PREDICTING LOAN DEFAULTERS TO REDUCE NPA

- HARSHIT PAUNIKAR (IIIT BANGALORE)

BUSINESS OBJECTIVES :

The loan providing companies find it hard to give loans to the people due to their insufficient or non-existent credit history. Because of that, some consumers use it as their advantage by becoming a defaulter. The project aim is to identify patterns which indicate if a client has difficulty paying their instalments which may be used for taking actions such as denying the loan, reducing the amount of loan, lending (to risky applicants) at a higher interest rate, etc. This will ensure that the consumers capable of repaying the loan are not rejected. Identification of such applicants using EDA is the aim of this case study.

APPROACH :

We performed following steps to work on this project:

- Understanding business problem
- Data Understanding
- Data Exploration
- Data cleaning and preparation
- Analyzing data using various graphs
- Results & Recommendations

DATA ANALYSIS:

- Univariate and segmented univariate analysis is done and appropriate realistic assumptions are made wherever required. The analyses successfully identify at least the 5 important driver variables (i.e. variables which are strong indicators of default).
- Business-driven, type-driven and data-driven metrics are created for the important variables and utilized for analysis. The explanation for creating the derived metrics is mentioned.
- Bivariate analysis is performed and is able to identify the important combinations of driver variables. The combinations of variables are chosen such that they make business or analytical sense.
- The most useful insights are explained.
- Appropriate plots are created to present the results of the analysis. The plots are clearly presented & the relevant insights are given. The axes and important data points are labelled.

Current Scenario : Using the application_data.csv file Normal checks

- *Import the Data:*

```python
# reading application_data.csv
df= pd.read_csv(r"D:\python\data\CR\application_data.csv")
applicationdata_df=df
```

- *Head of the data*

```python
df.head()
```

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CRE
0	100002	1	Cash loans	M	N	Y	0	202500.0	4065
1	100003	0	Cash loans	F	N	N	0	270000.0	12935
2	100004	0	Revolving loans	M	Y	Y	0	67500.0	1350
3	100006	0	Cash loans	F	N	Y	0	135000.0	3126
4	100007	0	Cash loans	M	N	Y	0	121500.0	5130

- *Tail of the data*

```python
df.tail()
```

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AM1
307506	456251	0	Cash loans	M	N	N	0	157500.0	
307507	456252	0	Cash loans	F	N	Y	0	72000.0	
307508	456253	0	Cash loans	F	N	Y	0	153000.0	
307509	456254	1	Cash loans	F	N	Y	0	171000.0	
307510	456255	0	Cash loans	F	N	N	0	157500.0	

- *Shape of the data:*

```python
df.shape
```

```
(307511, 122)
```

Current Scenario : Using the application_data.csv file Data Quality checks

- *Data Type verification:*
  - *All the data columns were Present as the required data format*

```
# check info
df.info(verbose = True)
```

```
SK_ID_CURR                      int64
TARGET                          int64
NAME_CONTRACT_TYPE              object
CODE_GENDER                     object
FLAG_OWN_CAR                    object
FLAG_OWN_REALTY                 object
CNT_CHILDREN                    int64
AMT_INCOME_TOTAL                float64
AMT_CREDIT                      float64
AMT_ANNUITY                     float64
AMT_GOODS_PRICE                 float64
NAME_TYPE_SUITE                 object
NAME_INCOME_TYPE                object
NAME_EDUCATION_TYPE             object
NAME_FAMILY_STATUS              object
NAME_HOUSING_TYPE               object
REGION_POPULATION_RELATIVE      float64
DAYS_BIRTH                      int64
DAYS_EMPLOYED                   int64
DAYS_REGISTRATION               float64
```

- *Data Percentage of missing values :*
  - *Below is the list of data column With the high missing value*
    - *COMMONAREA_MEDI*
    - *COMMONAREA_AVG*
    - *COMMONAREA_MODE*
    - *NONLIVINGAPARTMENTS_MODE*
    - *NONLIVINGAPARTMENTS_MEDI*
    - *NONLIVINGAPARTMENTS_AVG*
    - *FONDKAPREMONT_MODE*
    - *LIVINGAPARTMENTS_MEDI*
    - *LIVINGAPARTMENTS_MODE*
    - *LIVINGAPARTMENTS_AVG*
    - *FLOORSMIN_MEDI*
    - *FLOORSMIN_MODE*
    - *FLOORSMIN_AVG*
    - *YEARS_BUILD_MEDI*
    - *YEARS_BUILD_AVG*
    - *YEARS_BUILD_MODE*
    - *OWN_CAR_AGE*

```
missingvalues = df.count()/len(df)
missingvalues = (1-missingvalues)*100
```

```
missingvalues.sort_values(ascending=False)
```

```
COMMONAREA_MEDI                 69.872297
COMMONAREA_AVG                  69.872297
COMMONAREA_MODE                 69.872297
NONLIVINGAPARTMENTS_MODE        69.432963
NONLIVINGAPARTMENTS_MEDI        69.432963
NONLIVINGAPARTMENTS_AVG         69.432963
FONDKAPREMONT_MODE              68.386172
LIVINGAPARTMENTS_MEDI           68.354953
LIVINGAPARTMENTS_MODE           68.354953
LIVINGAPARTMENTS_AVG            68.354953
FLOORSMIN_MEDI                  67.848630
FLOORSMIN_MODE                  67.848630
FLOORSMIN_AVG                   67.848630
YEARS_BUILD_MEDI                66.497784
YEARS_BUILD_AVG                 66.497784
YEARS_BUILD_MODE                66.497784
OWN_CAR_AGE                     65.990810
LANDAREA_MODE                   59.376738
LANDAREA_AVG                    59.376738
LANDAREA_MEDI                   59.376738
```

