. What is the proportion of females in the applicant customer base?

```
Percentage of Female applicants: 66.99%
+-----------+-----+
|CODE_GENDER|count|
+-----------+-----+
|          F|24327|
|          M|11985|
+-----------+-----+
```



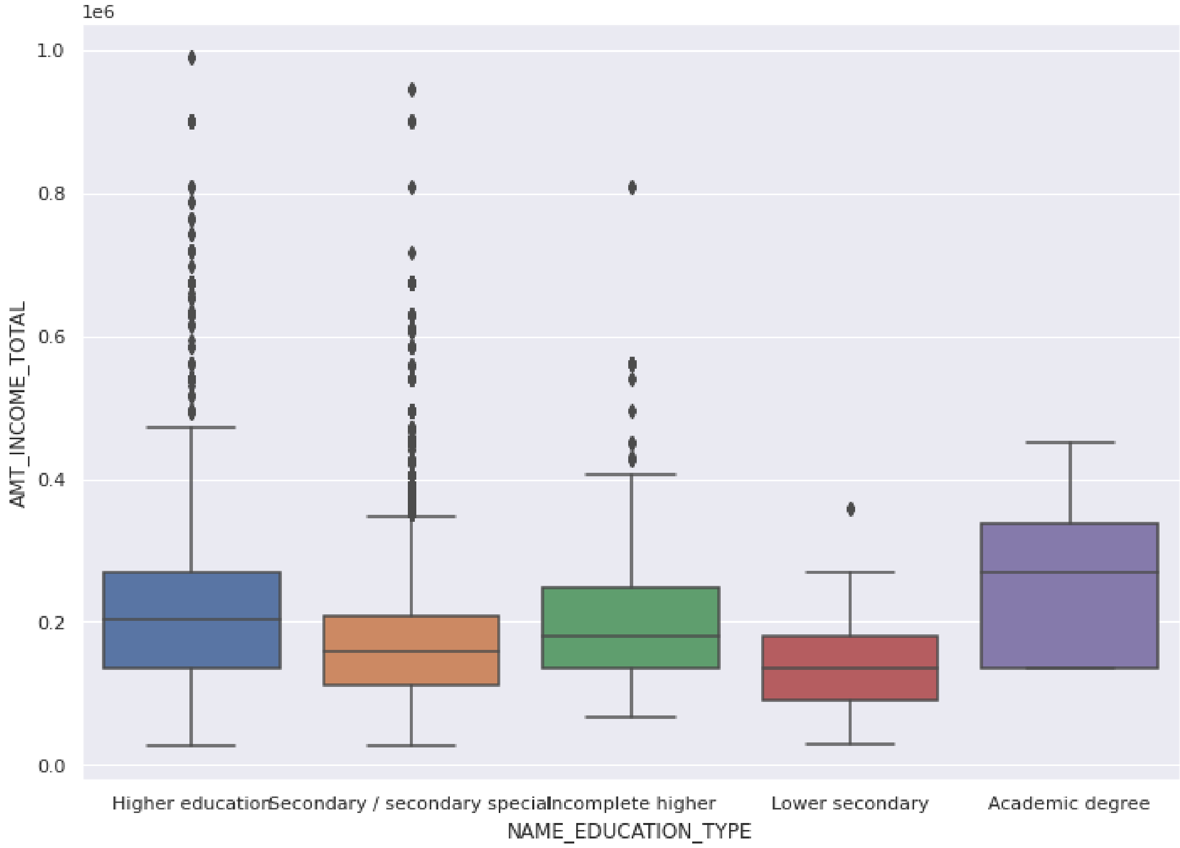
2. Is homeownership higher among male applicants or female applicants?

```
+-----------+---------------+-----+
|CODE_GENDER|FLAG_OWN_REALTY|count|
+-----------+---------------+-----+
|          F|              Y|16748|
|          M|              Y| 7652|
+-----------+---------------+-----+
```



3. Is there any correlation between the customer's income level and education level?

```
+------------------------------+-----------------------+
|NAME_EDUCATION_TYPE           |AMT_INCOME_TOTAL_MEDIAN|
+------------------------------+-----------------------+
|Academic degree               |270000.0               |
|Incomplete higher             |180000.0               |
|Secondary / secondary special|157500.0               |
|Lower secondary               |135000.0               |
|Higher education              |202500.0               |
+------------------------------+-----------------------+
```



