CREDIT EDA CASE STUDY

JAYAKRISHNAN RADHAKRISHNAN

PRABITHA BALAKRISHNAN

INTRODUCTION

This case study aims to give you an idea of applying EDA in a real time business scenario. In this case study of risk analytics in banking and financial services, we understand how data is used to minimize the risk of losing money while lending to customers.

BUSINESS UNDERSTANDING

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. Use EDA to analyze the patterns present in the data. This will ensure that the applicants capable of repaying the loan, are not rejected.

BUSINESS OBJECTIVES

This case study aims to identify patterns which indicate if a client has difficulty paying their installments, 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.

SOLUTION APPROACH

- We have followed following EDA approach for this solution.
- Data Understanding :
  - Sampling of data to find out data definitions.
  - Analyze the data types of each of the columns and if needed try to modify the data types suitable for our analysis.
  - Try to understand the all the columns that are available and try to identify the variables for our univariate and bivariate analysis.
- Data Cleaning :
  - As mentioned in the requirement we have removed the columns which has null value percentage more than 50% and we have impute the variables with zero which has null value more than 13% and again the variables which has less than 13% null value we have imputed with median for numerical variable.
  - For categorical variables the null value should be imputed with mode value of the respective column.

Reading of Data frame:

We have two data :current application data and Previous application data

In [2]:

```
# For displaying all rows and columns
pd.options.display.max_columns=None
pd.options.display.max_rows=None
app_data=pd.read_csv(r"C:\Users\Prabitha's PC\Documents\DS2021\EDA\Eda case study dataset\application_data.csv")
app_data.head(5)
```

Out[2]:

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

2.Inspecting Application_data

In [3]:

```
app_data.shape
```

Out[3]: (307511, 122)

[120]:

```
df_previous_app.head()
```

Out[120]:

	SK_ID_PREV	SK_ID_CURR	NAME_CONTRACT_TYPE	AMT_ANNUITY	AMT_APPLICATION	AMT_CREDIT	AMT_DOWN_PAYMENT	AMT_GOODS_PRICE	W
0	2030495	271877	Consumer loans	1730.430	17145.0	17145.0	0.0	17145.0	
1	2802425	108129	Cash loans	25188.615	607500.0	679671.0	NaN	607500.0	
2	2523466	122040	Cash loans	15060.735	112500.0	136444.5	NaN	112500.0	
3	2819243	176158	Cash loans	47041.335	450000.0	470790.0	NaN	450000.0	
4	1784265	202054	Cash loans	31924.395	337500.0	404055.0	NaN	337500.0	

[121]:

```
df_previous_app.shape
```

Out[121]: (1670214, 37)

Understanding the various Features of data

- FINDING NULL VALUES IN EACH COLUMN

3.1 checking for percentage of null values in each columns.

```
In [6]:  1 (app_data.isnull().sum()/len(app_data.index)*100).sort_values(ascending=False)

Out[6]: 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
```

- FINDING DATATYPE OF COLUMNS

```
1 app_data.info(verbose=True)

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307511 entries, 0 to 307510
Data columns (total 81 columns):
 #   Column                       Non-Null Count   Dtype
---  ------                       --------------   -----
 0   SK_ID_CURR                   307511 non-null  int64
 1   TARGET                       307511 non-null  int64
 2   NAME_CONTRACT_TYPE           307511 non-null  object
 3   CODE_GENDER                  307511 non-null  object
 4   FLAG_OWN_CAR                 307511 non-null  object
 5   FLAG_OWN_REALTY              307511 non-null  object
 6   CNT_CHILDREN                 307511 non-null  int64
 7   AMT_INCOME_TOTAL             307511 non-null  float64
 8   AMT_CREDIT                   307511 non-null  float64
 9   AMT_ANNUITY                  307499 non-null  float64
 10  AMT_GOODS_PRICE              307233 non-null  float64
 11  NAME_TYPE_SUITE              306219 non-null  object
 12  NAME_INCOME_TYPE             307511 non-null  object
 13  NAME_EDUCATION_TYPE          307511 non-null  object
 14  NAME_FAMILY_STATUS           307511 non-null  object
 15  NAME_HOUSING_TYPE            307511 non-null  object
 16  REGION_POPULATION_RELATIVE   307511 non-null  float64
 17  DAYS_BIRTH                   307511 non-null  int64
```

Data Cleaning

- As mentioned in the requirement we have removed the columns which has null value percentage more than 50% and we have impute the variables with zero which has null value more than 13% and again the variables which has less than 13% null value we have imputed with median for numerical variable.
- For categorical variables the null value should be imputed with mode value of the respective column.

Columns like COMMONAREA_MEDI,COMMONAREA_AVG,LANDAREA_MODE,ELEVATORS_MEDI,and many more have null values higher than shown above.Since they will not contribute much to further studies we can drop those columns.

```
## drop columns with nullvalues higher than 50%
app_data=app_data[app_data.columns[app_data.isnull().sum()/len(app_data.index)<=.50]]
```

```
# checking shape of dataframe after removing columns with null values more than 50%.We have lost 41 columns.
app_data.shape
```

8]: (307511, 81)

AMT_REQ_CREDIT_BUREAU_HOUR,AMT_REQ_CREDIT_BUREAU_DAY, AMT_REQ_CREDIT_BUREAU_WEEK,AMT_REQ_CR
AMT_REQ_CREDIT_BUREAU_QRT,AMT_REQ_CREDIT_BUREAU_YEAR

As we already seen that for these columns have same mean,median values.We can impute them using value of 0.00

n [32]:

```
app_subset.AMT_REQ_CREDIT_BUREAU_HOUR=app_subset.AMT_REQ_CREDIT_BUREAU_HOUR.replace(np.nan,0)
app_subset.AMT_REQ_CREDIT_BUREAU_DAY=app_subset.AMT_REQ_CREDIT_BUREAU_DAY.replace(np.nan,0)
app_subset.AMT_REQ_CREDIT_BUREAU_WEEK=app_subset.AMT_REQ_CREDIT_BUREAU_WEEK.replace(np.nan,0)
app_subset.AMT_REQ_CREDIT_BUREAU_MON=app_subset.AMT_REQ_CREDIT_BUREAU_MON.replace(np.nan,0)
app_subset.AMT_REQ_CREDIT_BUREAU_QRT=app_subset.AMT_REQ_CREDIT_BUREAU_QRT.replace(np.nan,0)
app_subset.AMT_REQ_CREDIT_BUREAU_YEAR=app_subset.AMT_REQ_CREDIT_BUREAU_YEAR.replace(np.nan,0)
```

2]:

```
## imputing null values in AMT_ANNUITY column
app_subset.AMT_ANNUITY.median()
```

ut[42]: 24903.0

3]:

```
app_subset.AMT_ANNUITY=app_subset.AMT_ANNUITY.replace(np.nan,app_subset.AMT_ANNUITY.median())
```

