Credit Default Risk: Predicting Client repayment ability

Introduction

In an economy, members of society commonly seek financial aid from financial, government, and private institutions to support their investments. These investments tend to be crucial in supporting people's everyday lives as well as their futures. People take out loans for home mortgages, car ownerships, or tuition. Credit bureaus are agencies that collect people's financial data from creditors and present this information to consumer reporting agencies. Based on the information, consumer reporting agencies create credit scores and provide them to lending institutions. Ultimately, credit scores can be a measure used to approve or decline people from obtaining loans. Unfortunately, many people struggle to obtain loans from said lending institutions due to insufficient or non-existing credit histories. In turn, people reach out to untrustworthy lenders where they are often mistreated and get taken advantage of. Are members of the unbanked population capable of repaying loans? institutions like Home Credit focus on financial inclusion and responsible lending by using a variety of alternative data to predict their client's repayment abilities and empower them to be successful. Classification methods will be used to predict a client's likelihood of repayment.

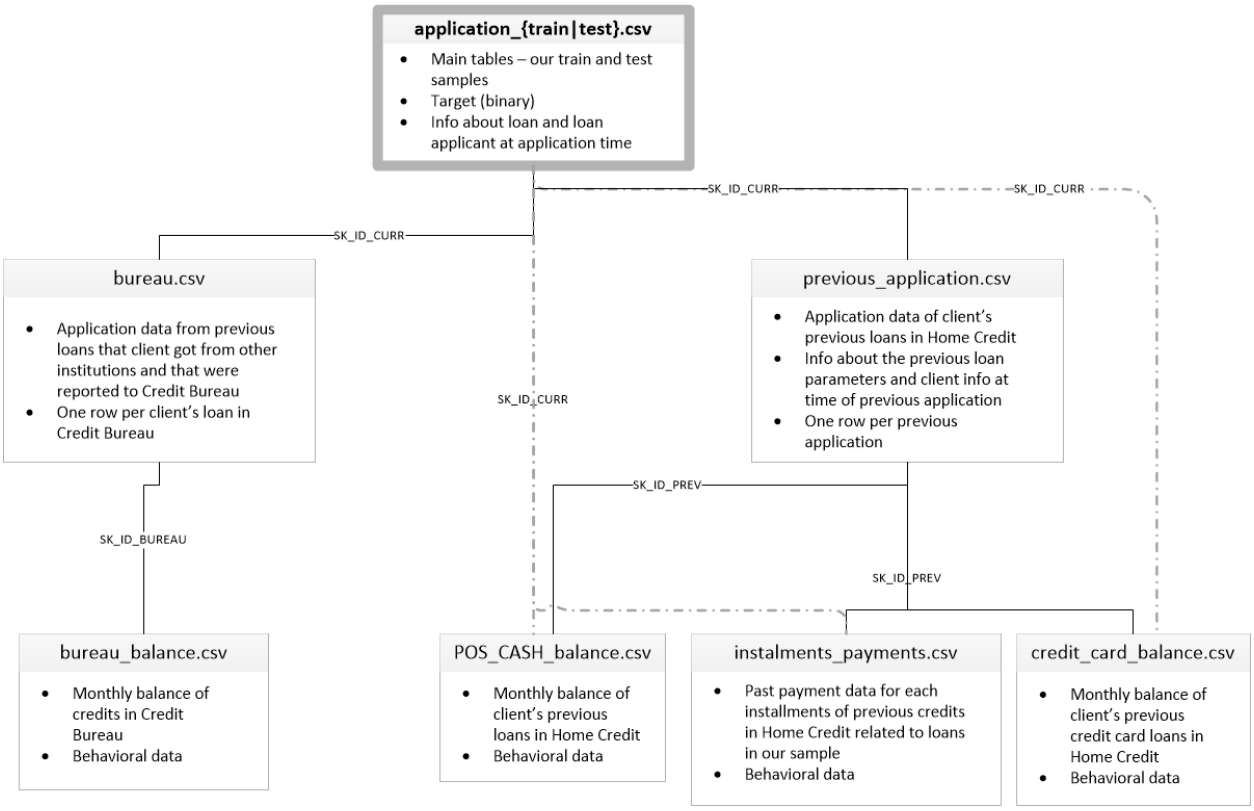
Literature Review

The importance of information sharing between borrowers and lenders is elemental to projects like these. Brown et al. argues that information sharing is key in increasing repayment as borrowers are incentivized to maintain a good credit record to increase their credit availability. The type of data shared ultimately influences how we extract powerful features. Empirical study conducted by Kao et al. demonstrates that demographic variables used in credit scoring models have little correlation to borrower's repayment behavior and that credit history information is worth the most importance. Other types data used in peer-peer lending are low- and high-level semantic information. Kim et al. scrap borrower's social network data from platforms like Twitter and Facebook. Mathematical, data mining, and statistical models are created to determine how likely loans are to be repaid. Wijewardhana made use of more traditional models like logistic regression, Artificial Neural Network (ANN), and affinity analysis. Deep sense convolutional networks were used in Kim et al. study to automatically extract important features while representing semantic borrower information. Novel advanced models are exciting however, they also need to be benchmarked. Lessmann et al. compares novel classification algorithms and provides a new baseline for which future approaches ca be compared to.

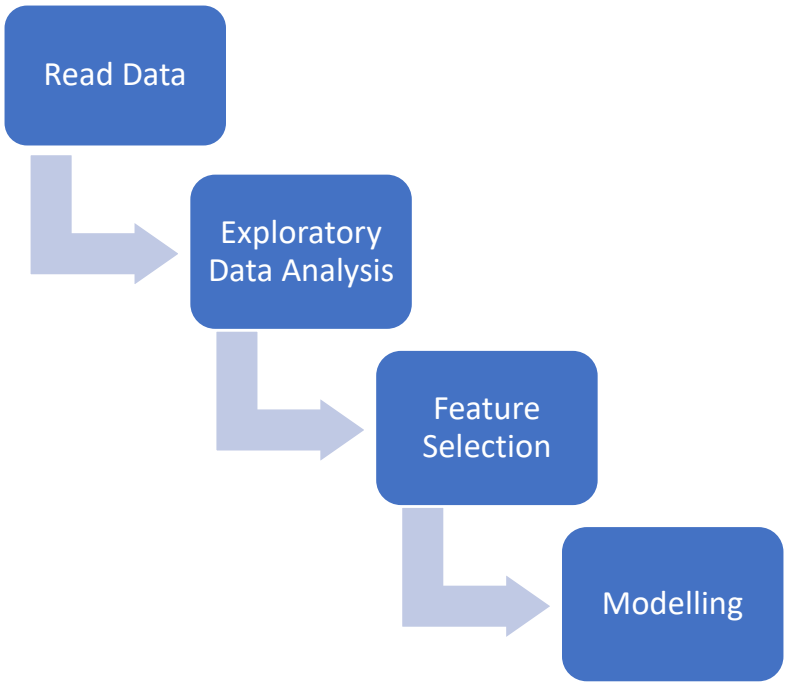
Dataset

The dataset used contains anonymized client loan applications. The primary table contains client information related to their properties (car, real estate), family, annual income, education, employment,