4. What is the average and median salary of the applicant base?

```
+---------------------+-----------------------+
|AMT_INCOME_TOTAL_MEAN|AMT_INCOME_TOTAL_MEDIAN|
+---------------------+-----------------------+
|186130.8493473232    |157500.0               |
+---------------------+-----------------------+
```

5. Is the proportion of bad customers higher for people who own cars?

```
+------------+----------+-----+
|FLAG_OWN_CAR|DELINQUENT|count|
+------------+----------+-----+
|           Y|         1|  224|
|           Y|         0|13517|
+------------+----------+-----+
```

6. Is the proportion of bad customers higher for those living on rent than the rest of the population?

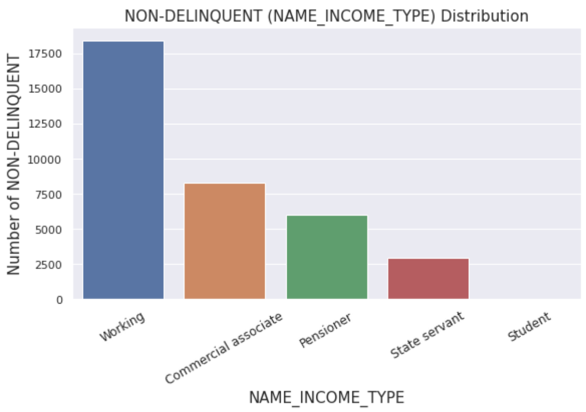
```
+-------------------+-----+------------------+
|  NAME_HOUSING_TYPE|count|    proportion_pct|
+-------------------+-----+------------------+
|  House / apartment|  539| 87.64227642276423|
|Municipal apartment|   30| 4.878048780487805|
|    Co-op apartment|    3|0.4878048780487805|
|   Rented apartment|    8|1.3008130081300813|
|   Office apartment|    9|1.4634146341463414|
```

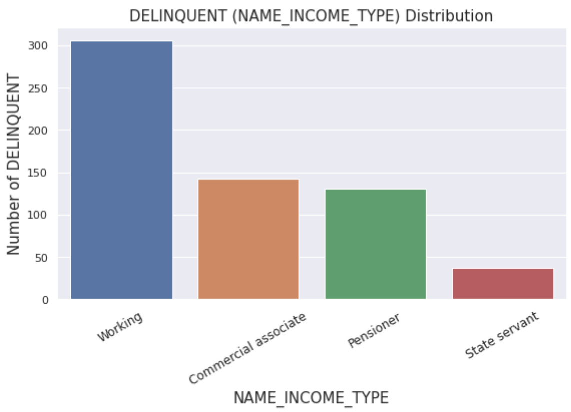
With parents	26	4.227642276422764

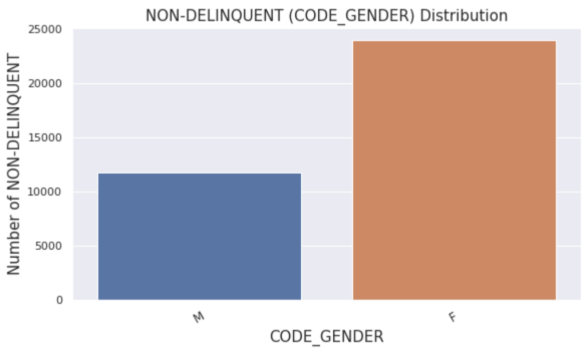
7. Is the proportion of bad customers higher for those who are single than married customers?