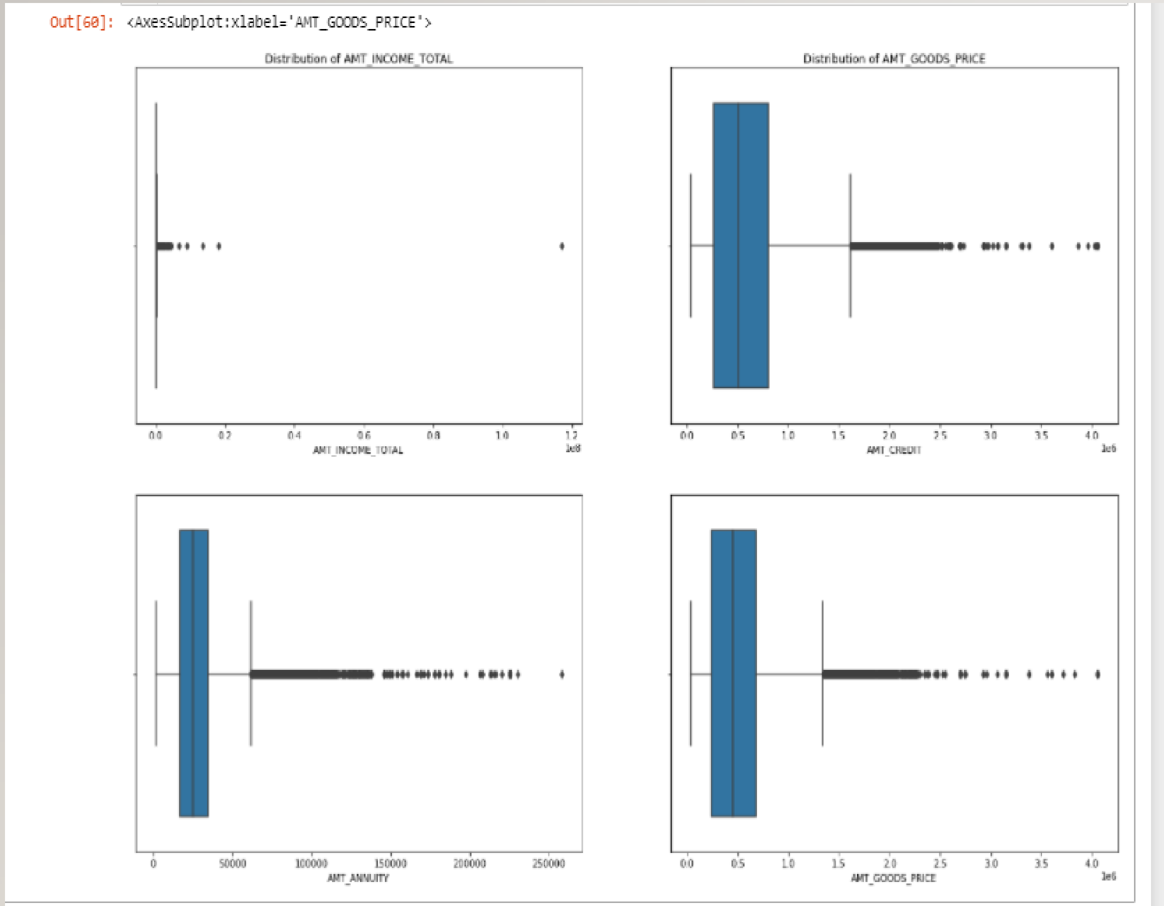
SUB SETTING OF COLUMNS

- For the purpose of study we considered following columns.

```
app_subset=app_data[['TARGET',
                     'SK_ID_CURR',
                     'NAME_CONTRACT_TYPE',
                     'CODE_GENDER',
                     'AMT_INCOME_TOTAL',
                     'NAME_TYPE_SUITE',
                     'NAME_INCOME_TYPE',
                     'NAME_EDUCATION_TYPE',
                     'DAYS_EMPLOYED',
                     'OCCUPATION_TYPE',
                     'ORGANIZATION_TYPE',
                     'AMT_REQ_CREDIT_BUREAU_HOUR',
                     'AMT_REQ_CREDIT_BUREAU_DAY',
                     'AMT_REQ_CREDIT_BUREAU_WEEK',
                     'AMT_REQ_CREDIT_BUREAU_MON',
                     'AMT_REQ_CREDIT_BUREAU_QRT',
                     'AMT_REQ_CREDIT_BUREAU_YEAR',
                     'AMT_GOODS_PRICE',
                     'AMT_CREDIT',
                     'AMT_ANNUITY',
                     'DAYS_BIRTH',
                     'EMERGENCYSTATE_MODE',
                     'NAME_FAMILY_STATUS',
                     'FLAG_OWN_REALTY',
                     'REGION_RATING_CLIENT_W_CITY',
                     'DEF_60_CNT_SOCIAL_CIRCLE',
                     'DEF_30_CNT_SOCIAL_CIRCLE',
                     'CNT_CHILDREN',
                     'CNT_FAM_MEMBERS']]
```

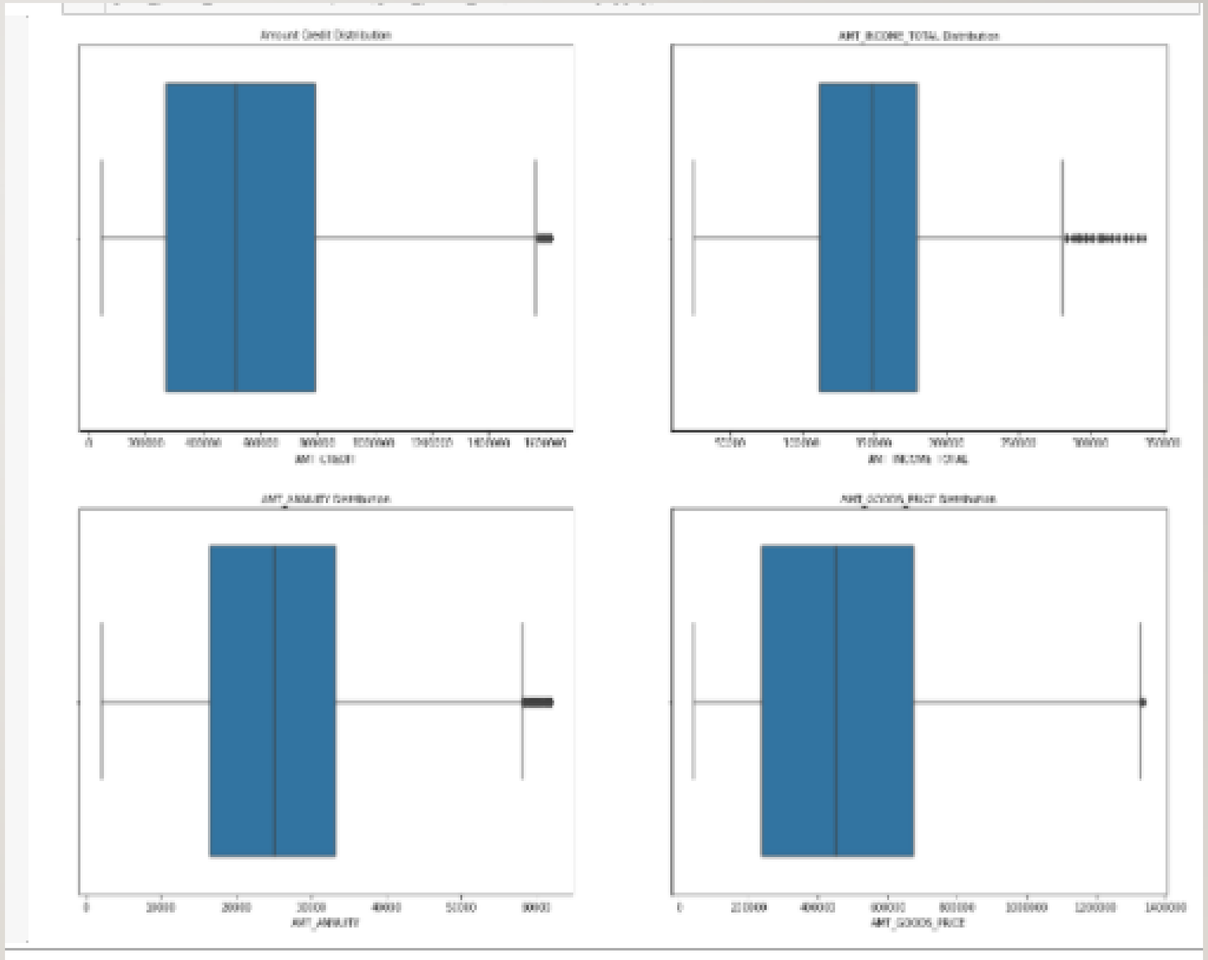
Univariate Analysis : Outlier Analysis

- Finding out outliers in application dataset



From the above boxplots it is clear that there are some outliers in these columns.So we have to treat these outliers.