credit bureau inquiries, and normalized scores from external data sources. Loans are identified with primary key “SK_ID_CURR”. The primary table is already split into train and test data with “TARGET” variable (1 = payment difficulty, 0 = payment) . The primary table contains 122 attributes for ≈ 300,000 loans (train) and ≈ 50,000 loans (test). Additional tables are available to use: previous credits provided by other financial institutions, monthly balances of previous credits, point of sales and cash loan applicants have had with Home Credit, monthly balance snapshots of previous credit cards the applicant has with Home Credit, all previous applications for Home Credit loans, and repayment history for previously disbursed credits in Home Credit related to loans. The additional tables can be joined on load ID “SK_ID_CURR” or “SK_ID_PREV” (for past loans).



Approach



Step 1: Read Data

- Identify available data files – 9 files: primary training including target variable, primary testing without target variable, 6 additional tables, and example submission

Step 2: Exploratory Data Analysis

- Analyze target variable and distribution
- Analyze missing values
- Identify and fix anomalies in the data
- Labeling and One hot encoding categorical values
- Identify and fix anomalies in the data

- Impute missing data

Step 3: Feature Selection

- Sample data for feature selection
- Filter: Pearson Correlation
- Wrapper: Recursive Feature Elimination
- Embedded: Random Forest
- Choose most significant features across the three feature selection methods for modelling

Step 4: Modelling

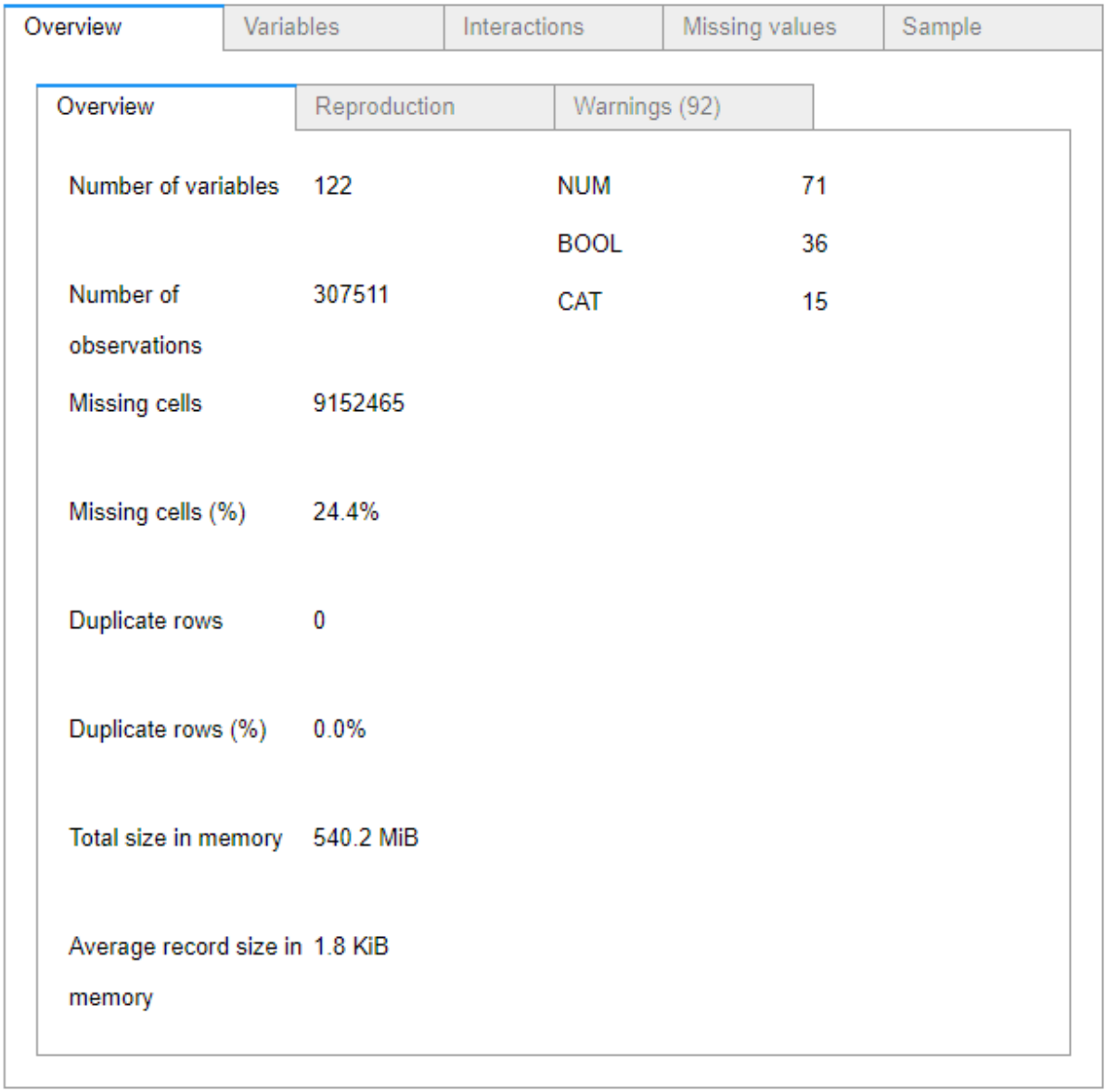
- Balance data using SMOTE
- Split data into Train and Test
- Classifier: Logistic Regression
    - Fit model to training data 10 fold
    - Model stability
    - Apply evaluation metrics
    - Classification on testing data
    - Apply evaluation metrics
- Classifier: Random Forest
    - Fit model to training data
    - Model stability
    - Apply evaluation metrics
    - Classification on testing data
    - Apply evaluation metrics
- Classifier: Light GBM
    - Fit model to training data 10 fold
    - Model stability
    - Apply evaluation metrics
    - Classification on testing data
    - Apply evaluation metrics
- Automated Hyperparameter Tuning

Results

Exploratory Data Analysis

The powerful package *pandas_profiling* was used to create a report that gave general information of the dataset including warnings for missing data, zeros and high cardinality. The report also provides detailed properties of each variable, visualization of distribution of each variable. Through inspection, observations of anomalies were made