Current Scenario : Using the application_data.csv file Data Quality checks

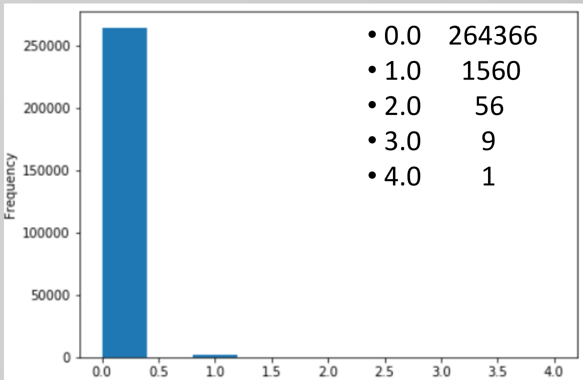
- *Data Percentage of missing values :*
  - *Below is the list of data column With the high missing value*
    - *AMT_REQ_CREDIT_BUREAU_QRT*
    - *AMT_REQ_CREDIT_BUREAU_YEAR*
    - *AMT_REQ_CREDIT_BUREAU_WEEK*
    - *AMT_REQ_CREDIT_BUREAU_MON*
    - *AMT_REQ_CREDIT_BUREAU_DAY*
    - *AMT_REQ_CREDIT_BUREAU_HOUR*
    - *NAME_TYPE_SUITE*
    - *OBS_30_CNT_SOCIAL_CIRCLE*
    - *OBS_60_CNT_SOCIAL_CIRCLE*
    - *DEF_60_CNT_SOCIAL_CIRCLE*
    - *DEF_30_CNT_SOCIAL_CIRCLE*

AMT_REQ_CREDIT_BUREAU_QRT	13.501631
AMT_REQ_CREDIT_BUREAU_YEAR	13.501631
AMT_REQ_CREDIT_BUREAU_WEEK	13.501631
AMT_REQ_CREDIT_BUREAU_MON	13.501631
AMT_REQ_CREDIT_BUREAU_DAY	13.501631
AMT_REQ_CREDIT_BUREAU_HOUR	13.501631
NAME_TYPE_SUITE	0.420148
OBS_30_CNT_SOCIAL_CIRCLE	0.332021
OBS_60_CNT_SOCIAL_CIRCLE	0.332021
DEF_60_CNT_SOCIAL_CIRCLE	0.332021
DEF_30_CNT_SOCIAL_CIRCLE	0.332021
EXT_SOURCE_2	0.214626
AMT_GOODS_PRICE	0.090403
AMT_ANNUITY	0.003902
CNT_FAM_MEMBERS	0.000650
DAYS_LAST_PHONE_CHANGE	0.000325
AMT_CREDIT	0.000000
FLAG_OWN_CAR	0.000000
FLAG_EMAIL	0.000000
TARGET	0.000000

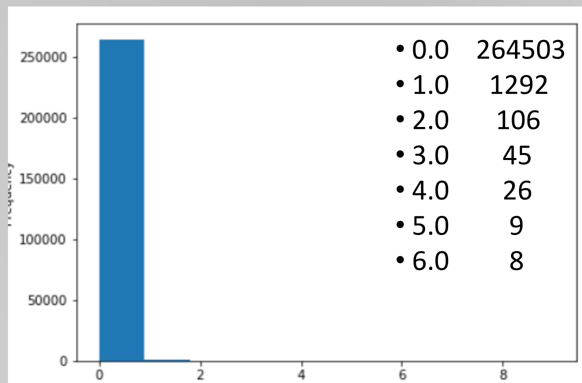
- *Imputing the Data*

  From the above data imputing is required only for the 5 columns, for which we plotted the frequency distribution as