- Outliers Treating Using IQR

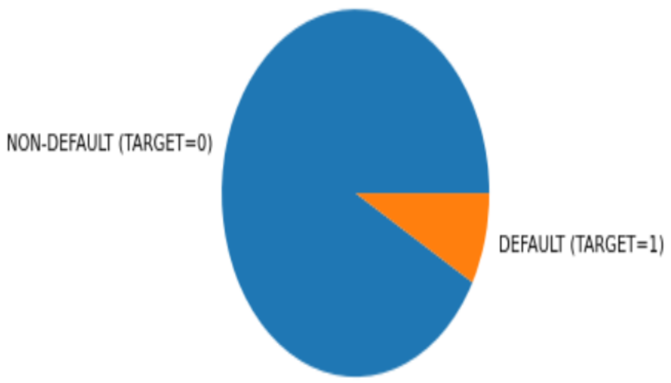


Checking For Imbalance

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

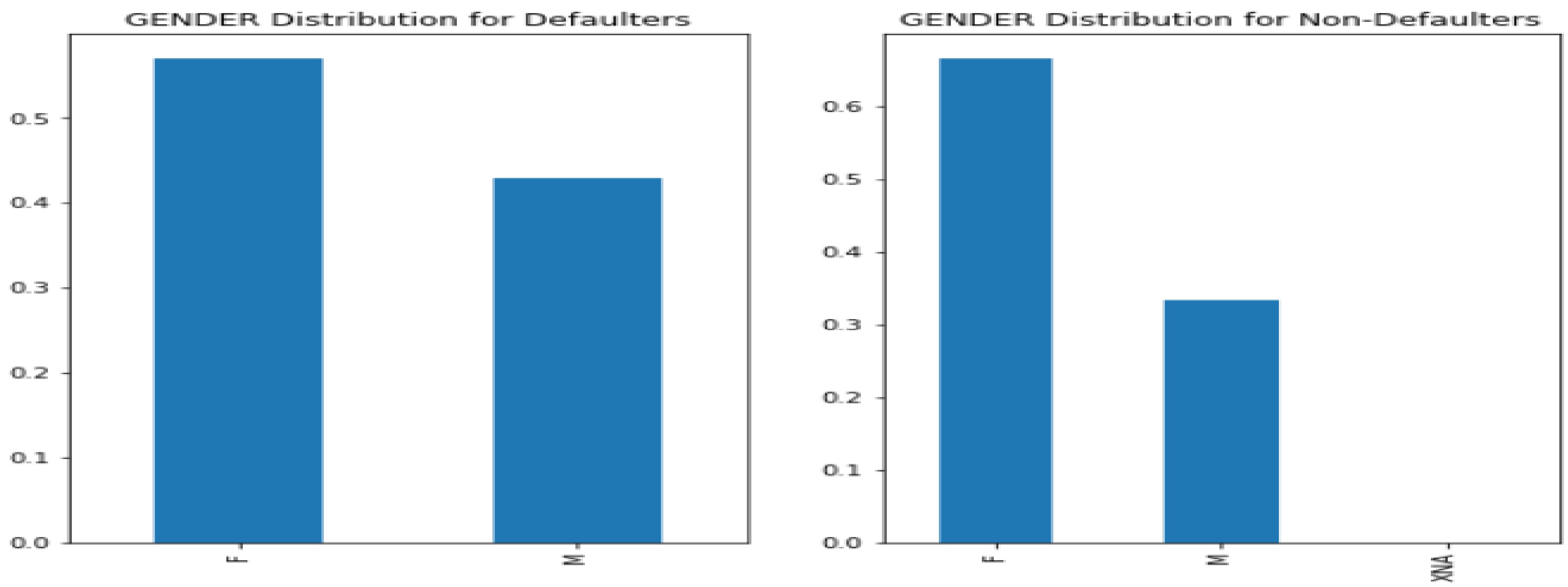
```
0    0.919271
1    0.080729
Name: TARGET, dtype: float64
```

```
plt.pie(app_subset.TARGET.value_counts(normalize=True),labels=['NON-DEFAULT (TARGET=0)','DEFAULT (TARGET=1)'])
plt.show()
```



- TARGET has 2 values 1 and 0.
- 1 shows - client with payment difficulties.
- 0 shows- client who will not make any default.
- Around 91% client will pay loan on time, while around 8% client shows tendency to default payment.