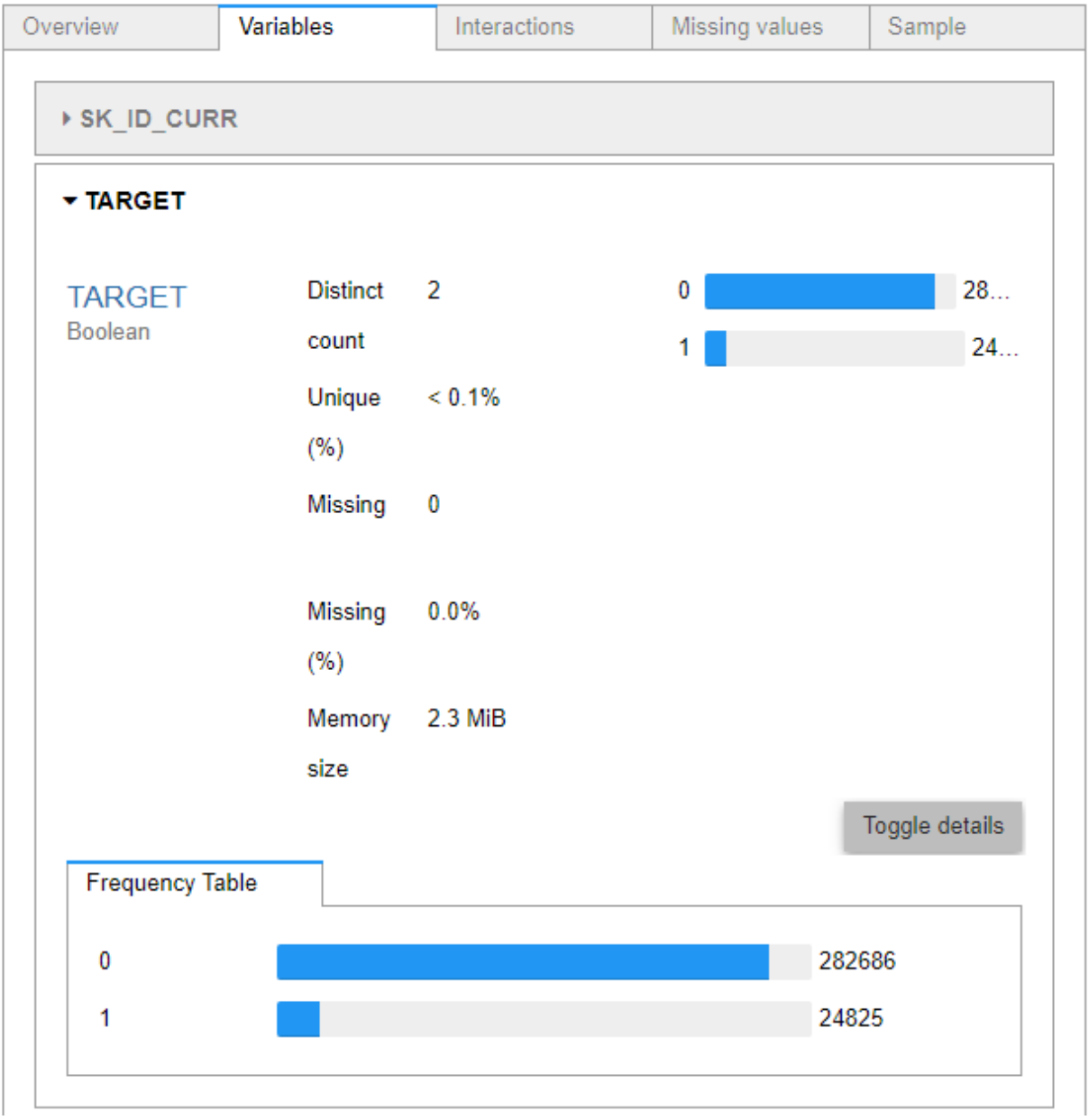
Below are few samples of generated report: (can be explored more in person via html knit or jnb)



Overview | Variables | Interactions | Missing values | Sample

Overview | Reproduction | Warnings (92)

Number of variables	122	NUM	71
		BOOL	36
Number of observations	307511	CAT	15
Missing cells	9152465		
Missing cells (%)	24.4%		
Duplicate rows	0		
Duplicate rows (%)	0.0%		
Total size in memory	540.2 MiB		
Average record size in memory	1.8 KiB		


Overview
Variables
Interactions
Missing values
Sample
SK_ID_CURR
TARGET
TARGET
Boolean
Distinct count 2
Unique (%) < 0.1%
Missing 0
Missing (%) 0.0%
Memory size 2.3 MiB
0 28...
1 24...
Toggle details
Frequency Table
0 282686
1 24825

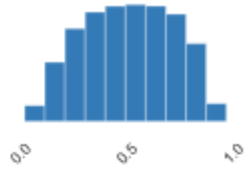
EXT_SOURCE_1

EXT_SOURCE_1

Real number ($\mathbb{R}_{\geq 0}$)

`MISSING`

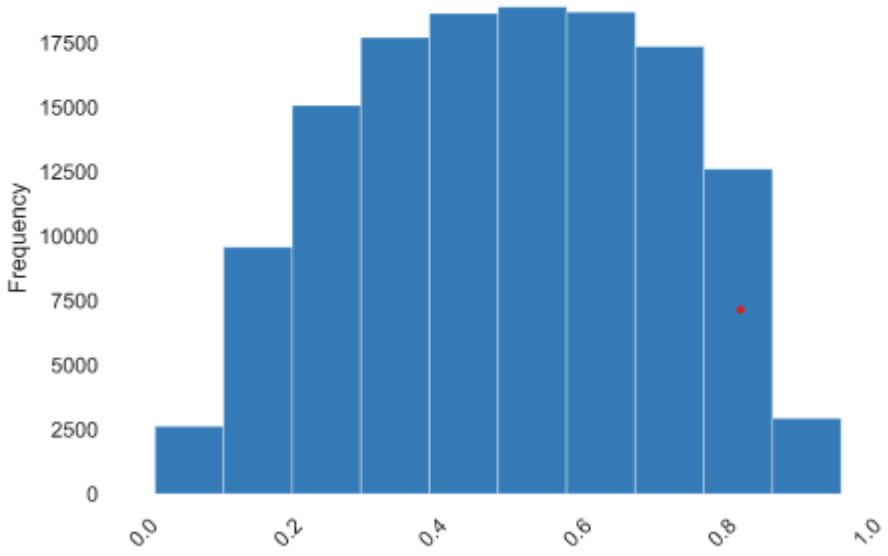
Distinct count	114584	Mean	0.5021298057
Unique (%)	85.4%	Minimum	0.01456813241
Missing	173378	Maximum	0.9626927706
Missing (%)	56.4%	Zeros	0
Infinite	0	Zeros (%)	0.0%
Infinite (%)	0.0%	Memory size	2.3 MiB