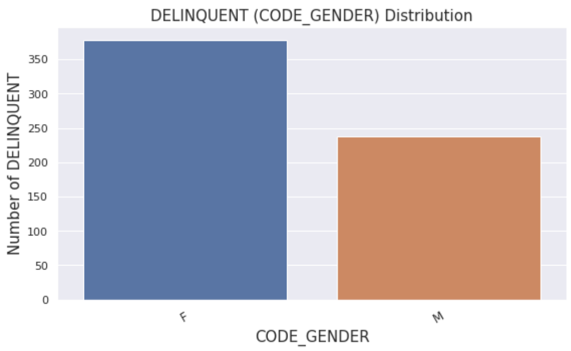
NAME_FAMILY_STATUS	count	proportion_pct
Separated	31	5.040650406504065
Married	392	63.739837398373986
Single / not married	101	16.422764227642276
Widow	45	7.317073170731708
Civil marriage	46	7.479674796747967

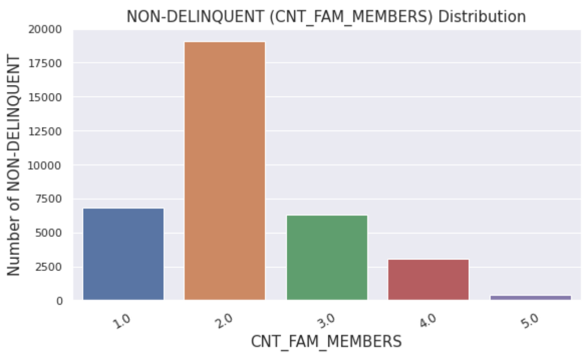
8. Plotted different categorical columns against the target column, DELINQUENT

