Credit EDA Case Study

By

AKSHAY MAGOTRA

&

SUDHANSHU PANDEY

Introduction

This case study is to predict the defaulters of a financial institution before approving them loans using past loan data through Exploratory Data Analysis using basic statistics.

The data given contains the information about the loan application at the time of applying for the loan. It contains two types of scenarios:

- The client with payment difficulties: he/she had late payment more than X days on at least one of the first Y instalments of the loan in our sample,
- All other cases: All other cases when the payment is paid on time.

When a client applies for a loan, there are four types of decisions that could be taken by the client/company):

1. Approved: The Company has approved loan Application.
2. Cancelled: The client cancelled the application sometime during approval. Either the client changed her/his mind about the loan or in some cases due to a higher risk of the client he received worse pricing which he did not want.
3. Refused: The company had rejected the loan (because the client does not meet their requirements etc.).
4. Unused offer: Loan has been cancelled by the client but on different stages of the process.

This dataset has 3 files as explained below:

1. 'application_data.csv’ contains all the information of the client at the time of application. The data is about whether a client has payment difficulties.

2. 'previous_application.csv' contains information about the client’s previous loan data. It contains the data whether the previous application had been Approved, Cancelled, Refused or Unused offer.

3. 'columns_description.csv' is data dictionary which describes the meaning of the variables.

Steps Involved in Analysis:

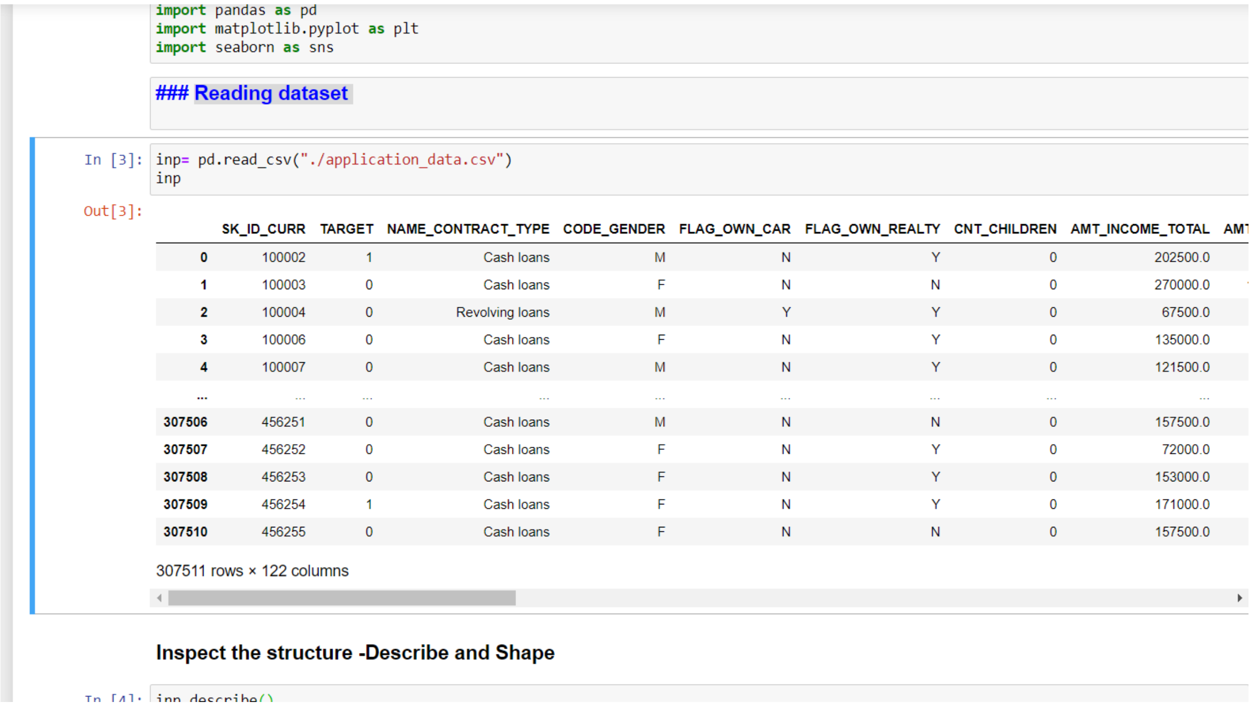
Prerequisite

Place 'application_data.csv' and 'previous_application_data.csv' input file at "../input" directory before running this code.

Please make sure that you have following python libraries imported/installed at your system:

1. numpy version : 1.12.1 or higher
2. pandas version : 0.20.3 or higher
3. seaborn version : 0.8.0 or higher
4. Import Libraries and set required parameters

Reading dataset



```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Reading dataset

```
In [3]: inp= pd.read_csv("./application_data.csv")
        inp
```

```
Out[3]:
         SK_ID_CURR  TARGET  NAME_CONTRACT_TYPE  CODE_GENDER  FLAG_OWN_CAR  FLAG_OWN_REALTY  CNT_CHILDREN  AMT_INCOME_TOTAL  AMT
      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
    ...         ...     ...                 ...          ...           ...              ...           ...               ...
 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

307511 rows × 122 columns
```

Inspect the structure -Describe and Shape

```
In [4]: inp.describe()
```

So We have some analysis, before we go for the cleaning of the datasets . We have 307511 applications for Loan recieved, and the average income of the loan applicants is near about Rs.1.68L, Also we can see that 8.07% of the applicants had a past history of payment difficulty.

Data Cleaning and Manipulation

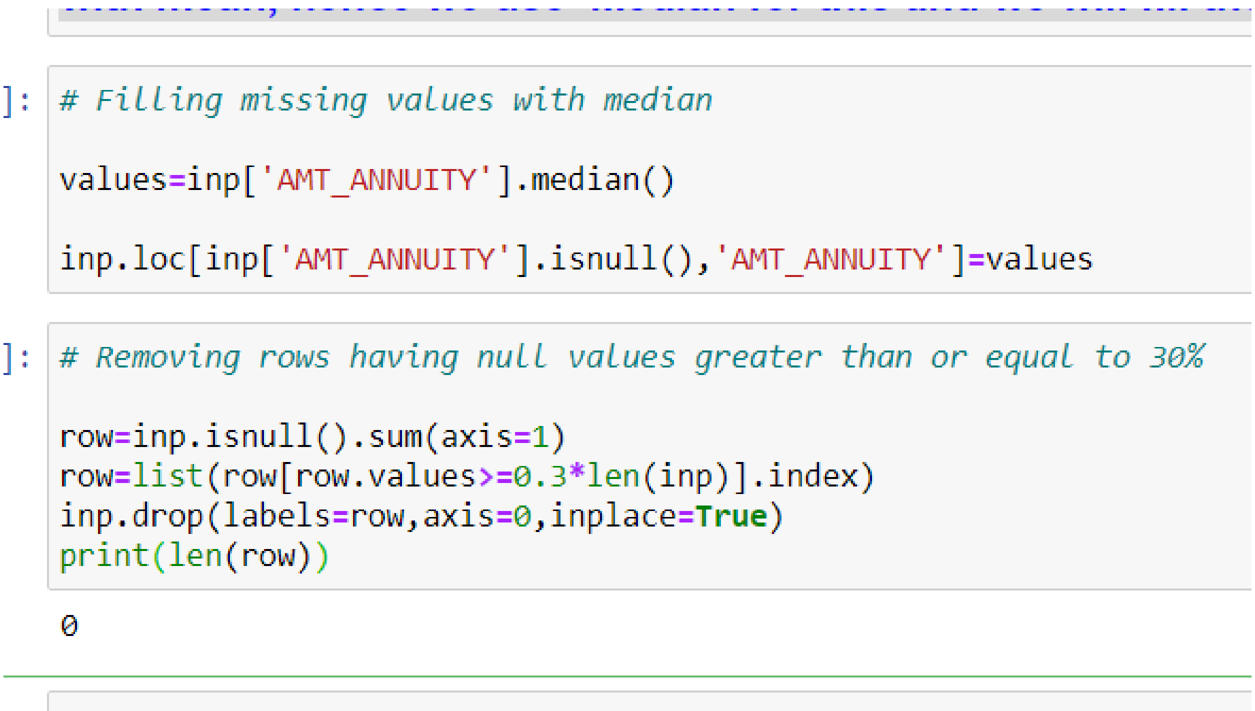
We have observed there were many missing values in this data set, so using the indexing and count of missing values in each column.And dropped all columns from data frame for which missing values is more than 50%. We have also found few more columns which are having around 47% missing values. Since these are almost around 50%, we have removed these columns as well.

- **We note that :**

  Annuity amount can decide whether the person can complete paying the annuity to replay the loan based on his/her total income and its relevant expenses like on family for which the number of family members is essential to know.

  As we can see that, 'AMT_ANNUITY' columns is having very few null values rows. Hence let's try to impute the missing values.

  Since this column is having an outlier which is very large it will be inappropriate to fill those missing values with mean, hence we use Median for this and we will fill those missing values with median value.



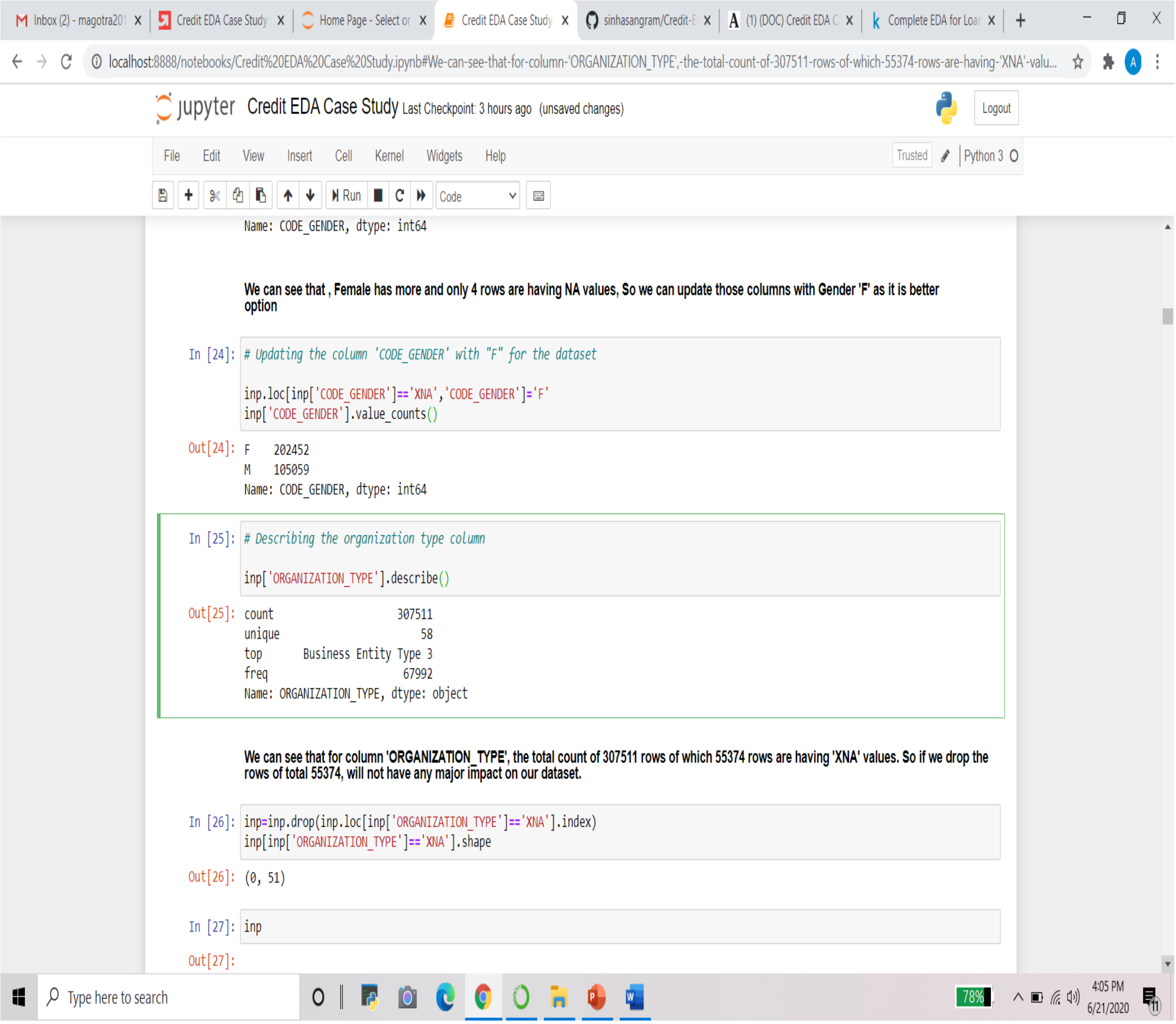
```
]: # Filling missing values with median

   values=inp['AMT_ANNUITY'].median()

   inp.loc[inp['AMT_ANNUITY'].isnull(),'AMT_ANNUITY']=values
```

```
]: # Removing rows having null values greater than or equal to 30%

   row=inp.isnull().sum(axis=1)
   row=list(row[row.values>=0.3*len(inp)].index)
   inp.drop(labels=row,axis=0,inplace=True)
   print(len(row))

   0
```

We have to remove the unwanted columns from the dataset ,in order to make it more appropriate.

Also, there are few columns in the data where the value is 'XNA' which means 'Not Available', So it is better to find the number of rows and columns and apply a best method on them to fill those missing values or to delete them.

1. We can see that , Female has more and only 4 rows are having NA values, So we can update those columns with Gender 'F' as it is better option.
2. We can see that for column 'ORGANIZATION_TYPE', the total count of 307511 rows of which 55374 rows are having 'XNA' values. So if we drop the rows of total 55374, will not have any major impact on our dataset.


Inbox (2) - magotra20
Credit EDA Case Study
Home Page - Select or
Credit EDA Case Study
sinhasangram/Credit-E
(1) (DOC) Credit EDA C
Complete EDA for Loar
localhost:8888/notebooks/Credit%20EDA%20Case%20Study.ipynb#We-can-see-that-for-column-'ORGANIZATION_TYPE',-the-total-count-of-307511-rows-of-which-55374-rows-are-having-'XNA'-valu...
jupyter Credit EDA Case Study Last Checkpoint: 3 hours ago (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code
Type here to search
78%
4:05 PM
6/21/2020
11