Toggle details

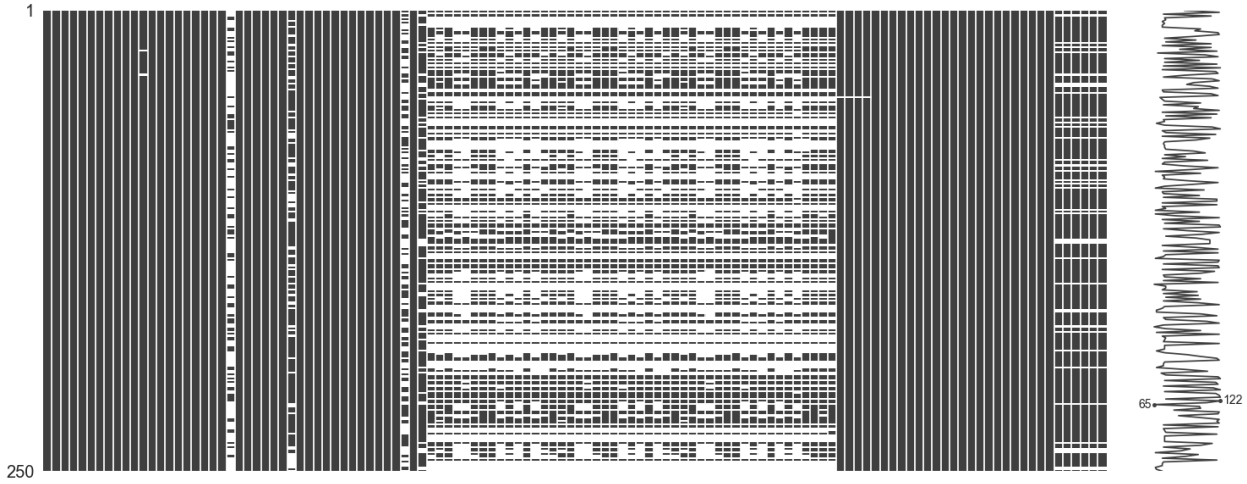
Statistics | Histogram(s) | Common values | Extreme values

Histogram

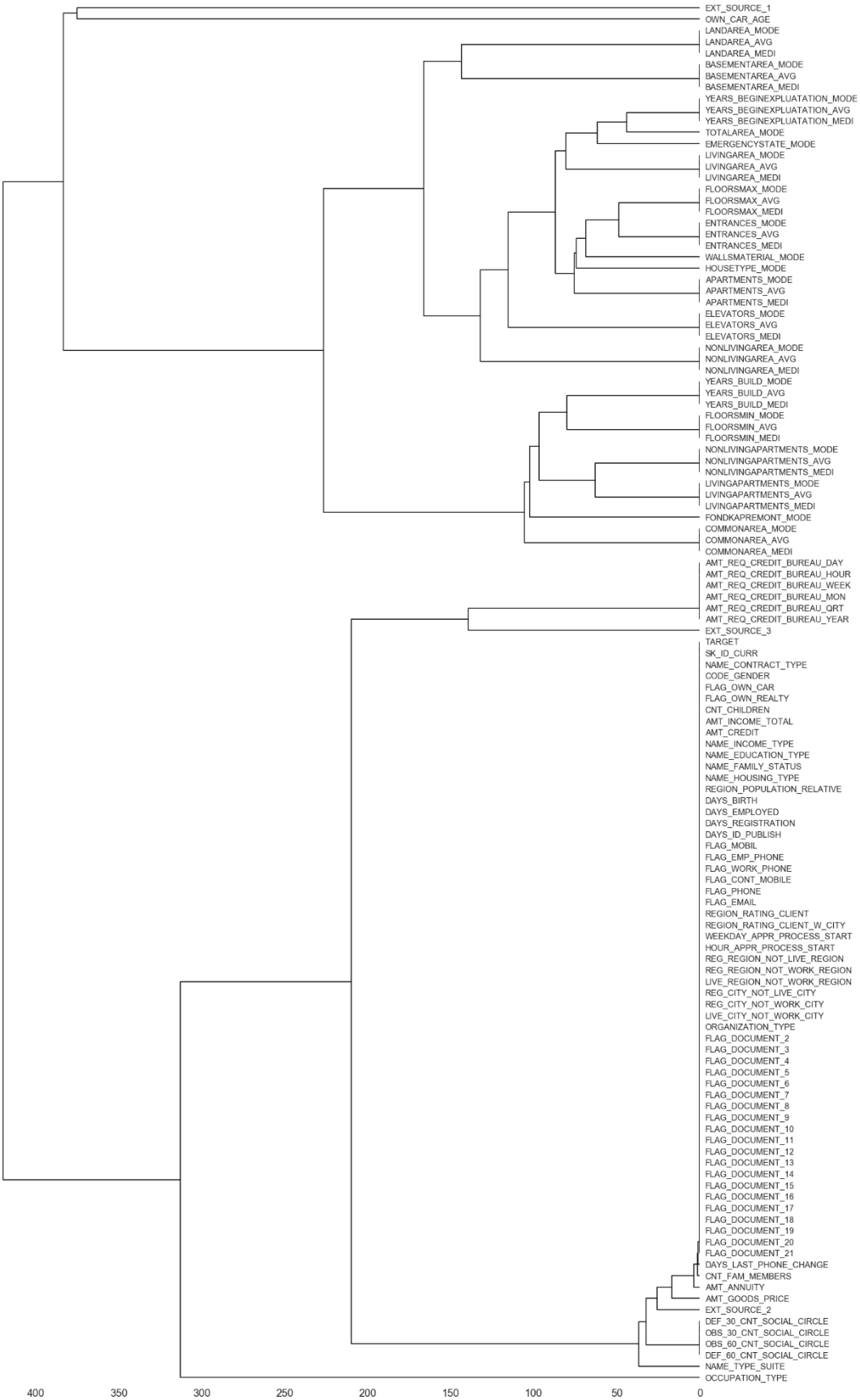


Histogram with fixed size bins (bins=10)

Another powerful package; *missingno* was used to visualize the missing data. The figure below gives an idea for which variables have missing data, through inspection it was clear that the variables that related to home features (e.g no of bedrooms) had that most missing values.