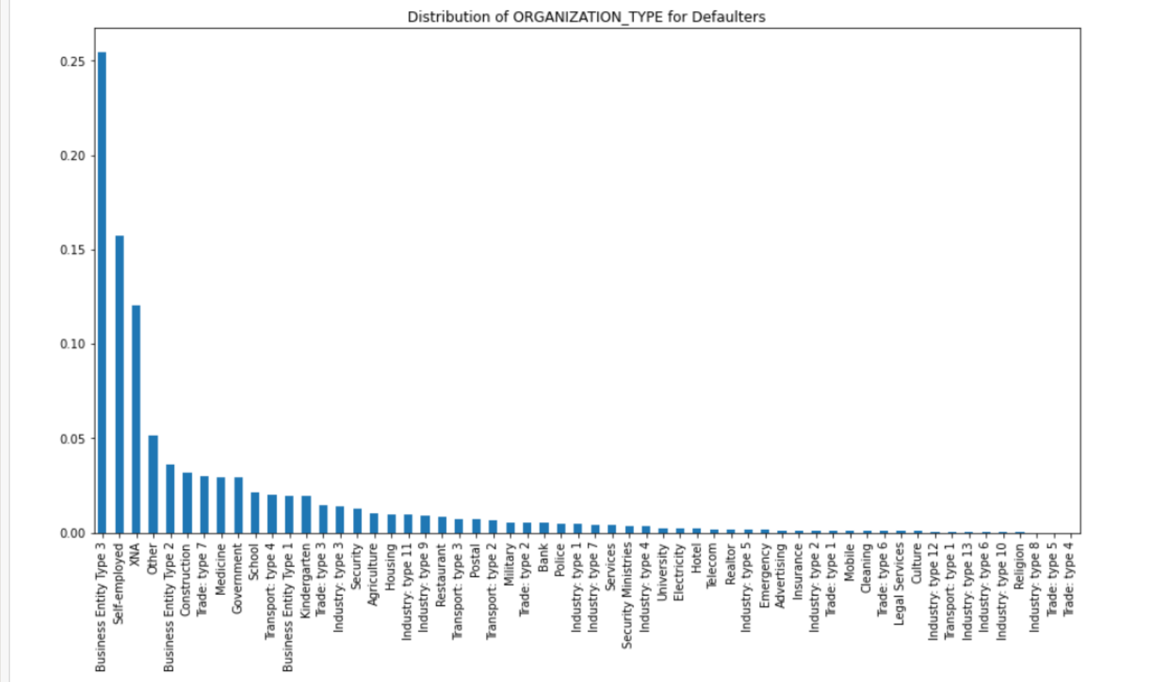
Analyzing Gender for Target variable

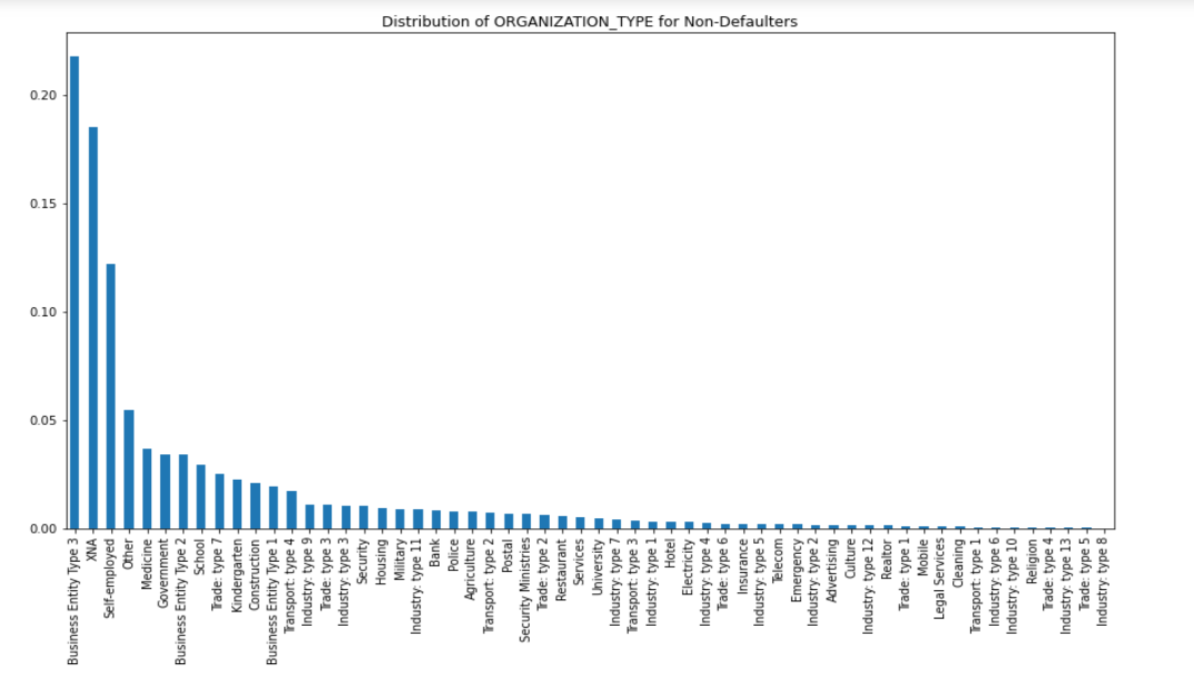


Obsevations:

67% Females contribute non-defaulters while 57% fail to make payment.
33% Male client contribute to non-defaulters while 42% fail to do payment.
We can see that more females are applying for loan.
Rate of defaulting is found to be high for males than females.

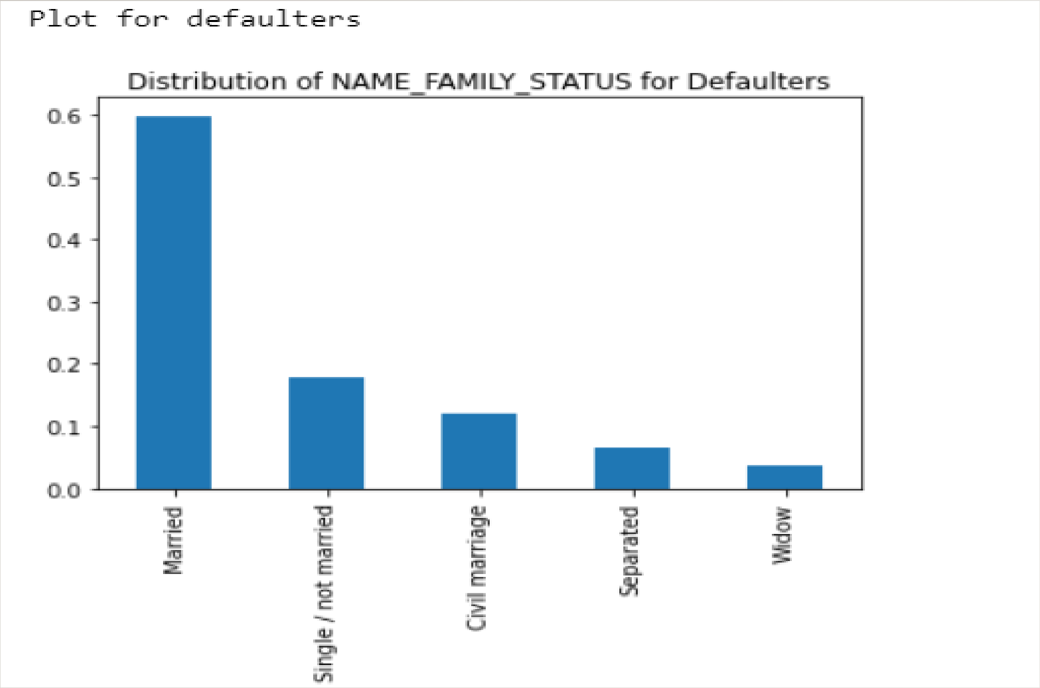
Analyzing ORGANIZATION_TYPE for Target variable

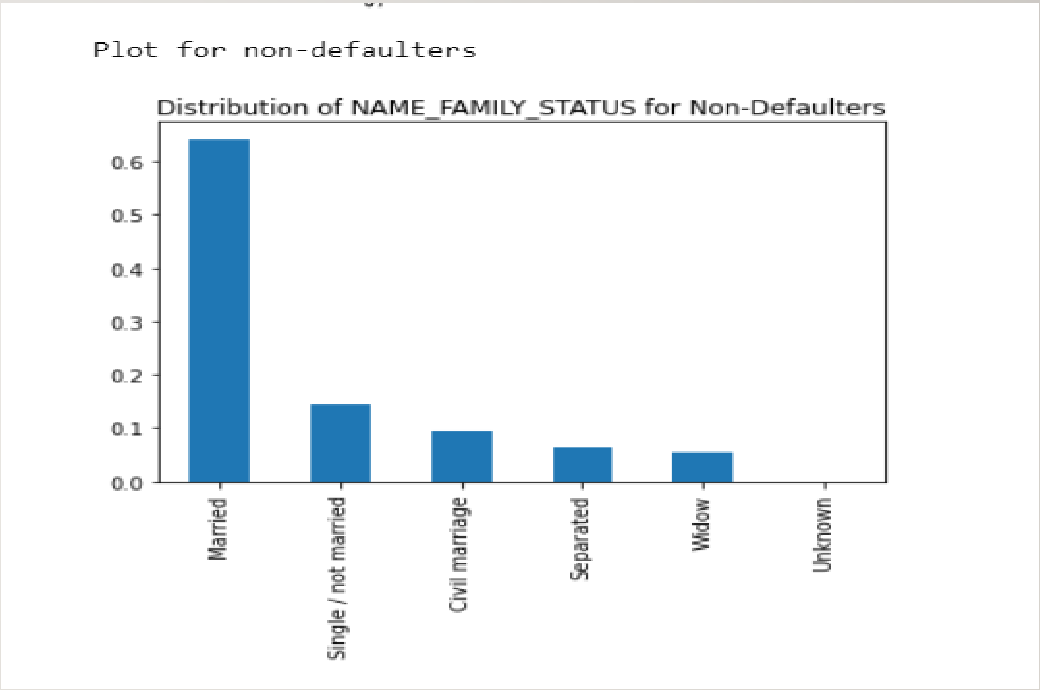




- Business Entity Type 3 contribute higher in defaulters, followed by self-employed.
- All other classes show almost similar distribution among defaulters and non-defaulters group.