```
Name: CODE_GENDER, dtype: int64
```

We can see that , Female has more and only 4 rows are having NA values, So we can update those columns with Gender 'F' as it is better option

In [24]:
```
# Updating the column 'CODE_GENDER' with "F" for the dataset

inp.loc[inp['CODE_GENDER']=='XNA','CODE_GENDER']='F'
inp['CODE_GENDER'].value_counts()
```

Out[24]:
```
F    202452
M    105059
Name: CODE_GENDER, dtype: int64
```

In [25]:
```
# Describing the organization type column

inp['ORGANIZATION_TYPE'].describe()
```

Out[25]:
```
count                    307511
unique                       58
top       Business Entity Type 3
freq                      67992
Name: ORGANIZATION_TYPE, dtype: object
```

We can see that for column 'ORGANIZATION_TYPE', the total count of 307511 rows of which 55374 rows are having 'XNA' values. So if we drop the rows of total 55374, will not have any major impact on our dataset.

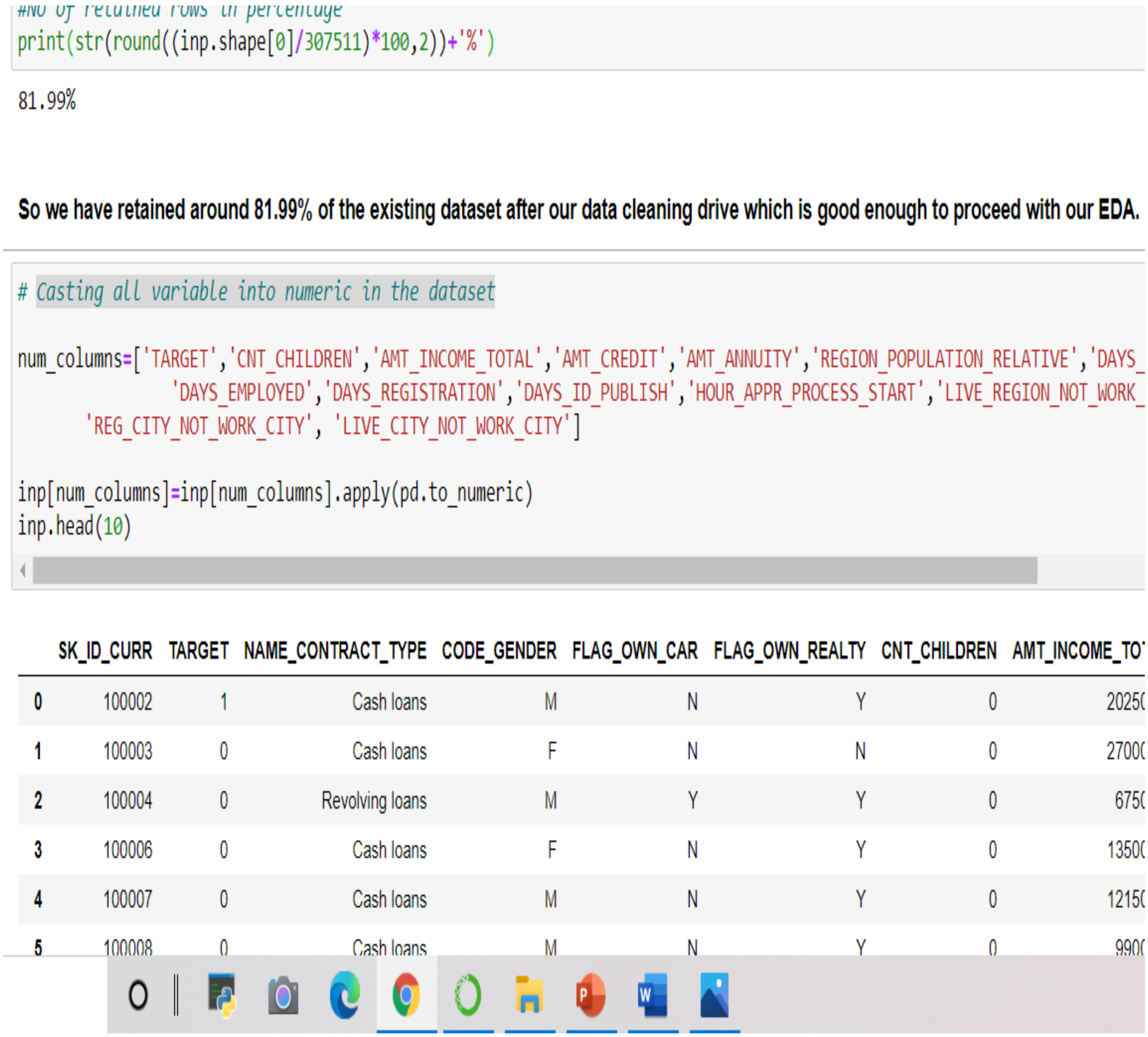
In [26]:
```
inp=inp.drop(inp.loc[inp['ORGANIZATION_TYPE']=='XNA'].index)
inp[inp['ORGANIZATION_TYPE']=='XNA'].shape
```

Out[26]:
```
(0, 51)
```

In [27]:
```
inp
```

Out[27]:

Casting all variable into numeric in the dataset.



```
#No of retained rows in percentage
print(str(round((inp.shape[0]/307511)*100,2))+'%')
```

```
81.99%
```

So we have retained around 81.99% of the existing dataset after our data cleaning drive which is good enough to proceed with our EDA.

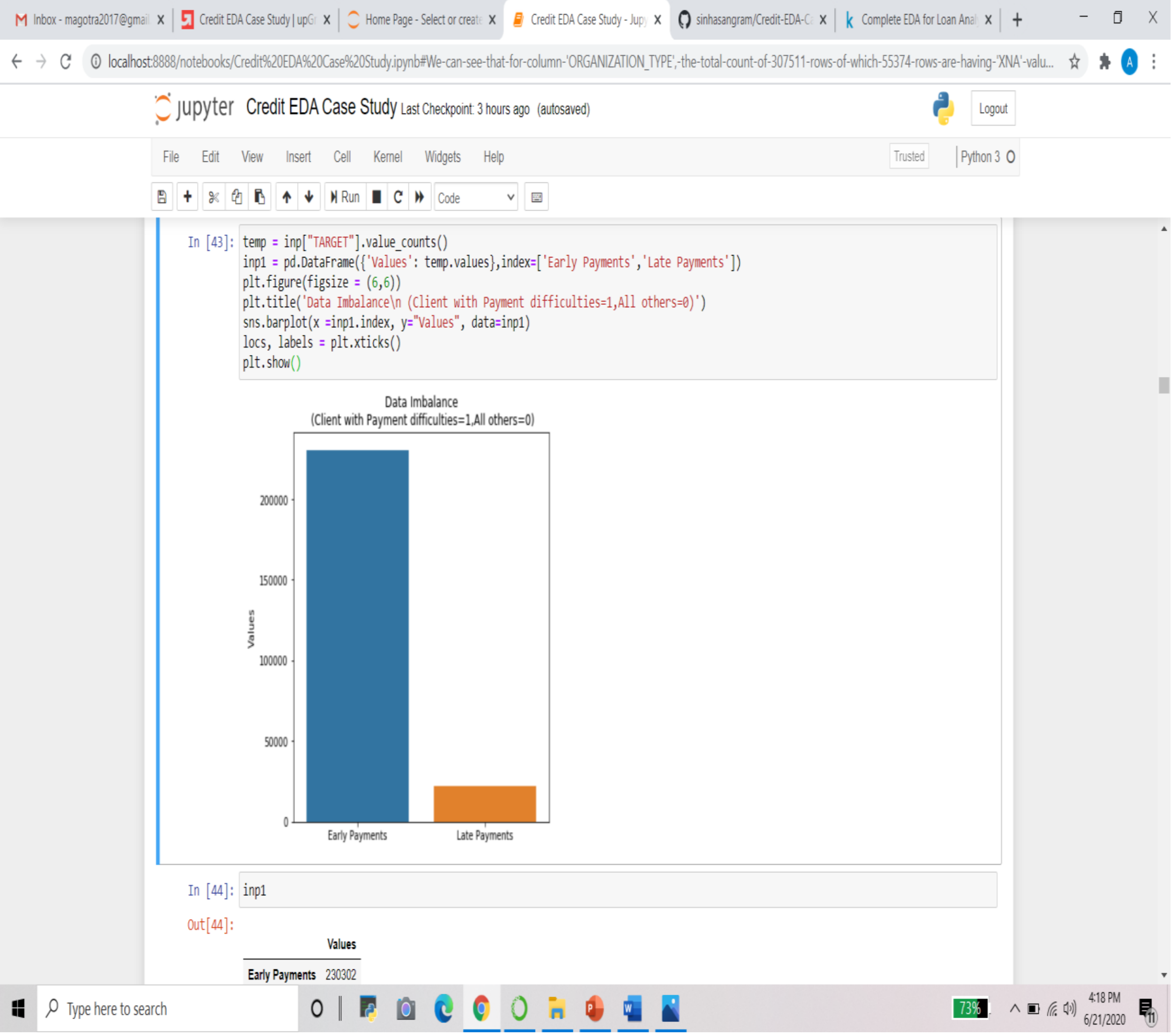
```
# Casting all variable into numeric in the dataset

num_columns=['TARGET','CNT_CHILDREN','AMT_INCOME_TOTAL','AMT_CREDIT','AMT_ANNUITY','REGION_POPULATION_RELATIVE','DAYS_
             'DAYS_EMPLOYED','DAYS_REGISTRATION','DAYS_ID_PUBLISH','HOUR_APPR_PROCESS_START','LIVE_REGION_NOT_WORK_
      'REG_CITY_NOT_WORK_CITY', 'LIVE_CITY_NOT_WORK_CITY']

inp[num_columns]=inp[num_columns].apply(pd.to_numeric)
inp.head(10)
```

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TO
0	100002	1	Cash loans	M	N	Y	0	2025
1	100003	0	Cash loans	F	N	N	0	2700
2	100004	0	Revolving loans	M	Y	Y	0	675
3	100006	0	Cash loans	F	N	Y	0	1350
4	100007	0	Cash loans	M	N	Y	0	1215
5	100008	0	Cash loans	M	N	Y	0	990

- ❖ Then we have to Change negative datatype to correct datatype.

DATA IMBALANCE

We have to Identify if there is data imbalance in the data. Find the ratio of data imbalance.

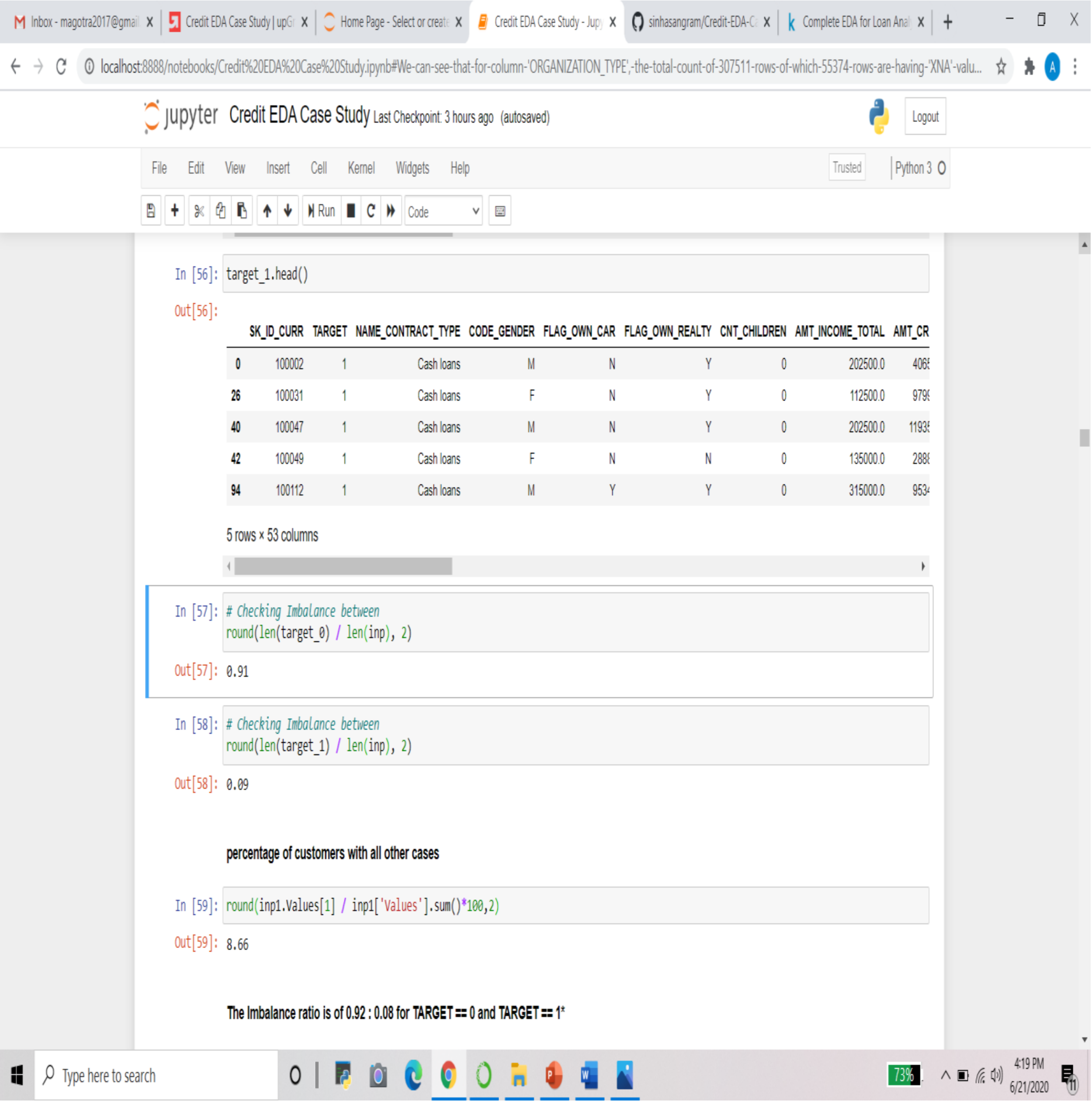

Inbox - magotra2017@gmai
Credit EDA Case Study | upGr
Home Page - Select or create
Credit EDA Case Study - Jupy
sinhasangram/Credit-EDA-C
Complete EDA for Loan Anal
localhost:8888/notebooks/Credit%20EDA%20Case%20Study.ipynb#We-can-see-that-for-column-'ORGANIZATION_TYPE',-the-total-count-of-307511-rows-of-which-55374-rows-are-having-'XNA'-valu...
jupyter Credit EDA Case Study Last Checkpoint: 3 hours ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code

```
In [43]: temp = inp["TARGET"].value_counts()
         inp1 = pd.DataFrame({'Values': temp.values},index=['Early Payments','Late Payments'])
         plt.figure(figsize = (6,6))
         plt.title('Data Imbalance\n (Client with Payment difficulties=1,All others=0)')
         sns.barplot(x =inp1.index, y="Values", data=inp1)
         locs, labels = plt.xticks()
         plt.show()
```

Data Imbalance
(Client with Payment difficulties=1,All others=0)
Values
200000
150000
100000
50000
0
Early Payments
Late Payments

```
In [44]: inp1

Out[44]:
                  Values
Early Payments    230302
```

Type here to search
73%
4:18 PM
6/21/2020


Inbox - magotra2017@gmai
Credit EDA Case Study | upGr
Home Page - Select or creat
Credit EDA Case Study - Jupy
sinhasangram/Credit-EDA-C
Complete EDA for Loan Anal
localhost:8888/notebooks/Credit%20EDA%20Case%20Study.ipynb#We-can-see-that-for-column-'ORGANIZATION_TYPE',-the-total-count-of-307511-rows-of-which-55374-rows-are-having-'XNA'-valu...
jupyter
Credit EDA Case Study Last Checkpoint: 3 hours ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code
Type here to search
73%
4:19 PM
6/21/2020

```
In [56]: target_1.head()
```

Out[56]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CR
0	100002	1	Cash loans	M	N	Y	0	202500.0	4065
26	100031	1	Cash loans	F	N	Y	0	112500.0	9799
40	100047	1	Cash loans	M	N	Y	0	202500.0	11935
42	100049	1	Cash loans	F	N	N	0	135000.0	2888
94	100112	1	Cash loans	M	Y	Y	0	315000.0	9534

5 rows × 53 columns

```
In [57]: # Checking Imbalance between
         round(len(target_0) / len(inp), 2)

Out[57]: 0.91
```

```
In [58]: # Checking Imbalance between
         round(len(target_1) / len(inp), 2)

Out[58]: 0.09
```

percentage of customers with all other cases

```
In [59]: round(inp1.Values[1] / inp1['Values'].sum()*100,2)

