Credit EDA Case Study

PRESENTED BY

RISHITA SINHA

ROHAN PUJARI

Contents

- Problem Statement
- Overview of the data - Reading & Inspecting the data
- Renaming and arranging values
- Missing Values
- Data Imputing
- Imbalance Calculation
- Identifying Outliers
- Correlation
- Univariate Analysis
- Bivariate Analysis
- Insights drawn

Problem Statement:

- Loan lending companies are facing issues in providing loans to the customers who are not defaulters.
- By analyzing the patterns present in the data, we need to ensure that the applicants capable of repaying the loan are not rejected.
- Insights should explain why the variable is important for differentiating the clients with payment difficulties with all other cases.

Overview of the data – Reading & Inspecting the data

This case study aims 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.

The number of rows and columns in the data frame is (307511,12)

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT
0	100002	1	Cash loans	M	N	Y	0	
1	100003	0	Cash loans	F	N	N	0	
2	100004	0	Revolving loans	M	Y	Y	0	
3	100006	0	Cash loans	F	N	Y	0	
4	100007	0	Cash loans	M	N	Y	0	

5 rows × 122 columns

Overview of the data – Reading & Inspecting the data

```
]: pre_application.dtypes

]: SK_ID_PREV                    int64
   SK_ID_CURR                    int64
   NAME_CONTRACT_TYPE           object
   AMT_ANNUITY                 float64
   AMT_APPLICATION             float64
   AMT_CREDIT                  float64
   AMT_DOWN_PAYMENT            float64
   AMT_GOODS_PRICE             float64
   WEEKDAY_APPR_PROCESS_START   object
   HOUR_APPR_PROCESS_START       int64
   FLAG_LAST_APPL_PER_CONTRACT  object
   NFLAG_LAST_APPL_IN_DAY        int64
   RATE_DOWN_PAYMENT           float64
   RATE_INTEREST_PRIMARY       float64
   RATE_INTEREST_PRIVILEGED    float64
   NAME_CASH_LOAN_PURPOSE       object
   NAME_CONTRACT_STATUS         object
   DAYS_DECISION                 int64
   NAME_PAYMENT_TYPE            object
   CODE_REJECT_REASON           object
   NAME_TYPE_SUITE              object
```

Inspected the dataframe – we have Int, Object, float and Boolean type of data. Out of that many of them have null and NA values, we have dropped them with *percentage of null values > 50% - 60%* len

Renaming and arranging values.

In the cells shown below we have Changed the 'DAYS_BIRTH', 'DAYS_EMPLOYED', 'DAYS_REGISTRATION', 'DAYS_ID_PUBLISH' and 'DAYS_LAST_PHONE_CHANGE' which had negative or mixed values and imputed them with absolute values for our analysis.

```
# Checking the values present in columns starting with 'DAYS'
print(appdata['DAYS_BIRTH'].unique())
print(appdata['DAYS_EMPLOYED'].unique())
print(appdata['DAYS_REGISTRATION'].unique())
print(appdata['DAYS_ID_PUBLISH'].unique())
print(appdata['DAYS_LAST_PHONE_CHANGE'].unique())
```

Missing Values

We have dropped variables with missing values ranging from 30% - 50% and the count of columns after dropping all the required columns is (307511, 51)

Missing Values

```
[22]: #Dropping columing with missing values range from 30-50%

[23]: # Dropping the columns which end with _AVG,_MODE,MEDI
      application.drop(application.filter(regex="_AVG").columns, axis=1, inplace=True)

[24]: application.drop(application.filter(regex="_MODE").columns, axis=1, inplace=True)

[25]: application.drop(application.filter(regex="_MEDI").columns, axis=1, inplace=True)

[26]: # Count of columns after dropping above data
      application.shape

t[26]: (307511, 75)

[27]: # Dropping all the verification documents column taken while applying for the loan
      application.drop(application.filter(regex="FLAG_DOCUMENT_").columns,axis=1, inplace=True)

[28]: application.drop(application.filter(regex="_SOCIAL_CIRCLE").columns,axis=1, inplace=True)

[29]: # Count of columns after dropping all the required columns
      application.shape

t[29]: (307511, 51)
```

Data Imputing

- *We have merged the EXT_SOURCE columns into one column(EXT_SOURCE_AVG) after taking mean of the 3 columns.*

