EDA Assignment

By
Karthik Mishra

Problem Statement

- To perform Exploratory Data Analysis and Data cleaning of a Dataset comprising of two Tables, one on New applications for Loan at a Bank, and second containing previous applications data of loans at the same Banking Institution

Methodology

1. Load and Inspect the two Tables
2. Handling of Null Values
3. Detection of Datatype mismatches, Outliers, and Data Imbalance
4. Univariate, Segmented Univariate and Bivariate Analysis
5. Analysis on Merged Data

Handling Null Values

- For inpO table, or application_data.csv
  - Deleted columns having greater than 40% null values
  - For the remaining columns the null values were handled as below

OCCUPATION_TYPE	31.345545	NaN is making the dtype to int, replace NaN with the value "Unkonwn"
EXT_SOURCE_3	19.825307	replaced with mean, a low value will be incorrect, and a highest value might be incorrect, so filled with mean
AMT_REQ_CREDIT_BUREAU_HOUR	13.501631	Set the default value to 0, which is the mode and median
AMT_REQ_CREDIT_BUREAU_DAY	13.501631	Set the default value to 0, which is the mode and median
AMT_REQ_CREDIT_BUREAU_WEEK	13.501631	Set the default value to 0, which is the mode and median
AMT_REQ_CREDIT_BUREAU_MON	13.501631	Set the default value to 0, which is the mode and median
AMT_REQ_CREDIT_BUREAU_QRT	13.501631	Set the default value to 0, which is the mode and median
AMT_REQ_CREDIT_BUREAU_YEAR	13.501631	Set the default value to 0, which is the mode and median
NAME_TYPE_SUITE	0.420148	Filled with the mode, and median value, unaccompanied
OBS_30_CNT_SOCIAL_CIRCLE	0.332021	Dropped the 1000 rows which didn't contain these values
DEF_30_CNT_SOCIAL_CIRCLE	0.332021	Dropped the 1000 rows which didn't contain these values
OBS_60_CNT_SOCIAL_CIRCLE	0.332021	Dropped the 1000 rows which didn't contain these values
DEF_60_CNT_SOCIAL_CIRCLE	0.332021	Dropped the 1000 rows which didn't contain these values
EXT_SOURCE_2	0.214626	656 values filled with median
AMT_GOODS_PRICE	0.090403	filled nan with value in amt_credit
AMT_ANNUITY	0.003902	filled with 0
CNT_FAM_MEMBERS	0.00065	filled with 0
DAYS_LAST_PHONE_CHANGE	0.000325	phone provided was work phone so column not applicabed, filled with 0

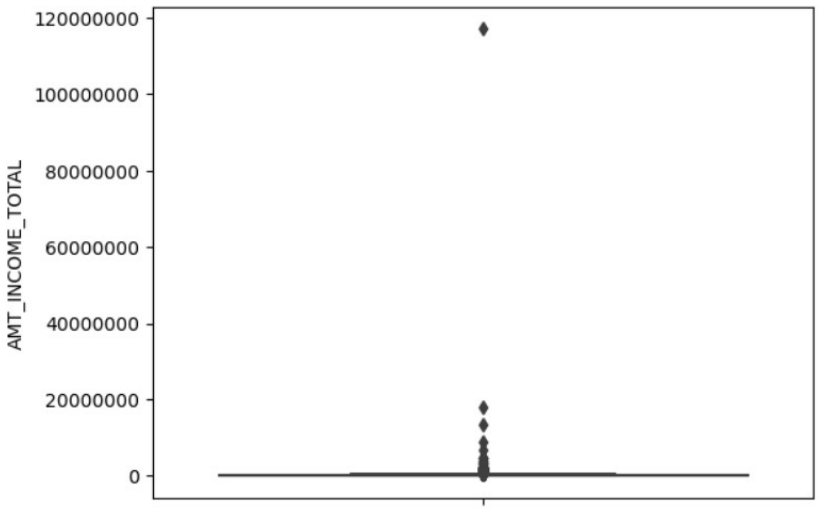
- For inp1 table, or previous_application.csv
  - Deleted columns having greater than 40% null values
  - For the remaining columns the null values were handled as below