Out[59]: 8.66
```

The Imbalance ratio is of 0.92 : 0.08 for TARGET == 0 and TARGET == 1*

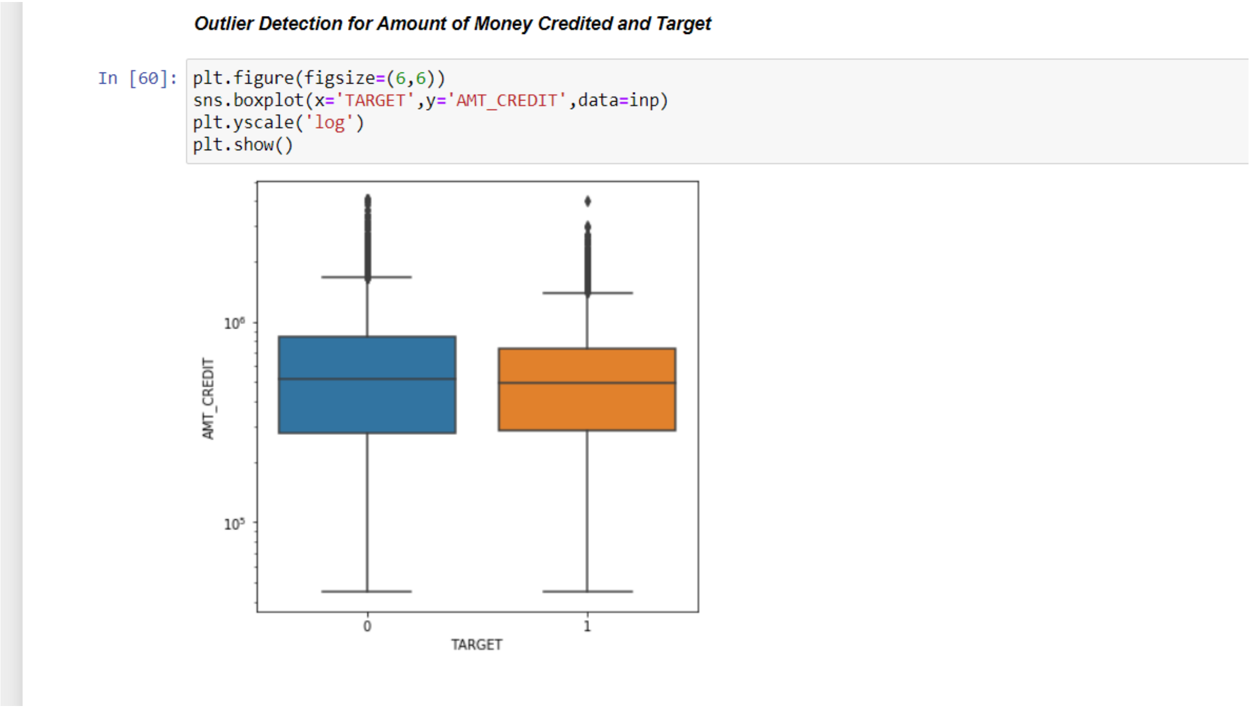
***The Imbalance ratio is of 0.92 : 0.08 for TARGET == 0 and TARGET == 1*

Analysis of Data Imbalance for Target

We can find a a data imbalance here with respect to the Target variable of the dataset since the data points for other i.e applicants with payment difficulties is at around 8.66% only compared to the total number of applicants.

OUTLIER DETECTION:

We have to Identify if there are outliers in the dataset. Also, mention why do you think it is an outlier. Again, remember that for this exercise, it is not necessary to remove any data points.



Outlier Detection for Amount of Money Credited and Target

```
In [60]: plt.figure(figsize=(6,6))
         sns.boxplot(x='TARGET',y='AMT_CREDIT',data=inp)
         plt.yscale('log')
         plt.show()
```

Here from the above boxplot between the Target and the Income of the client, we can find, the income for the clients who are having payment difficulties and other cases both are in similar range with median value almost same. But most importantly, what we can notice is outlier values present in the dataset. There are outlier values present for both the Target cases, but we can easily spot a High-Income client present who had previously a payment difficulty**.**

<u>UNIVARIATE ANALYSIS</u> :

Categorical Univariate Analysis Target 1

Distribution of Income range

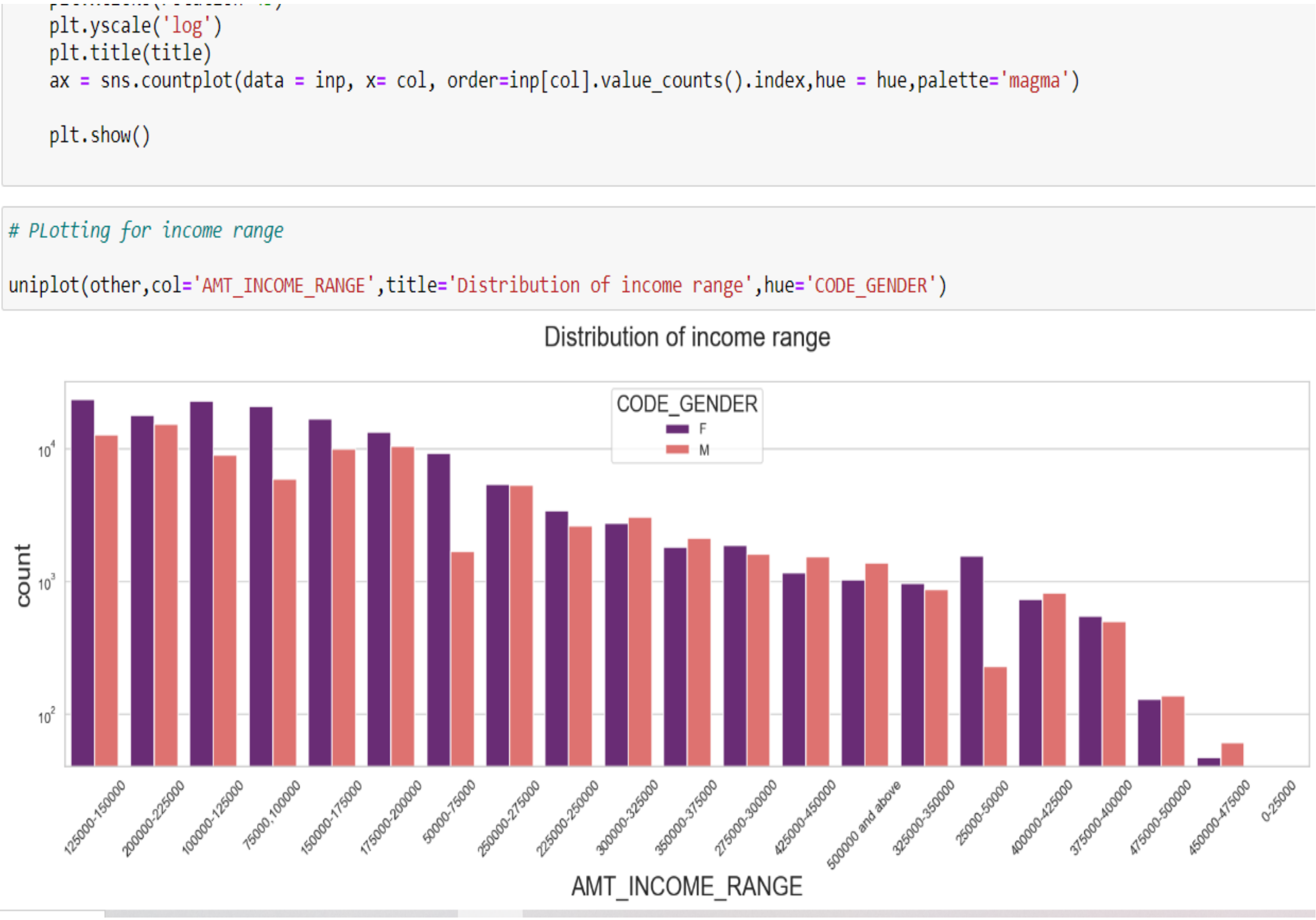
Points to Note from above graph.

a) Females counts are higher.

b) Income range from 100000 to 200000 is having more no. of credits.

c) This graph show that females are more than male in having credits for that range.

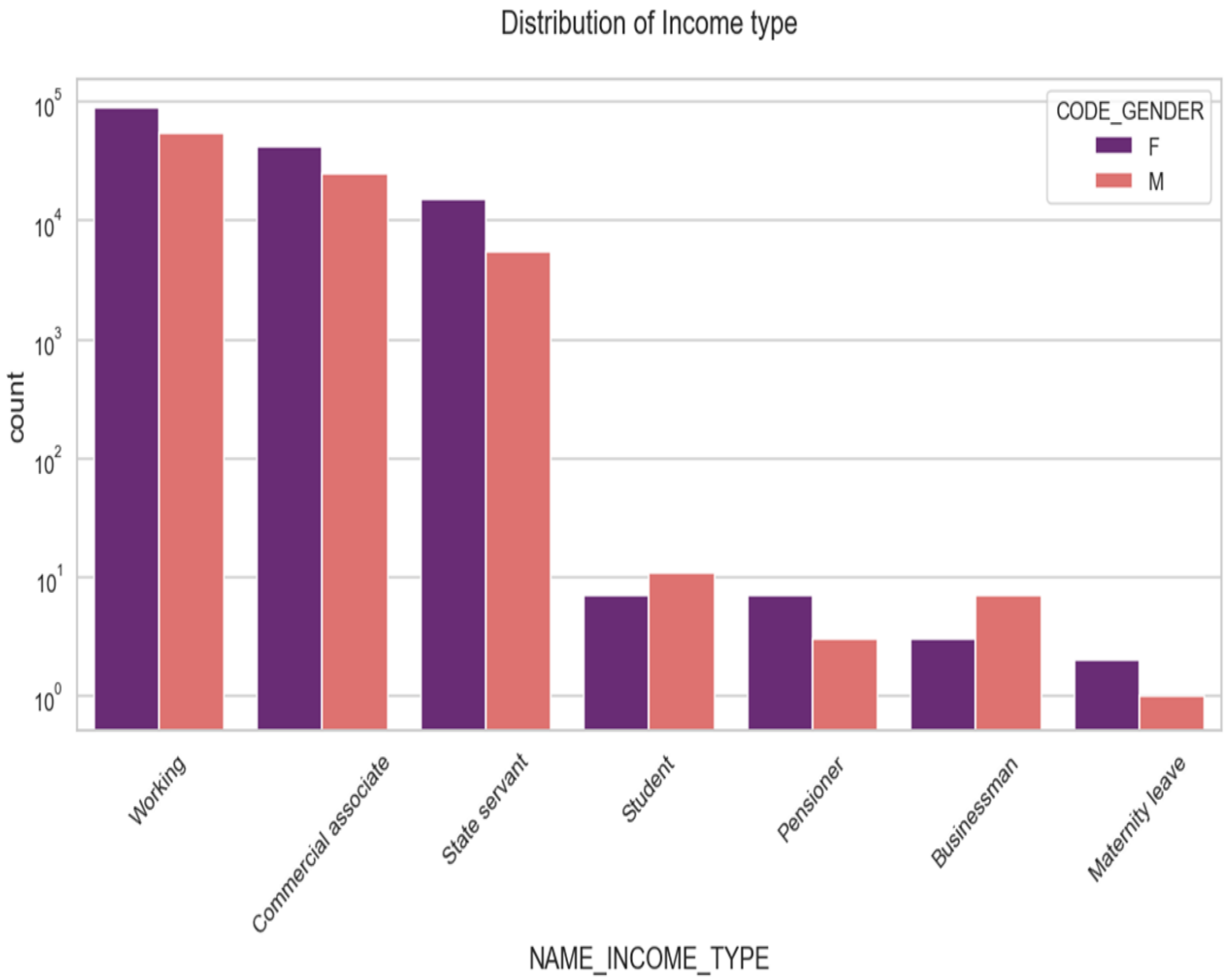
d) Very less count for income range 400000 and above



```
    plt.yscale('log')
    plt.title(title)
    ax = sns.countplot(data = inp, x= col, order=inp[col].value_counts().index,hue = hue,palette='magma')