- *We have also changed the Categorical columns to Numerical columns.*

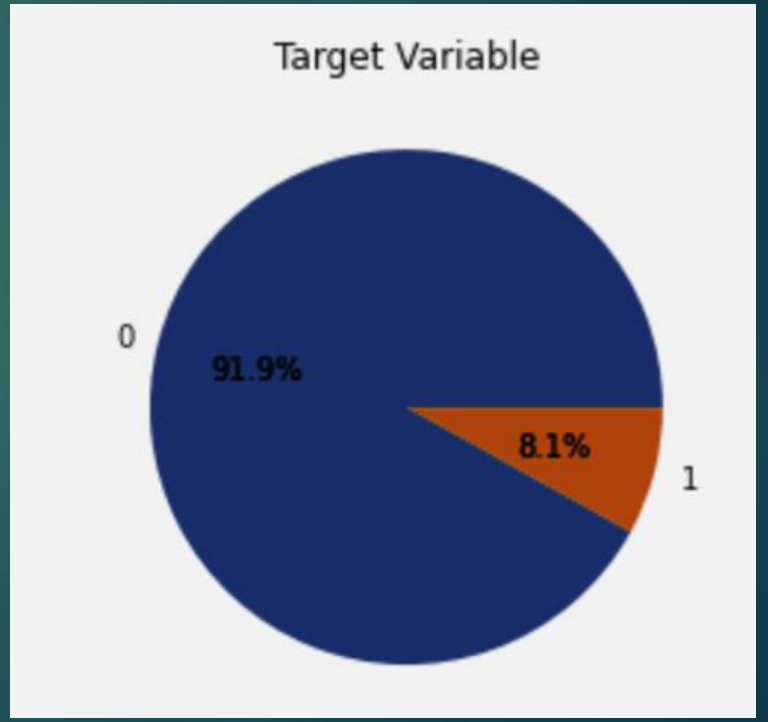
Class Imbalance

- representing 2 target variables from TARGET column with each of their percentages and displaying as Pie graph. 0 – 91.9% & 1 – 8.1%

Imbalance Calculation

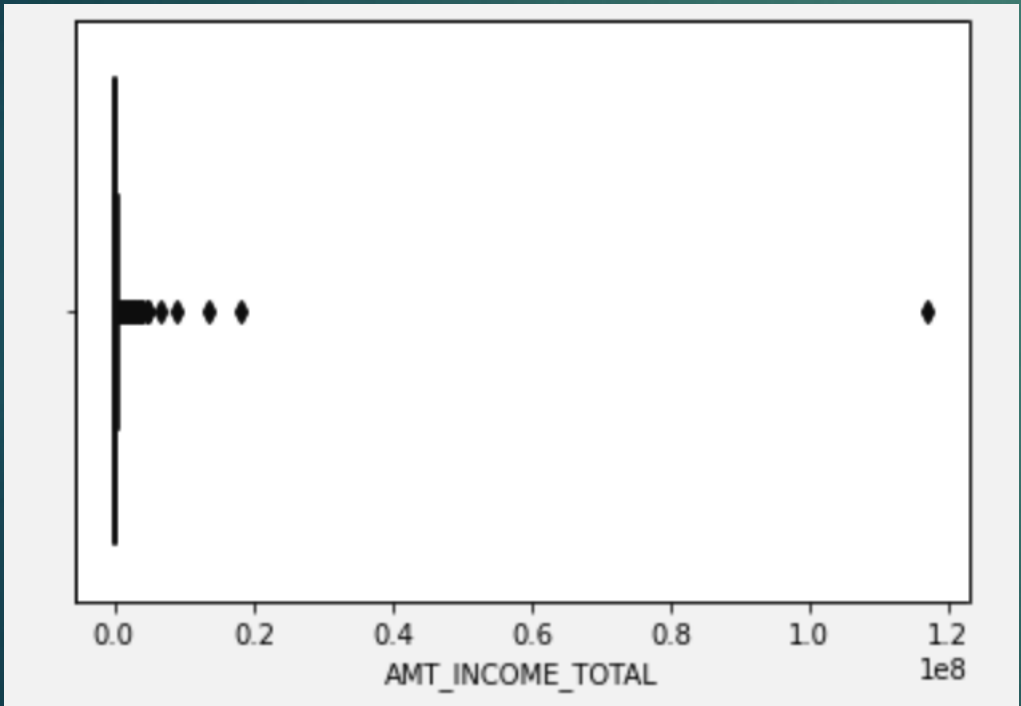
```
abc=round(application["TARGET"].value_counts(normalize=True)*100,2)
print(abc)
plt.pie(abc,labels=abc.index,autopct='%1.1f%%')
plt.title("Target Variable")
plt.show()
```

```
0    91.93
1     8.07
Name: TARGET, dtype: float64
```