Analyzing NAME_FAMILY_STATUS for Target variable

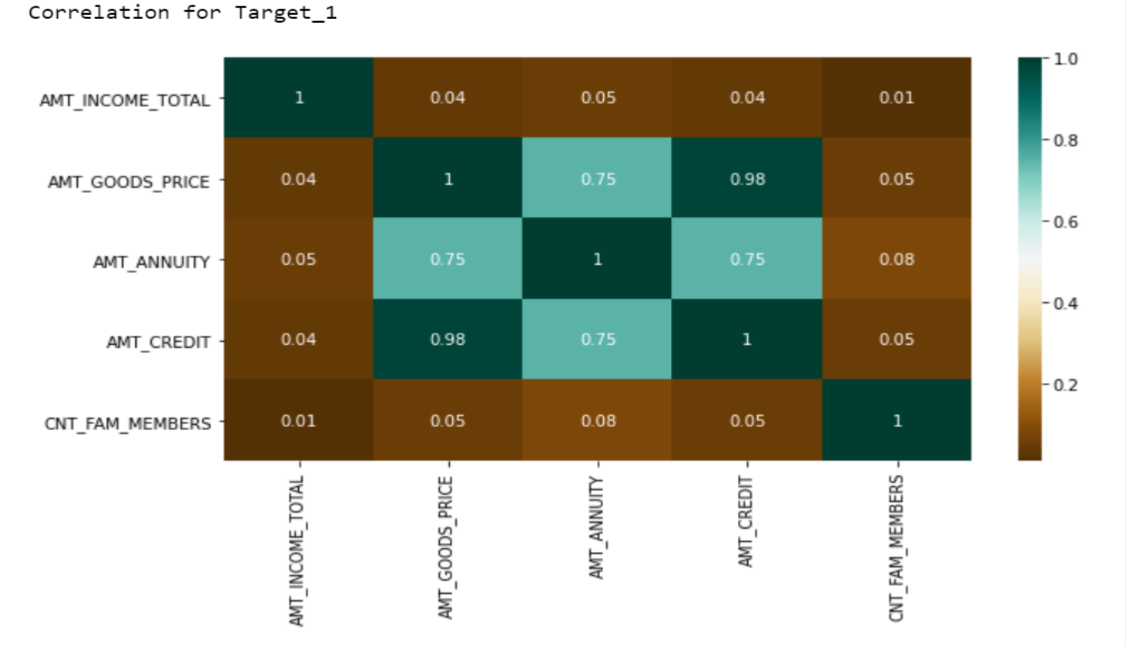




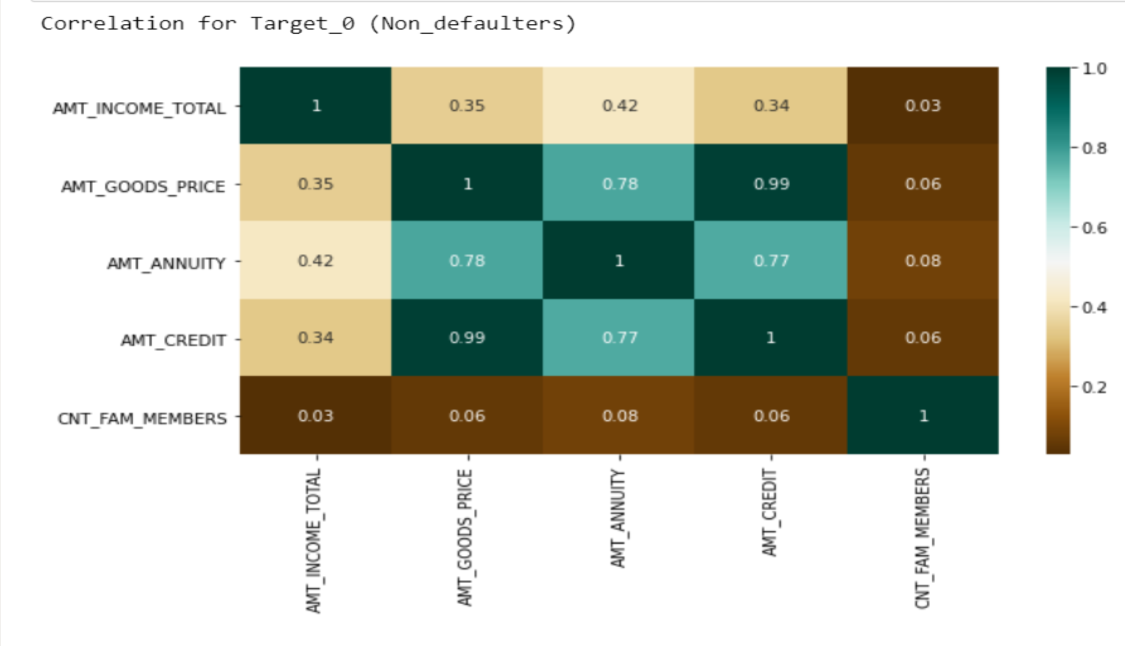
- Married people contribute high in both cases, It means that they take more loan compared to others. But rate of defaulting is less.
- Single/nonmarried class contribute 17% defaulters ,so more risk is associated with them.

Finding the correlation between continuous variables

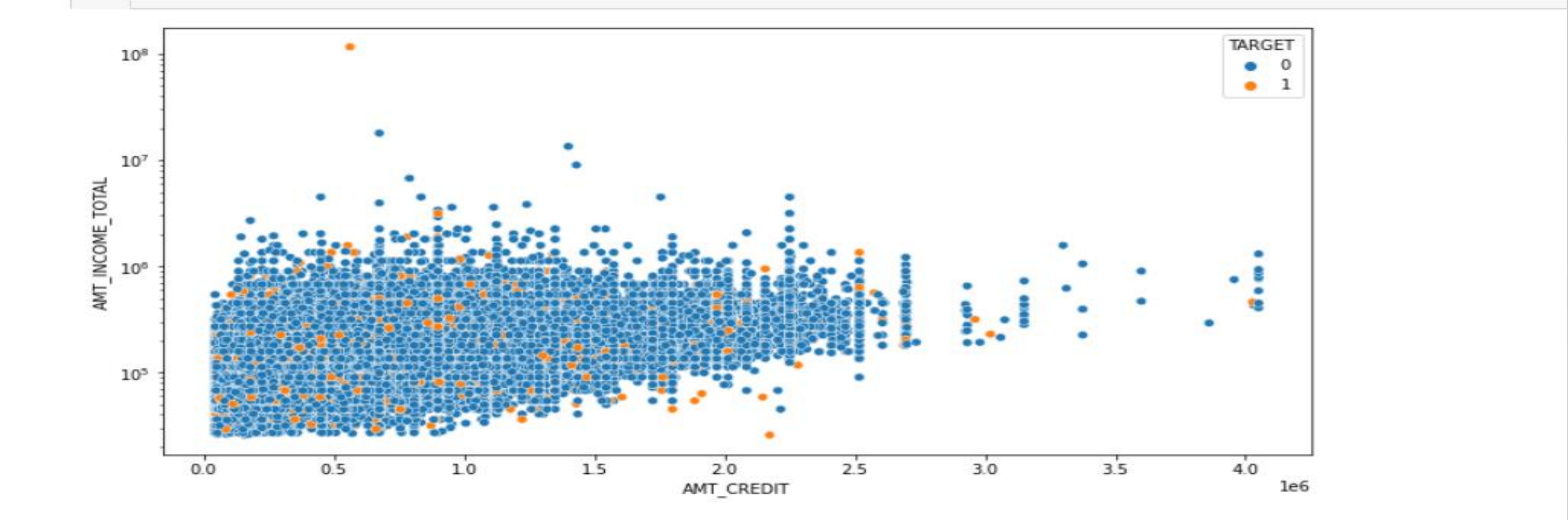
- Correlation for Target_1 (Defaulters)