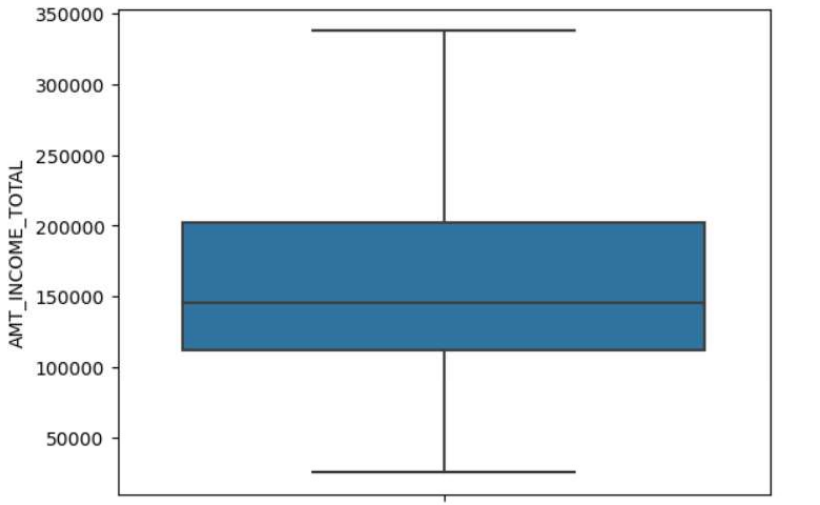
AMT_GOODS_PRICE	23.081773	Filled NaN with value from AMT_CREDIT
AMT_ANNUITY	22.286665	Dropped column, because for NA values both AMT_ANNUITY and CNT_PAYMENT are missing, hence values can't be effectively estimated.
CNT_PAYMENT	22.286366	Dropped column, because for NA values both AMT_ANNUITY and CNT_PAYMENT are missing, hence values can't be effectively estimated.
PRODUCT_COMBINATION	0.020716	Filled NaN with Unknown
AMT_CREDIT	0.00006	Dropped the one row containing NaN

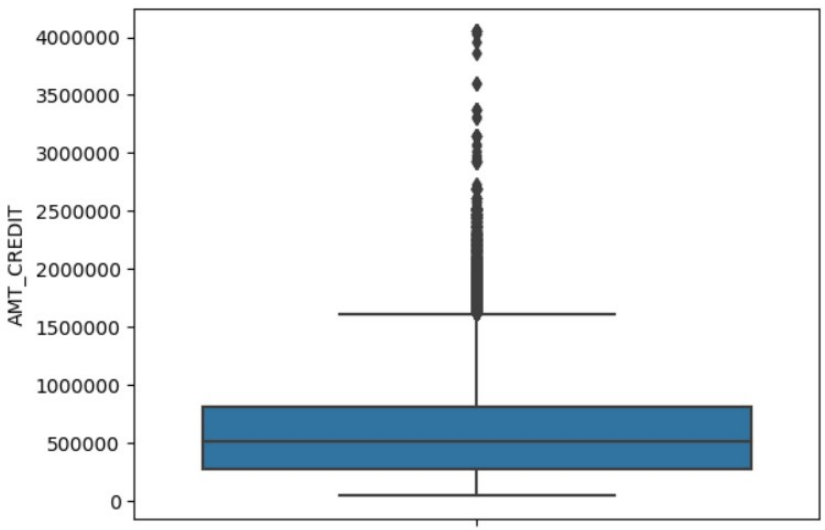
Detection of Datatype mismatches, Outliers, and Data Imbalance