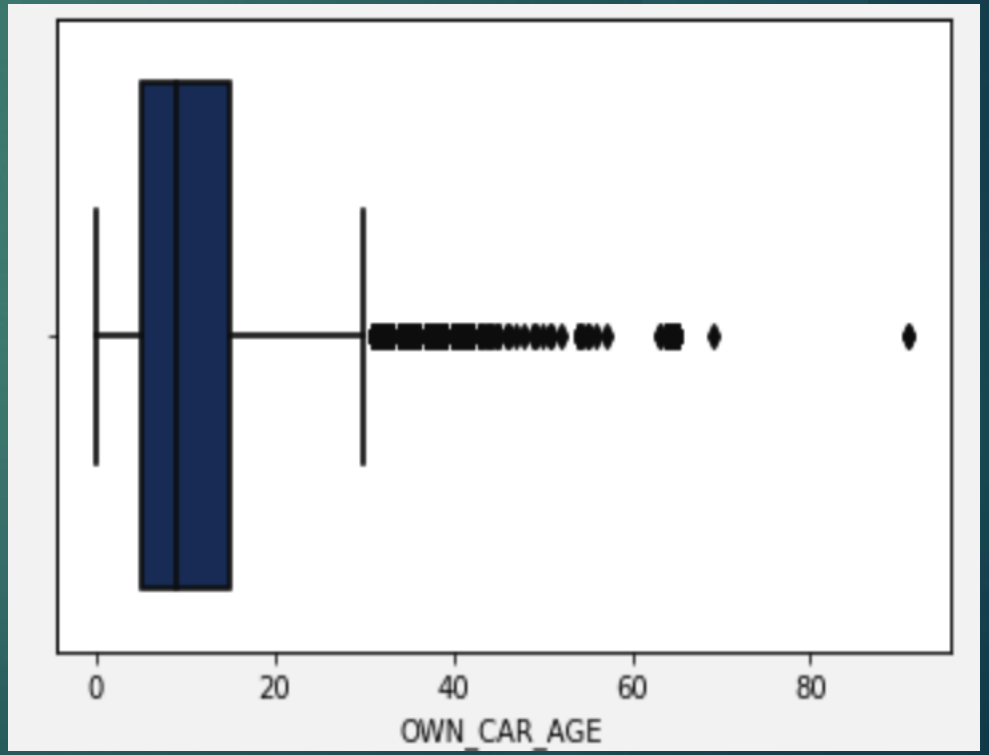


Outlier Detection

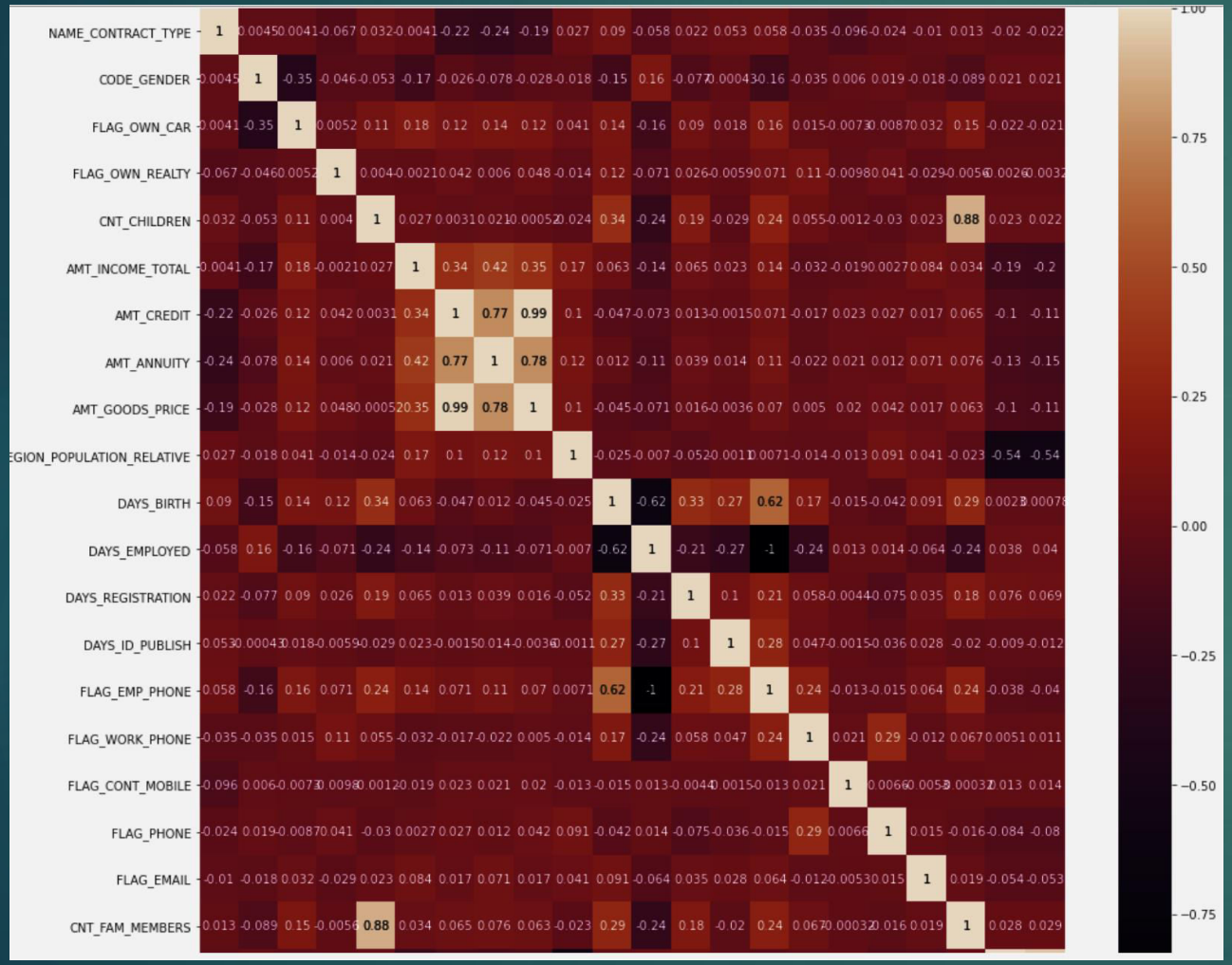
this variable indicates the Income of the client. as we can see from the plot there is one value which is too high compared to others. hence it is an outlier. '''



