Home Credit Default Risk (Kaggle Competition) - Milestone Report

Problem Statement:

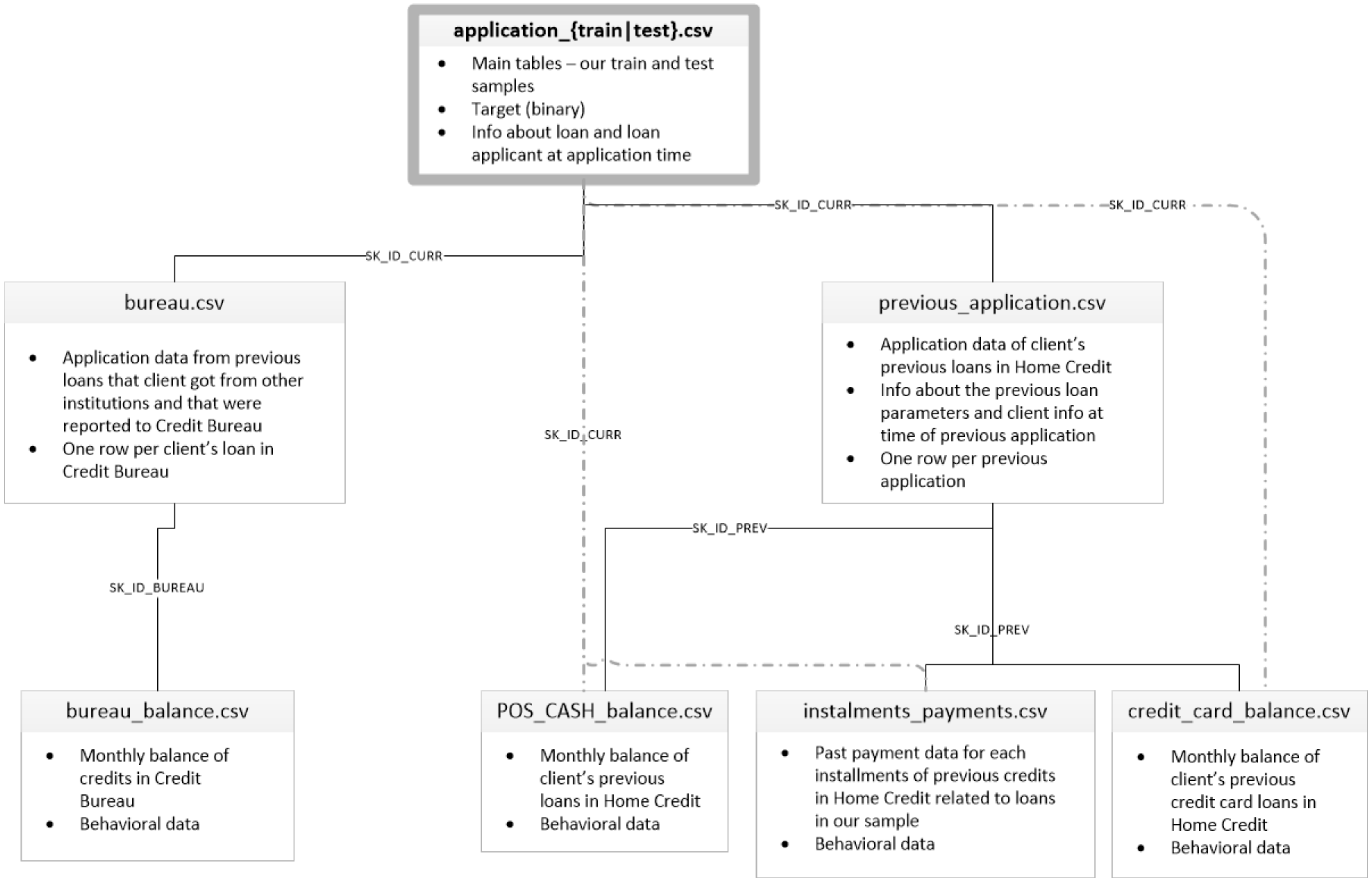
Home Credit is an organization that serves the unbanked population with access to loans. Such individuals that do not have a built-up credit score have a challenging time securing loans from financial institutions. Home Credit are trying to address this challenge by utilizing various other types of information sources to make their decision on offering a loan to a prospect. This though, comes with risk as a client's ability to repay is more difficult to discern. As such, Home Credit are seeking to understand whether a client is likely to be able to repay a loan and subsequently make a decision to offer or reject the loan or perhaps adjust some of the loan conditions. Being able to better predict the repayment outcomes offers confidence to the financial institutions and empowers it to to be able to offer much needed opportunities to their clients.