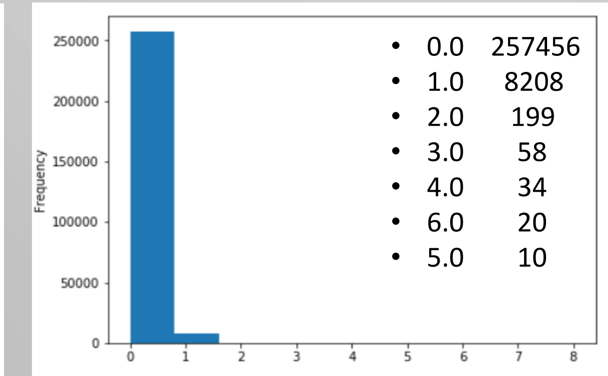
- AMT_REQ_CREDIT_BUREAU_HOUR



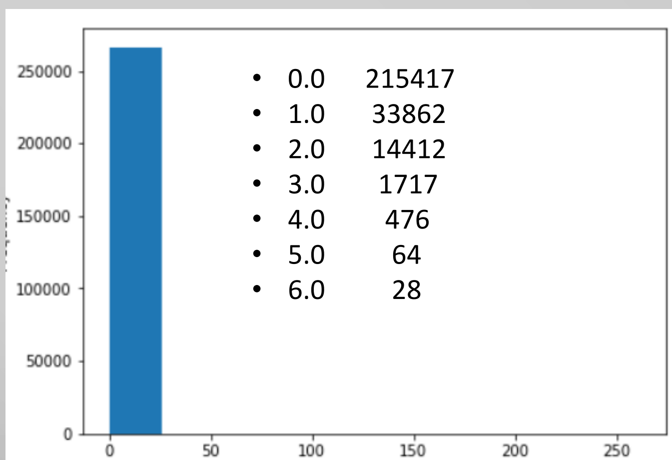
- AMT_REQ_CREDIT_BUREAU_HOUR



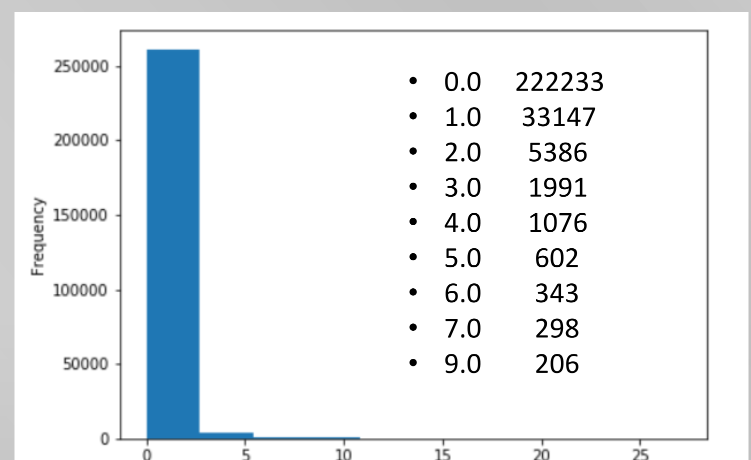
- AMT_REQ_CREDIT_BUREAU_WEEK



- AMT_REQ_CREDIT_BUREAU_QRT



- AMT_REQ_CREDIT_BUREAU_MON

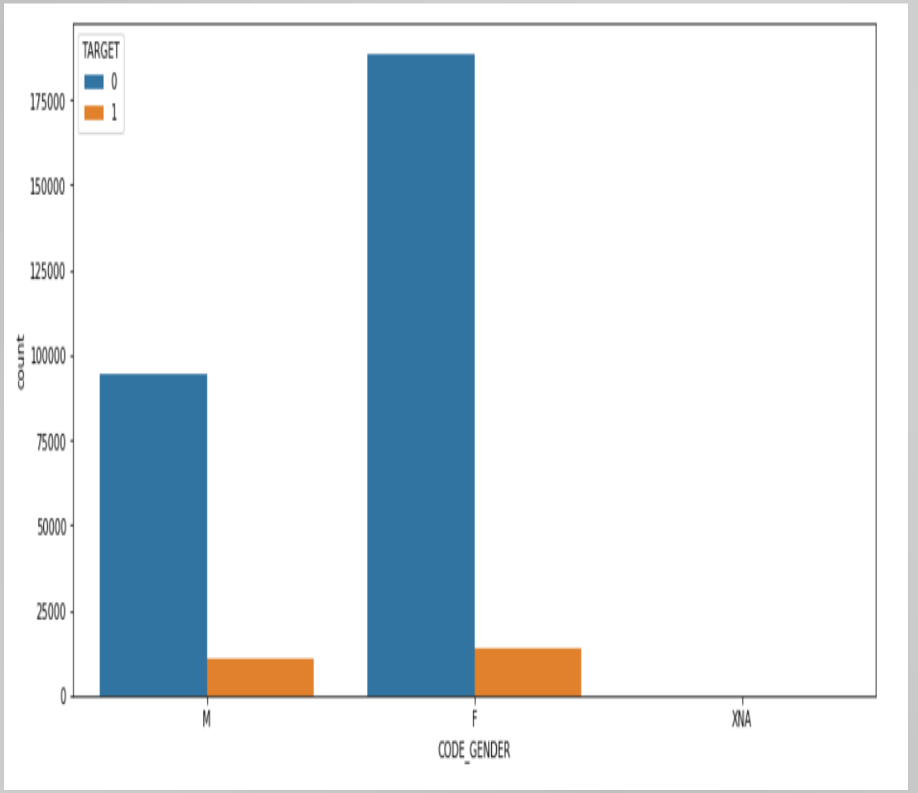


Conclusion: from the above data distribution we can concluded not to impute the data since the ~ 97% values as zero (0).

Univariate analysis : CODE_GENDER

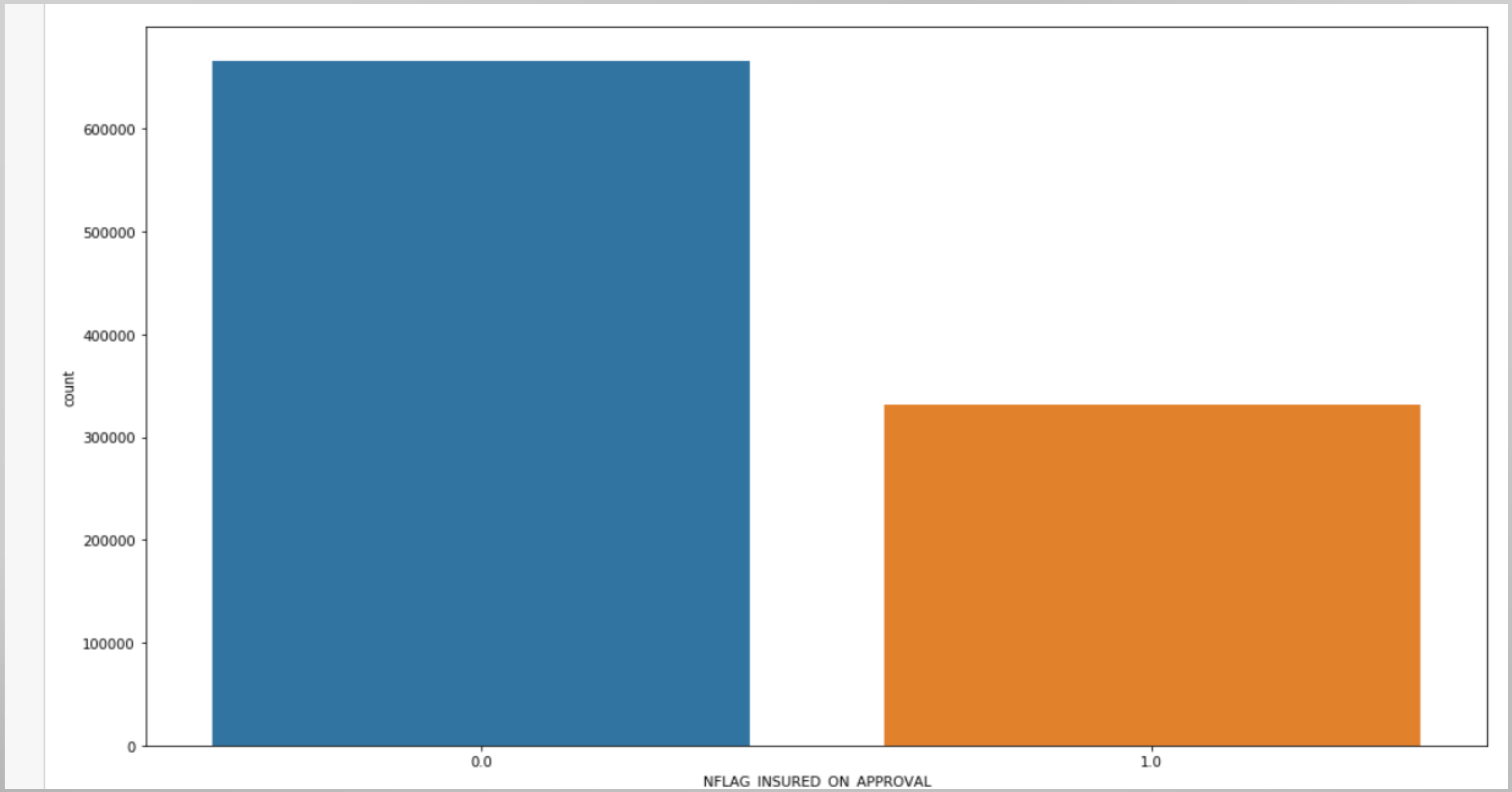
From the data distribution plot we can say that the number of male applicant is much higher than the number of female applicant. Also the same pattern is followed by both the target values (0,1).