this variable indicates the Car Age of the client. we can see from the plot That the value is comparatively high and it is aslso an outlier



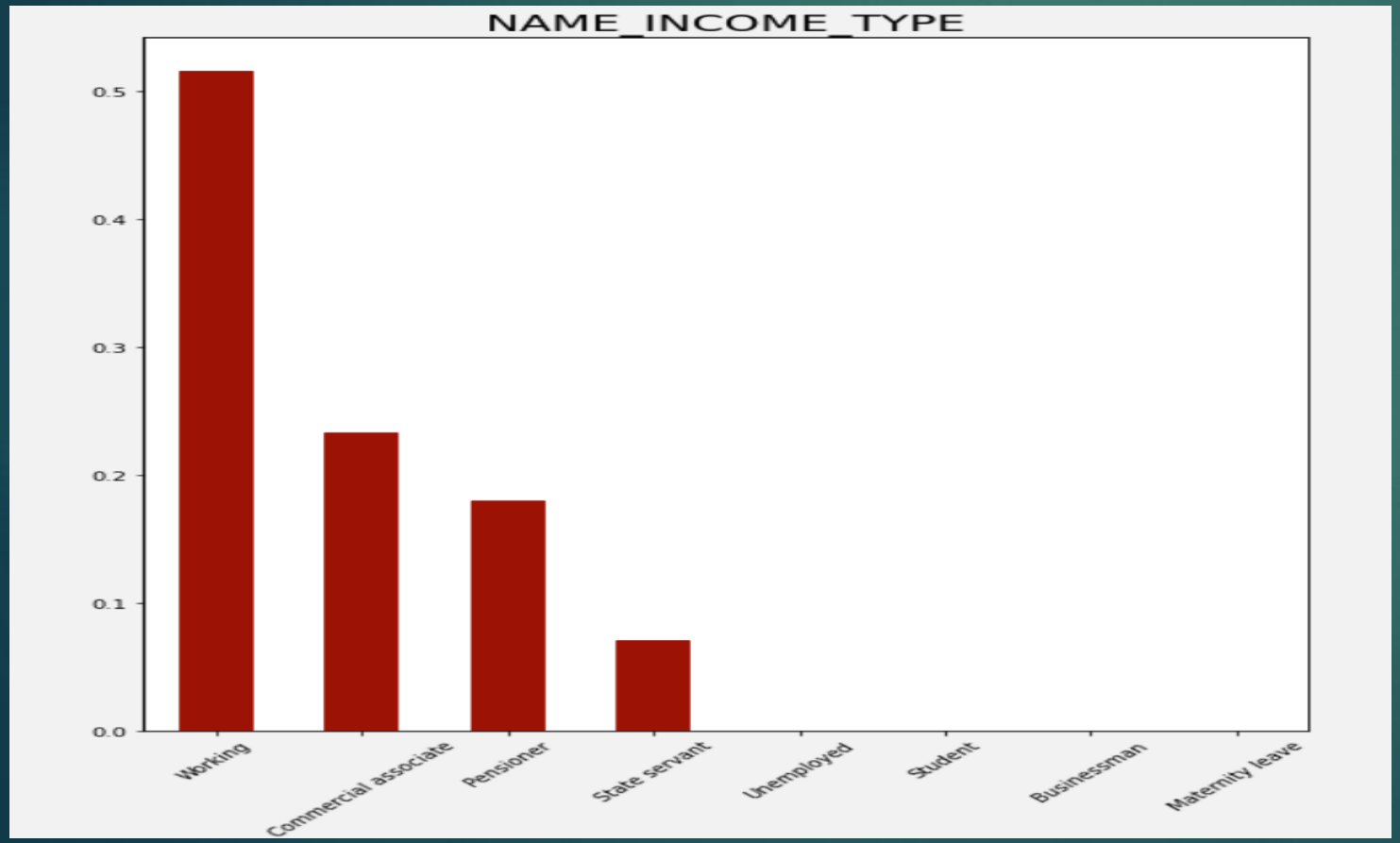
Correlation



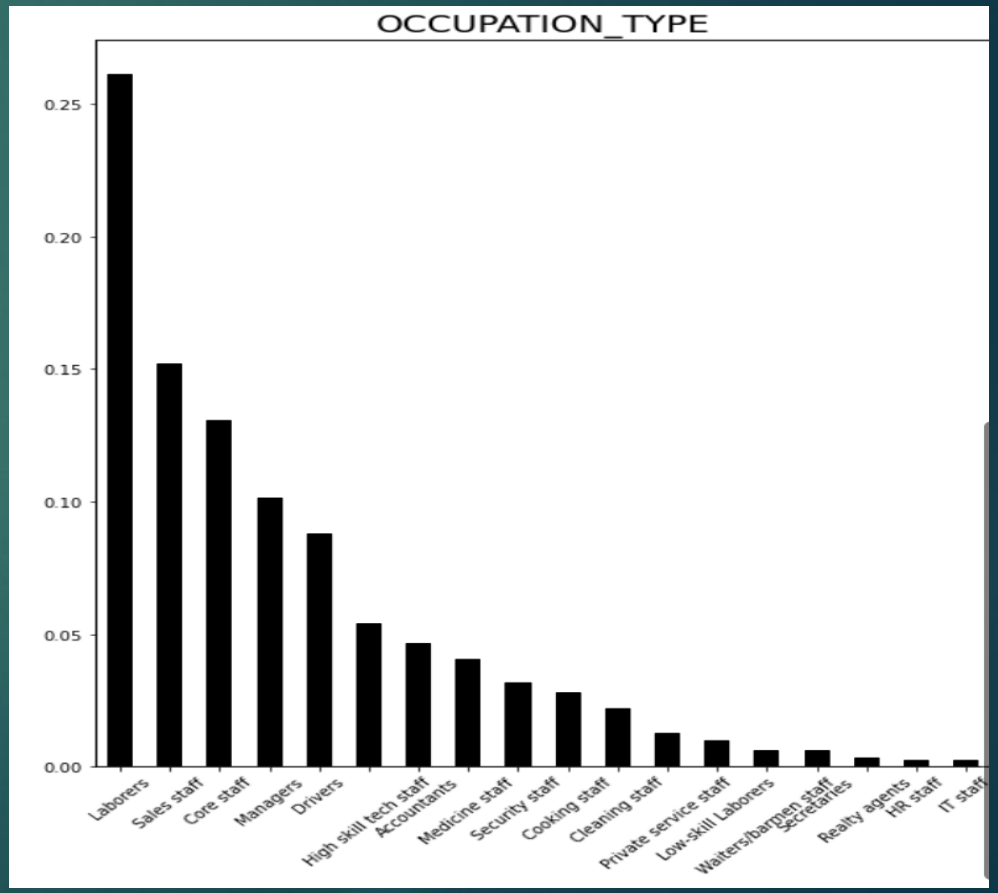
We observe that there is a high correlation between credit amount and goods price. There appears to be some deviancies in the correlation of Loan-Payment Difficulties and Loan- Non Payment Difficulties such as credit amount v/s income.