Dataset Description:

There are seven data sets that are at the disposal of Home Credit:

> **Application_train.csv** - this is the principal table and presents all of the application information. There is a single row per application, which has a unique identifier.
>
> **previous_application.csv** - this file presents previous applications for people in the sample through Home Credit. There is a row per each application.
>
> **installments_payments.csv** - this is the repayment history on loans given out through Home Credit for people in the sample. Each row is a made or missed payment.
>
> **bureau.csv** - credit information from other financial institutions that were reported to Home Credit. Each row represents a credit that was given to an individual in the sample.
>
> **bureau_balance.csv** - monthly information per credit, per loan for users in the sample. This is a long data set as it has (number of loans*credits associated for those loans*month duration for each of those credits) rows.
>
> **POS_CASH_balance.csv** - similar to bureau_balance.csv, this data set is the internal version of the previous monthly breakdown of balances for consumer credit and cash loans that were taken out through Home Credit.
>
> **credit_card_calance.csv** - each row in this data set represents a monthly balance of credit cards that were issues to applicants in the sample through Home Credit.

Links between the 7 data files can be seen here:



application_{train|test}.csv

- Main tables – our train and test samples
- Target (binary)
- Info about loan and loan applicant at application time

bureau.csv

- Application data from previous loans that client got from other institutions and that were reported to Credit Bureau
- One row per client's loan in Credit Bureau

previous_application.csv

- Application data of client's previous loans in Home Credit
- Info about the previous loan parameters and client info at time of previous application
- One row per previous application

bureau_balance.csv

- Monthly balance of credits in Credit Bureau
- Behavioral data

POS_CASH_balance.csv

- Monthly balance of client's previous loans in Home Credit
- Behavioral data

instalments_payments.csv

- Past payment data for each installments of previous credits in Home Credit related to loans in our sample
- Behavioral data

credit_card_balance.csv

- Monthly balance of client's previous credit card loans in Home Credit
- Behavioral data

Data Wrangling and Cleaning:

There was significant preparatory work necessary in order to combine all of the data into a single workable file. Since the data in the application_train data set is on the application level, it was necessary to group data in the other data sets to the same level.

Here are the steps that were taken to gather the data into the same file:

- Beginning with the ***bureau_balance file***, the past application information was grouped so that there existed only a single application identifier per row. This means that the months balance information was summed to give an idea for length of loan. Additionally the status of the loan was encoded into dummy variables to give a cumulative figure of the state of such an application. The values for all of these statuses for each application add up to 1 for each row (application).

- The ***bureau*** file had a number of categorical variables that were also encoded as dummies. These included the credit type, status and currency. Once those were encoded, the data was also grouped on the ID of the loan in summation.

- ***credit_card_balance*** was next to be addressed and this data was mostly continuous variables for each of the balances. There was one exception, which was the status of the balance for a specific month and these were also encoded as dummies. As with the bureau files, these were grouped by summation on the unique identifier for previous transaction.

- ***POS_CASH_balance*** was treated similarly to the the cc_balance file above in that only the single status variable was encoded and the data was grouped. The ***previous_application*** was also grouped in the same way, although required encoding many more variables, such as type of the product, yield group, seller industry and more.

- While no encoding was necessary on the ***installments_payments*** file, it was important to create a new variable that denoted the number of times a lonee was late on a payment. The variable ('late') was created using the installment due date in number of days from application and the corresponding payment. This was important as grouping such data would have likely made it difficult to unbundle the number of times that a loanee was late on a payment. This variable would later show to be of importance as it frequently was of the top 10 model feature importances.

Once grouped, these data-sets were joined as per the above links graphic to be finally joined to the main application file by the Id of the current application. Because most applications that are present in the main file do not have previous available history, there were several empty values for each of the applications.

Columns with over 35% of missing data were removed from the data set, which cut down the dataframe from 339 to 291 columns.