The dendrogram figure below shows the missing value hierarchical relationship between variables. This as useful in identifying some variables that did not have a description such as EXT_SOURCE_1. For example, through inspection, the relationship of the variable EXT_SOURCE_1 with OWN_CAR flag attribute was significant which led to believe that the EXT_SROUCE_1 variable perhaps a credit score inquired at the time of purchasing a car.

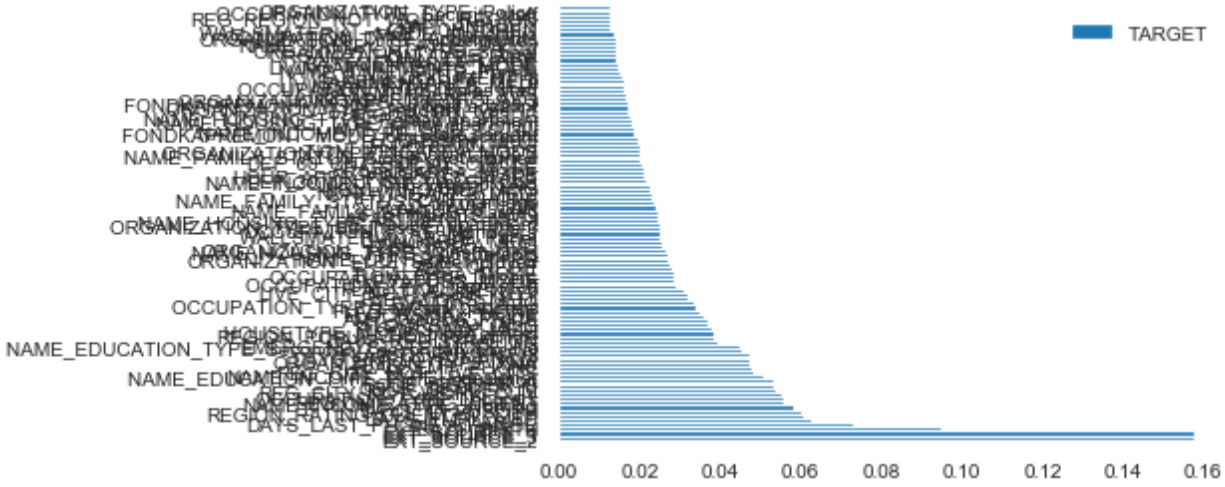

EXT_SOURCE_1
OWN_CAR_AGE
LANDAREA_MODE
LANDAREA_AVG
LANDAREA_MEDI
BASEMENTAREA_MODE
BASEMENTAREA_AVG
BASEMENTAREA_MEDI
YEARS_BEGINEXPLUATATION_MODE
YEARS_BEGINEXPLUATATION_AVG
YEARS_BEGINEXPLUATATION_MEDI
TOTALAREA_MODE
EMERGENCYSTATE_MODE
LIVINGAREA_MODE
LIVINGAREA_AVG
LIVINGAREA_MEDI
FLOORSMAX_MODE
FLOORSMAX_AVG
FLOORSMAX_MEDI
ENTRANCES_MODE
ENTRANCES_AVG
ENTRANCES_MEDI
WALLSMATERIAL_MODE
HOUSETYPE_MODE
APARTMENTS_MODE
APARTMENTS_AVG
APARTMENTS_MEDI
ELEVATORS_MODE
ELEVATORS_AVG
ELEVATORS_MEDI
NONLIVINGAREA_MODE
NONLIVINGAREA_AVG
NONLIVINGAREA_MEDI
YEARS_BUILD_MODE
YEARS_BUILD_AVG
YEARS_BUILD_MEDI
FLOORSMIN_MODE
FLOORSMIN_AVG
FLOORSMIN_MEDI
NONLIVINGAPARTMENTS_MODE
NONLIVINGAPARTMENTS_AVG
NONLIVINGAPARTMENTS_MEDI
LIVINGAPARTMENTS_MODE
LIVINGAPARTMENTS_AVG
LIVINGAPARTMENTS_MEDI
FONDKAPREMONT_MODE
COMMONAREA_MODE
COMMONAREA_AVG
COMMONAREA_MEDI
AMT_REQ_CREDIT_BUREAU_DAY
AMT_REQ_CREDIT_BUREAU_HOUR
AMT_REQ_CREDIT_BUREAU_WEEK
AMT_REQ_CREDIT_BUREAU_MON
AMT_REQ_CREDIT_BUREAU_QRT
AMT_REQ_CREDIT_BUREAU_YEAR
EXT_SOURCE_3
TARGET
SK_ID_CURR
NAME_CONTRACT_TYPE
CODE_GENDER
FLAG_OWN_CAR
FLAG_OWN_REALTY
CNT_CHILDREN
AMT_INCOME_TOTAL
AMT_CREDIT
NAME_INCOME_TYPE
NAME_EDUCATION_TYPE
NAME_FAMILY_STATUS
NAME_HOUSING_TYPE
REGION_POPULATION_RELATIVE
DAYS_BIRTH
DAYS_EMPLOYED
DAYS_REGISTRATION
DAYS_ID_PUBLISH
FLAG_MOBIL
FLAG_EMP_PHONE
FLAG_WORK_PHONE
FLAG_CONT_MOBILE
FLAG_PHONE
FLAG_EMAIL
REGION_RATING_CLIENT
REGION_RATING_CLIENT_W_CITY
WEEKDAY_APPR_PROCESS_START
HOUR_APPR_PROCESS_START
REG_REGION_NOT_LIVE_REGION
REG_REGION_NOT_WORK_REGION
LIVE_REGION_NOT_WORK_REGION
REG_CITY_NOT_LIVE_CITY
REG_CITY_NOT_WORK_CITY
LIVE_CITY_NOT_WORK_CITY
ORGANIZATION_TYPE
FLAG_DOCUMENT_2
FLAG_DOCUMENT_3
FLAG_DOCUMENT_4
FLAG_DOCUMENT_5
FLAG_DOCUMENT_6
FLAG_DOCUMENT_7
FLAG_DOCUMENT_8
FLAG_DOCUMENT_9
FLAG_DOCUMENT_10
FLAG_DOCUMENT_11
FLAG_DOCUMENT_12
FLAG_DOCUMENT_13
FLAG_DOCUMENT_14
FLAG_DOCUMENT_15
FLAG_DOCUMENT_16
FLAG_DOCUMENT_17
FLAG_DOCUMENT_18
FLAG_DOCUMENT_19
FLAG_DOCUMENT_20
FLAG_DOCUMENT_21
DAYS_LAST_PHONE_CHANGE
CNT_FAM_MEMBERS
AMT_ANNUITY
AMT_GOODS_PRICE
EXT_SOURCE_2
DEF_30_CNT_SOCIAL_CIRCLE
OBS_30_CNT_SOCIAL_CIRCLE
OBS_60_CNT_SOCIAL_CIRCLE
DEF_60_CNT_SOCIAL_CIRCLE
NAME_TYPE_SUITE
OCCUPATION_TYPE
400
350
300
250
200
150
100
50
0

Prior to imputing missing variables, variables that had anomalies were manipulated to reflect correct information. Label encoding for categorical variables that had 2 or less unique categories was implemented while one-hot encoding for categorical variables that had more than 2 unique categories was implemented. This had increased our variables from 122 variables to 241. This would make our imputations more effective, simpler to implement and without information loss by using the median strategy.