- Correlation for Target_0 (Non_defaulters)

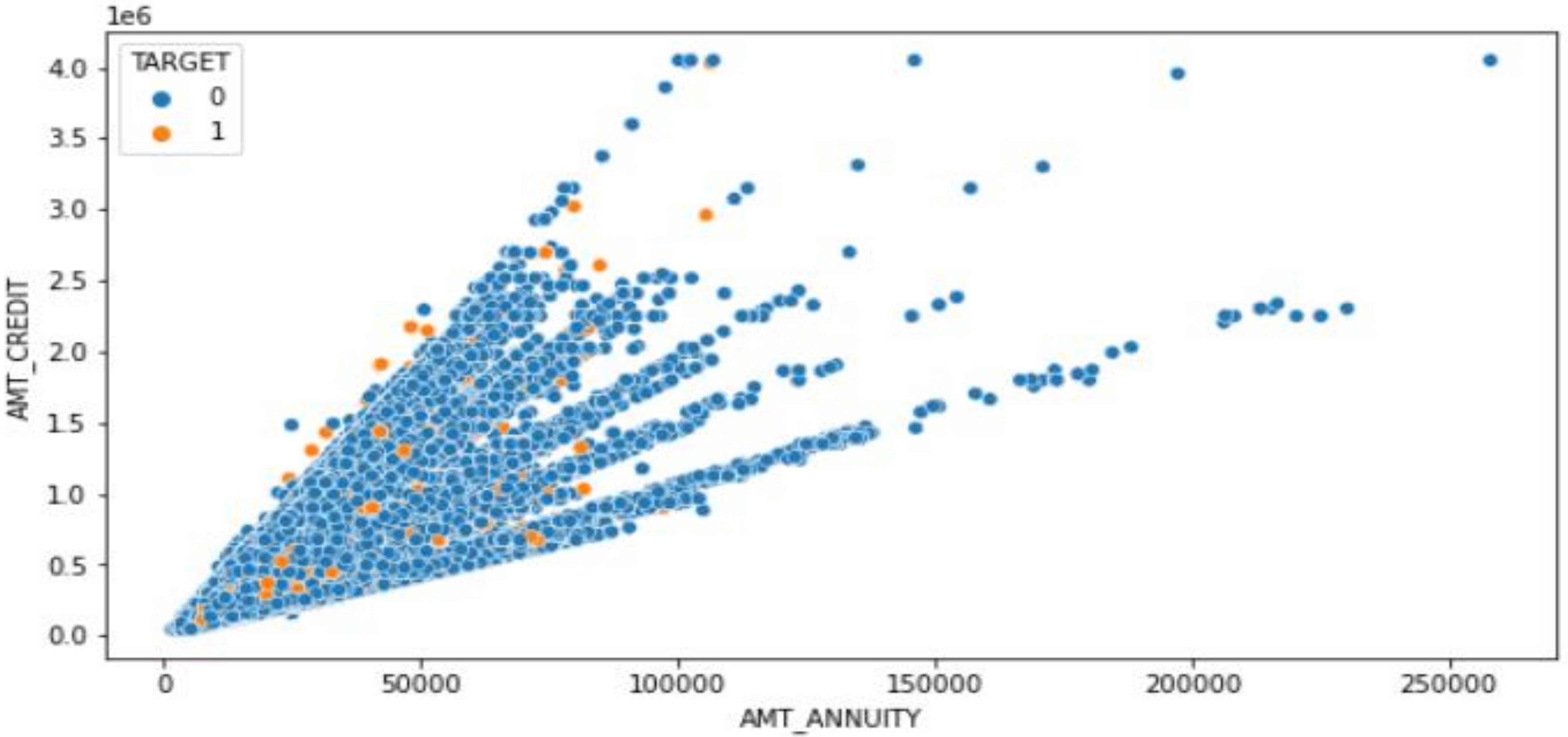


BIVARIATE VARIABLE ANALYSIS : INCOME TOTAL VS AMOUNT CREDIT

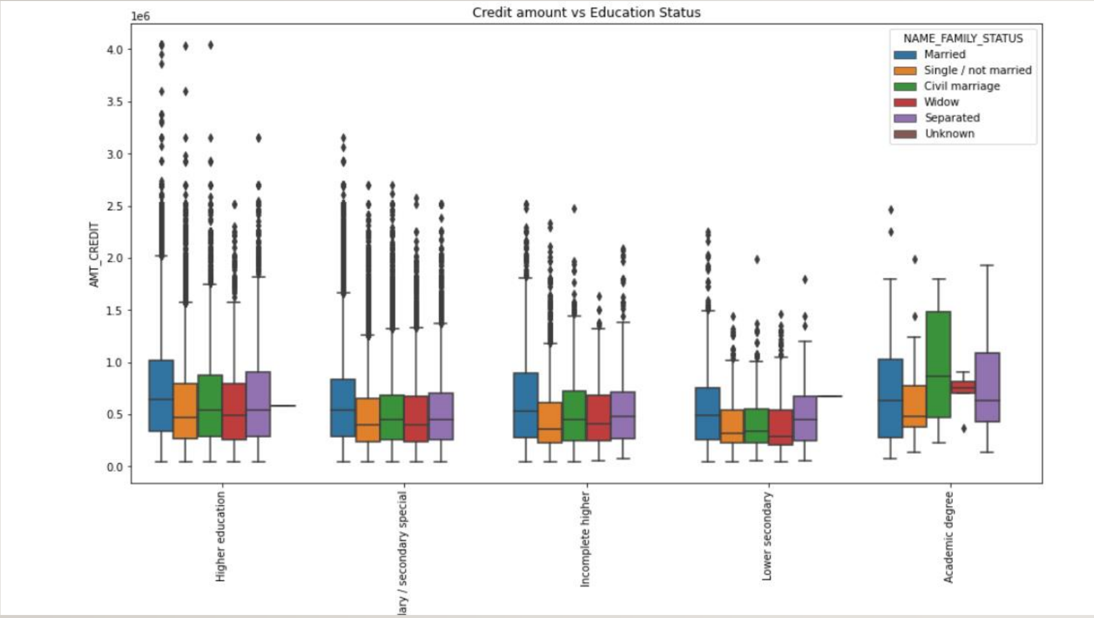


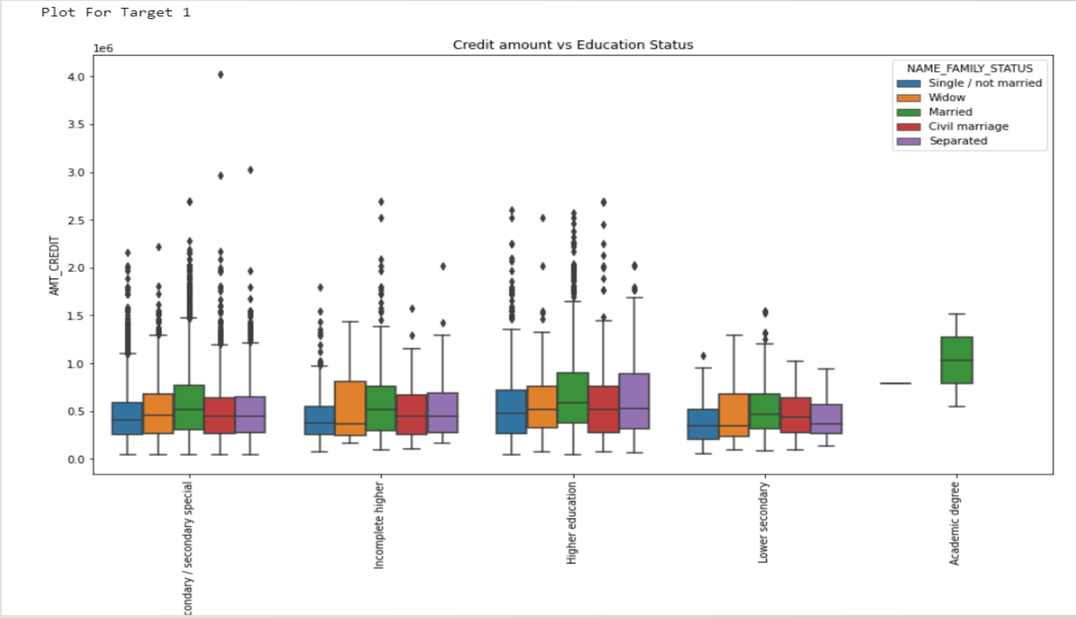
- We can observe more values in between 0 and 2500000.
- Very few outliers are in where loan is paid on time for higher total income above 100000
- Beyond credit amount 2500000 ,we can see less default.

BIVARIATE VARIABLE ANALYSIS : AMOUNT CREDIT VS AMOUNT ANNUITY