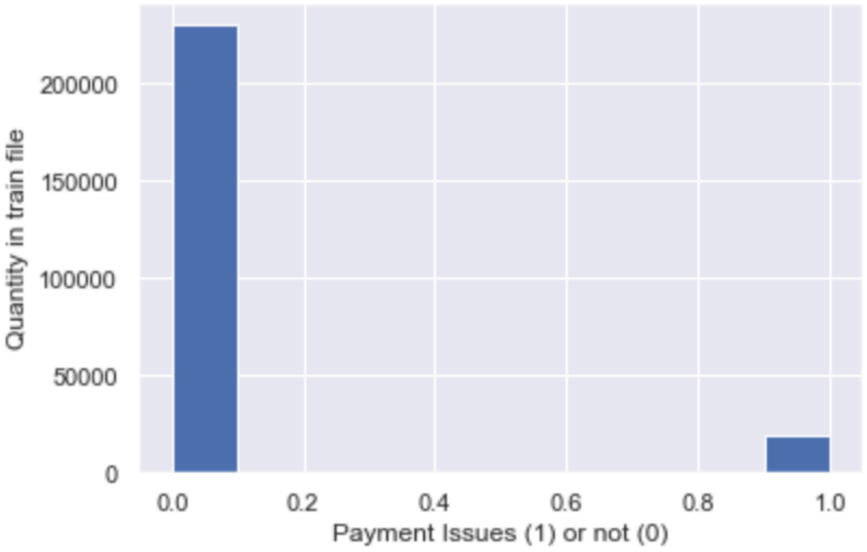
Subsequently, there were approximately 19% of the data that had the majority of data missing - this is because of the inability to match the current application with this previous application information. This data was discarded due to a having a large enough data-set and the ratio of the defaulters to non-defaulters was similar after discarding these applications.

Some other important decisions were taken place before we moved to exploring the data:

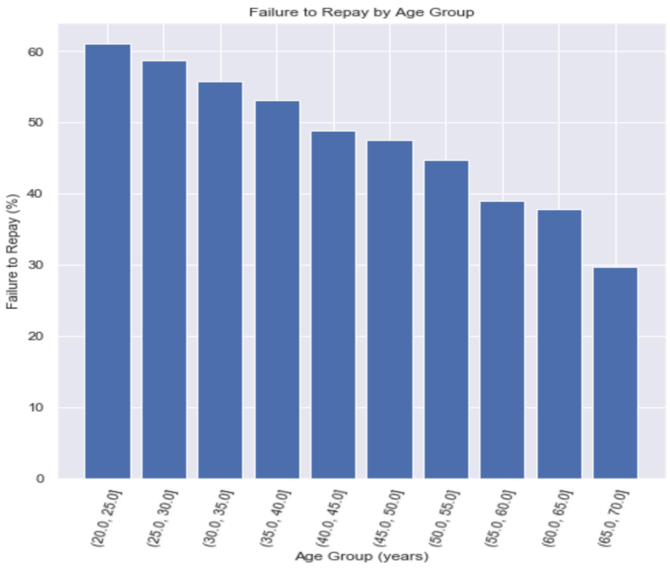
- Nan occupation type was presumed to be unemployed and was marked as such
- 3 key external sources (which later proved to be critical for modeling) had a number of missing values, which were imputed through the mean. These were EXT_SOURCE_1, EXT_SOURCE_2 and EXT_SOURCE_3.
- Values that were blank for noting who the applicant was accompanied by were marked as 'unaccompanied'.

Data Exploration:

The distribution of the target variable was important to review to decide whether upsampling or downsampling will be needed in order to balance the proportion of instances with payment issues to those without. Indeed, due the ratio of the quantities in the train file, downsampling was later utilized.



Distribution plots as well as pair-plots helped identify any immediate correlations as well as outliers. Of the variables that were explored, most seemed reasonably distributed. For example, the age of the applicant variables did not seem to offer any unexpected observations, while it did later show a strong correlation to the target variable.