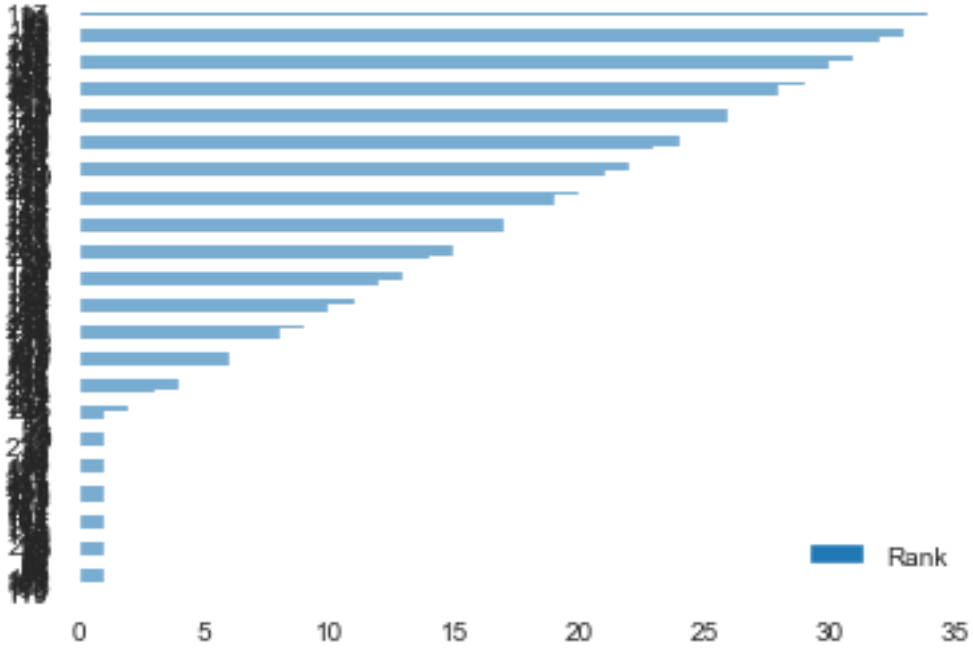
Feature Selection

The strategy for feature selection was to implement three different methods, find which variables were significant throughout the three methods and select them as our features. This would reduce our dimensions and thus reduce the computation cost of modeling with minimal information loss.

First up is Pearson correlation; a filter-based method that uses statistical scores to score the variable dependence of the input variables with the target variable. The figure below plots the sorted correlation coefficients versus its respective variable. The higher the better.



Second is Recursive Feature Elimination (RME); a wrapper method which fits a model and removes the weakest features. The features are initially ranked with their correlation scores. Low scoring variables are recursively eliminated after each loop thus eliminating collinearity. The figure below plots the count of features by their rank. The features at the bottom are of most significance.



Lastly, Random forest, an embedded method which is similar to the previous methods by combining their advantages with the difference of an intrinsic model building metric being used during learning

Below is a sample of the table created that entails which features have been “classified” as significant across the three previous methods. Feature selection was brought down from 241 to 75.

	Feature	Pearson	RFE	Random Forest	Total
1	TOTALAREA_MODE	TRUE	TRUE	TRUE	3
2	REGION_RATING_CLIENT_W_CITY	TRUE	TRUE	TRUE	3
3	NONLIVINGAREA_MODE	TRUE	TRUE	TRUE	3
4	NAME_EDUCATION_TYPE_Secondary / secondary special	TRUE	TRUE	TRUE	3
5	FLOORSMAX_MODE	TRUE	TRUE	TRUE	3
6	FLAG_OWN_CAR	TRUE	TRUE	TRUE	3
7	FLAG_DOCUMENT_3	TRUE	TRUE	TRUE	3
8	EXT_SOURCE_3	TRUE	TRUE	TRUE	3
9	EXT_SOURCE_2	TRUE	TRUE	TRUE	3
10	EXT_SOURCE_1	TRUE	TRUE	TRUE	3
11	DEF_60_CNT_SOCIAL_CIRCLE	TRUE	TRUE	TRUE	3
12	DAYS_LAST_PHONE_CHANGE	TRUE	TRUE	TRUE	3
13	DAYS_EMPLOYED	TRUE	TRUE	TRUE	3
14	COMMONAREA_MODE	TRUE	TRUE	TRUE	3
15	COMMONAREA_MEDI	TRUE	TRUE	TRUE	3
16	COMMONAREA_AVG	TRUE	TRUE	TRUE	3
17	CODE_GENDER_F	TRUE	TRUE	TRUE	3
18	AMT_INCOME_TOTAL	TRUE	TRUE	TRUE	3
19	AMT_GOODS_PRICE	TRUE	TRUE	TRUE	3
20	AMT_CREDIT	TRUE	TRUE	TRUE	3
21	YEARS_BEGINEXPLUATATION_MODE	FALSE	TRUE	TRUE	2
22	REG_CITY_NOT_WORK_CITY	TRUE	FALSE	TRUE	2
23	REG_CITY_NOT_LIVE_CITY	TRUE	FALSE	TRUE	2
24	REGION_RATING_CLIENT	TRUE	FALSE	TRUE	2
25	REGION_POPULATION_RELATIVE	TRUE	FALSE	TRUE	2
26	OWN_CAR_AGE	FALSE	TRUE	TRUE	2
27	ORGANIZATION_TYPE_Self-employed	TRUE	FALSE	TRUE	2

Modelling

A simple 10 fold linear regression model was fit twice. Once with the imbalanced data (as is) and once after balancing the data. A slight improvement was noted on the balanced data and significantly computation speed was observed for the balanced data. Hence going forward all the models were fit on the balanced data.