    plt.show()
```

```
# PLotting for income range

uniplot(other,col='AMT_INCOME_RANGE',title='Distribution of income range',hue='CODE_GENDER')
```

Distribution of income type:

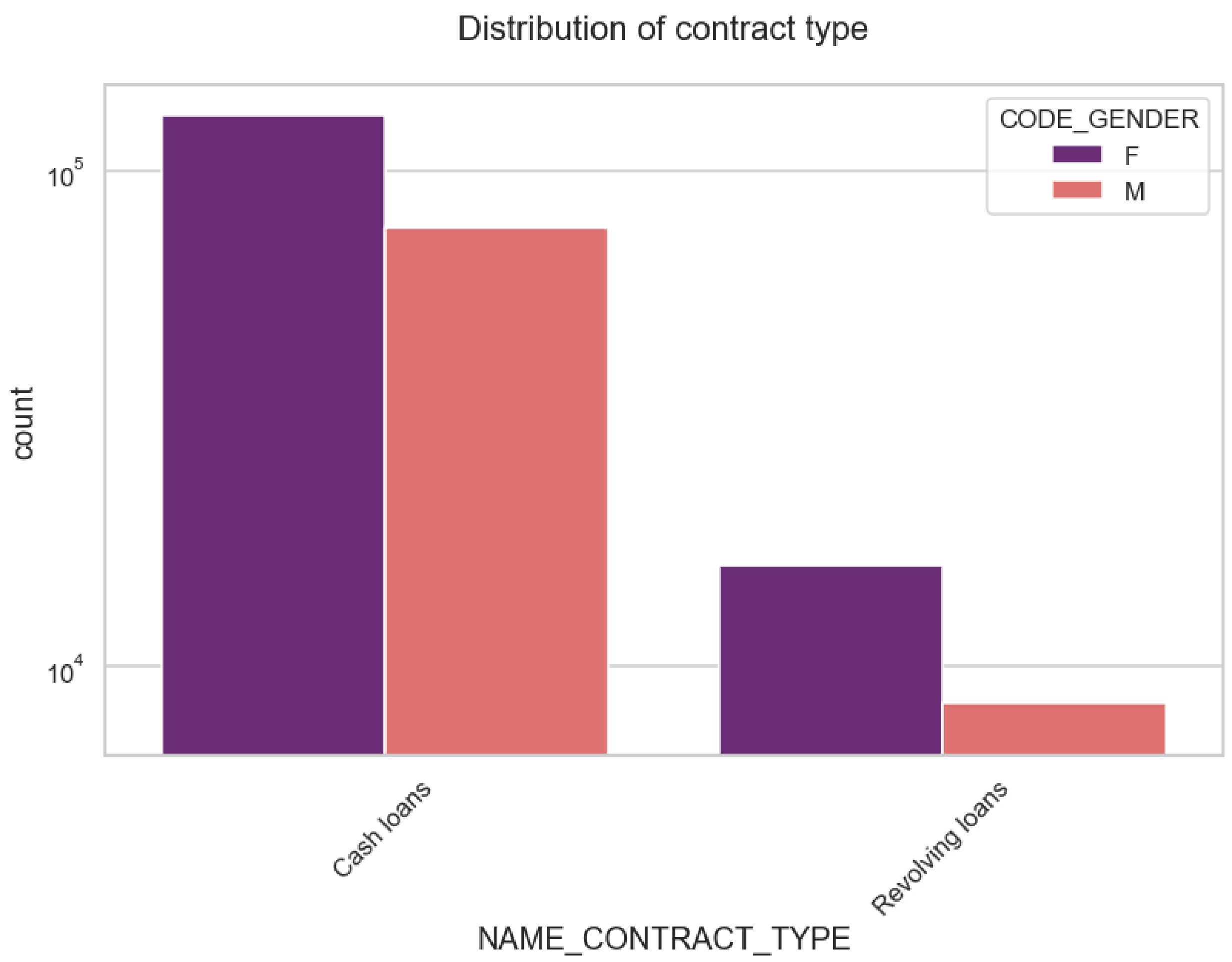


For income type ‘working’, ’commercial associate’, and ‘State Servant’ the number of credits is higher than others. For this Females are having a greater number of credits than male. Less number of credits for income type ‘student’, ‘pensioner’, ‘Businessman’ and ‘Maternity leave’.

Distribution for contract type :

Points to be concluded from the graph on the right.

- **For contract type ‘cash loans’ is having higher number of credits than ‘Revolving loans’ contract type.**
- **For this also Female is leading for applying credits.**

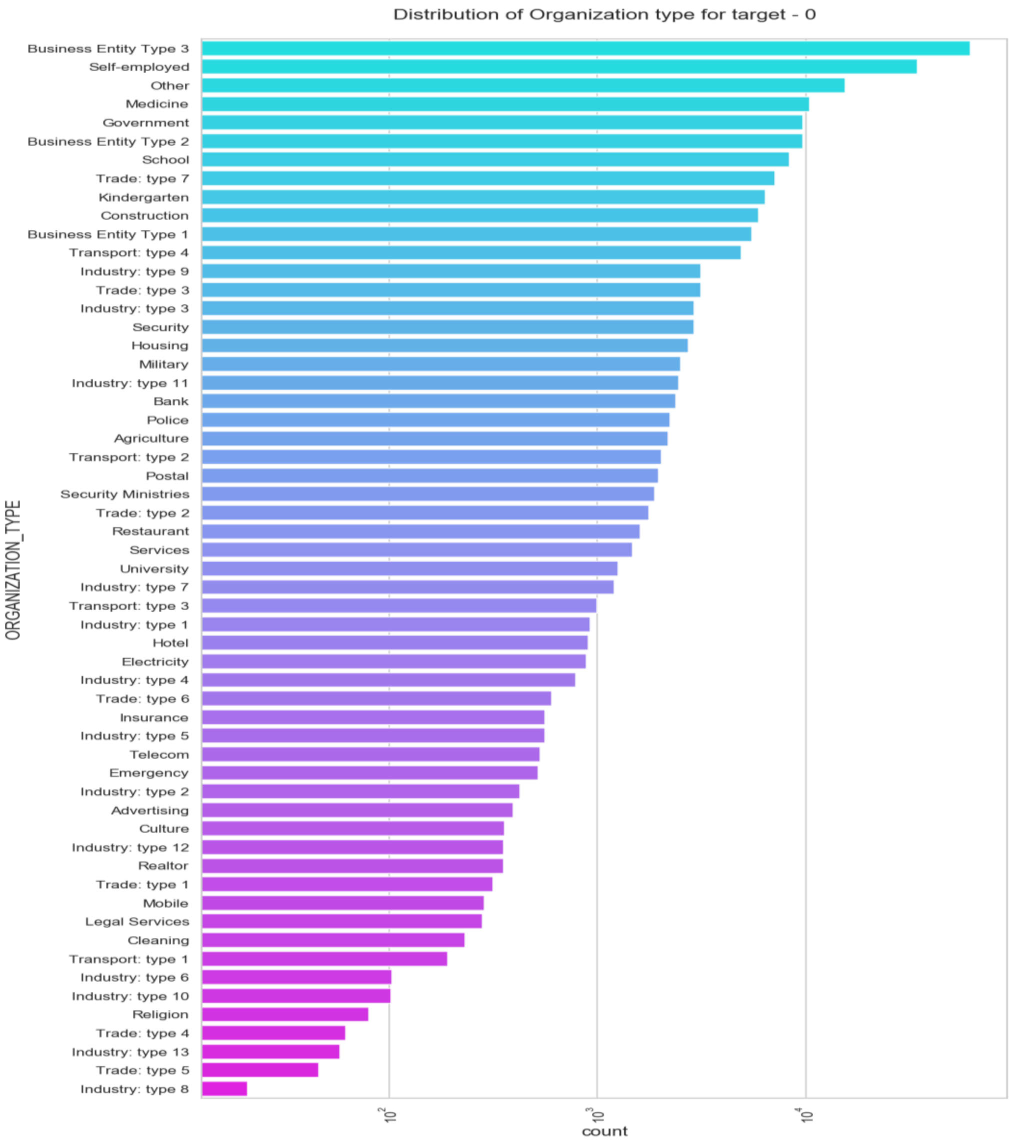


Points to Note:

- Around 91% of the client Loans are Cash Loans. Rest 9% are the only revolving Loans.
- Based on the Analysis Done above we find that the Number of People who don't have monetary problems are higher than the ones who have monetary Problems.
- But, unfortunately the Number of Cash Loans given out to Customers are higher than Revolving Loans. Revolving Loans has few benefits for both customers as well as Banks.
  - ***a) For contract type ‘cash loans’ is having higher number of credits than ‘Revolving loans’ contract type.***
  - ***b) For this also Female is leading for applying credits***

Plotting for Organization type in logarithmic scale

Distribution of organization type:



Points to be concluded from the graph on the right.

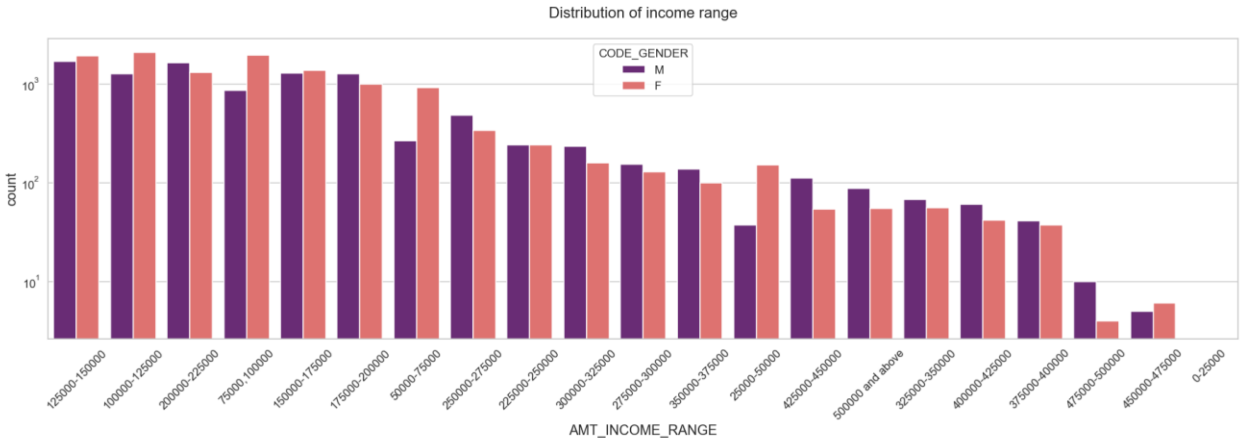
- Clients which have applied for credits are from most of the organization type ‘Business entity

- Less clients are from Industry type 8,type 6, type 10, religion and trade type 5, type 4.

Categorical Univariate Analysis Target 1

Distribution of Income range :

- Points to learn
- Income range from 100000 to 200000 is having more number of credits.
- This graph show that males are more than female in having credits for that range.

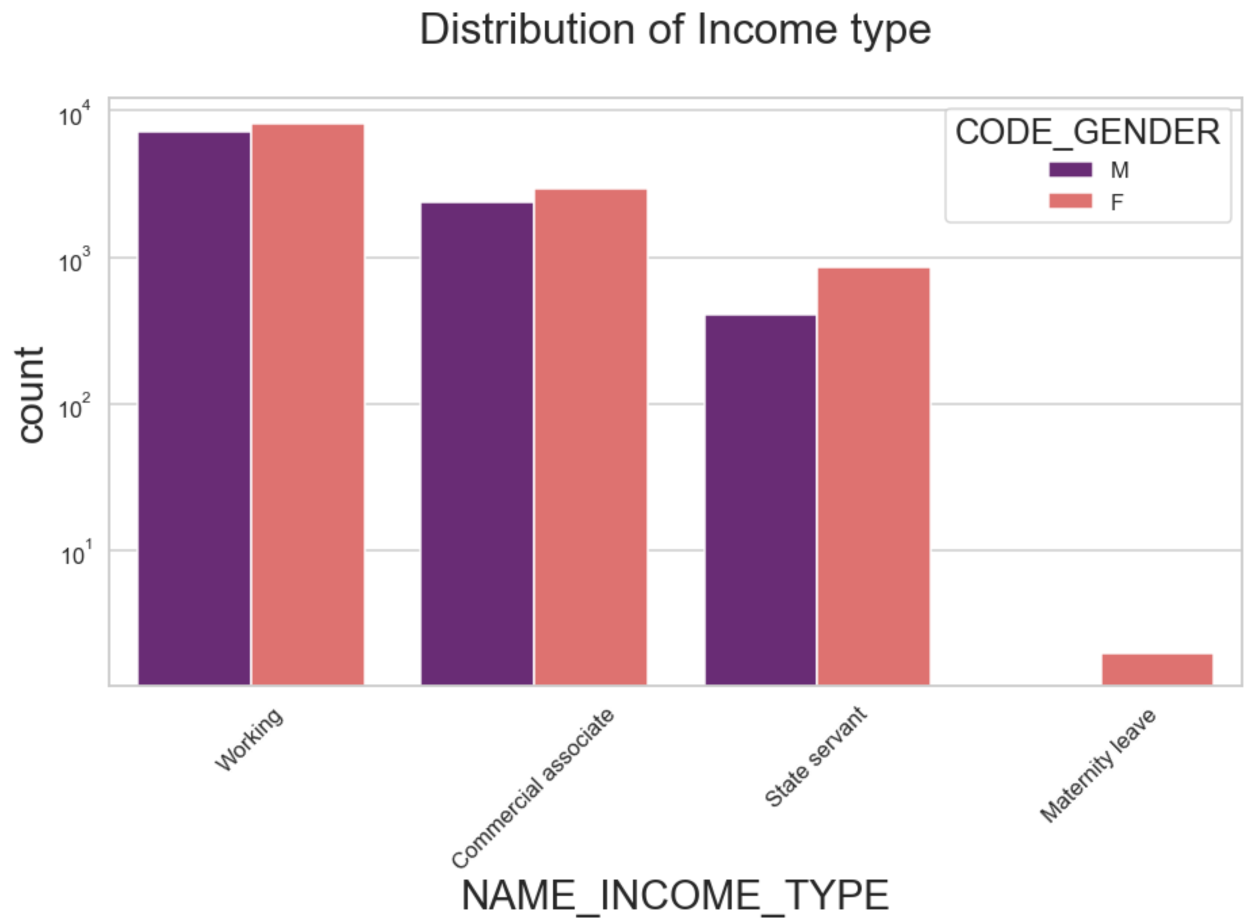


Distribution of Income type

Points to Note :

a) For income type ‘working’, ’commercial associate’, and ‘State Servant’ the number of credits are higher than other i.e. ‘Maternity leave.

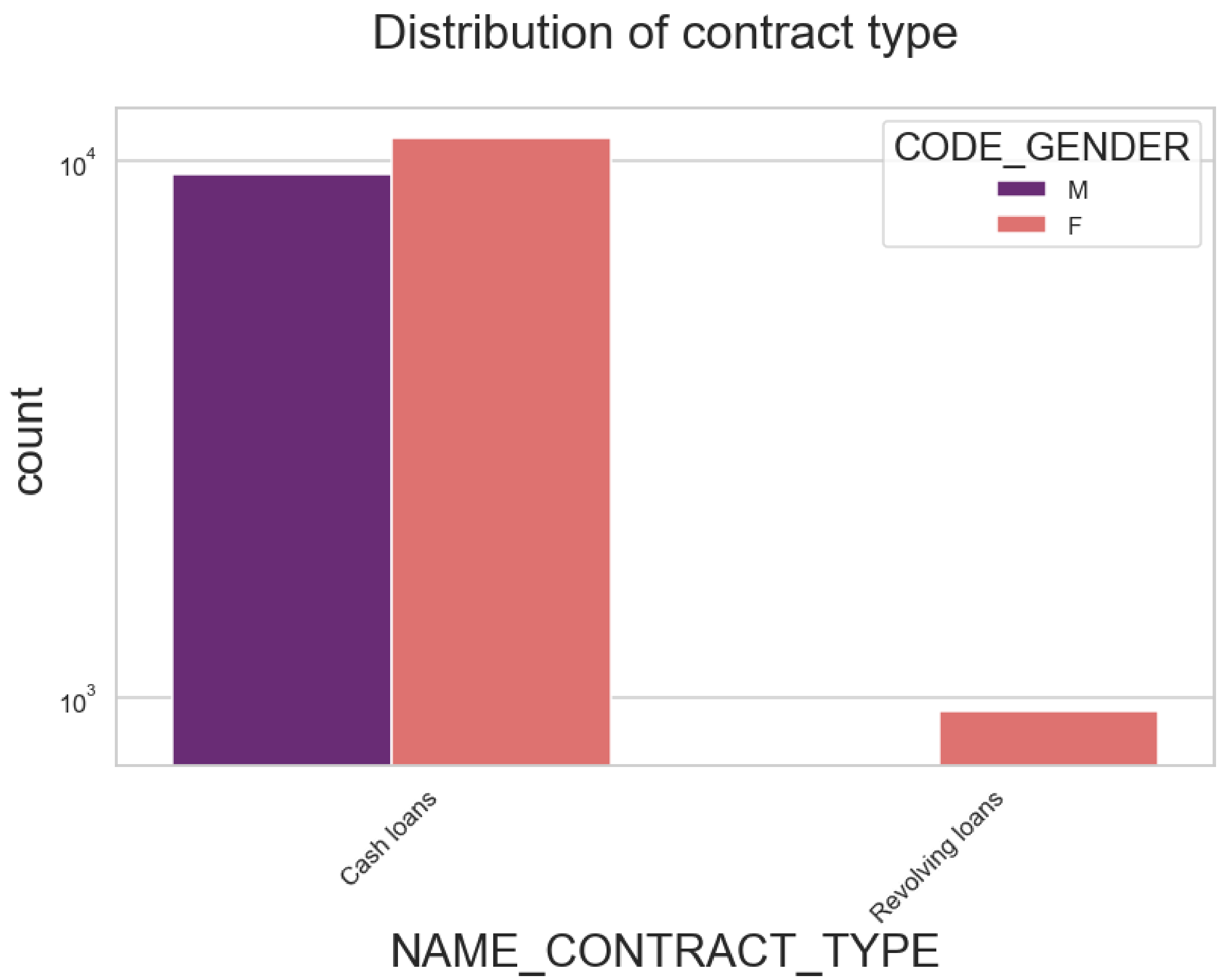
b) For this Females are having more number of credits than male.



Distribution of contract type :

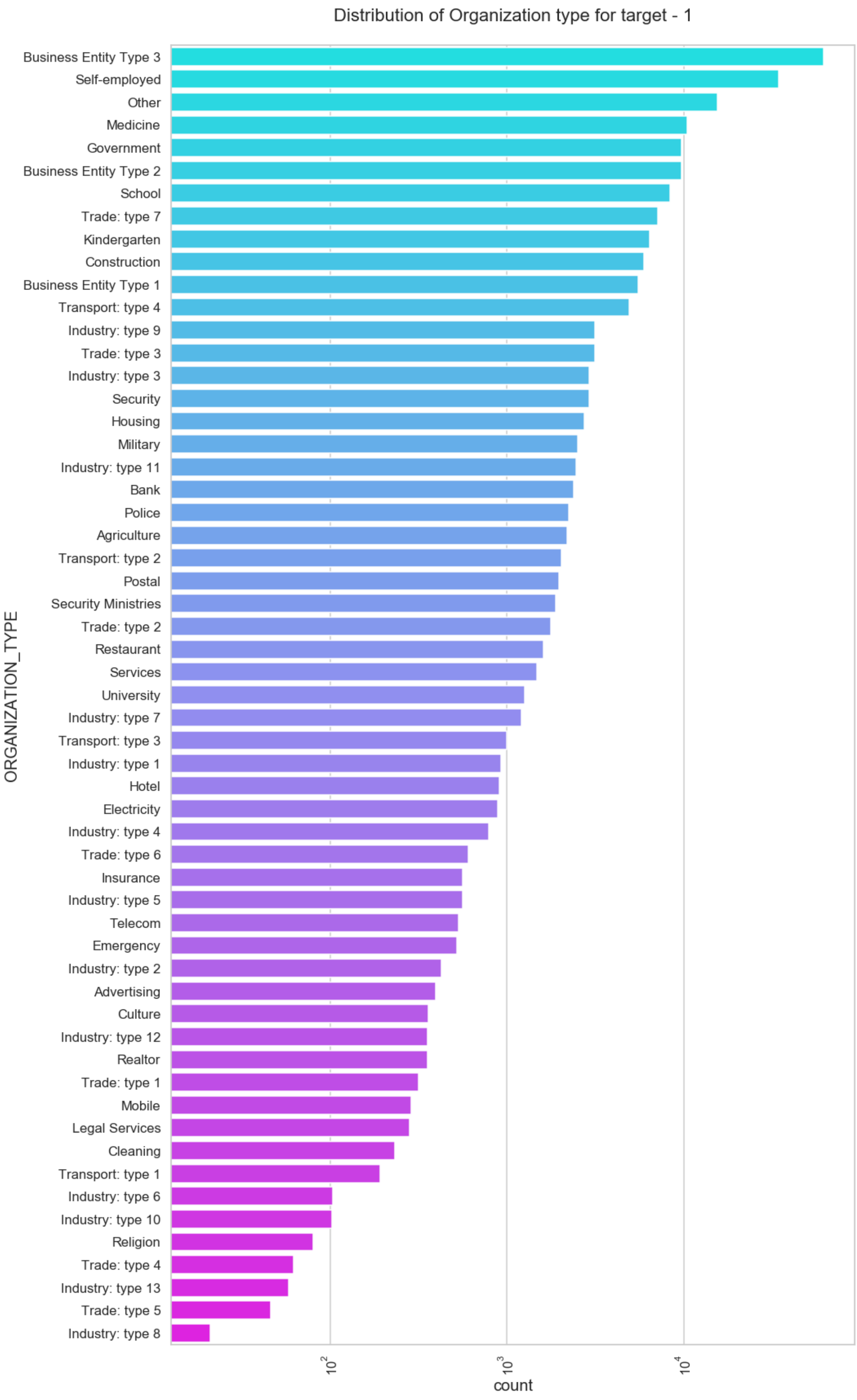
Points to be concluded from the graph on the right.

- For contract type ‘cash loans’ is having higher number of credits than ‘Revolving loans’ contract type.
- For this also Female is leading for applying credits.
- For type 1 : there is only Female Revolving loans.

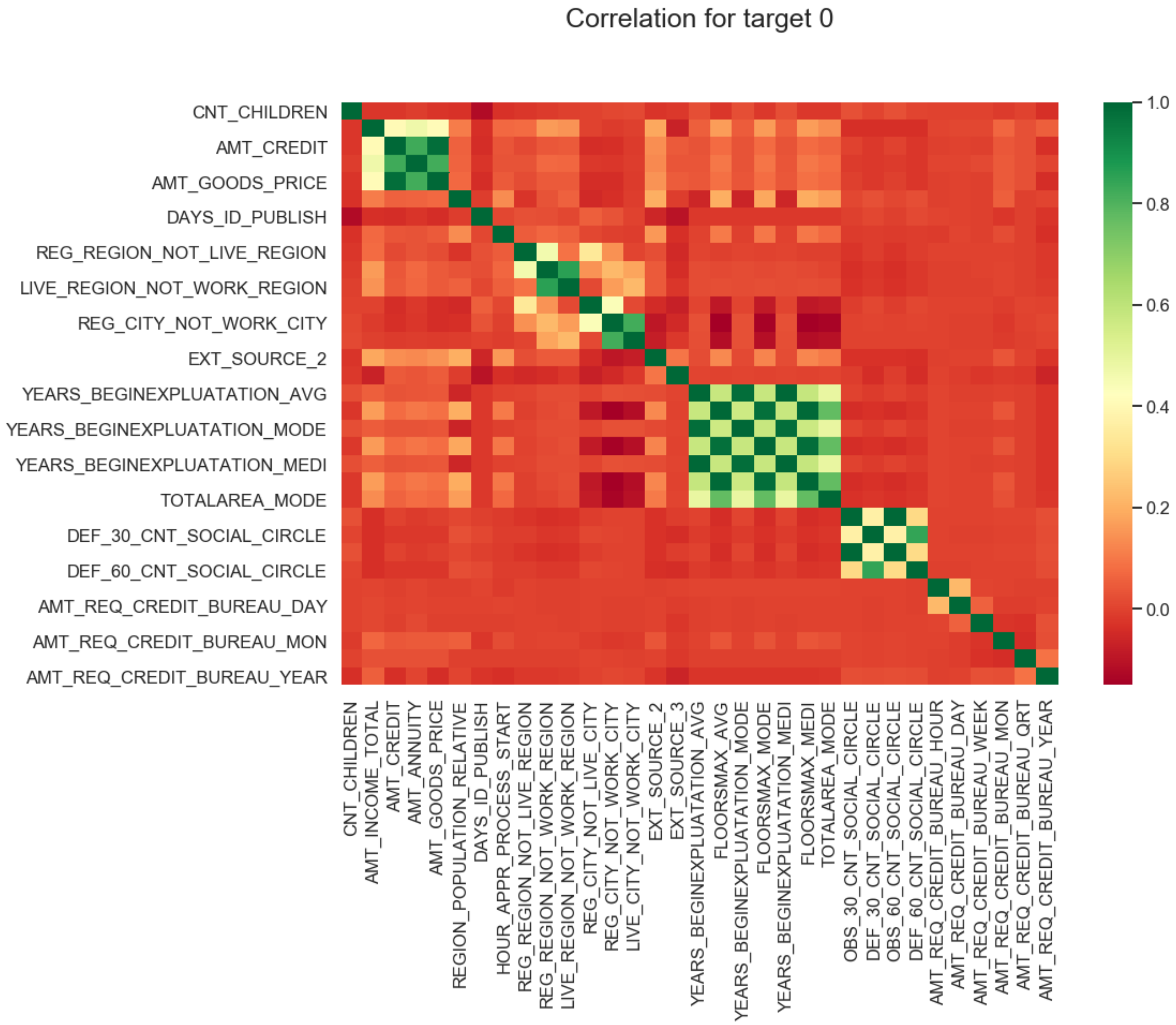