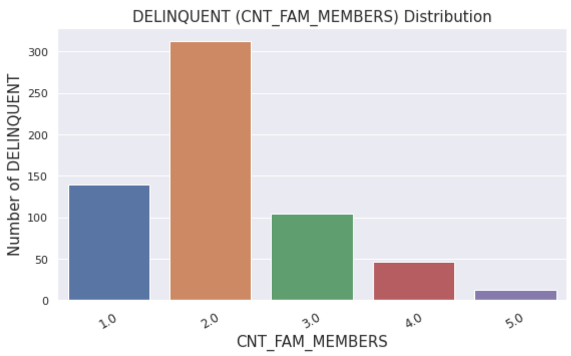


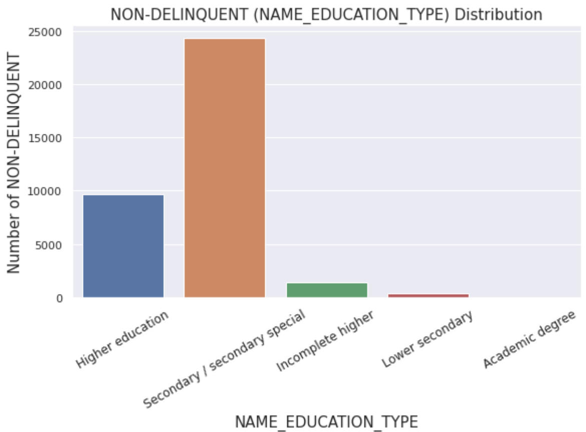


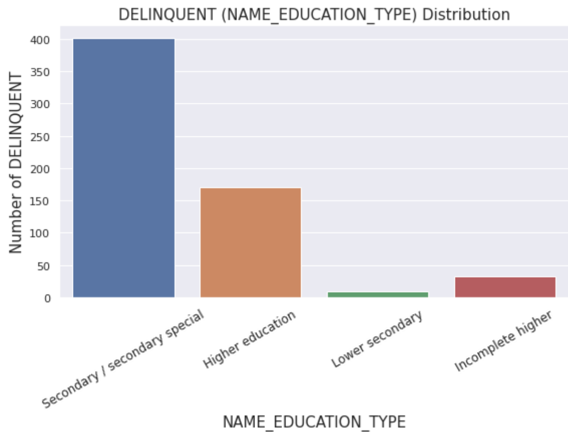


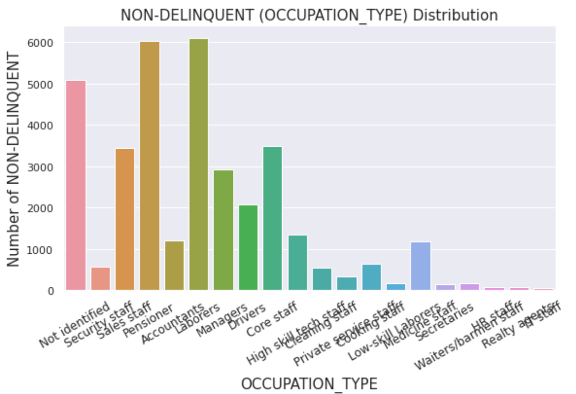


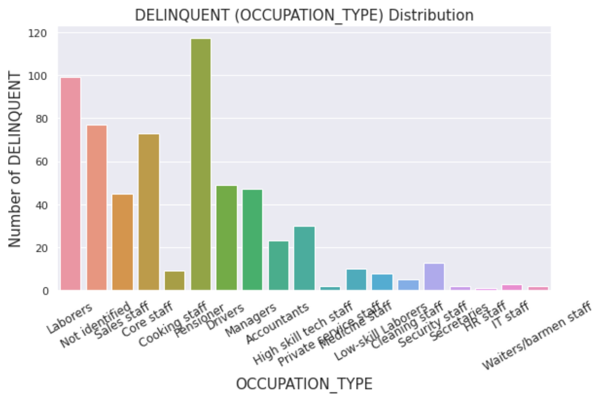


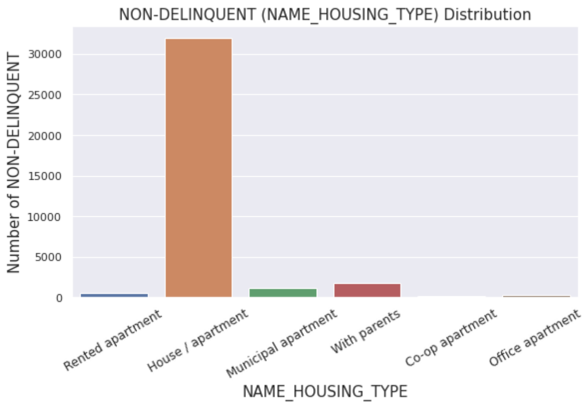


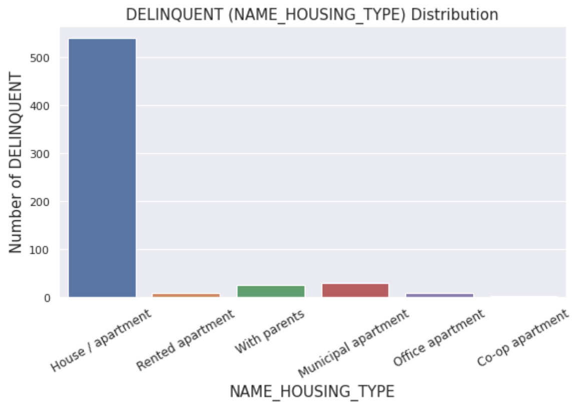

NON-DELINQUENT (NAME_EDUCATION_TYPE) Distribution
Number of NON-DELINQUENT
0
5000
10000
15000
20000
25000
Higher education
Secondary / secondary special
Incomplete higher
Lower secondary
Academic degree
NAME_EDUCATION_TYPE

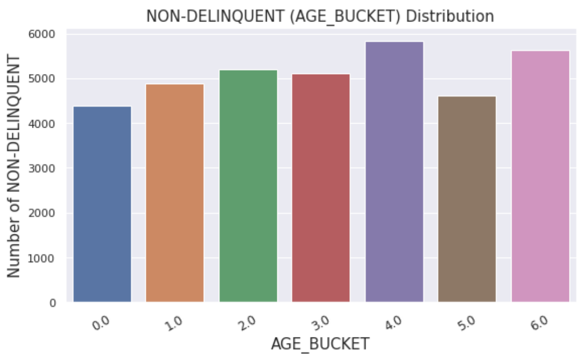

DELINQUENT (NAME_EDUCATION_TYPE) Distribution
Number of DELINQUENT
0
50
100
150
200
250
300
350
400
Secondary / secondary special
Higher education
Lower secondary
Incomplete higher
NAME_EDUCATION_TYPE