We can conclude that this column Code_Gender wont give any substantial evidence whether the Bank approve the loan or not.



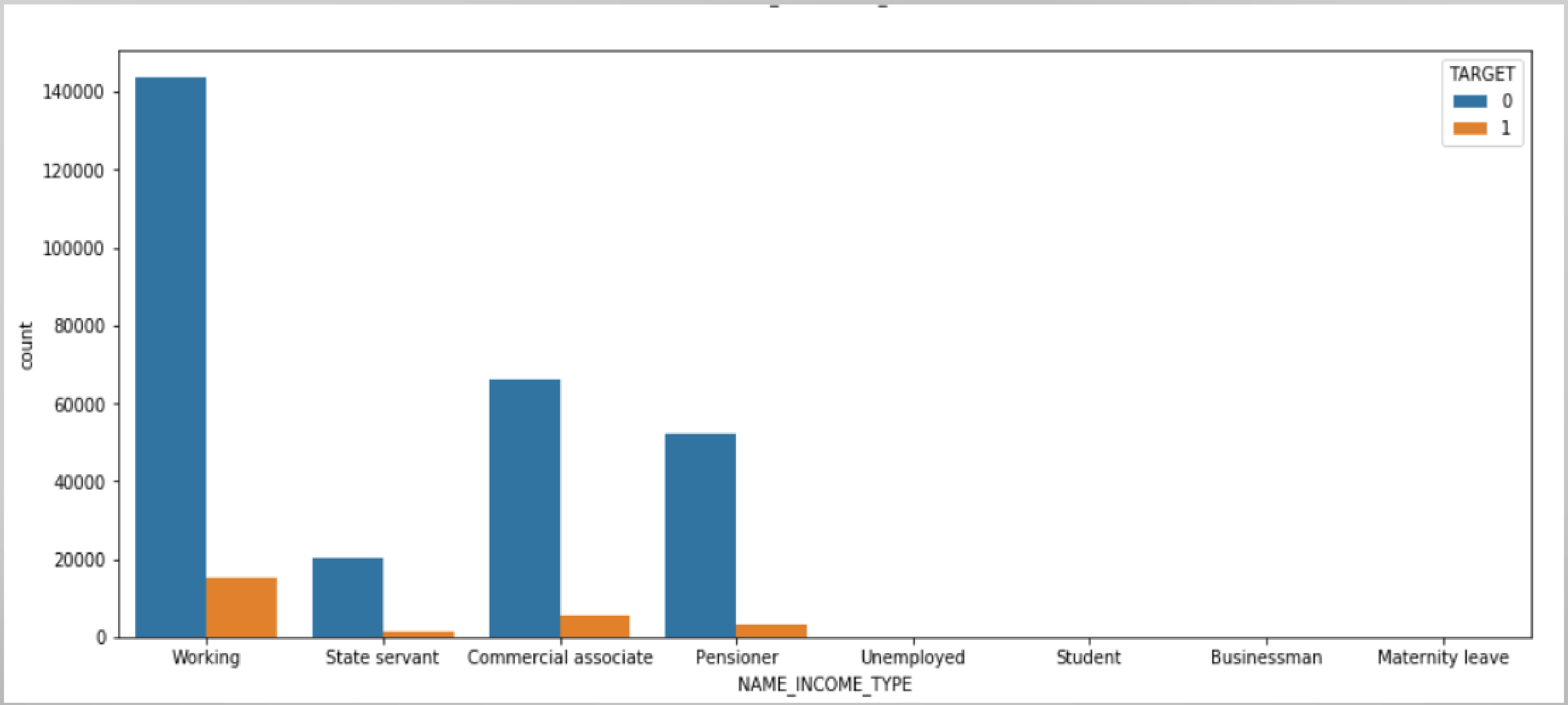
Univariate analysis : NFLAG_INSURED_ON_APPROVAL

From the data distribution plot the people facing challenges in repaying the loan are not opting for the insurance . Certainly, this is an indicator before approving.