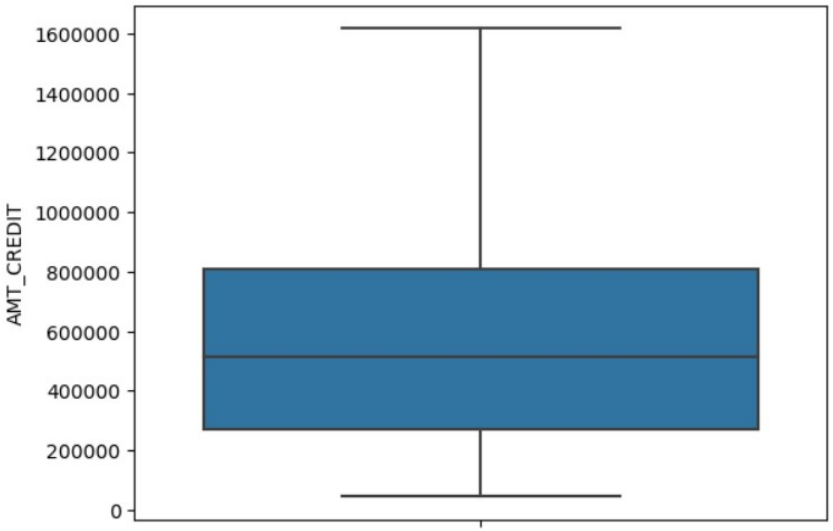
- **Datatype mismatches** were rare, and the only cases were caused by the presence of NaN values
- Filling of NaN values fixed datatype mismatches
- The values of the column, WEEKDAY_APPR_PROCESS_START, decision was made to retain the values in object datatype, and not change to datetime, because the benefits were minimal

- Outliers are present in multiple columns of both dataframes as evidenced by output of several columns with and with outliers









- A complete plot of outliers for all the columns in inp0 and inp1 can be viewed in outputs to line 88 and 89, and 116 from the Jupyter notebook
- For the present data cleaning and EDA task, these outliers have just been identified and ignored. However based on the kind of analysis required in further machine learning task these outliers can be dealt with by capping the data, binning the data, dropping the outliers, etc, based on the task expected.

- Data Imbalance is present in the target variable of this dataset, with the negative cases present **<u>11.3639</u>** times positive cases

```
inp0.TARGET.value_counts()
```

```
0    281701
1     24789
Name: TARGET, dtype: int64
```

```
inp0.TARGET.value_counts(normalize=True)
```

```
0    0.91912
1    0.08088
Name: TARGET, dtype: float64
```

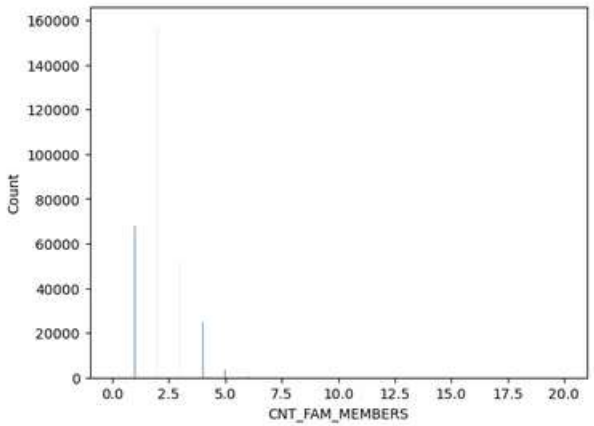
Univariate, Segmented Univariate, and Bivariate Analysis

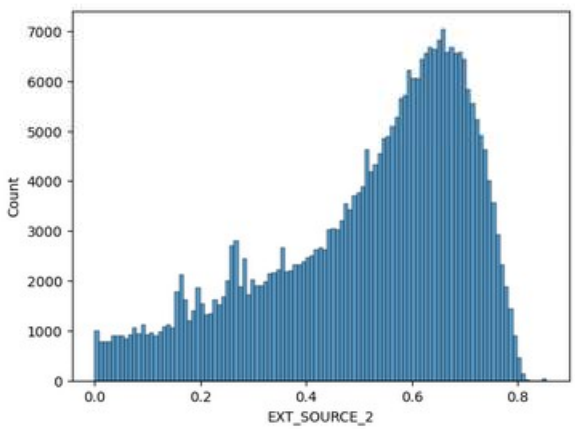
- To perform analysis on this data, both dataframe's columns are divided into 3 different lists, comprising of float variables (for categorical data), float variables (for numerical continuous data) and int variables (for numerical discrete data)
- Based on this the following observations can be made:

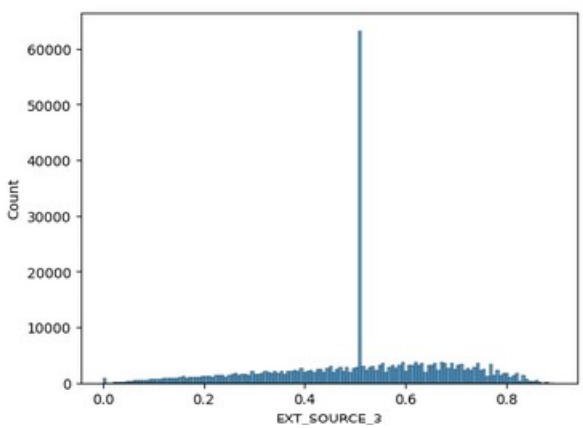
```
NAME_CONTRACT_TYPE  TARGET
Cash loans          0        0.917
                    1        0.083
Revolving loans     0        0.945
                    1        0.055
Name: TARGET, dtype: float64
CODE_GENDER  TARGET
F            0        0.930
             1        0.070
M            0        0.898
             1        0.102
XNA          0        1.000
Name: TARGET, dtype: float64
FLAG_OWN_CAR  TARGET
N             0        0.915
              1        0.085
Y             0        0.927
              1        0.073
Name: TARGET, dtype: float64
FLAG_OWN_REALTY  TARGET
N                0        0.917
                 1        0.083
Y                0        0.920
                 1        0.080
Name: TARGET, dtype: float64
```

```
NAME_TYPE_SUITE  TARGET
Children         0        0.926
                 1        0.074
Family           0        0.925
                 1        0.075
Group of people  0        0.914
                 1        0.086
Other_A          0        0.912
                 1        0.088
Other_B          0        0.901
                 1        0.099
Spouse, partner  0        0.921
                 1        0.079
Unaccompanied    0        0.918
                 1        0.082
Name: TARGET, dtype: float64
NAME_INCOME_TYPE      TARGET
Businessman           0        1.000
Commercial associate  0        0.925
                      1        0.075
Maternity leave       0        0.600
                      1        0.400
Pensioner             0        0.946
                      1        0.054
State servant         0        0.942
                      1        0.058
Student               0        1.000
Unemployed            0        0.579
                      1        0.421
Working               0        0.904
                      1        0.096
Name: TARGET, dtype: float64
```

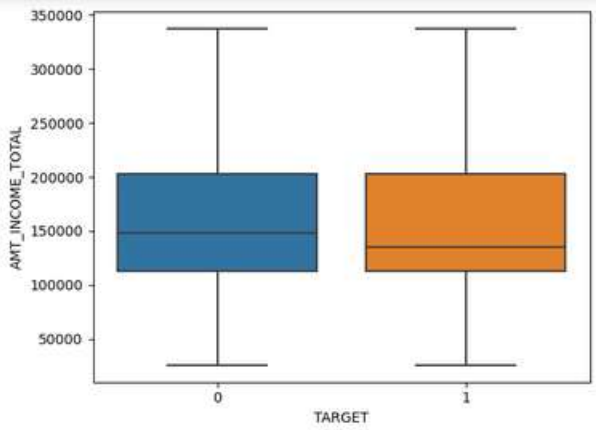
Complete Analysis can be found at the output of line 87 in the Jupyter notebook