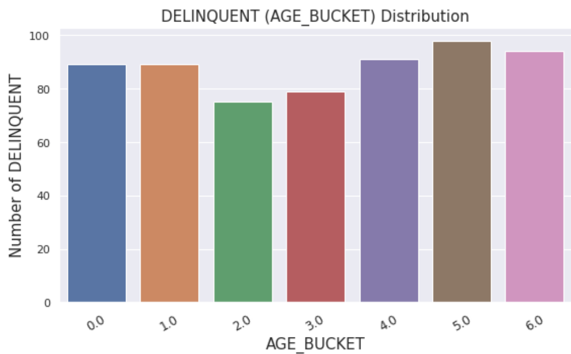

NON-DELINQUENT (OCCUPATION_TYPE) Distribution
Number of NON-DELINQUENT
0
1000
2000
3000
4000
5000
6000
Not identified
Security staff
Sales staff
Pensioner
Accountants
Laborers
Managers
Drivers
Core staff
High skill tech staff
Cleaning staff
Private service staff
Cooking staff
Low-skill Laborers
Medicine staff
Secretaries
Waiters/barmen staff
HR staff
Realty agents
IT staff
OCCUPATION_TYPE

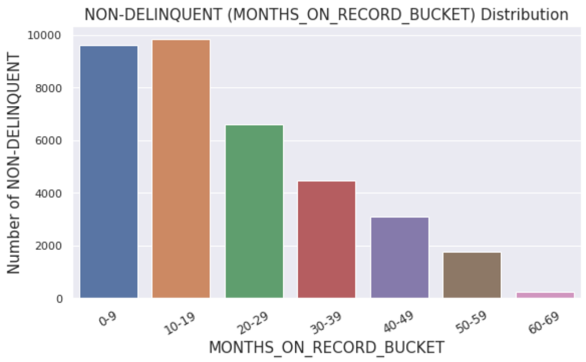

DELINQUENT (OCCUPATION_TYPE) Distribution
Number of DELINQUENT
0
20
40
60
80
100
120
Laborers
Not identified
Sales staff
Core staff
Cooking staff
Pensioner
Drivers
Managers
Accountants
High skill tech staff
Private service staff
Medicine staff
Low-skill Laborers
Cleaning staff
Security staff
Secretaries
HR staff
IT staff
Waiters/barmen staff
OCCUPATION_TYPE

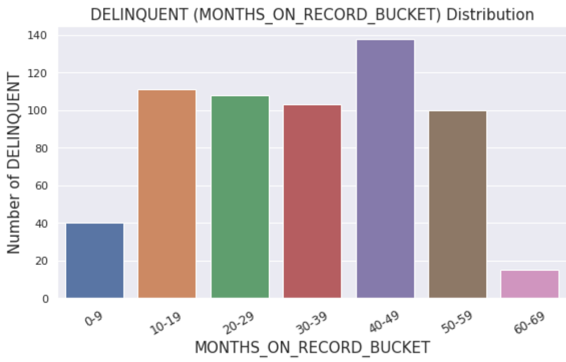

NON-DELINQUENT (NAME_HOUSING_TYPE) Distribution
Number of NON-DELINQUENT
0
5000
10000
15000
20000
25000
30000
Rented apartment
House / apartment
Municipal apartment
With parents
Co-op apartment
Office apartment
NAME_HOUSING_TYPE


DELINQUENT (NAME_HOUSING_TYPE) Distribution
Number of DELINQUENT
0
100
200
300
400
500
House / apartment
Rented apartment
With parents
Municipal apartment
Office apartment
Co-op apartment
NAME_HOUSING_TYPE


NON-DELINQUENT (AGE_BUCKET) Distribution
Number of NON-DELINQUENT
0
1000
2000
3000
4000
5000
6000
0.0
1.0
2.0
3.0
4.0
5.0
6.0
AGE_BUCKET


DELINQUENT (AGE_BUCKET) Distribution
Number of DELINQUENT
0
20
40
60
80
100
0.0
1.0
2.0
3.0
4.0
5.0
6.0
AGE_BUCKET





Weight of Evidence (WOE) and Information Value (IV)

1. Calculated the WOE and IV for all discretized columns.
2. Using IV and the threshold of 0.002 found the insignificant columns
    a. FLAG_OWN_CAR
    b. FLAG_WORK_PHONE
    c. FLAG_PHONE
    d. FLAG_EMAIL
3. Dropped all significant columns
4. Dropped all discretized/bucketed columns created as we can use regular numerical columns for Modelling