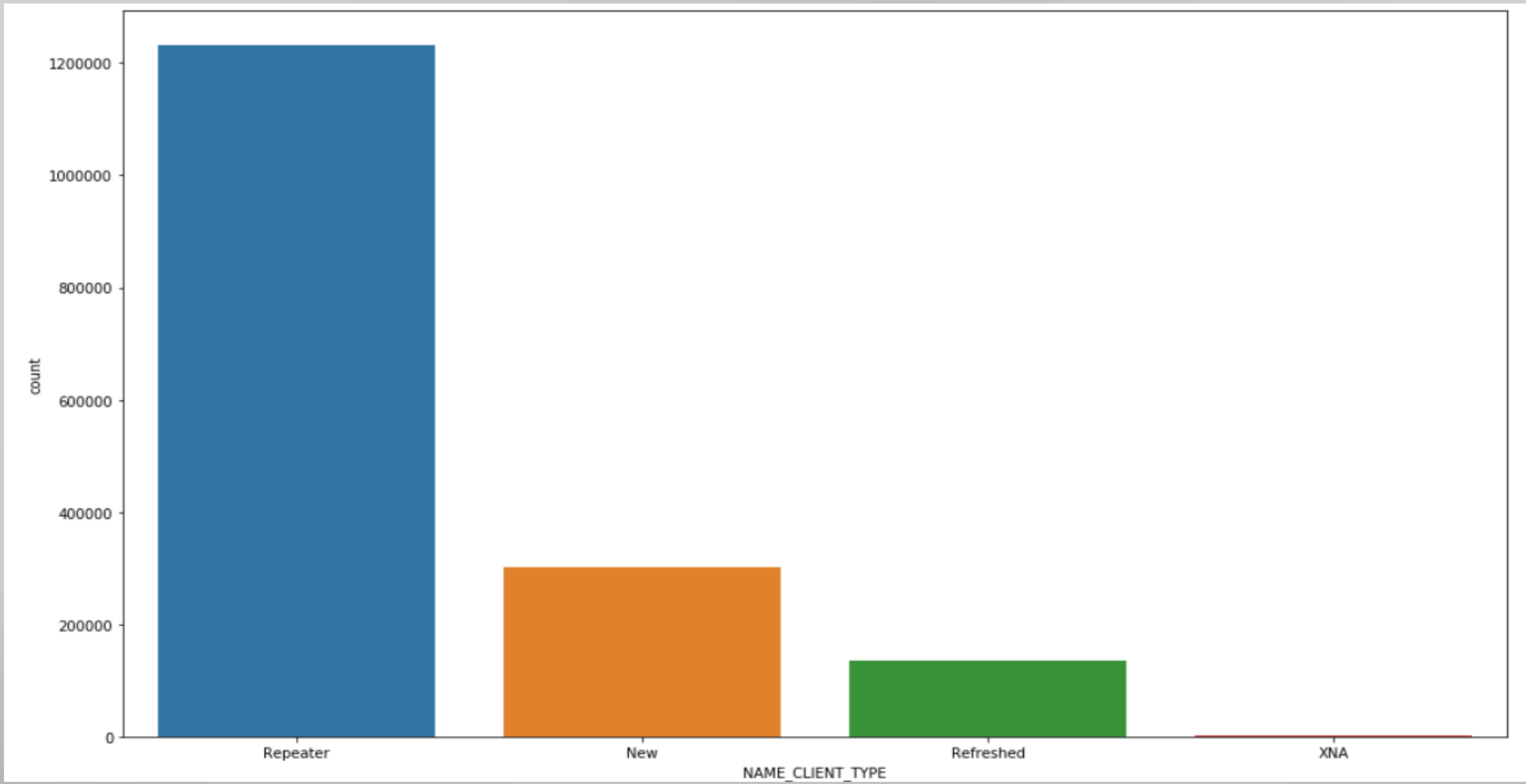
Univariate analysis : NAME_INCOME_TYPE

From the data distribution plot we can say the people working people, commercial associate and Pensioner follows trend for both the values of target where as the State servant category is marginally low. Also the defaulting in the working category is very low , so we can should consider this column approving or denying the loan