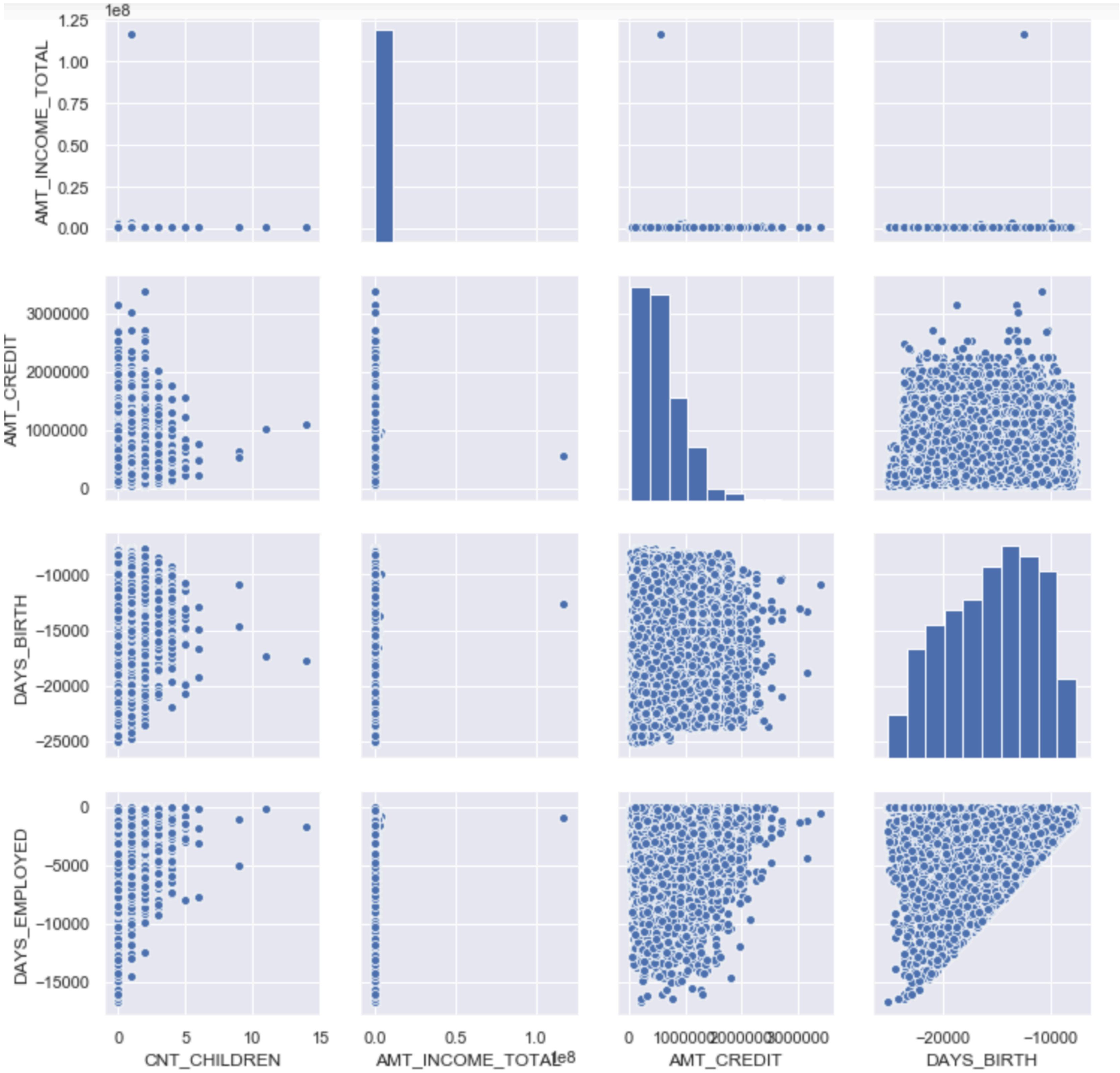


There were a few exceptions however that were adjusted:

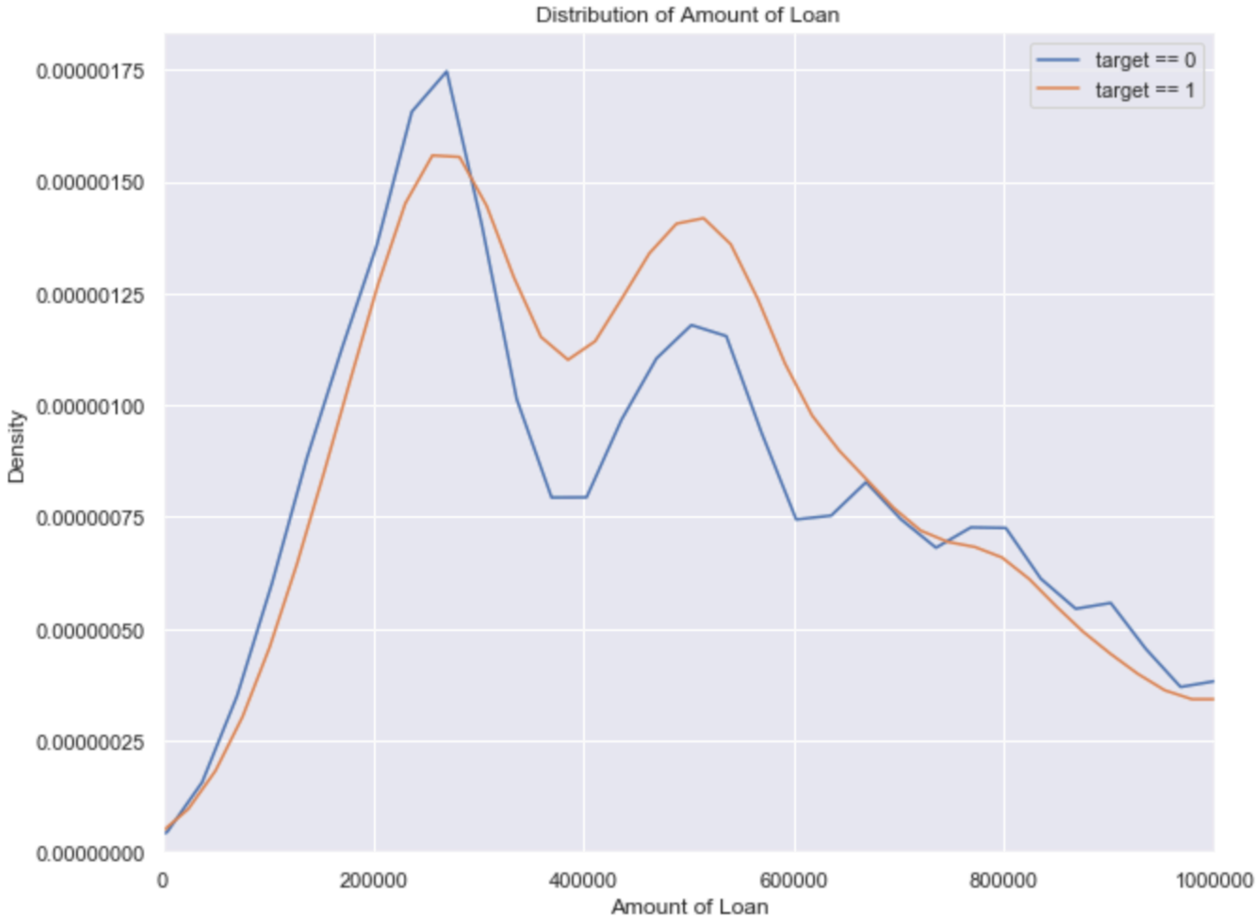
- The ['DAYS_EMPLOYED'] variable, which denoted the number of days an applicant was employed prior to the application date had 18% of the records with -1000 years of employment. This must have been a coding issue, which I decided to address with the assumption that these were unemployed and changed those records to zeros.

- The ['AMT_INCOME_TOTAL'] variable had an extremely long tail on the distribution with the maximum value of $117M. Given the extremely large an atypical salary, I decided to cut down the data to within three standard deviations from the mean.

Pair-plots:



Subsequent exploration included looking at correlation heat maps between variables in the train file to identify potential interesting patterns. Similarly, distribution differences in relation to the target variable helped give an idea of the types of variables that could be important for distinguishing the difference in our subsequent models. Once such visible difference can be seen in the Amount of Loan variable.



Before moving on to modeling, the data was cut down to address collinearity through Variance Inflation Factoring. This helped eliminate some of the duplicate variables that resulted in the merging of the data as well as some highly collinear variables.
While the initial Linear Regression models excluded some variables, such as EXT_Source_1, EXT_Source_2 and EXT_Source_3, these later proved to be very significant in the Random Forest and GXBoost Models. As such, these variables were reintroduced into the data set for the Regression models as well.