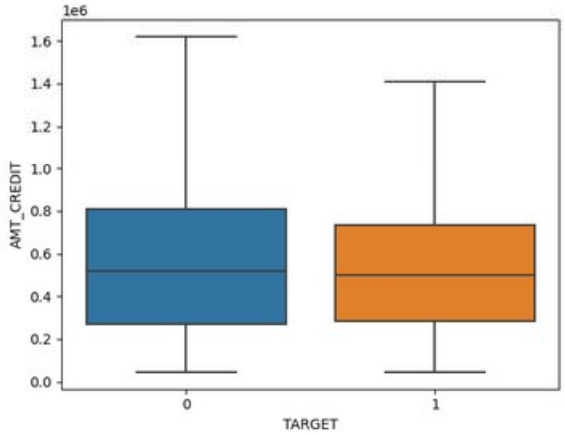


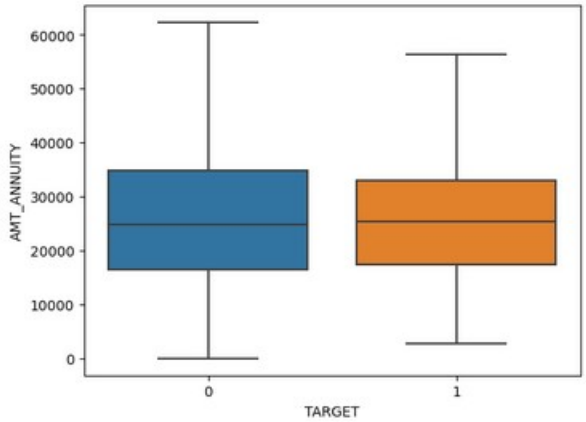




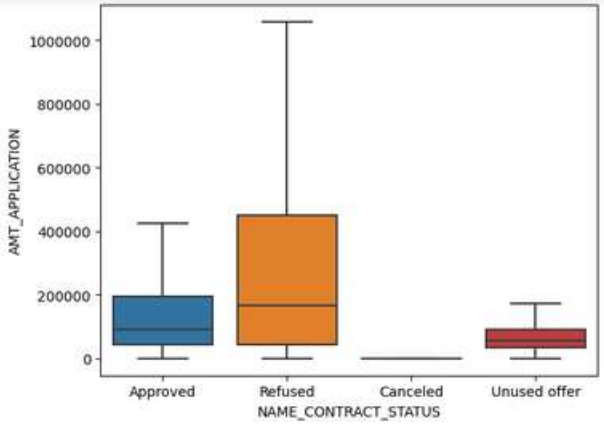
Complete Analysis can be found at the output of line 120 in the Jupyter notebook

