

Credit_Score_EDA_Report_Keerthana

November 5, 2022

```
[222]: #imports
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
# Filtering Warnings
import warnings
warnings.filterwarnings('ignore')
```

```
[223]: # Reading application history dataset

df=pd.read_csv("application_data.csv")
```

```
[224]: # shape of the dataset

df.shape
```

```
[224]: (307511, 122)
```

```
[225]: # removing the missing data

# printing the columns having more than 30% null values

emptycol=df.isnull().sum()
emptycol=emptycol[emptycol.values>(0.3*len(emptycol))]
len(emptycol)
```

```
[225]: 64
```

```
[226]: # Removing columns those are having null values
emptycol = list(emptycol[emptycol.values>=0.3].index)
df.drop(labels=emptycol,axis=1,inplace=True)
print(len(emptycol))
```

```
64
```

```
[227]: # validationg the less null percentage columns

df.isnull().sum()/len(df)*100
```

[227]: SK_ID_CURR	0.000000
TARGET	0.000000
NAME_CONTRACT_TYPE	0.000000
CODE_GENDER	0.000000
FLAG_OWN_CAR	0.000000
FLAG_OWN_REALTY	0.000000
CNT_CHILDREN	0.000000
AMT_INCOME_TOTAL	0.000000
AMT_CREDIT	0.000000
AMT_ANNUITY	0.003902
NAME_INCOME_TYPE	0.000000
NAME_EDUCATION_TYPE	0.000000
NAME_FAMILY_STATUS	0.000000
NAME_HOUSING_TYPE	0.000000
REGION_POPULATION_RELATIVE	0.000000
DAYS_BIRTH	0.000000
DAYS_EMPLOYED	0.000000
DAYS_REGISTRATION	0.000000
DAYS_ID_PUBLISH	0.000000
FLAG_MOBIL	0.000000
FLAG_EMP_PHONE	0.000000
FLAG_WORK_PHONE	0.000000
FLAG_CONT_MOBILE	0.000000
FLAG_PHONE	0.000000
FLAG_EMAIL	0.000000
CNT_FAM_MEMBERS	0.000650
REGION_RATING_CLIENT	0.000000
REGION_RATING_CLIENT_W_CITY	0.000000
WEEKDAY_APPR_PROCESS_START	0.000000
HOUR_APPR_PROCESS_START	0.000000
REG_REGION_NOT_LIVE_REGION	0.000000
REG_REGION_NOT_WORK_REGION	0.000000
LIVE_REGION_NOT_WORK_REGION	0.000000
REG_CITY_NOT_LIVE_CITY	0.000000
REG_CITY_NOT_WORK_CITY	0.000000
LIVE_CITY_NOT_WORK_CITY	0.000000
ORGANIZATION_TYPE	0.000000
DAYS_LAST_PHONE_CHANGE	0.000325
FLAG_DOCUMENT_2	0.000000
FLAG_DOCUMENT_3	0.000000
FLAG_DOCUMENT_4	0.000000
FLAG_DOCUMENT_5	0.000000
FLAG_DOCUMENT_6	0.000000
FLAG_DOCUMENT_7	0.000000
FLAG_DOCUMENT_8	0.000000
FLAG_DOCUMENT_9	0.000000
FLAG_DOCUMENT_10	0.000000

```

FLAG_DOCUMENT_11      0.000000
FLAG_DOCUMENT_12      0.000000
FLAG_DOCUMENT_13      0.000000
FLAG_DOCUMENT_14      0.000000
FLAG_DOCUMENT_15      0.000000
FLAG_DOCUMENT_16      0.000000
FLAG_DOCUMENT_17      0.000000
FLAG_DOCUMENT_18      0.000000
FLAG_DOCUMENT_19      0.000000
FLAG_DOCUMENT_20      0.000000
FLAG_DOCUMENT_21      0.000000
dtype: float64

```

```

[228]: # Inserting missing values with median

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

df.loc[df['AMT_ANNUITY'].isnull(), 'AMT_ANNUITY']=values# Inserting missing_
↳ values with median

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

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

```

```

[229]: # Checking the null values column

df.isnull().sum()