Since the target variable was significantly imbalanced (more 1 than 0), the data was balanced using SMOTE. SMOTE adds new synthesized examples from the existing examples to the imbalanced class. Prior to SMOTE:

```
Counter({0.0: 282686, 1.0: 24825})
```

Post SMOTE:

```
Counter({0.0: 56536, 1.0: 28268})
```

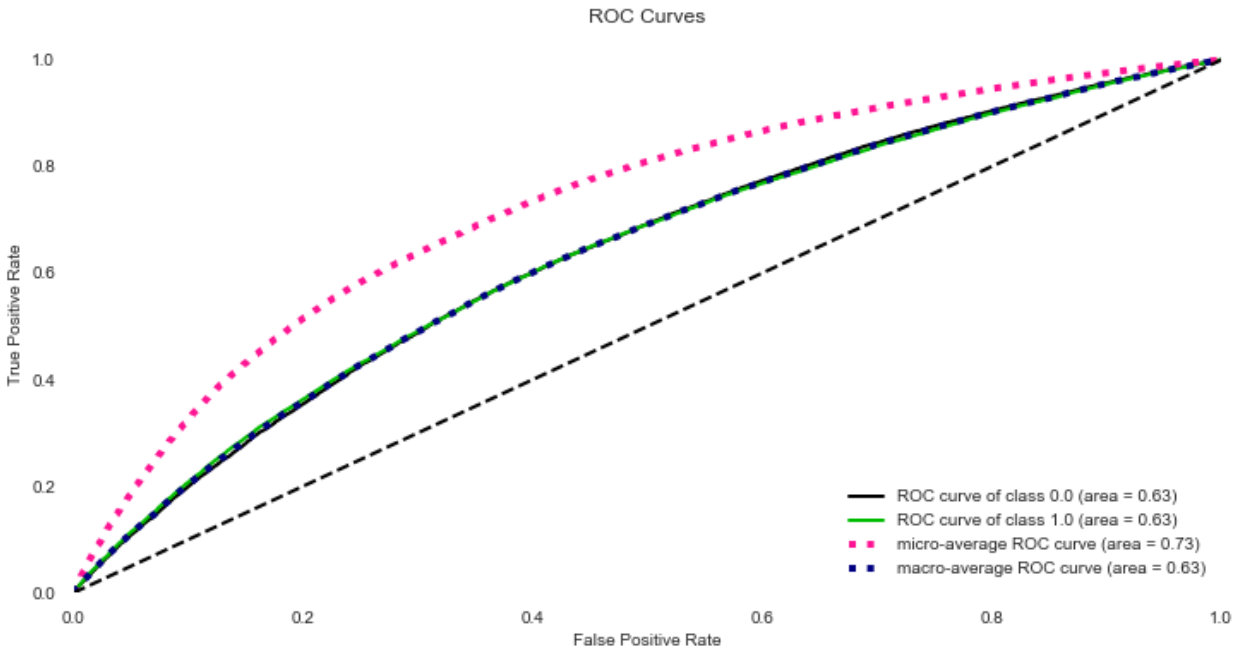
Following this is splitting our data into a train and a test set to a ratio of 1:0.8

For each model the following was done; the classifier was declared with mostly the default parameters. Then the training data was fit to the model. This process was repeated 10 times to check for stability via computation time. Once the model computed the classification prediction the training set, a confusion matrix as generated to further compute statistical measure for classification. The statistical measures were Accuracy which measures how close the prediction was to the targe, weighted precision which measures the ability of the classifier not to label a positive sample that is negative, weighted recall is the ability of the classifier to final all the positive samples, weighted f1 score which is a harmonic mean of the precision and recall. The weighted variation of those measures take into account class imbalance and hence score better. Our final measure is the Area Under the Receiver Operating Characteristics (AUROC) which simply put tells how much a model can distinguish between classes. It is also useful in that we can plot a curve since it is the False Positive Rate (FPR) plotted against the True Positive Rate (TPR). All these measures score between 0 and 1; the higher the better

Below is an example of the confusion matrix for the linear regression classifier along with computed measures and ROC curve

	True Positive	True Negative
Predicted Positive	819	610
Predicted Negative	21770	44644

```
Train AUC Score for probability of Target = 1: 0.6353972874810894
Train AUC Score for class prediction: 0.5113885568177933
Train Accuracy Score: 0.5113885568177933
Weighted Train Precision Score: 0.6392181293296968
Weighted Train Recall Score: 0.6701207198974102
Weighted Train f1 Score: 0.5560623795600497
```



The two tables below compile the results obtained for the model performance

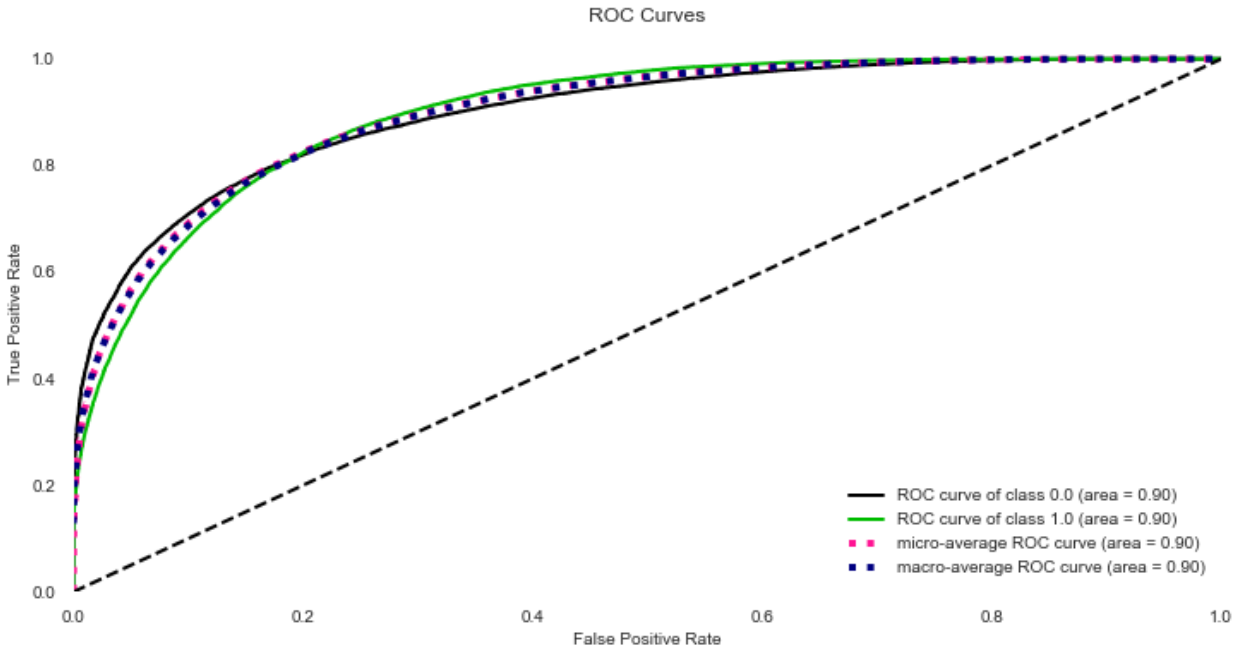
Train	Logistic Regression	Random Forest	Light GBM
ROC AUC (prob)	0.635397287	0.653565519	0.8237005
ROC AUC (pred)	0.511388557	NA	0.6993584
Accuracy	0.511388557	0.653565519	0.6993584
Precision	0.639218129	0.735116845	0.768832
Recall	0.67012072	0.741815663	0.7731822
f1	0.55606238	0.71745465	0.7579317