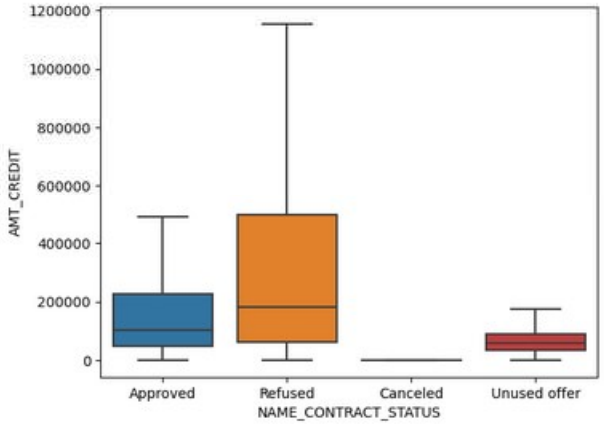


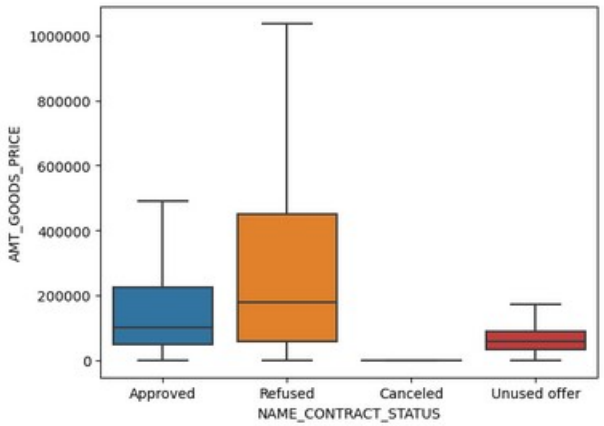




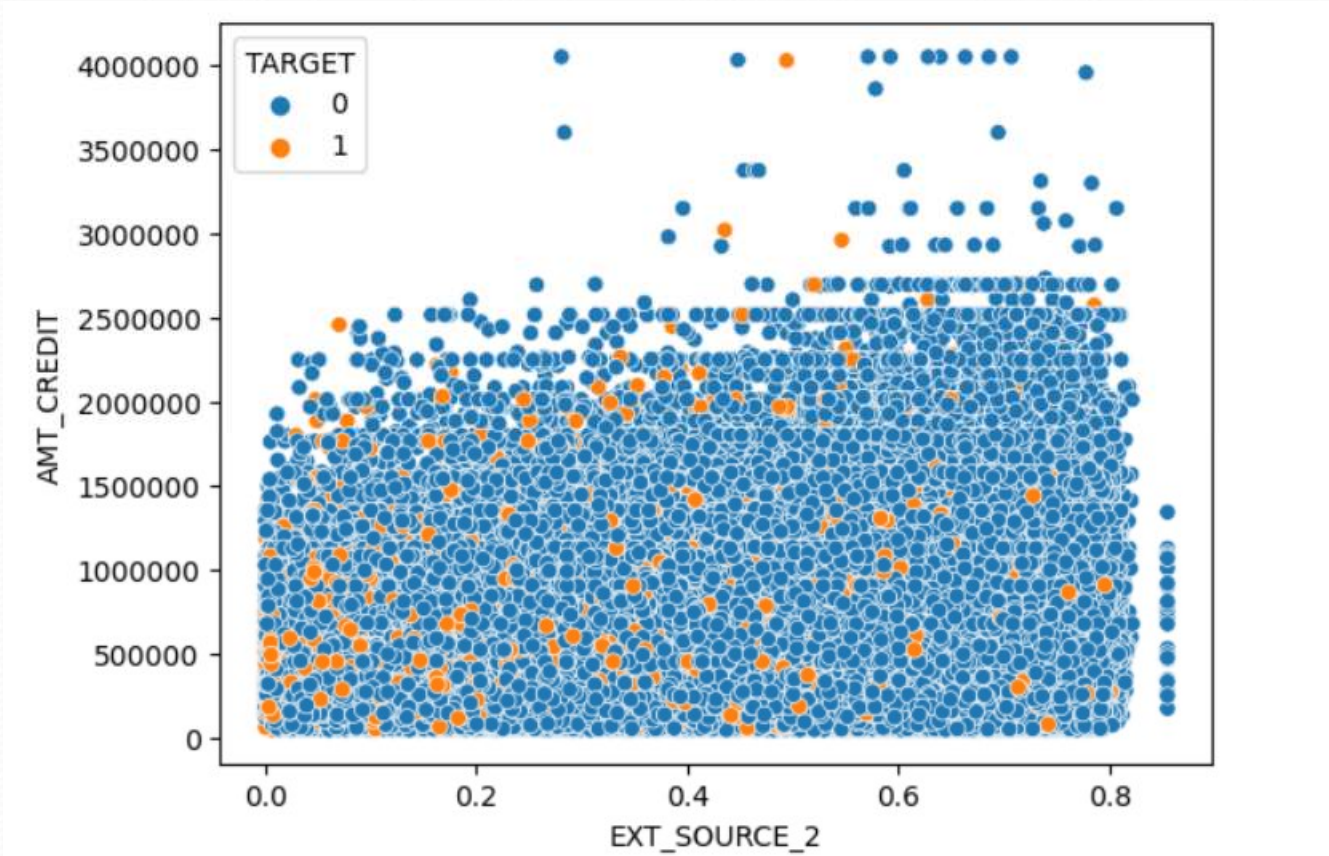
Complete Analysis can be found at the output of line 91 in the Jupyter notebook