Univariate Analysis

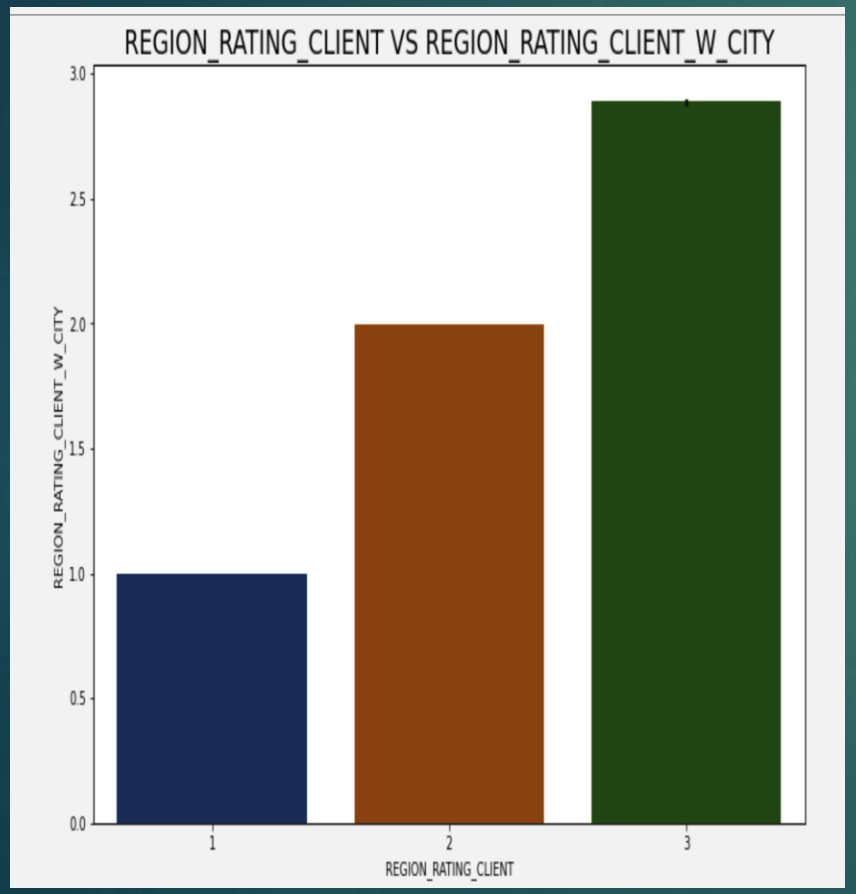
From the plot above we can say that clients with 'Maternity leave' Income type have maximum % of Loan-Payment Difficulties.