Distribution of Organization type for target – 1:

Clients which have applied for credits are from most of the organization type ‘Business entity Type 3’ , ‘Self-employed’ , ‘Other’ , ‘Medicine’ and ‘Government’. Less clients are from Industry type 8,type 6, type 10, religion and trade type 5, type 4.Same as type 0 in distribution of organization type.


Distribution of Organization type for target - 1
ORGANIZATION_TYPE
Business Entity Type 3
Self-employed
Other
Medicine
Government
Business Entity Type 2
School
Trade: type 7
Kindergarten
Construction
Business Entity Type 1
Transport: type 4
Industry: type 9
Trade: type 3
Industry: type 3
Security
Housing
Military
Industry: type 11
Bank
Police
Agriculture
Transport: type 2
Postal
Security Ministries
Trade: type 2
Restaurant
Services
University
Industry: type 7
Transport: type 3
Industry: type 1
Hotel
Electricity
Industry: type 4
Trade: type 6
Insurance
Industry: type 5
Telecom
Emergency
Industry: type 2
Advertising
Culture
Industry: type 12
Realtor
Trade: type 1
Mobile
Legal Services
Cleaning
Transport: type 1
Industry: type 6
Industry: type 10
Religion
Trade: type 4
Industry: type 13
Trade: type 5
Industry: type 8
10^2
10^3
10^4
count

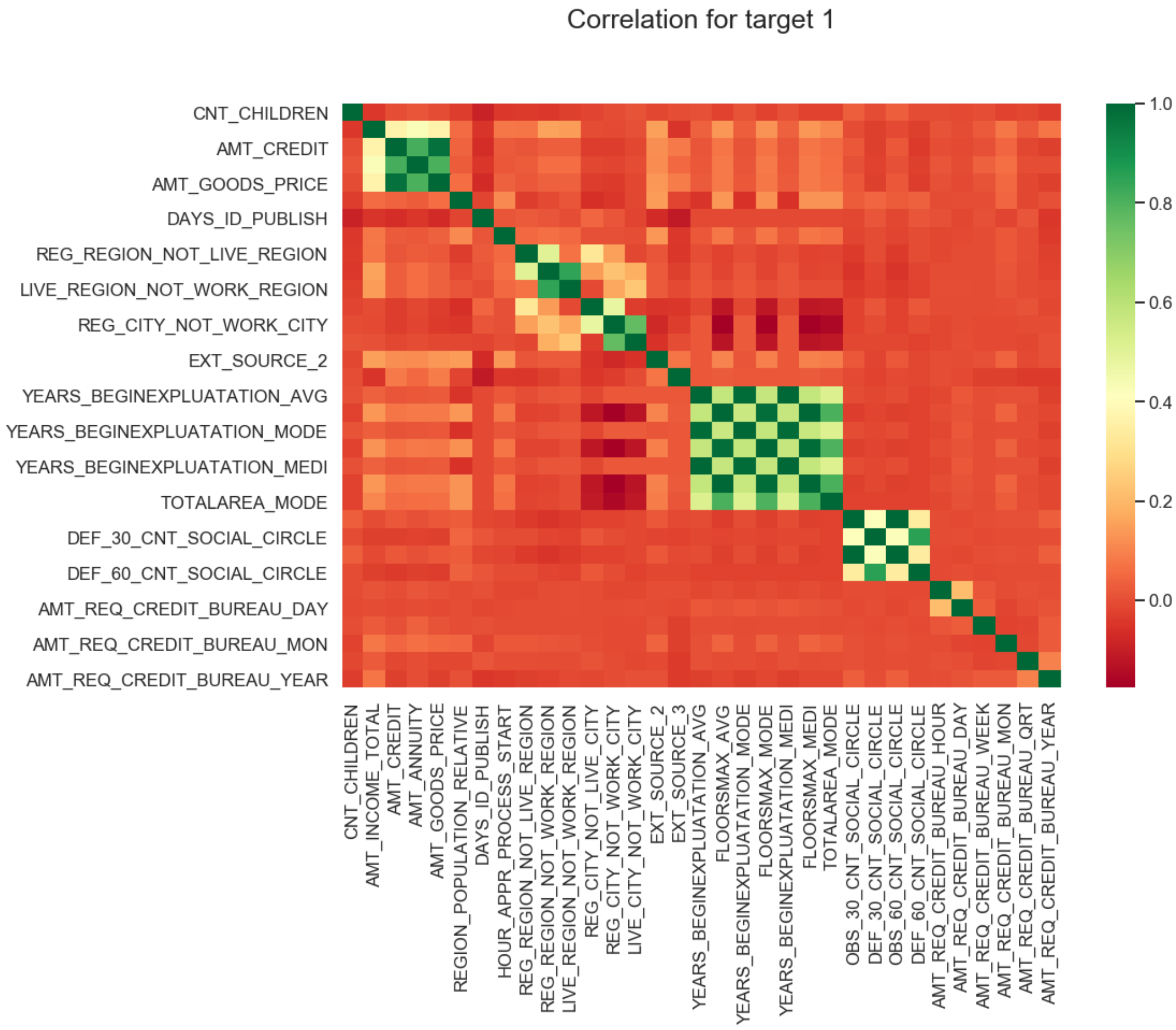
CORRELATION OF TARGET 0 :



Points to be concluded from the graph presented before.

- Credit amount is inversely proportional to the date of birth, which means Credit amount is higher for low age and vice-versa.
- Credit amount is inversely proportional to the number of children client have, means Credit amount is higher for less children count client have and vice-versa.
- Income amount is inversely proportional to the number of children client have, means more income for less children client have and vice-versa.
- less children client have in densely populated area.
- Credit amount is higher to densely populated area.
- The income is also higher in densely populated area.

Correlation for target 1 :



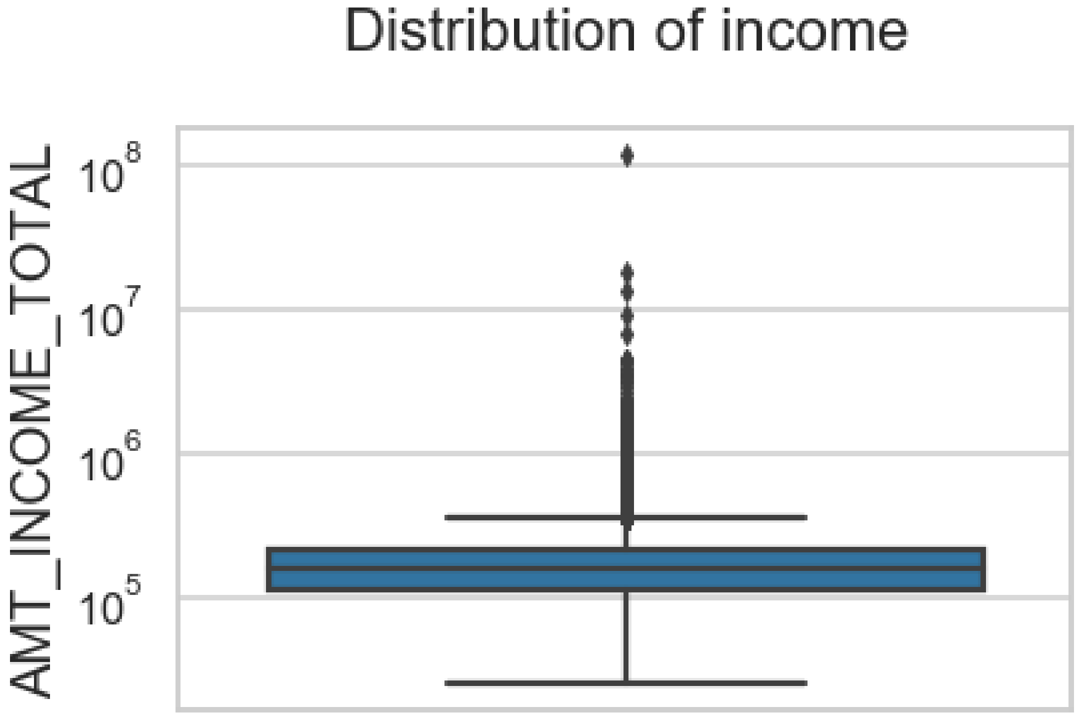
This heat map for Target 1 is also having quite a same observation just like Target 0.Excepet few one

The client's permanent address does not match contact address are having less children and vice-versa.and the client's permanent address does not match work address are having less children and vice-versa.

Boxplot for income amount

Few points can be concluded from the graph.

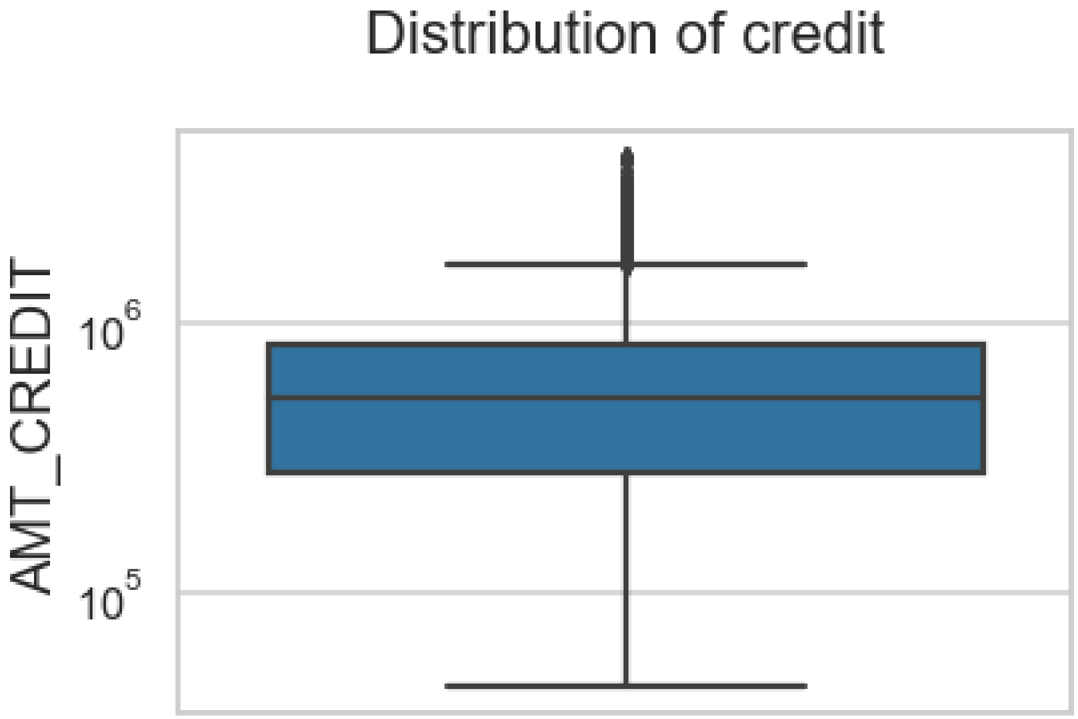
- Some outliers are noticed in income amount.
- The third quartiles is very slim for income amount.



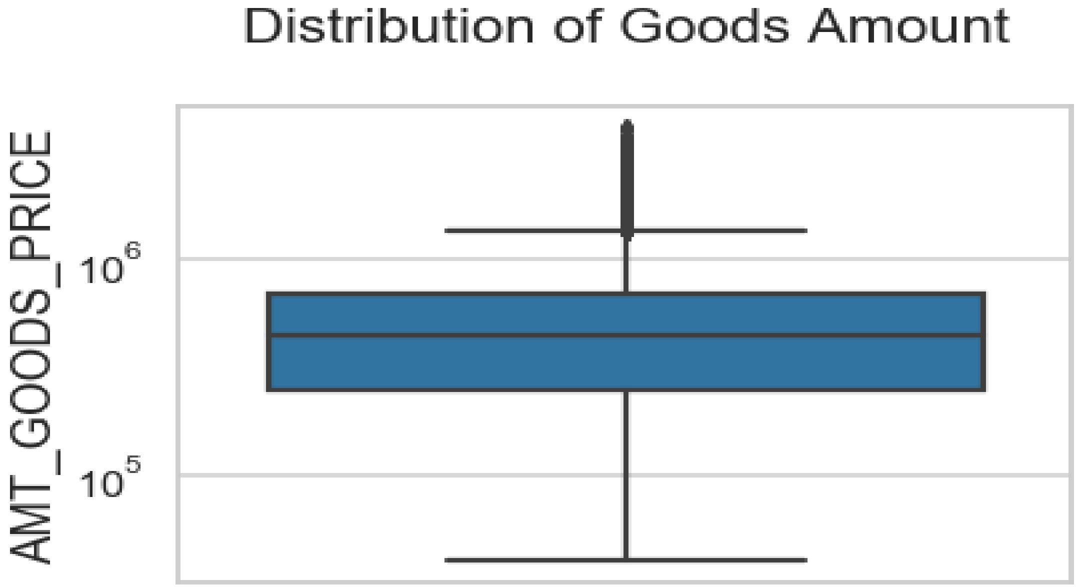
Distribution of credit:

Few points can be concluded from the graph.

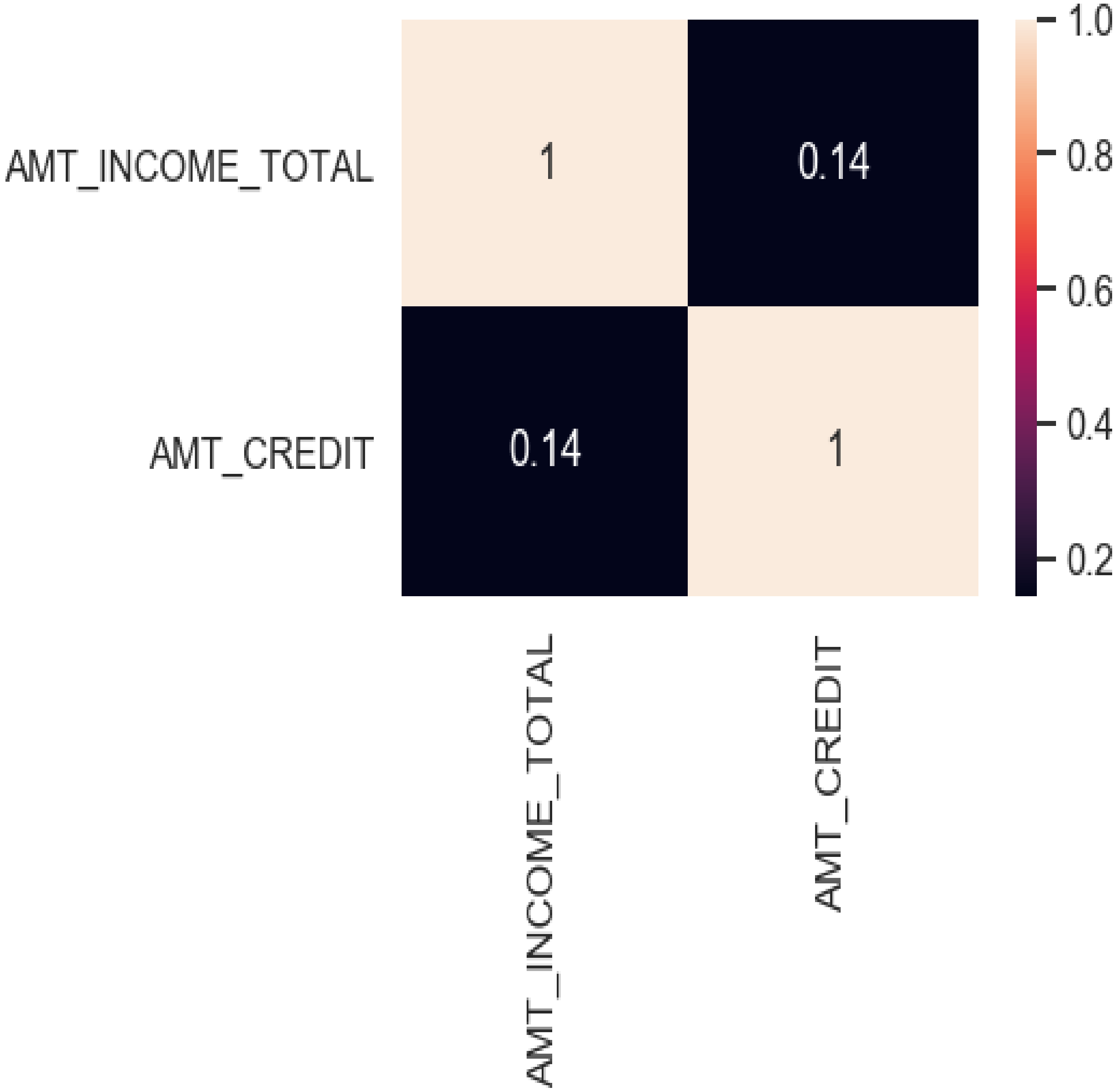
- Some outliers are noticed in credit amount.
- The first quartile is bigger than third quartile for credit amount which means most of the credits of clients are present in the first quartile.