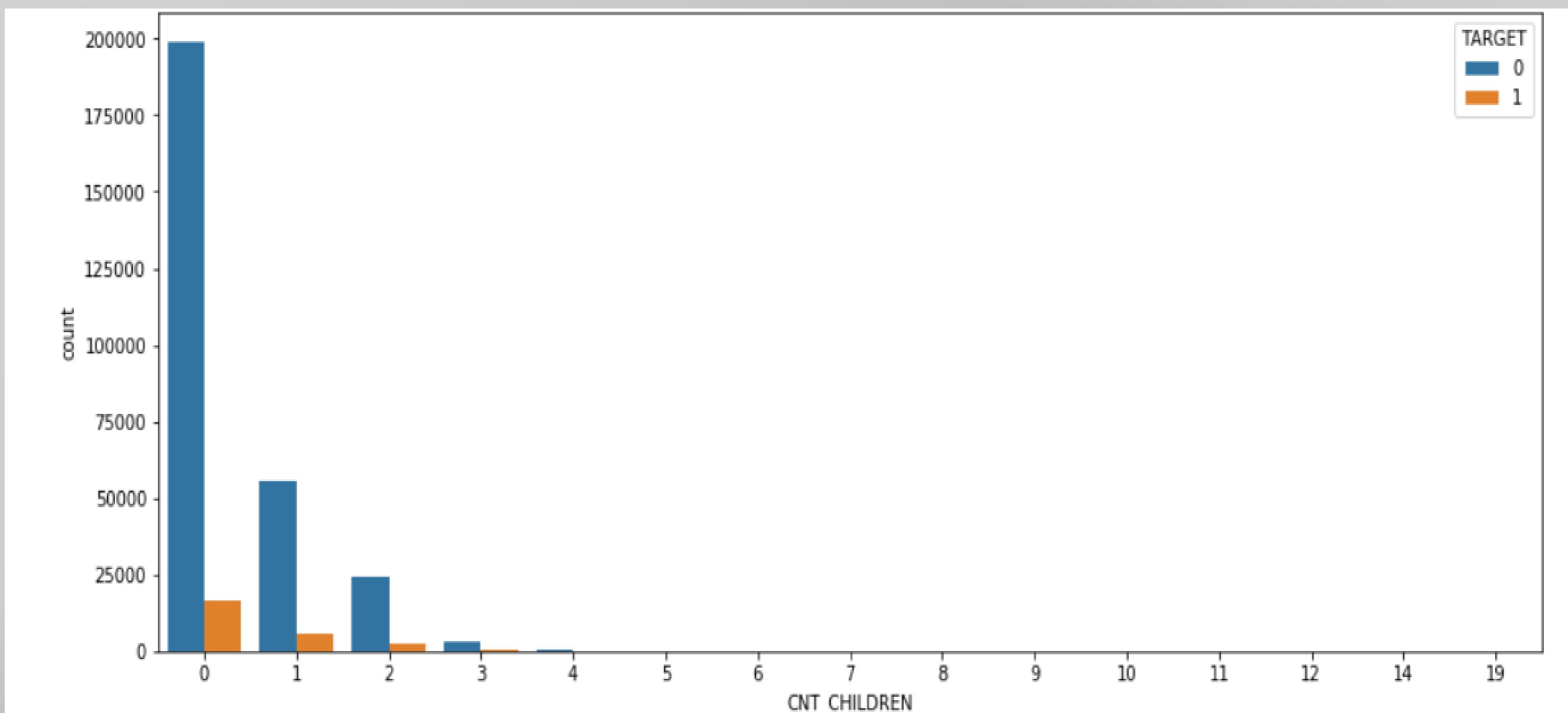
Univariate analysis : NAME_CLIENT_TYPE

From the data distribution plot we can say the higher rete of people getting loan is from the Repeater category i.e. existing customers are asking for another loan. This variable useful since we have track record for the customer.



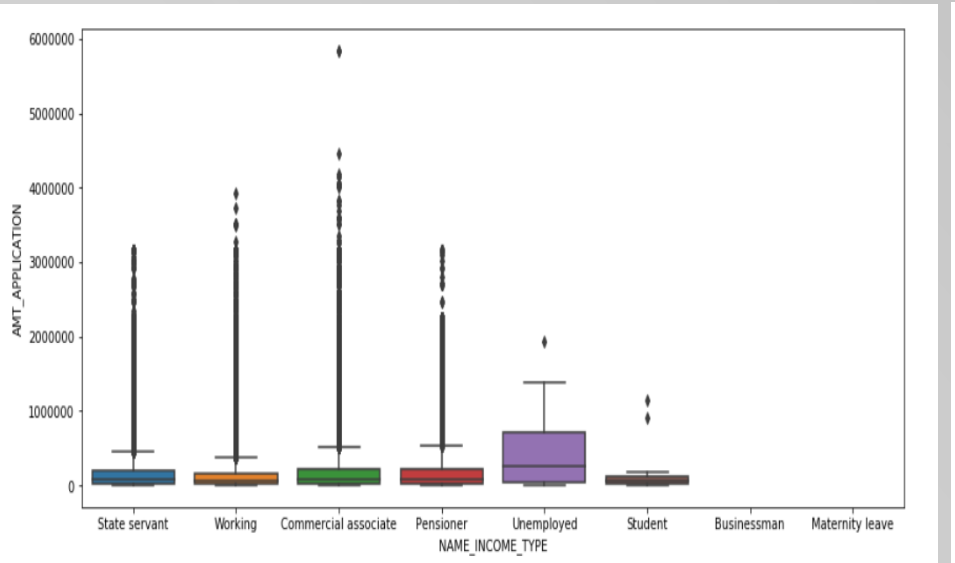
Univariate analysis :CNT_Children

From the data distribution plot we can say the higher rete of people applying for the loan with having children 1 or 0.

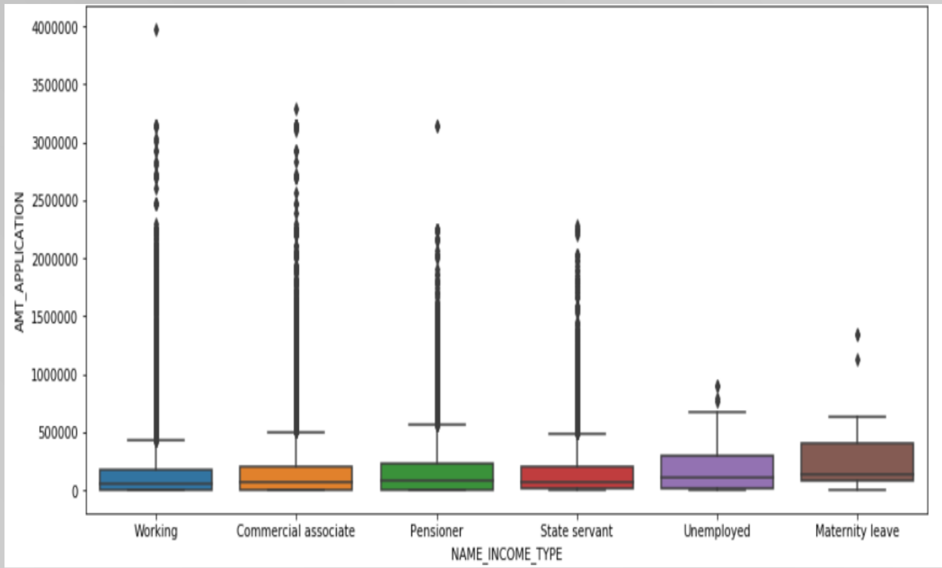


Bivariate analysis : NAME_INCOME_TYPE & AMT_APPLICATION

From the data distribution plot we can say that the commercial associated are asking for the higher value of loan followed by working and State servant. Also earlier we saw the density of the working category is high but the volume of commercial associated is high