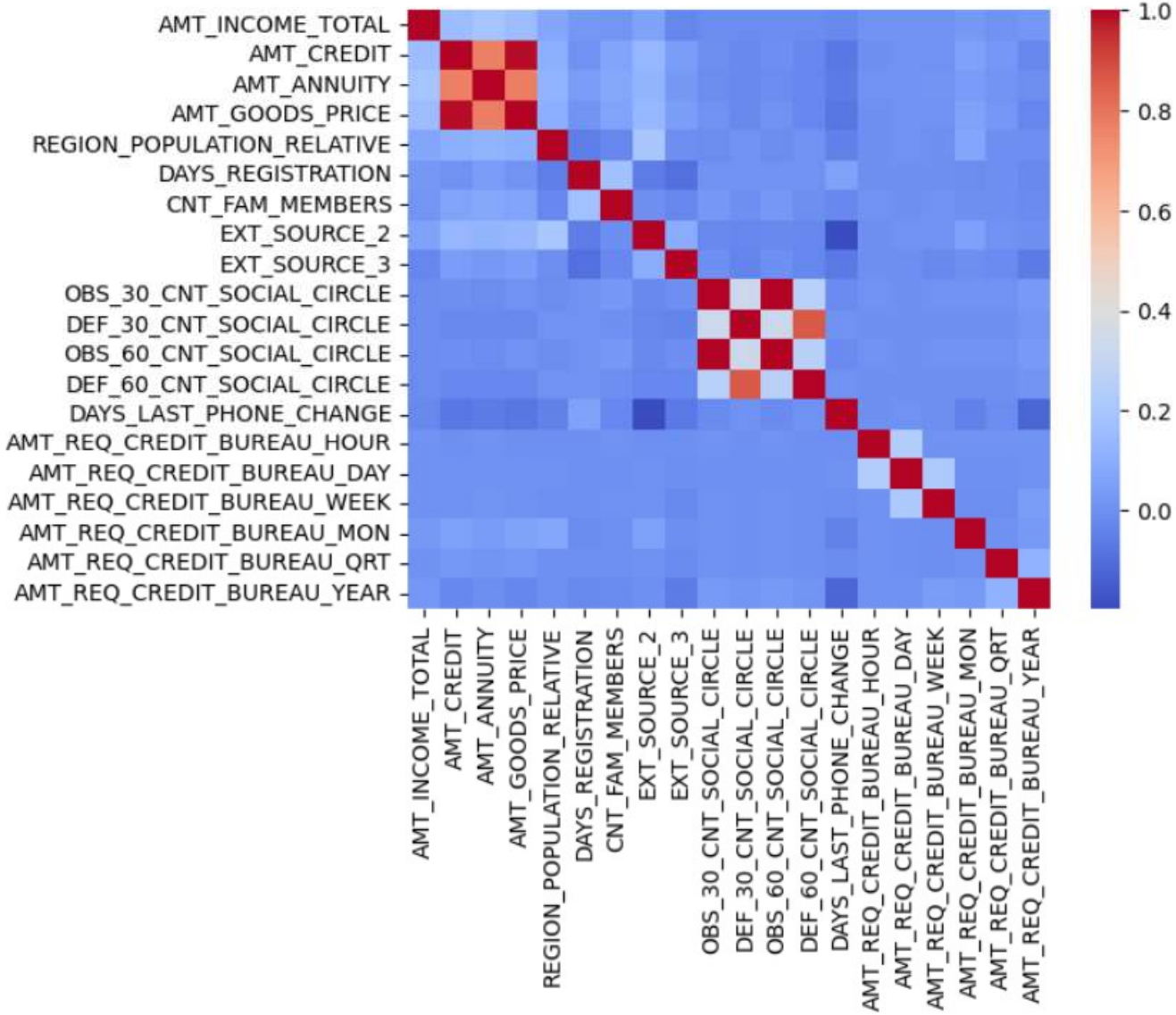




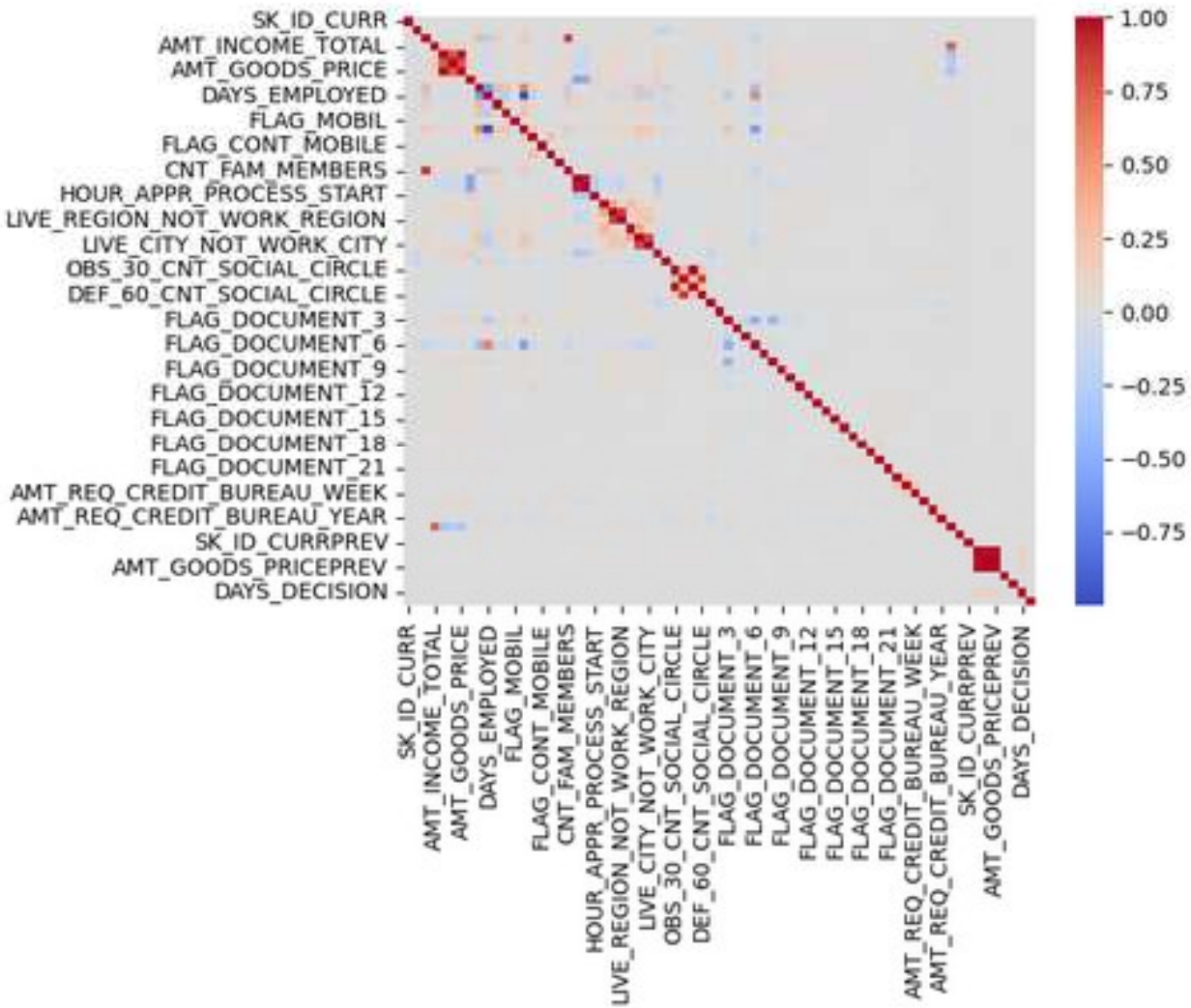
Complete Analysis can be found at the output of line 103 in the Jupyter notebook



Complete Analysis can be found at the output of line 96 in the Jupyter notebook



Similar heatmap were drawn for both the tables as well as the merged table, on line numbers 97, 105, and 118 of the Jupyter Notebook



Heatmap for the Merged Dataframe

Conclusions

- From the heat maps and the graphs plotted as part of EDA, it can be observed that very low correlation is observed between large part of the dataset.
- Among the components that significantly affect the target outcome, the following are most relevant

Column	Observation
AMT_CREDIT	Higher value implies lower positive outcome
AMT_ANNUITY	Higher value implies lower positive outcome
AMT_GOODS_PRICE	Higher value implies lower positive outcome
REGION_POPULATION_RELATIVE	Higher value implies lower positive outcome
DAYS_REGISTRATION	Higher negative value implies lower positive outcome
EXT_SOURCE_2	Higher value implies lower positive outcome
EXT_SOURCE_3	Higher value implies lower positive outcome

- And the following columns exhibit high correlating among each other:
  - AMT_CREDIT, AMT_ANNUITY, AMT_GOODS_PRICE
  - DEF_60_CNT_SOCIAL_CIRCLE, DEF_30_CNT_SOCIAL_CIRCLE
  - OBS_60_CNT_SOCIAL_CIRCLE, OBS_30_CNT_SOCIAL_CIRCLE

Conclusions

- And the following column values are also observed to have a ***slight*** affect on the outcome of the target variable:
  - NAME_CONTRACT_TYPE
  - CODE_GENDER
  - NAME_TYPE_SUITE
  - NAME_INCOME_TYPE
  - NAME_EDUCATION_TYPE
  - NAME_FAMILY_STATUS

Thank You!