Distribution of Goods Amount :

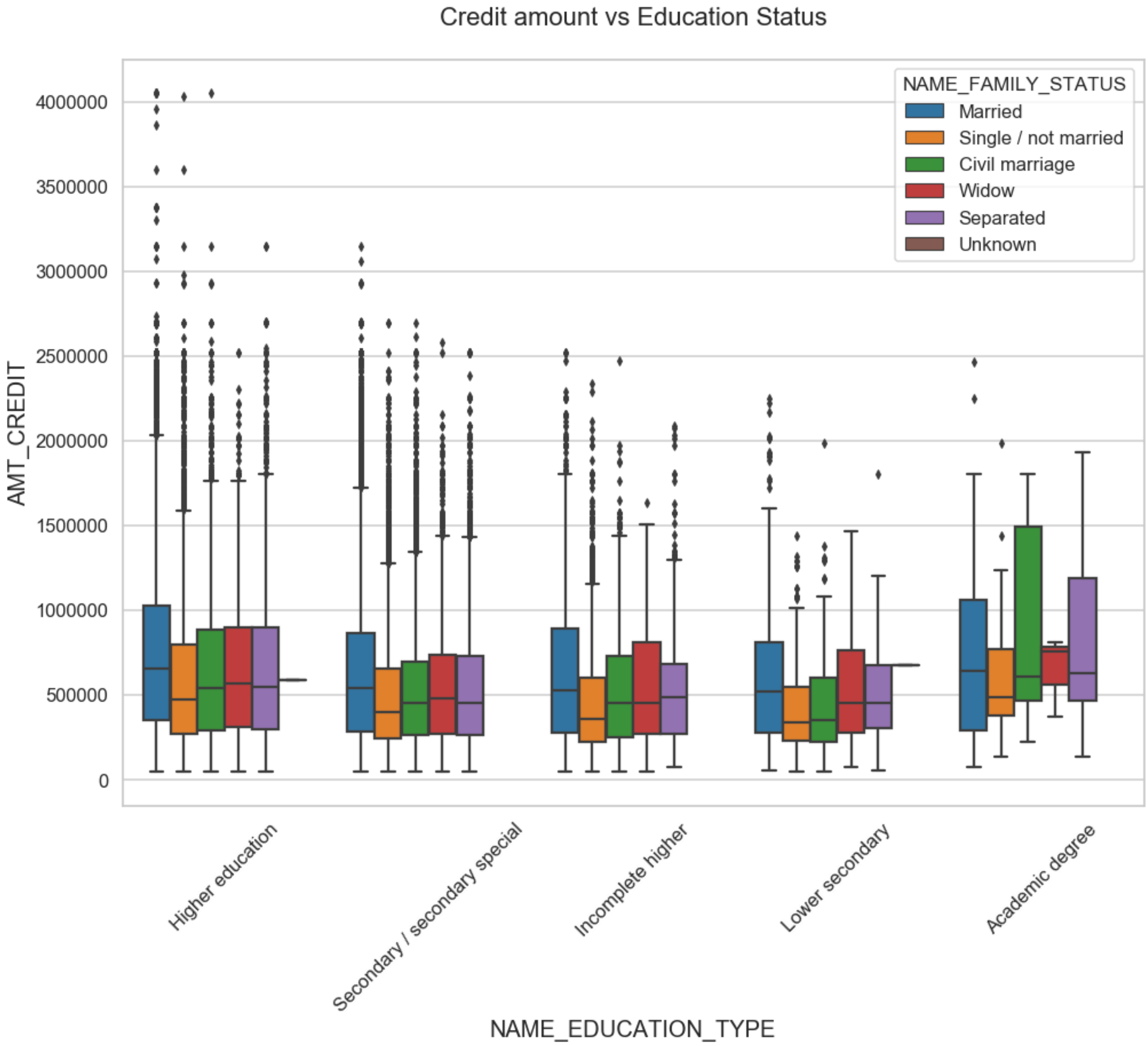


We also find out the correlation between the income of the Clients and the Credit amount of the Loan.

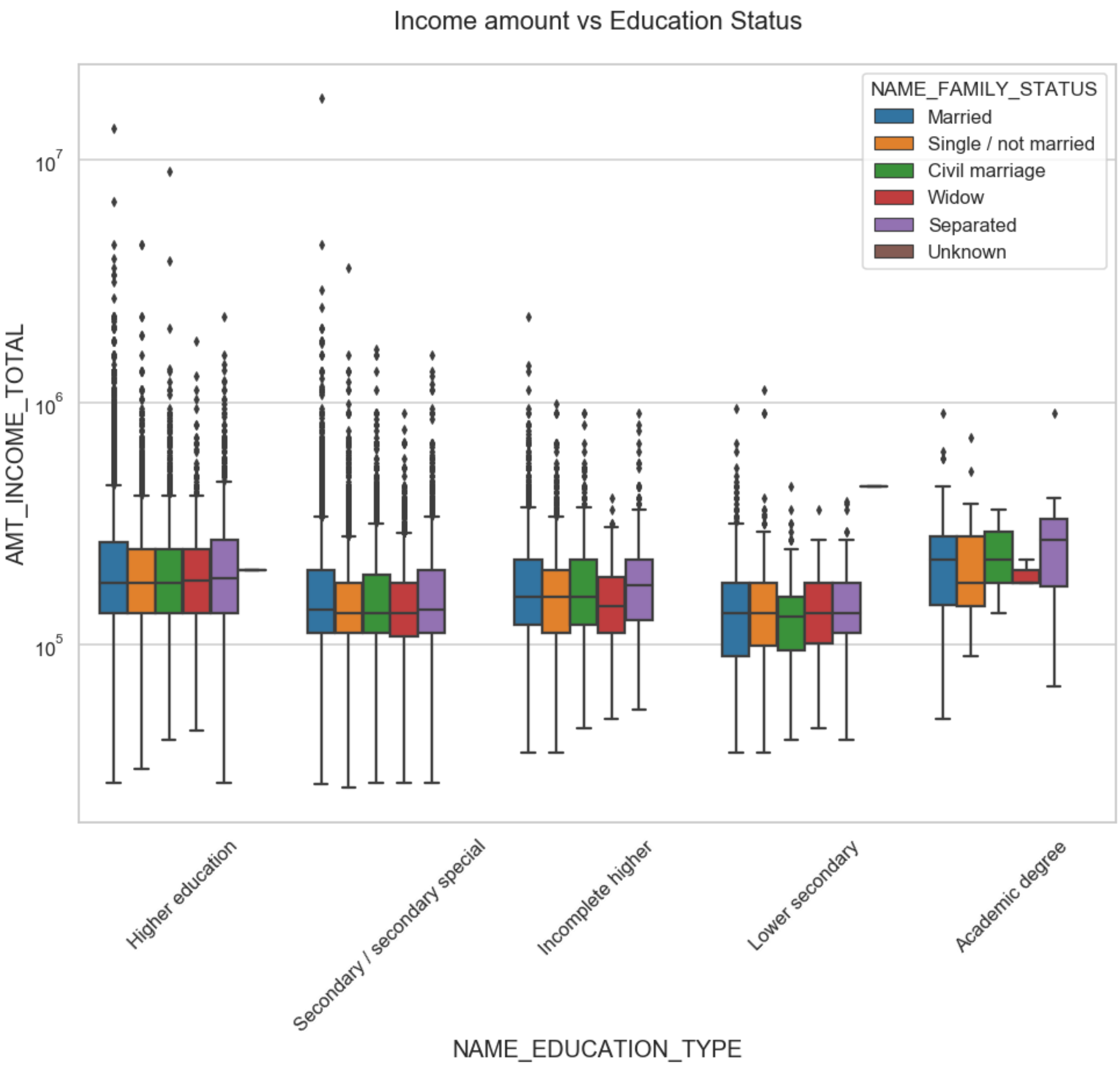


The above Heatmap shows very low correlation between the Income of the Applicants and the Loan credited amount which in turn implies that even less income applicants have opted out for a large Loan amount and also high income applicant would have opted out for Small Loan amounts.

Bivariate analysis for type 0

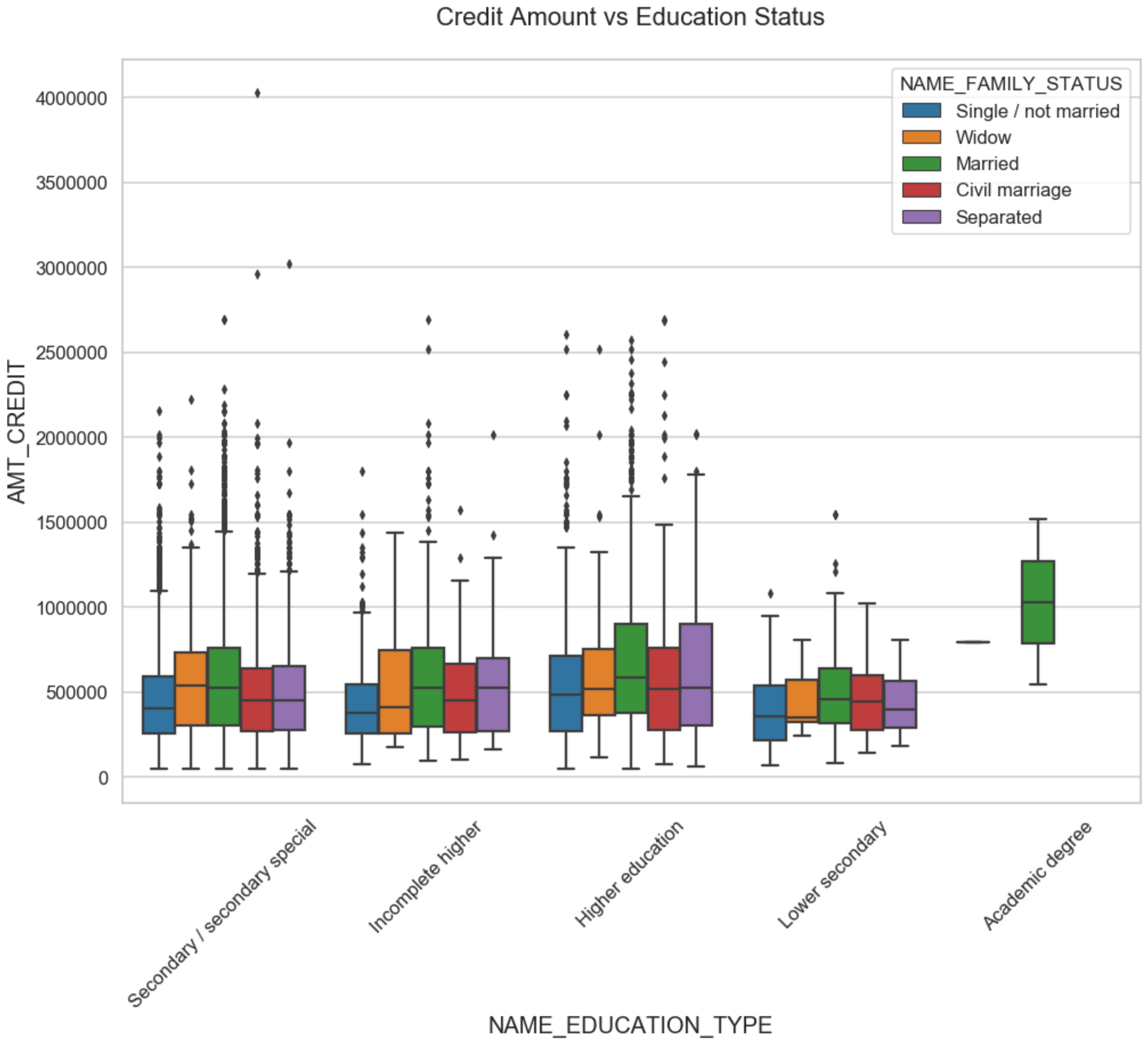


From the above box plot we can conclude that Family status of 'civil marriage', 'marriage' and 'separated' of Academic degree education are having higher number of credits than others. Also, higher education of family status of 'marriage', 'single' and 'civil marriage' are having more outliers. Civil marriage for Academic degree is having most of the credits in the third quartile**.**



From above boxplot for Education type 'Higher education' the income amount is mostly equal with family status. It does contain many outliers. Less outlier are having for Academic degree but there income amount is little higher that Higher education. Lower secondary of civil marriage family status are have less income amount than others.

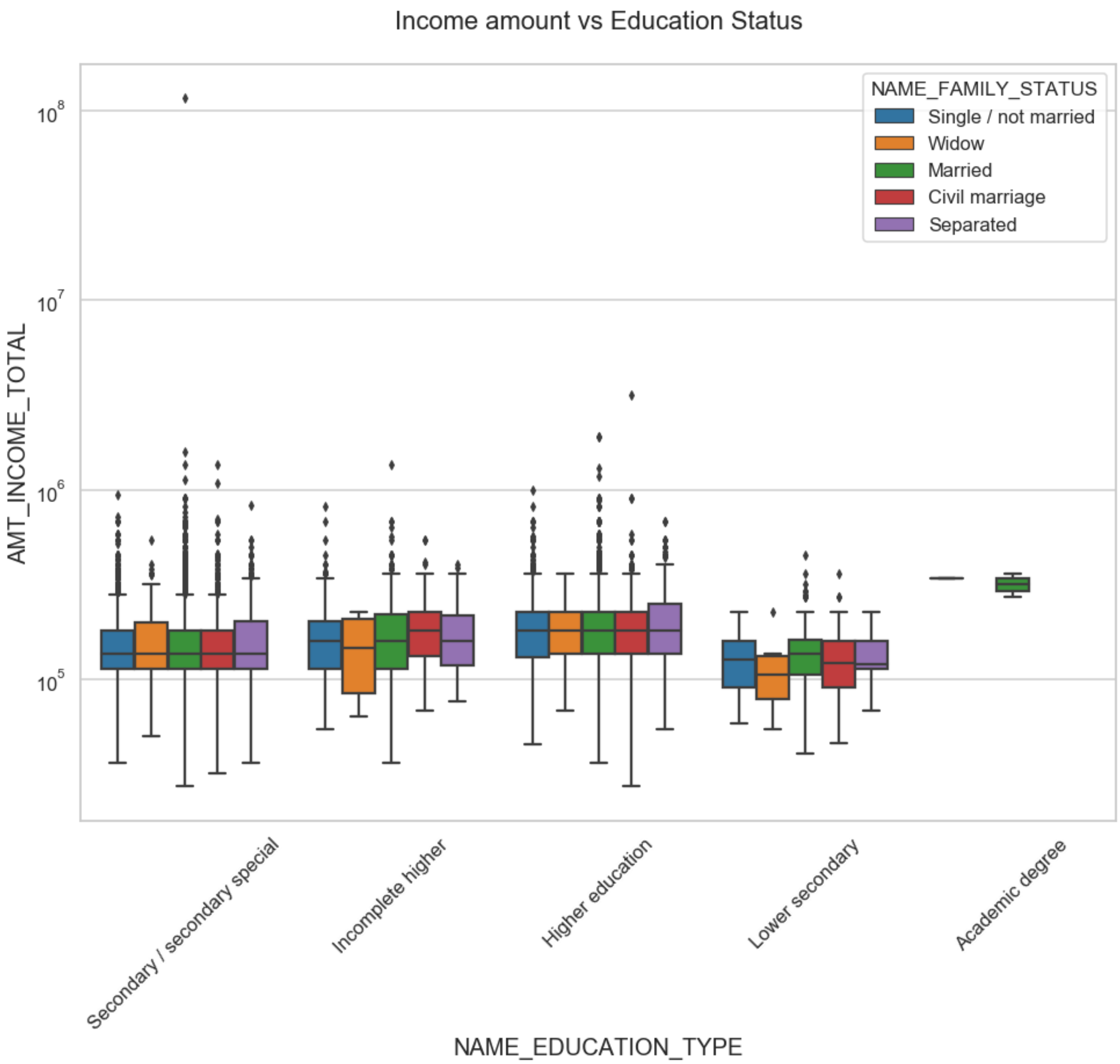
Bivariate analysis for type 1



Few points can be concluded from the graph.

- Family status of 'civil marriage', 'marriage' and 'separated' of Academic degree education are having higher number of credits than others.
- Higher education of family status of 'marriage', 'single' and 'civil marriage' are having more outliers.

Civil marriage for Academic degree is having most of the credits in the third quartile

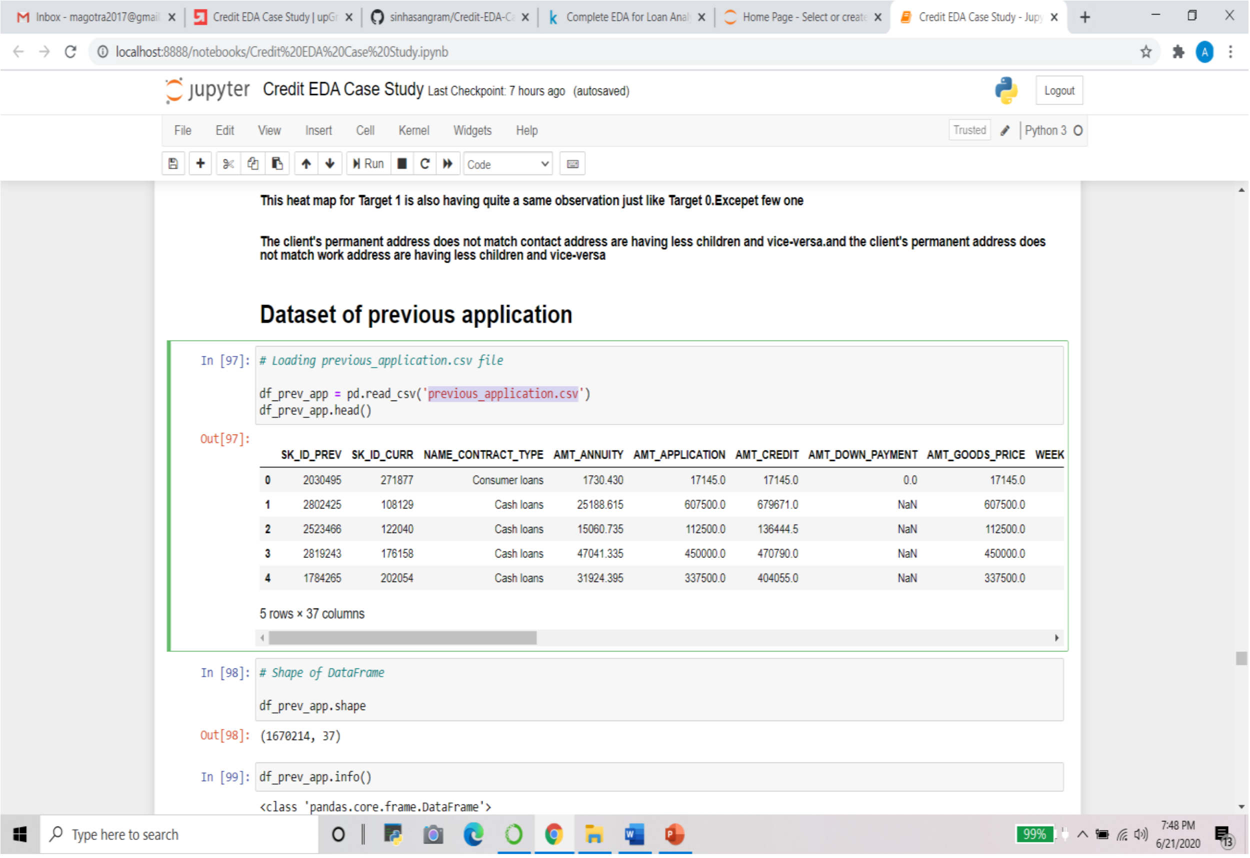


Few points can be concluded from the graph.

- For Education type 'Higher education' the income amount mean is mostly equal with family status. It does contain many outliers.
- Less outlier are having for Academic degree but they are having the income amount is little higher that Higher education.
- Lower secondary of civil marriage family status are have less income amount than others.

Dataset of previous application:

Now we have to import the data for ‘previous_application.csv’



jupyter
Credit EDA Case Study Last Checkpoint: 7 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

This heat map for Target 1 is also having quite a same observation just like Target 0.Excepet few one

The client's permanent address does not match contact address are having less children and vice-versa.and the client's permanent address does not match work address are having less children and vice-versa

Dataset of previous application