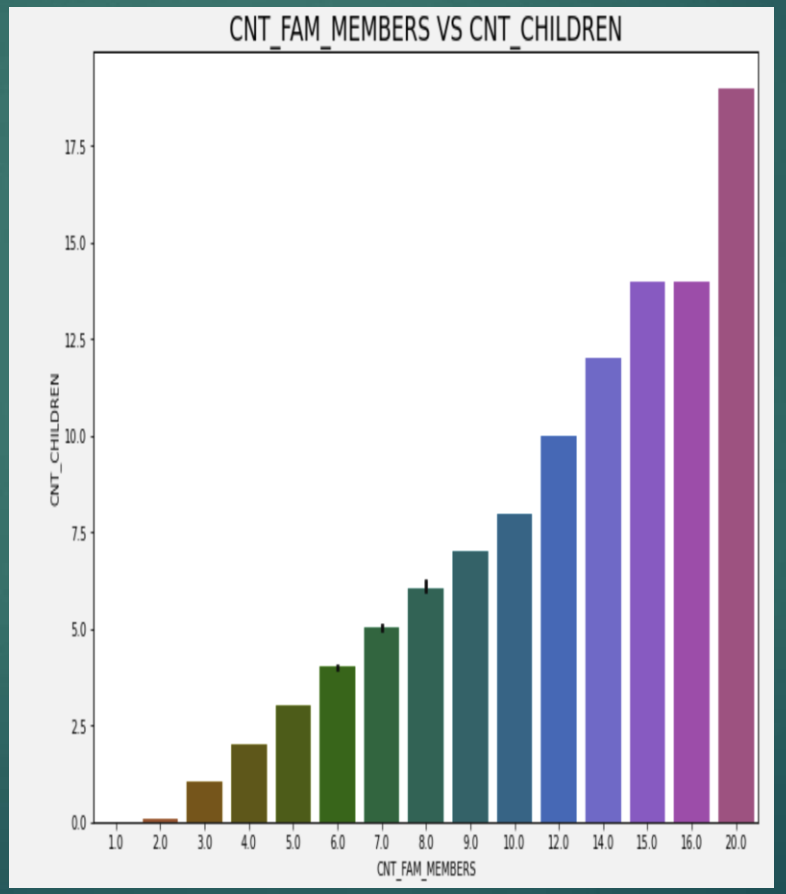


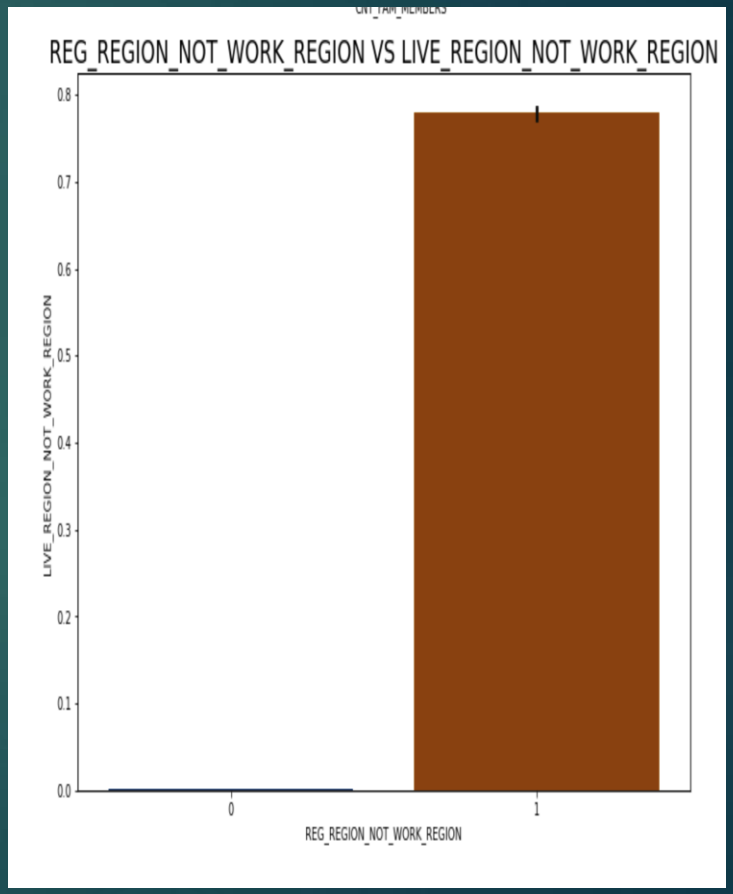
From the plot above we can s ay that clients with 'Lower skill