Modelling

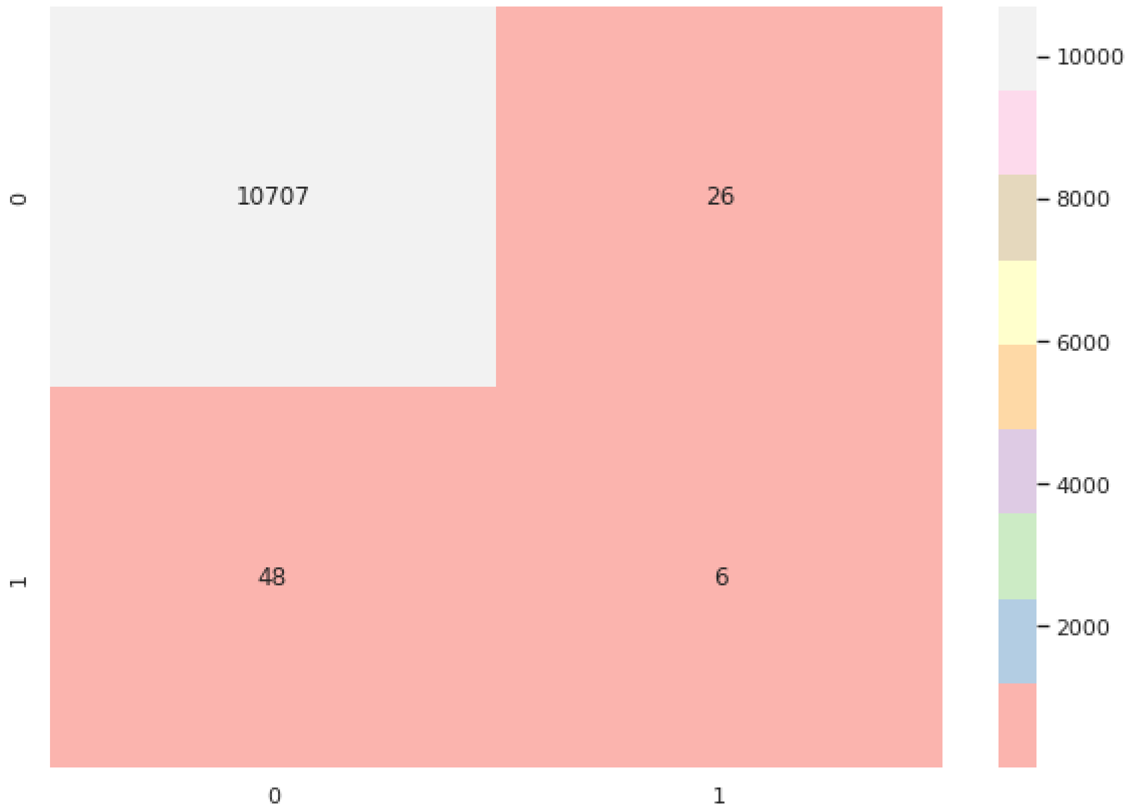
Data Estimators & Transformers

1. Created the following Estimator & Transformer stages.
    a. StringIndexer - For categorical columns with string values
    b. OneHotEncoder - For all categorical columns
    e. Vector Assembler - Convert all columns to features column of type Vector
2. Used a Pipeline Estimator to combine all the above Estimators and Transformers to modify data.
3. Since the data is heavily imbalanced, used SMOTE to bring some balance.
4. Used stratified split to split the dataset to training – 70% and test – 30% data with **random seed as 2018**
5. Ran vanilla Logistic Regression
6. Calculated Precision Recall graph
7. Used CrossValidator to find the right hyperparameters
    a. max iterations = 20
    b. regularization parameter = 0.0001
    c. threshold = 0.25
8. Ran Logistic Regression with the above parameters
9. Calculated recallByLabel, K2 stat, area under curve, precision, recall, and confusion matrix