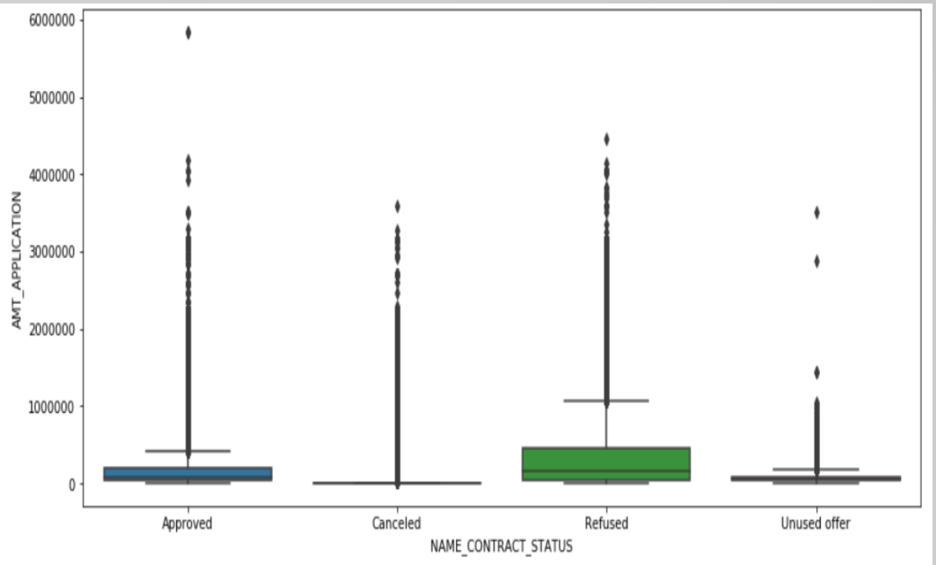
Target variable 0



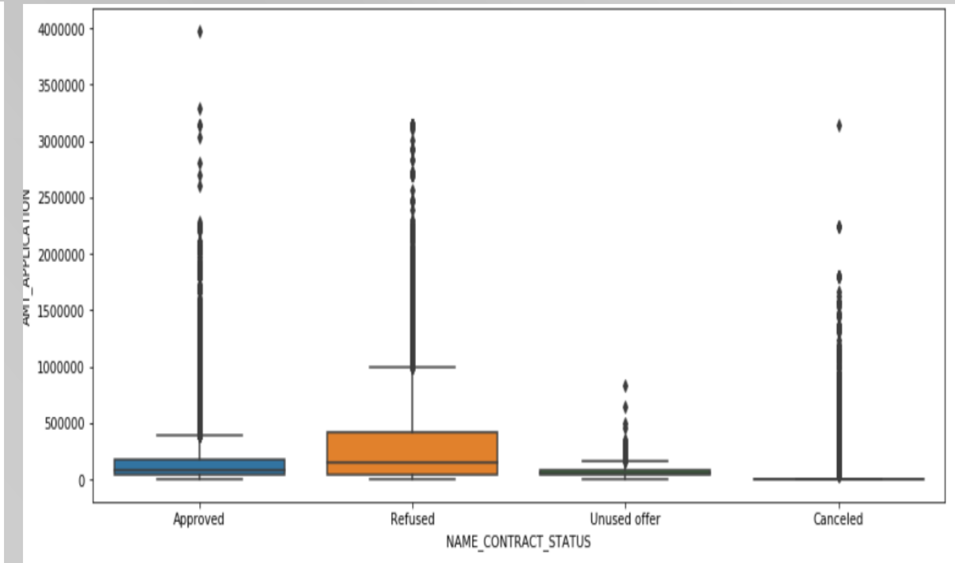
Target variable 1

Bivariate analysis : NAME_CONTRACT_STATUS & AMT_APPLICATION

From the data distribution plot we can say that in the Refused category is having much density over the approved in the lover amount of application since their behaviour as per the target column i.e. they faces difficulties in repaying in previous records . Also many offers were remain unused in the lover amount.