```
In [97]: # Loading previous_application.csv file

         df_prev_app = pd.read_csv('previous_application.csv')
         df_prev_app.head()
```

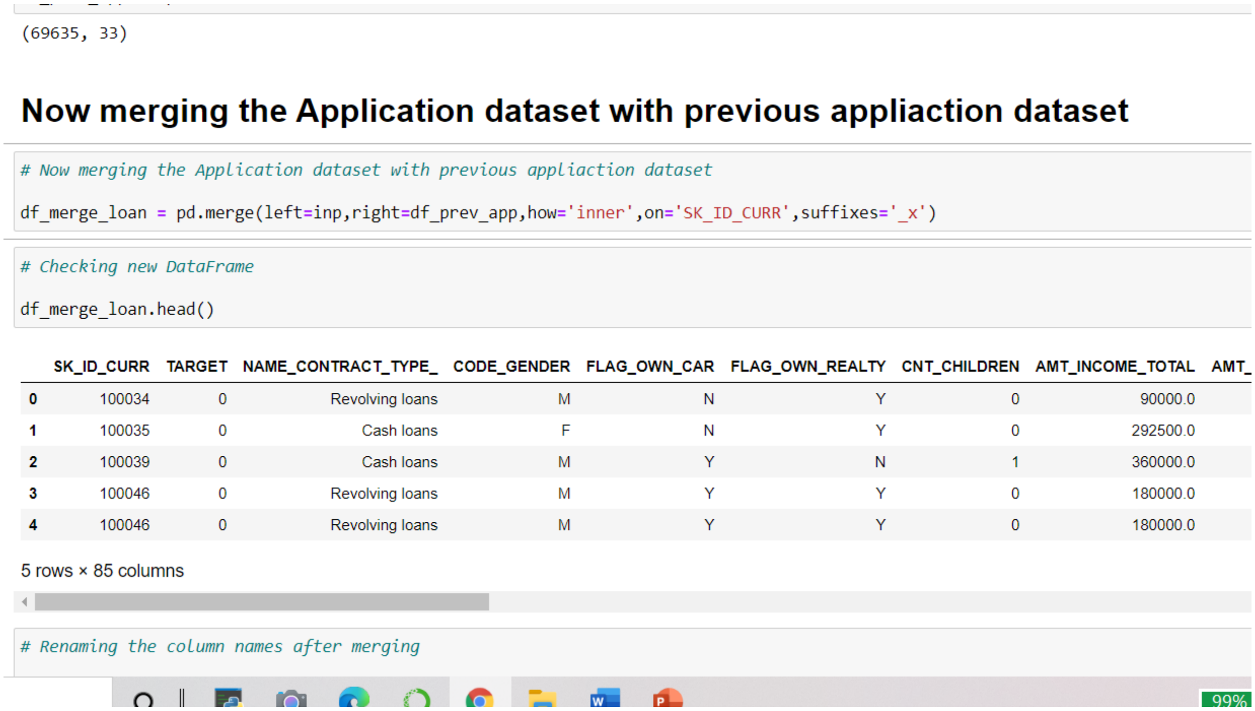
Out[97]:

	SK_ID_PREV	SK_ID_CURR	NAME_CONTRACT_TYPE	AMT_ANNUITY	AMT_APPLICATION	AMT_CREDIT	AMT_DOWN_PAYMENT	AMT_GOODS_PRICE	WEEK
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	

5 rows × 37 columns

```
In [98]: # Shape of DataFrame

         df_prev_app.shape

Out[98]: (1670214, 37)

In [99]: df_prev_app.info()

         <class 'pandas.core.frame.DataFrame'>
```

1. Then Cleaning the missing data, and drop Columns calculated from above.
2. Removing the column values of 'XNA' and 'XAP'.
3. After that merging the Application dataset with previous application dataset



```
(69635, 33)
```

Now merging the Application dataset with previous appliaction dataset

```
# Now merging the Application dataset with previous appliaction dataset

df_merge_loan = pd.merge(left=inp,right=df_prev_app,how='inner',on='SK_ID_CURR',suffixes='_x')
```

```
# Checking new DataFrame

df_merge_loan.head()
```

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE_	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_
0	100034	0	Revolving loans	M	N	Y	0	90000.0	
1	100035	0	Cash loans	F	N	Y	0	292500.0	
2	100039	0	Cash loans	M	Y	N	1	360000.0	
3	100046	0	Revolving loans	M	Y	Y	0	180000.0	
4	100046	0	Revolving loans	M	Y	Y	0	180000.0	

5 rows × 85 columns