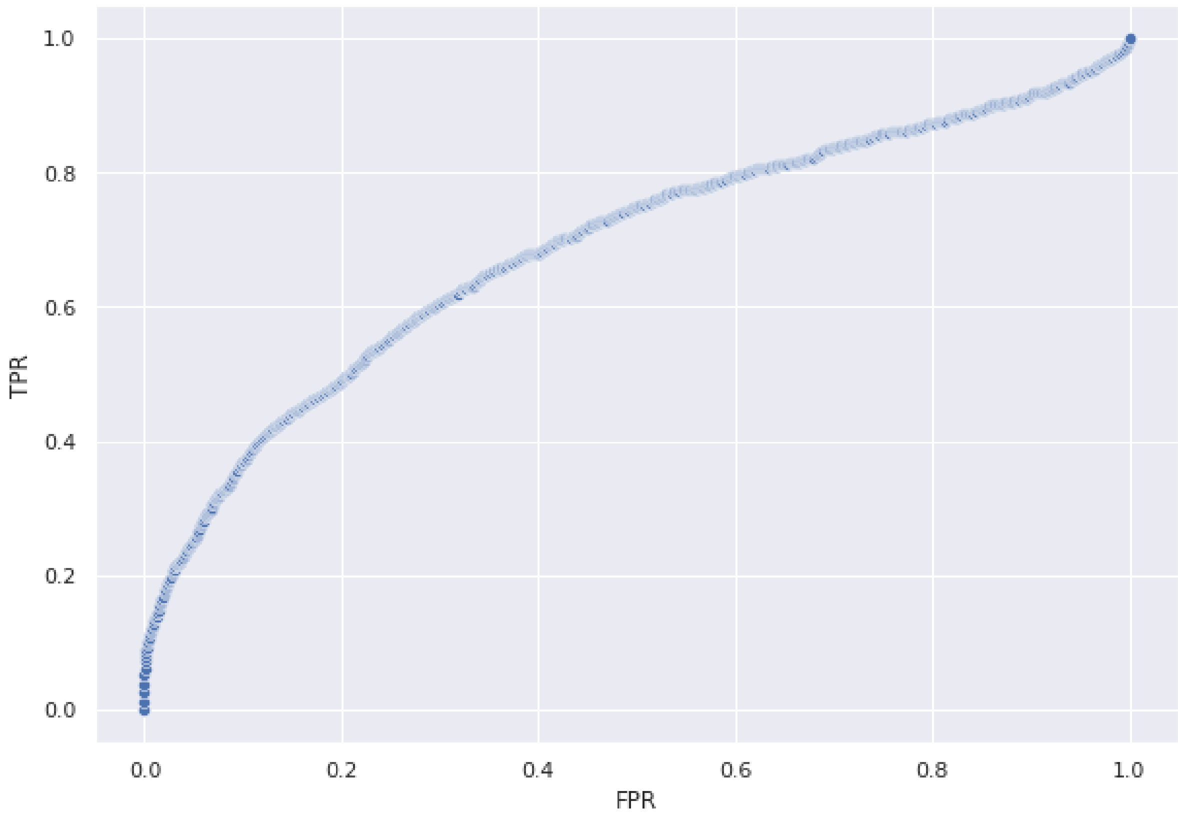
Metrics of the Final Model:

- Confusion Matrix

```
+-----------------+----------+-----+
|DELINQUENT_smoted|prediction|count|
+-----------------+----------+-----+
|                1|         0|   47|
|                1|         1|    7|
|                0|         0|10707|
|                0|         1|   26|
+-----------------+----------+-----+
```



- recallByLabel = [0.9974843939252772, 0.06521739130434782]
- Receiver Operating Characteristic (ROC)



- Area under ROC: 0.6884198236195141
- Recall/Sensitivity : 0.1111111111111111
- Precision : 0.1875
- Specificity : 0.9975775645206373
- Negative Predictive Value: 0.99553695595536959
- F1 Score : 0.13953488372093023
- Negative F1 Score : 0.9965562174236783
- KstestResult(statistic=0.9776536312849162, pvalue=5.773710670458929e-291)

The model is predicting the recall for the negative class really well but not so well for the delinquent class. This is because of the high imbalance in the dataset. Even after lowering the threshold from 0.5 to 0.3 the recall didn't improve much. Ideally this should have increased the True positives and therefore the recall. But the model isn't predicting the actual positives that well.

areaUnderROC – 0.6884198236195141

This is good. The area under the curve is more than 50%, which means that the true positive rate is higher than the false positive rate.

KS stat - 0.97765

This means that the fit of the model is good. The number of goods is much higher than the number of bads.