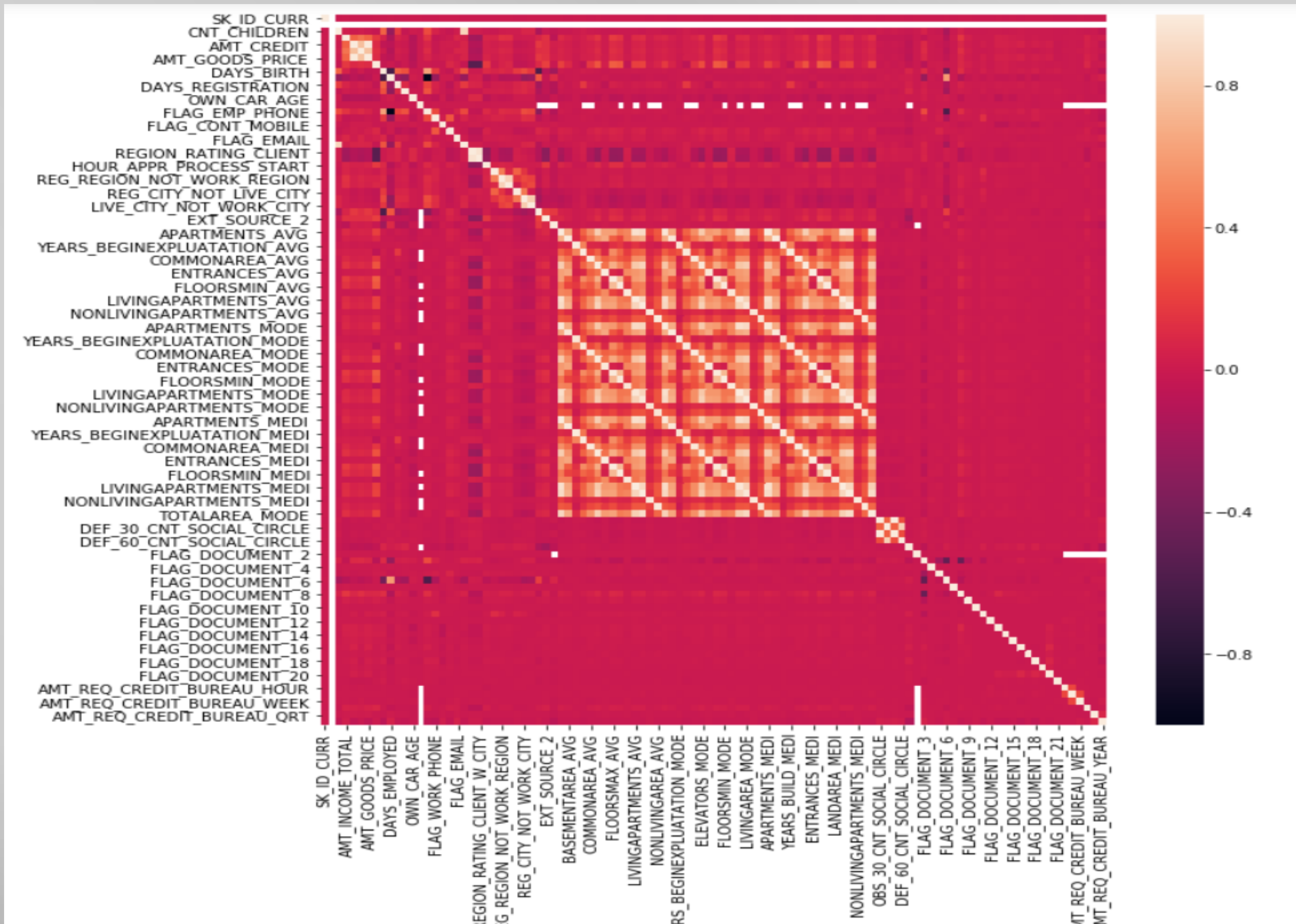


Target variable 0

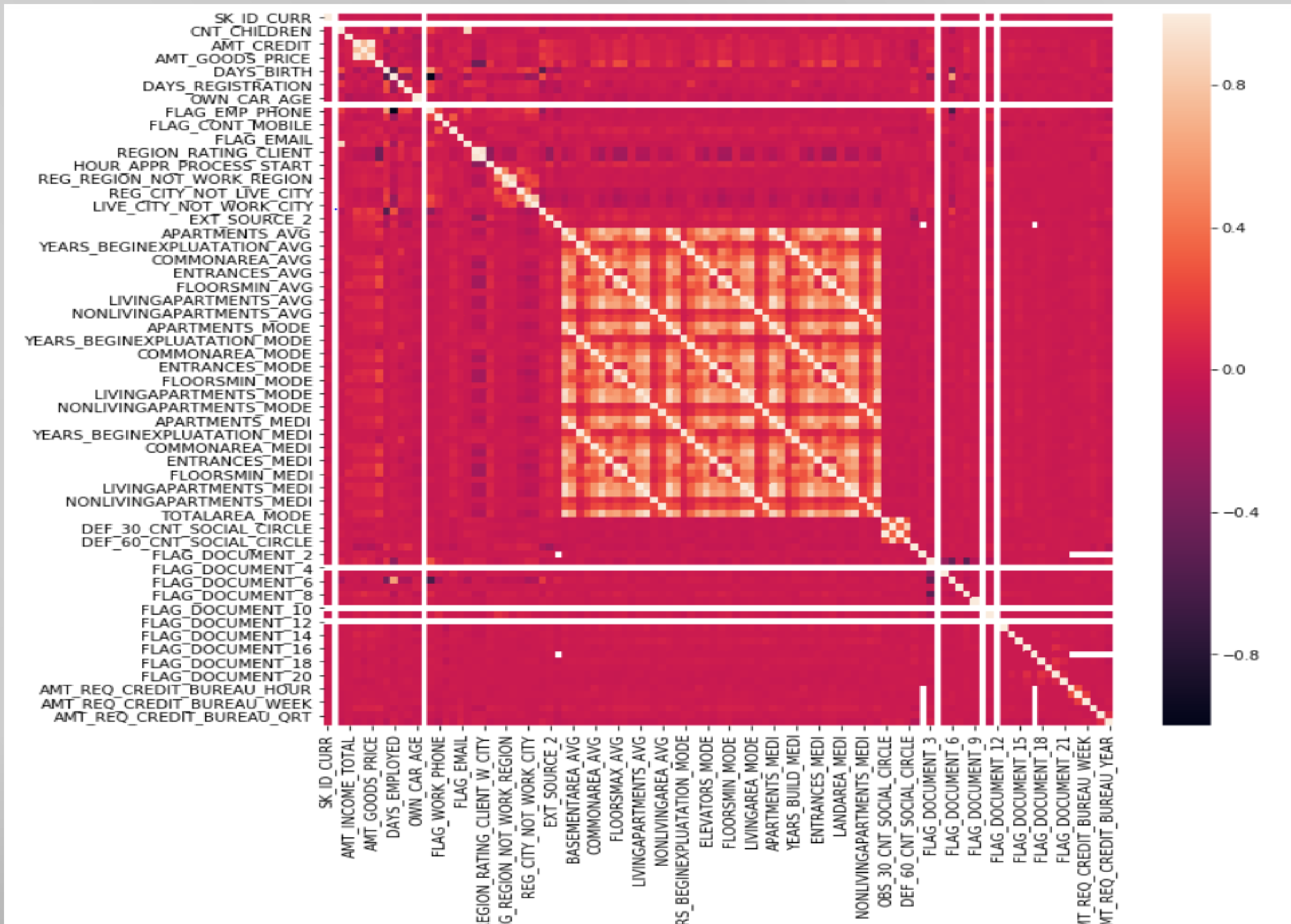


Target variable 1

Correlation for Target variable value 1



Correlation for Target variable value 0



Conclusion:

- *After the entire exercise we concluded the below observation*
  - *The density of loan getting approved by bank is much higher in working category but the volume is much higher in commercial associated*
  - *Bank are preferring to pay loan to the people with less dependencies ie more loan are getting approved for people having children 2 or less*
  - *Insurance is the key factor to be consider while approving the loan as the population facing challenges in replay have not opted for insurance.*
  - *Also Bank prefers to approve/offer the loan for existing population ie. their own customer.*