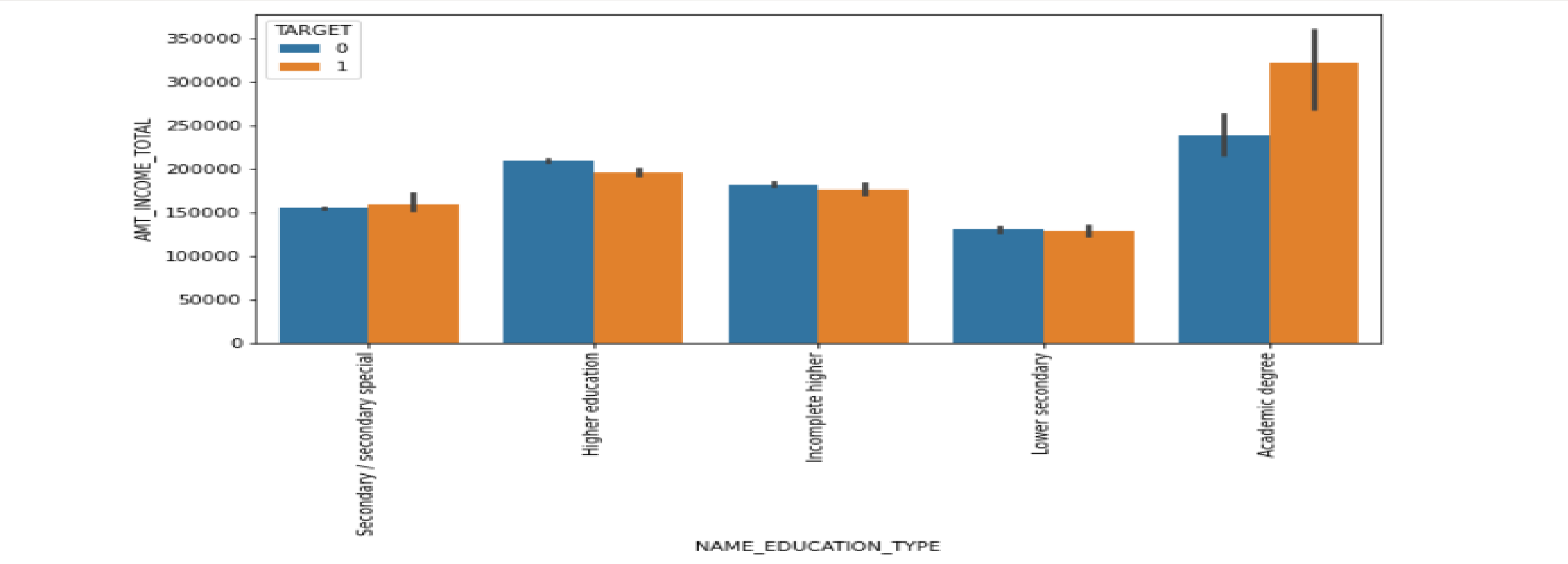
BIVARIATE VARIABLE ANALYSIS: CREDIT AMOUNT WITH EDUCATION STATUS





1. Family status of "Married","civil marriage",and "seperated" with higher education background have more outliers.
2. People holding academic degree and civil marriage status have most of values in third quartile.
3. Family status of 'civil marriage', 'marriage' and 'separated' of Academic degree education are having higher number of credits than others.