```

```

[229]: SK_ID_CURR      0
TARGET      0
NAME_CONTRACT_TYPE    0
CODE_GENDER      0
FLAG_OWN_CAR      0
FLAG_OWN_REALTY      0
CNT_CHILDREN      0
AMT_INCOME_TOTAL      0
AMT_CREDIT      0
AMT_ANNUITY      0
NAME_INCOME_TYPE      0
NAME_EDUCATION_TYPE    0
NAME_FAMILY_STATUS      0
NAME_HOUSING_TYPE      0
REGION_POPULATION_RELATIVE  0
DAYS_BIRTH      0
DAYS_EMPLOYED      0
DAYS_REGISTRATION      0
DAYS_ID_PUBLISH      0

```

```

FLAG_MOBIL          0
FLAG_EMP_PHONE      0
FLAG_WORK_PHONE     0
FLAG_CONT_MOBILE    0
FLAG_PHONE          0
FLAG_EMAIL          0
CNT_FAM_MEMBERS     2
REGION_RATING_CLIENT 0
REGION_RATING_CLIENT_W_CITY 0
WEEKDAY_APPR_PROCESS_START 0
HOUR_APPR_PROCESS_START 0
REG_REGION_NOT_LIVE_REGION 0
REG_REGION_NOT_WORK_REGION 0
LIVE_REGION_NOT_WORK_REGION 0
REG_CITY_NOT_LIVE_CITY 0
REG_CITY_NOT_WORK_CITY 0
LIVE_CITY_NOT_WORK_CITY 0
ORGANIZATION_TYPE   0
DAYS_LAST_PHONE_CHANGE 1
FLAG_DOCUMENT_2      0
FLAG_DOCUMENT_3      0
FLAG_DOCUMENT_4      0
FLAG_DOCUMENT_5      0
FLAG_DOCUMENT_6      0
FLAG_DOCUMENT_7      0
FLAG_DOCUMENT_8      0
FLAG_DOCUMENT_9      0
FLAG_DOCUMENT_10     0
FLAG_DOCUMENT_11     0
FLAG_DOCUMENT_12     0
FLAG_DOCUMENT_13     0
FLAG_DOCUMENT_14     0
FLAG_DOCUMENT_15     0
FLAG_DOCUMENT_16     0
FLAG_DOCUMENT_17     0
FLAG_DOCUMENT_18     0
FLAG_DOCUMENT_19     0
FLAG_DOCUMENT_20     0
FLAG_DOCUMENT_21     0
dtype: int64

```

```

[230]: # Cleaning rows those are having null values which are greater than or equal to
      ↪ 30%

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

```

```
print(len(emptyrow))
```

0

```
[231]: # remove unnecessary columns from this dataset
```

```
unwanted=['FLAG_MOBIL', 'FLAG_EMP_PHONE', 'FLAG_WORK_PHONE', 'FLAG_CONT_MOBILE',  
          'FLAG_PHONE',  
          ↪ 'FLAG_EMAIL', 'REGION_RATING_CLIENT', 'REGION_RATING_CLIENT_W_CITY', 'FLAG_EMAIL', 'CNT_FAM_MEM',  
          ↪ 'REGION_RATING_CLIENT',  
          'REGION_RATING_CLIENT_W_CITY', 'DAYS_LAST_PHONE_CHANGE',  
          ↪ 'FLAG_DOCUMENT_2', 'FLAG_DOCUMENT_3', 'FLAG_DOCUMENT_4', 'FLAG_DOCUMENT_5',  
          ↪ 'FLAG_DOCUMENT_6',  
          'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_8',  
          ↪ 'FLAG_DOCUMENT_9', 'FLAG_DOCUMENT_10', 'FLAG_DOCUMENT_11', 'FLAG_DOCUMENT_12',  
          'FLAG_DOCUMENT_13', 'FLAG_DOCUMENT_14',  
          ↪ 'FLAG_DOCUMENT_15', 'FLAG_DOCUMENT_16', 'FLAG_DOCUMENT_17',  
          ↪ 'FLAG_DOCUMENT_18',  
          'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'FLAG_DOCUMENT_21']  
  
df.drop(labels=unwanted,axis=1,inplace=True)
```

```
[232]: # Finding categorical columns having these 'XNA' values
```

```
# For Gender column
```

```
df[df['CODE_GENDER']=='XNA'].shape
```

```
[232]: (4, 28)
```

```
[233]: # Organization column
```

```
df[df['ORGANIZATION_TYPE']=='XNA'].shape
```

```
[233]: (55374, 28)
```

```
[234]: # Gender column to check the number of females and males
```

```
df['CODE_GENDER'].value_counts()
```

```
[234]: F      202448  
      M      105059  
      XNA         4  
      Name: CODE_GENDER, dtype: int64
```

```
[235]: # Updating the column 'CODE_GENDER' 'XNA' to "F" for the dataset
```

```
df.loc[df['CODE_GENDER']=='XNA','CODE_GENDER']='F'
```

```
df['CODE_GENDER'].value_counts()
```