```
# Renaming the column names after merging
```

Univariate analysis after merging previous data

Distribution of contract status with purposes

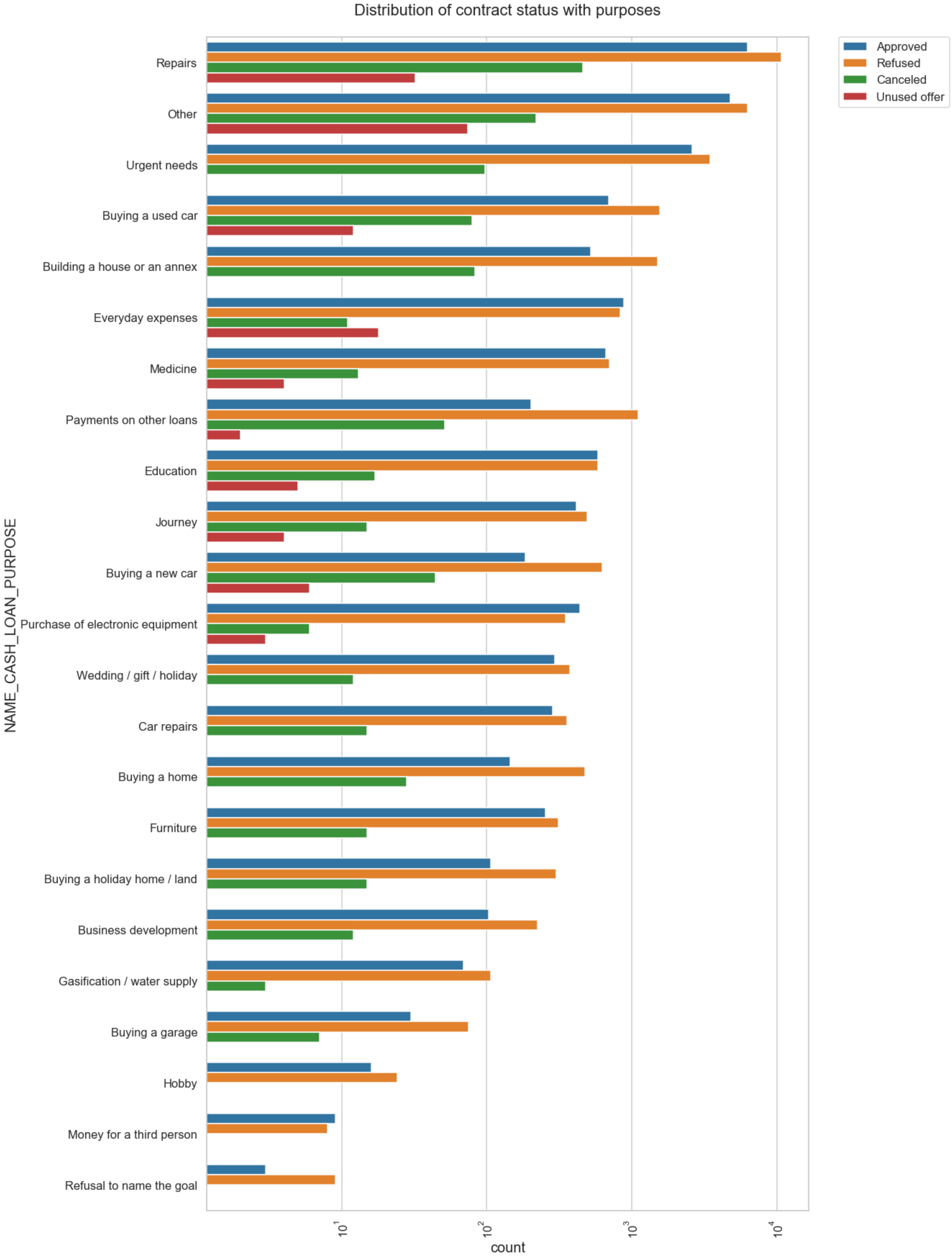
Conclusion:

1. *Most rejection of loans came from purpose 'repairs'.*

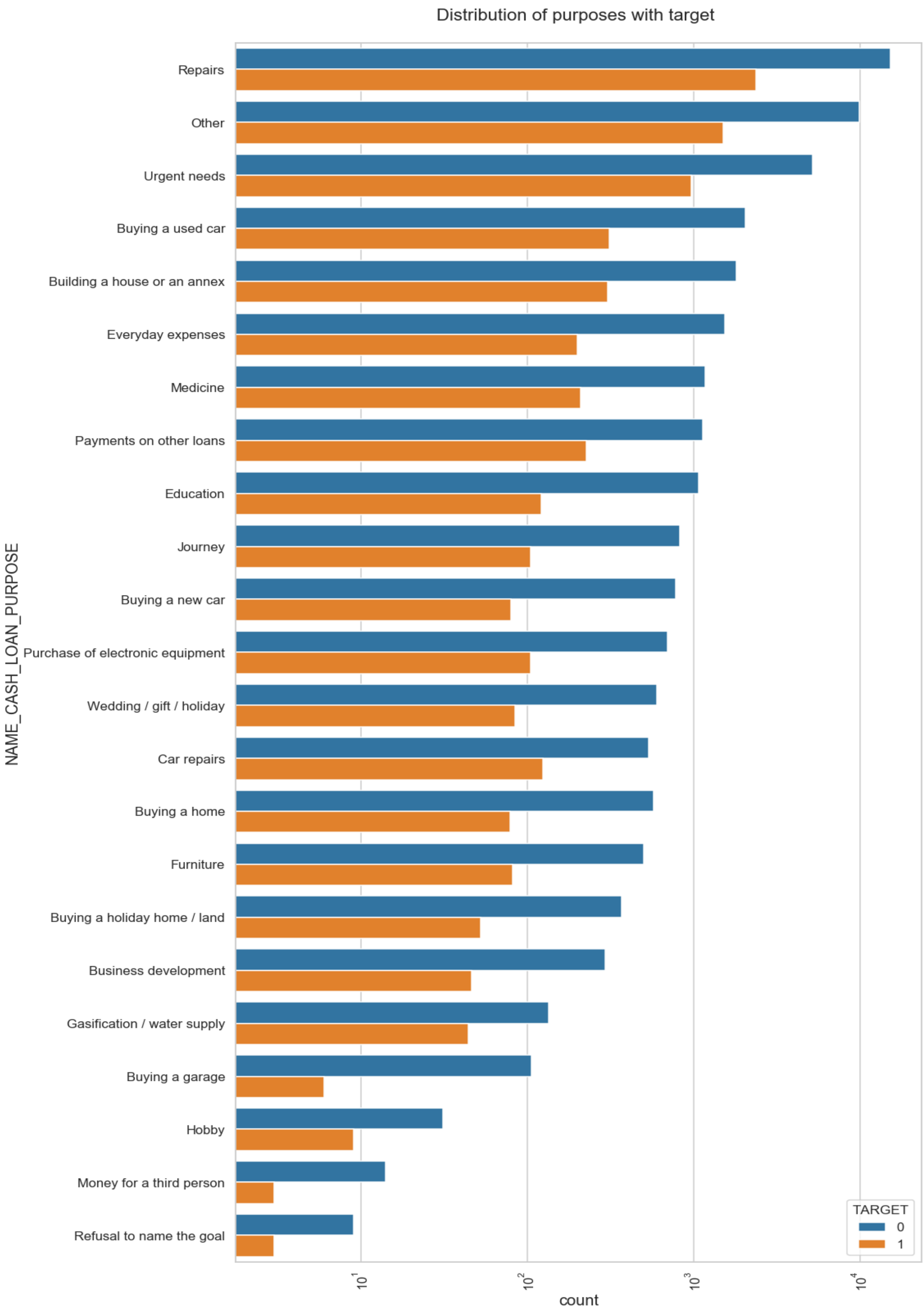
2. *For education purposes we have equal number of approves and rejection*

3. *Paying other loans and buying a new car is having significant higher rejection than approves.*

4. *Less approvals are for where person Refused to name the goal*


Distribution of contract status with purposes
Approved
Refused
Canceled
Unused offer
NAME_CASH_LOAN_PURPOSE
Repairs
Other
Urgent needs
Buying a used car
Building a house or an annex
Everyday expenses
Medicine
Payments on other loans
Education
Journey
Buying a new car
Purchase of electronic equipment
Wedding / gift / holiday
Car repairs
Buying a home
Furniture
Buying a holiday home / land
Business development
Gasification / water supply
Buying a garage
Hobby
Money for a third person
Refusal to name the goal
10^{1}
10^{2}
10^{3}
10^{4}
count

Distribution of purposes with target

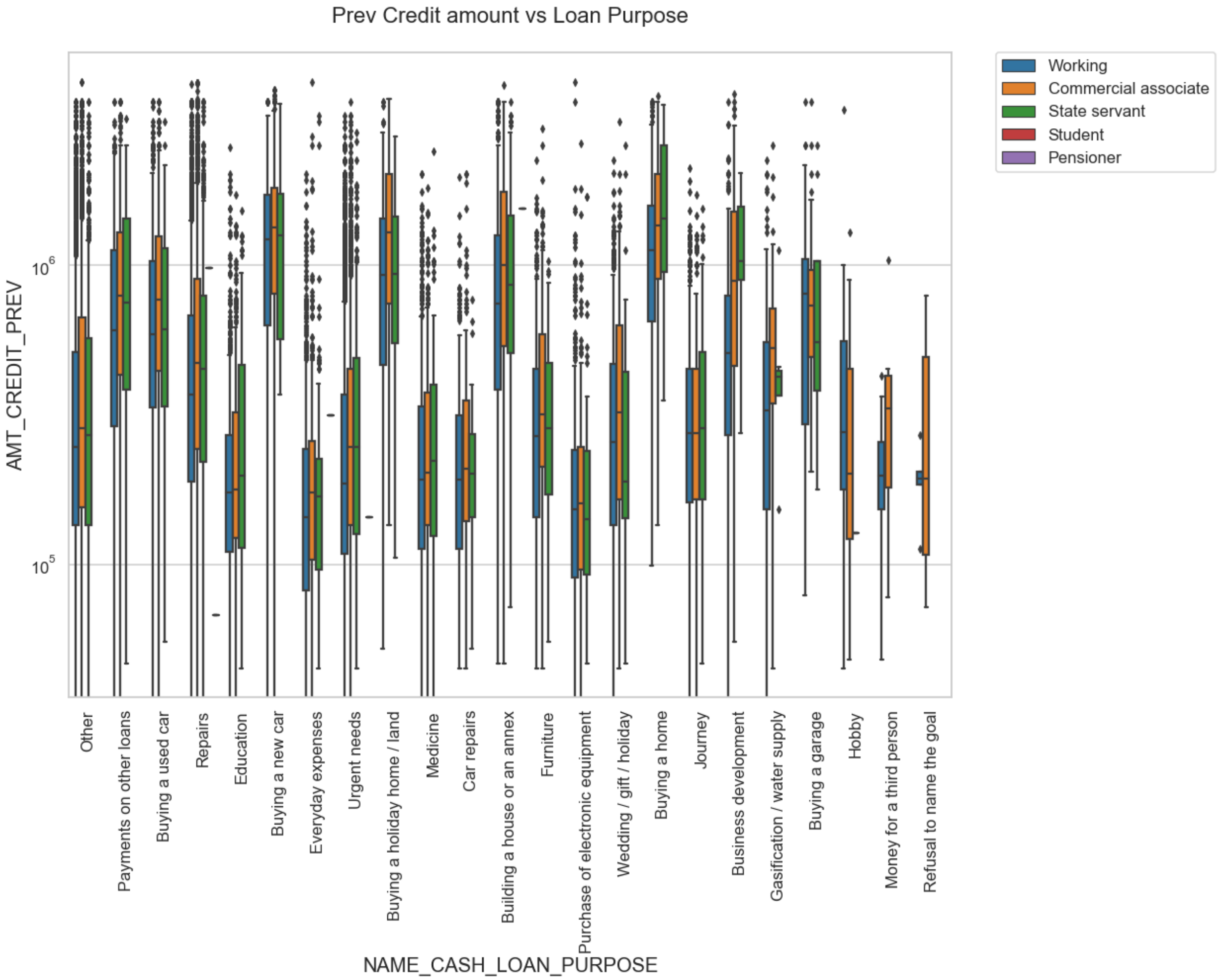


Conclusion

1. Loan purposes with 'Repairs' are facing more difficulites in payment on time.
2. There are few places where loan payment is significant higher than facing difficulties. They are 'Buying a garage', 'Business developemt', 'Buying land','Buying a new car' and 'Education' Hence we can focus on these purposes for which the client is having for minimal payment difficulties.

Performing bivariate analysis

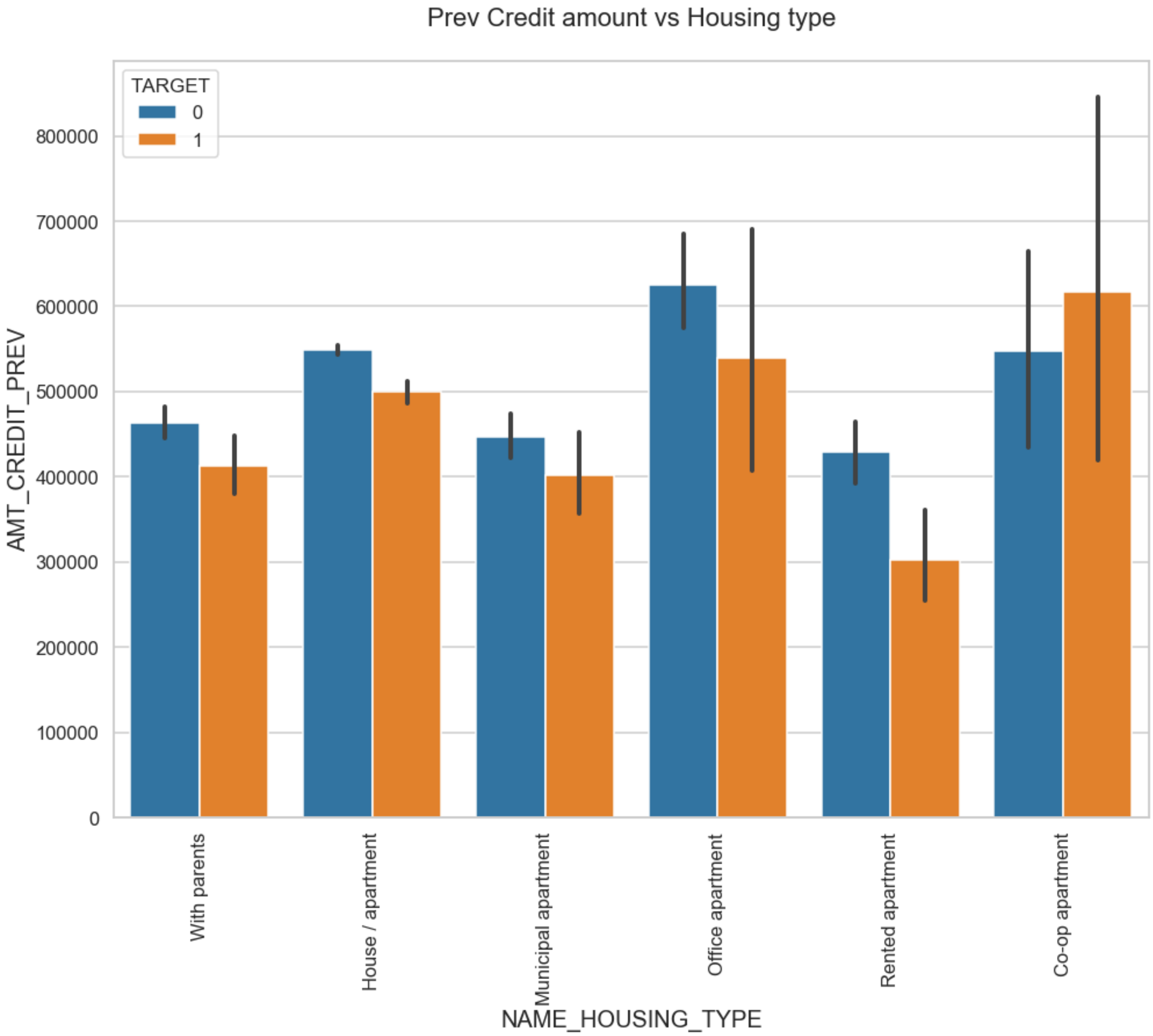
a) Previous Credit amount vs Loan Purpose :



Conclusion

1. The credit amount of Loan purposes like 'Buying a home','Buying a land','Buying a new car' and'Building a house' is higher.
2. Income type of state servants have a significant amount of credit applied
3. Money for third person or a Hobby is having less credits applied for.
4. Ignore XNA and XAP as they are representing null values.

b) Credit amount prev vs Housing type in logarithmic scale



For Housing type,Muncipal apartment and office appartment is having higher credit of target 0 and office apartment is having higher credit of target 1. So, we can conclude that bank should avoid giving loans to the housing type of office apartment as they are having difficulties in payment. Bank can focus mostly on housing type with parents or House\appartment or muuncipal appartment for successful payments.

CONCLUSION

1. Banks should focus more on contract type ‘Student’ ,’pensioner’ and ‘Businessman’ with housing ‘type other than ‘Co-op apartment’ for successful payments.

2. Banks should focus less on income type ‘Working’ as they are having most number of unsuccessful payments.

3. Also with loan purpose ‘Repair’ is having higher number of unsuccessful payments on time.

4. Get as much as clients from housing type ‘With parents’ as they are having least number of unsuccessful payments.¶