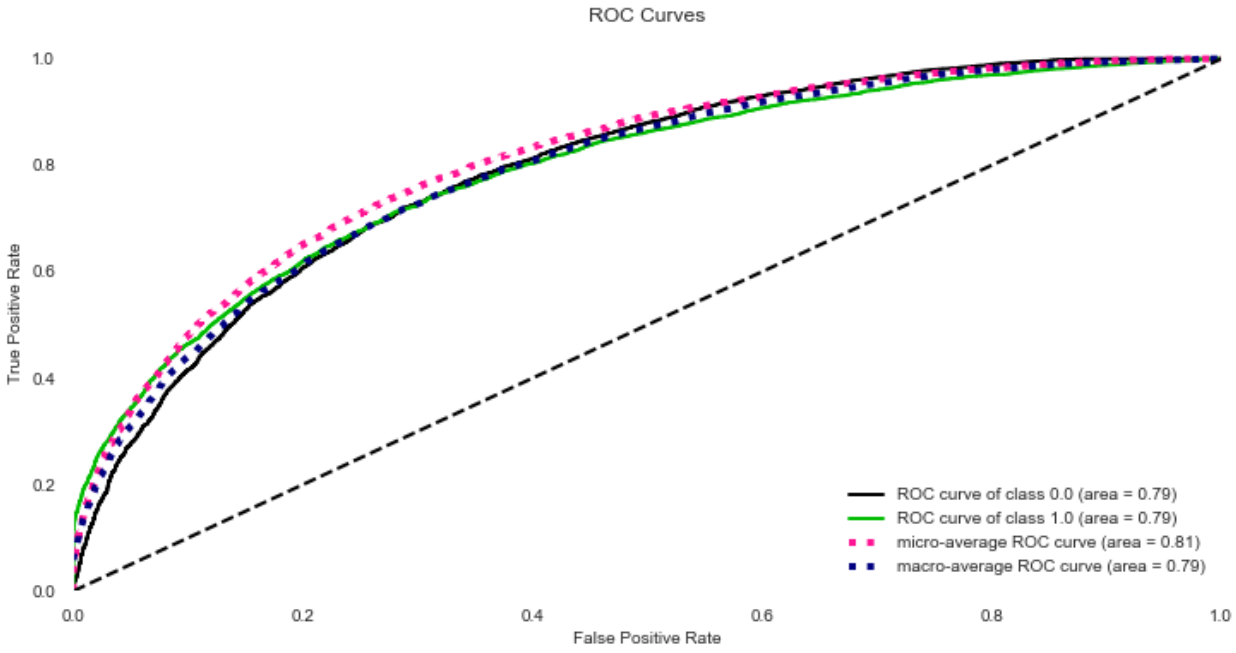
	Logistic Regression	Random Forest	Light GBM
ROC AUC (prob)	0.632746052	0.773952902	0.7873304
ROC AUC (pred)	0.510389582	0.661782602	0.6787574
Accuracy	0.510389582	0.661782602	0.6787574
Precision	0.635232815	0.74036512	0.7441483
Recall	0.667826189	0.747243677	0.7503095
f1	0.552235089	0.72524207	0.733334

Following this there was enough time to make use of an fantastic package, *hyperopt* which is used for automated hyperparameter tuning. This computation was arguably had the highest computation as its run time ran for 2 hours and 15 minutes. The tuning was run on the Light GBM model to tune its hyperparameters by running exhaustive computations of different combinations of hyperparameter values while learning from each computation. This is called Bayesian Optimization. The table below returns the top parameter choices based on the highest cross validation score

	boosting_type	colsample_bytree	is_unbalance	learning_rate	min_child_samples	num_leaves	reg_alpha	reg_lambda	subsampl
0	gbdt	0.628081	TRUE	0.014549	35	127	0.757313		
1	dart	0.862645	FALSE	0.012325	340	74	0.920178		
2	gbdt	0.700654	TRUE	0.403179	105	88	0.908288		
3	dart	0.748835	FALSE	0.20512	45	60	0.293528		
4	dart	0.879481	TRUE	0.102988	125	96	0.558871		
5	dart	0.738548	FALSE	0.195828	335	75	0.010304		

The optimized parameters returned the best metrics on training and testing, however the model appeared to overfit on the training data as we can compare from the two ROC curves for train and test sequentially.





Conclusion

In this study, data exploratory skills were implemented in getting a feel for the data and finding outliers and obvious “surface” relationships in order to optimize the data for further analysis. This analysis

utilized feature selection for dimension reduction. The modelling techniques used were put to use and tested vigorously to ensure stability. While we have learned that linear classifiers were not the most optimal choice, more interest shifted towards decision tree based classifiers and novel boost classifiers.

This project has put a fair test of the skills learned over the course of this certificate while there is a much room for improvement by participating in more challenges and learning novel ideas.

References

Lessmann, S., Baesens, B., Seow, H.-V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, *247*(1), 124–136. doi: 10.1016/j.ejor.2015.05.030

Kim, J. Y., & Cho, S. B. (2019). Predicting repayment of borrows in peer-to-peer social lending with deep dense convolutional network. *Expert Systems*, *36*(4). doi: 10.1111/exsy.12403

Wijewardhana, U. (2018). A Mathematical Model for Predicting Debt Repayment: A Technical Note. *Australasian Accounting, Business and Finance Journal*, *12*(3), 107–115. doi: 10.14453/aabfj.v12i3.8

Kao, L.-J., Chiu, C.-C., & Chiu, F.-Y. (2012). A Bayesian latent variable model with classification and regression tree approach for behavior and credit scoring. *Knowledge-Based Systems*, *36*, 245–252. doi: 10.1016/j.knosys.2012.07.004

Brown, M., & Zehnder, C. (2007). Credit Reporting, Relationship Banking, and Loan Repayment. *Journal of Money, Credit and Banking*, *39*(8), 1883–1918. doi: 10.1111/j.1538-4616.2007.00092.x