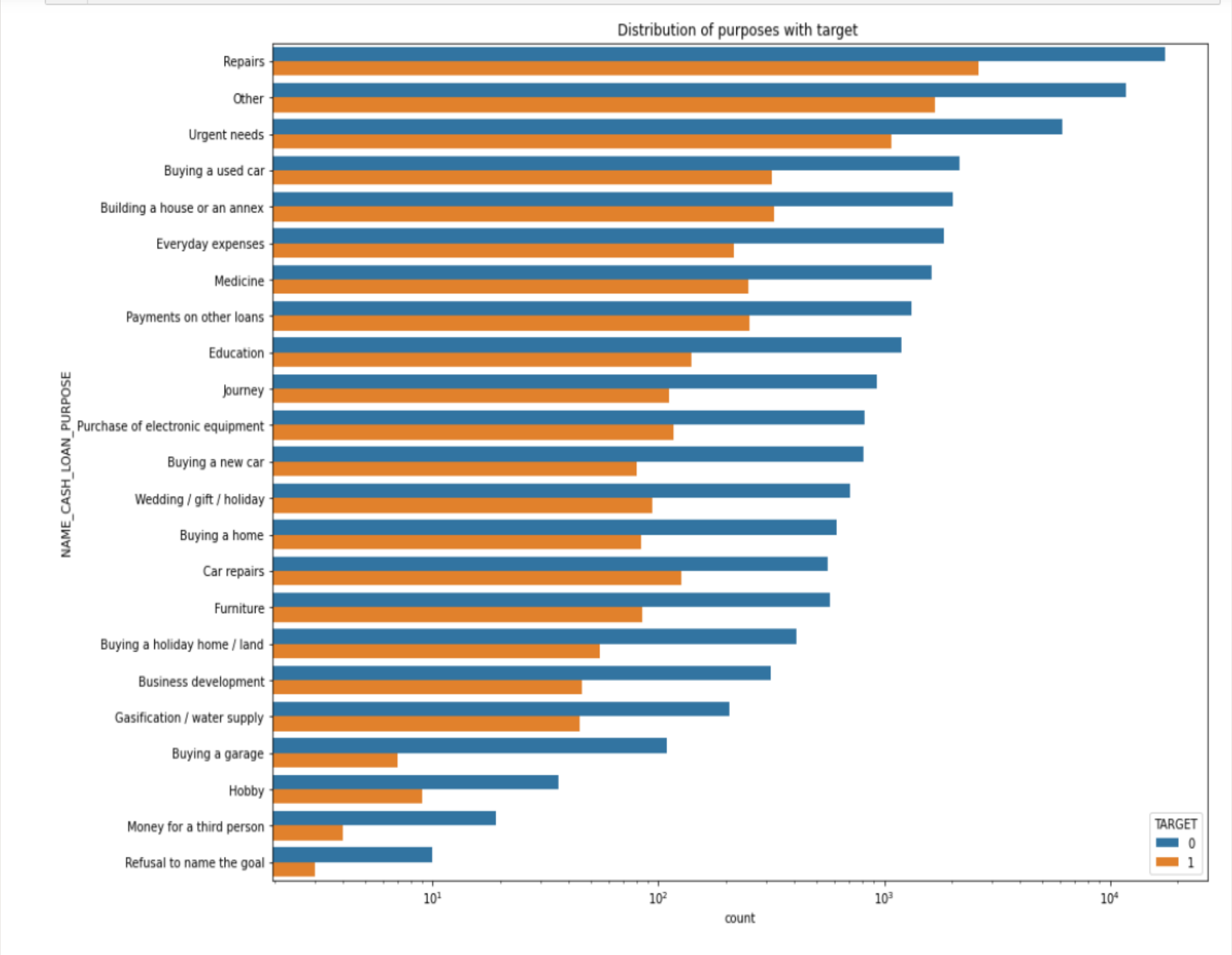
BIVARIATE VARIABLE ANALYSIS: INCOME TOTAL WITH EDUCATION STATUS



1. People with academic degree have more income compared to others and they show more tendency to make default.
2. People with 'Lower secondary' education have less income amount than others.
3. People with 'Higher education' will pay loan on time.

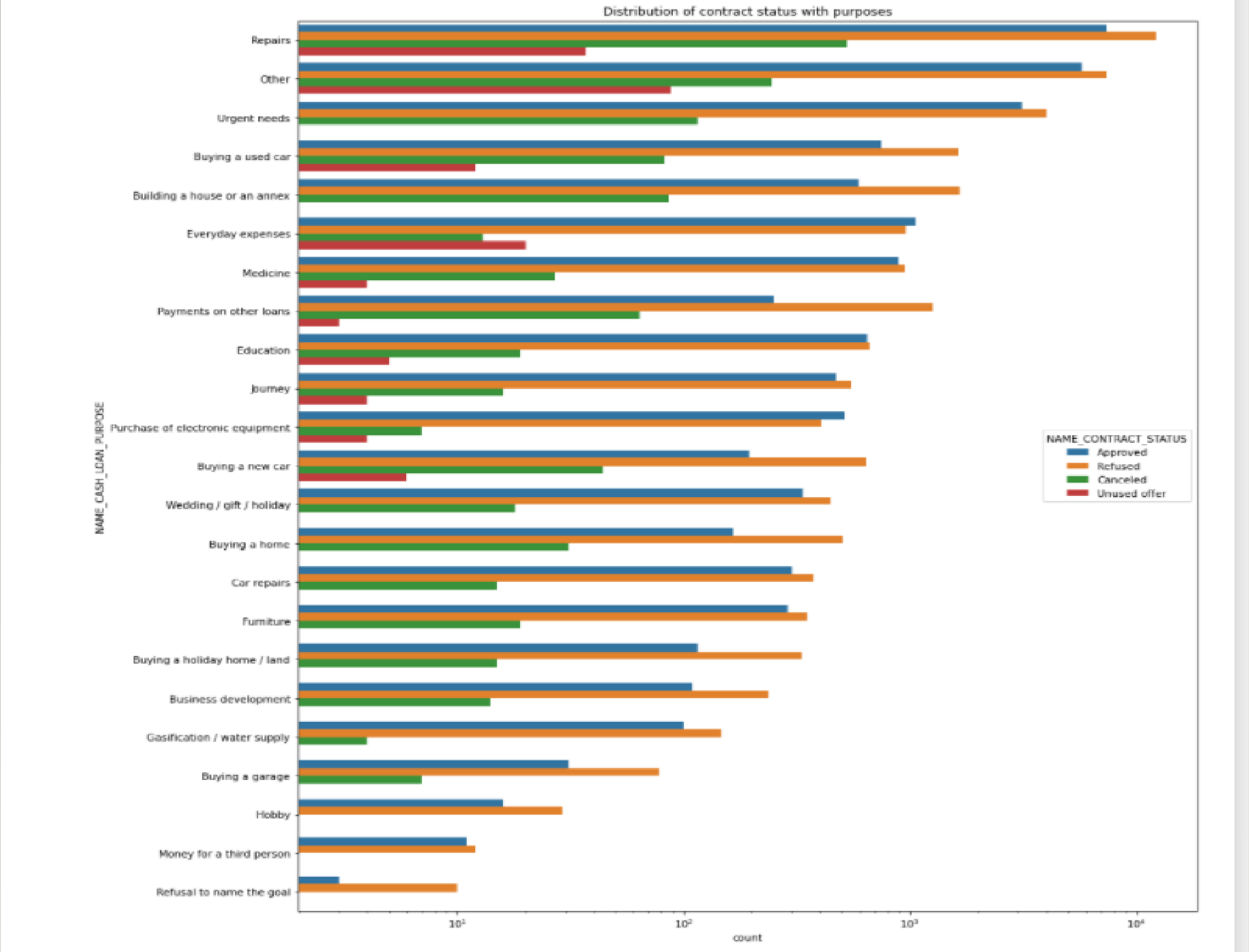
CATEGORICAL VARIABLE: PREVIOUS LOAN APPLICATION STATUS

Distribution of purposes with target



- Loan taken for purpose of 'repair' faces more difficulty in payment on time.
- In some cases such as 'Education', 'Buying a garage', 'Business development', 'Buying land', and 'Buying car' etc. we can see that loan payment is significantly higher than facing difficulties.

Distribution of contract status with purposes



- Loan taken for purpose of 'repair' faces more rejection.
- For 'Education purpose' we can see almost same number of approval and rejection.
- 'Buying new car' and 'paying other loan' have more rate of rejection than approval.

CONCLUSION

- 1. Bank should focus on `working 'with less income, as they have made more unsuccessful payments.
- 2. Bank can provide more loan to ‘Student’ ,’pensioner’ and ‘Businessman as they have made more successful payment.
- 3. People with Academic Degree are more likely to repay the loan only 0.0198% have not repaid the loan.
- 4. Loan taken for the purpose ‘Repair’ is having higher number of unsuccessful payments on time.
- 5. Single/nonmarried class contribute 17% defaulters ,so more risk is associated with them.
- 6. Family status of 'civil marriage', 'marriage' and 'separated' of Academic degree education are having higher number of credits than others.