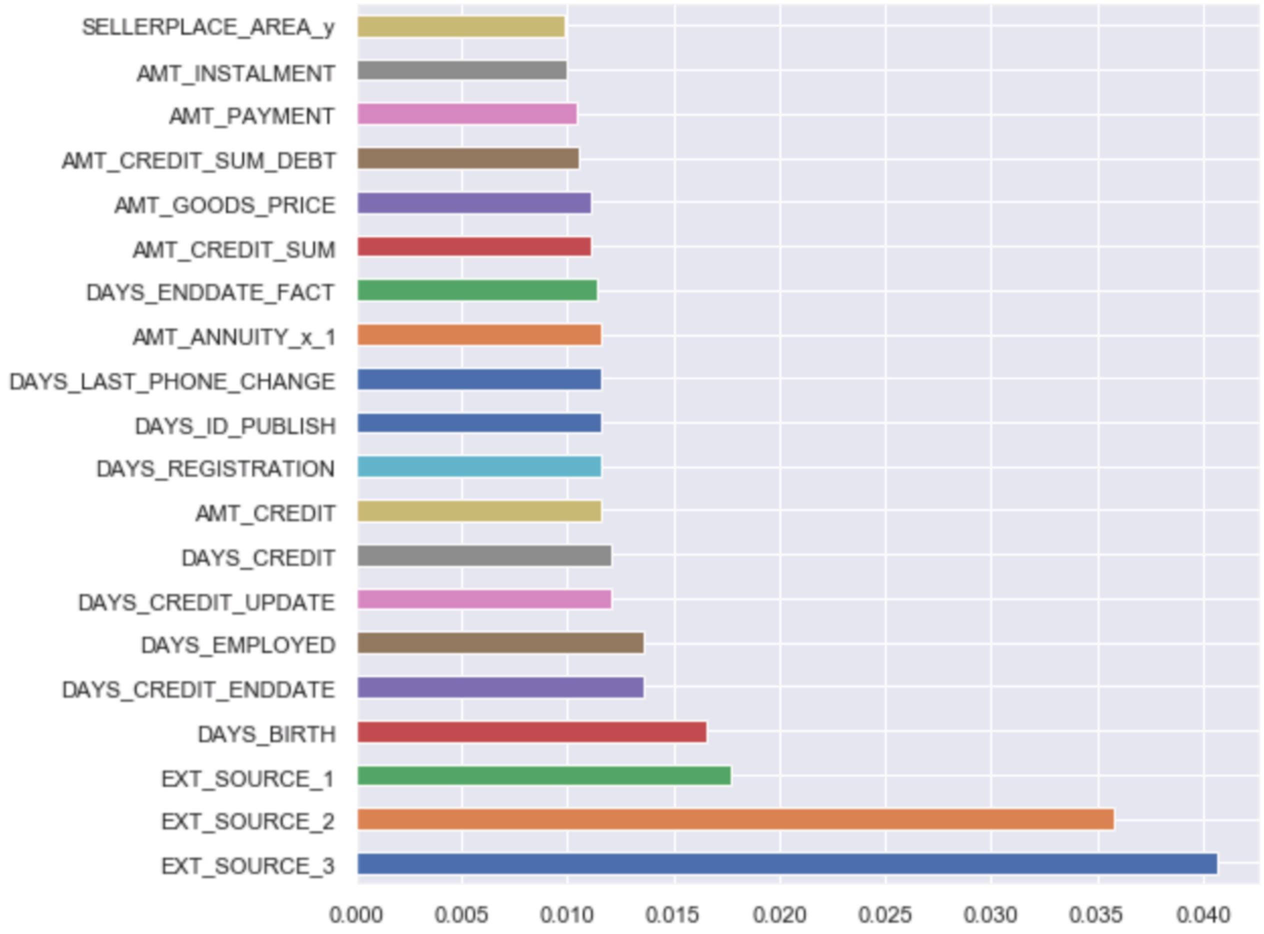
Early Models:

The measuring criteria for this competition is the AUC ROC, which is a measure of the ration of the True Positive to the False positive observations in our test data.

The early results in terms of the accuracy of logistic regression classifier on test set was an underwhelming 0.57. Subsequent models using a Random Forest classifier gave AUC scores as high as 0.697 after some iterations of hyperparameter tuning. Lastly, using the XGBClassifier, results as high as .704 were achieved - again after some hyperparameter tuning.

The image below shows the feature importances of that last model. It is interesting to see how the EXT_SOURCE variables proved to have a heavy weight in these models. Also, it is interesting to see that the above mentioned age variable (DAYS_BIRTH) was the fourth highest feature in terms of its importance for the model.

Feature Importances:



Further Exploration and Modeling:

While there are some nice models that are able to generate good accuracy scores, there are a few things that could be done to attempt to better these models. All of these will focus on feature engineering and the creation of new variables to input into the model:

1. Grouping supplemental variables through other statistics as the initial attempt grouped variables almost exclusively through mean grouping. Looking at the Max and Min values of past transactions as well as the count of those transactions could prove fruitful.

2. Subject Matter variables - It is possible to create variables as a combination of one or more currently available variables if those present a stronger correlation with the likelihood to have challenges repaying a loan. An example of such a variable is the ratio of the individuals pay to the loan amount as this would be an interesting presentation of the hardship that the loan is for that individual.

3. Grouping continuous variables - just as we saw with the age variable, an exploration about binning variables might uncover some patterns that a continuous distribution might not.