Laborers' occupation type have maximum % of Loan-Payment Difficulties.



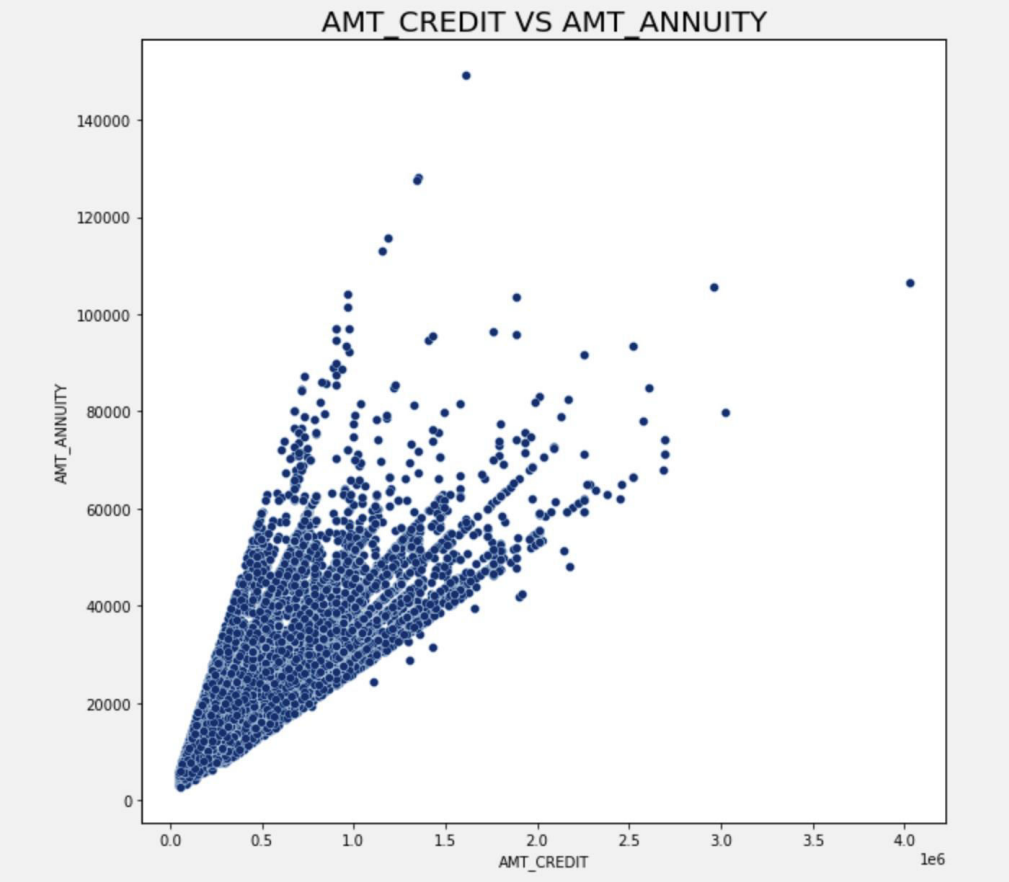
Bivariate Analysis – For Categorical Variables

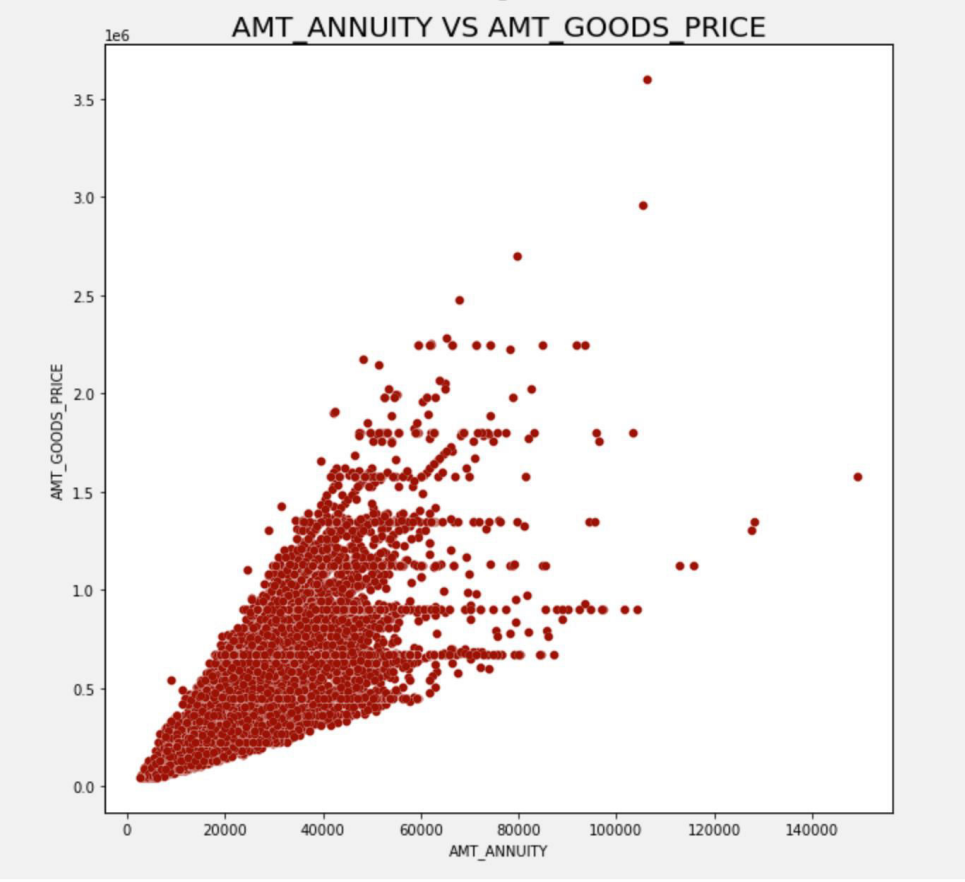






Bivariate Analysis – For Continuous Variables





Insights drawn

- The count of ‘Maternity Leave’ in ‘NAME_INCOME_TYPE’ is very less and it also has maximum % of payment difficulties- around 40%. Hence, client with income type as ‘Maternity leave’ are the driving factors for Loan Defaulters.
- The count of ‘Low skilled Laborers’ in ‘OCCUPATION_TYPE’ is comparatively very less and it also has maximum % of payment difficulties- around 17%. Hence, client with occupation type as ‘Low skilled Laborers’ are the driving factors for Loan Defaulters.
- The count of ‘Lower Secondary’ in ‘NAME_EDUCATION_TYPE’ is comparatively very less and it also has maximum % of payment difficulties- around 11%. Hence, client with education type as ‘Lower Secondary’ are the driving factors for Loan Defaulters.

Thank You