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

```
[236]: # Describing the organization  
  
df['ORGANIZATION_TYPE'].describe()
```

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

```
[237]: # dropping the rows of total 55374 have 'XNA' values in the organization type  
  
df=df.drop(df.loc[df['ORGANIZATION_TYPE']=='XNA'].index)  
df[df['ORGANIZATION_TYPE']=='XNA'].shape
```

```
[237]: (0, 28)
```

```
[238]: # Casting all variable into numeric in the dataset  
  
numeric_columns=['TARGET', 'CNT_CHILDREN', 'AMT_INCOME_TOTAL', 'AMT_CREDIT', 'AMT_ANNUITY', 'REGION_RATING_CODE',  
                 'DAYS_EMPLOYED', 'DAYS_REGISTRATION', 'DAYS_ID_PUBLISH', 'HOUR_APPR_PROCESS_START', 'LIVE_REGION_CODE',  
                 'REG_CITY_NOT_LIVE_CITY',  
                 'REG_CITY_NOT_WORK_CITY', 'LIVE_CITY_NOT_WORK_CITY']  
  
df[numeric_columns]=df[numeric_columns].apply(pd.to_numeric)  
df.head(5)
```

```
[238]: SK_ID_CURR  TARGET  NAME_CONTRACT_TYPE  CODE_GENDER  FLAG_OWN_CAR  \  
0      100002      1      Cash loans          M          N  
1      100003      0      Cash loans          F          N  
2      100004      0      Revolving loans       M          Y  
3      100006      0      Cash loans          F          N  
4      100007      0      Cash loans          M          N  
  
      FLAG_OWN_REALTY  CNT_CHILDREN  AMT_INCOME_TOTAL  AMT_CREDIT  AMT_ANNUITY  \  
0          Y          0      202500.0      406597.5      24700.5  
1          N          0      270000.0      1293502.5      35698.5  
2          Y          0       67500.0      135000.0       6750.0  
3          Y          0      135000.0      312682.5      29686.5  
4          Y          0      121500.0      513000.0      21865.5
```

	... DAYS_ID_PUBLISH	WEEKDAY_APPR_PROCESS_START	HOUR_APPR_PROCESS_START	\
0	...	-2120	WEDNESDAY	10
1	...	-291	MONDAY	11
2	...	-2531	MONDAY	9
3	...	-2437	WEDNESDAY	17
4	...	-3458	THURSDAY	11

	REG_REGION_NOT_LIVE_REGION	REG_REGION_NOT_WORK_REGION	\
0	0	0	
1	0	0	
2	0	0	
3	0	0	
4	0	0	

	LIVE_REGION_NOT_WORK_REGION	REG_CITY_NOT_LIVE_CITY	\
0	0	0	
1	0	0	
2	0	0	
3	0	0	
4	0	0	

	REG_CITY_NOT_WORK_CITY	LIVE_CITY_NOT_WORK_CITY	ORGANIZATION_TYPE
0	0	0	Business Entity Type 3
1	0	0	School
2	0	0	Government
3	0	0	Business Entity Type 3
4	1	1	Religion

[5 rows x 28 columns]

[239]: # bins for income amount

```
bins = [0,25000,50000,75000,100000,125000,150000,175000,200000,225000,250000,275000,300000,325000,350000]
slot = ['0-25000', '25000-50000', '50000-75000', '75000-100000', '100000-125000', '125000-150000', '150000-175000', '175000-200000', '200000-225000', '225000-250000', '250000-275000', '275000-300000', '300000-325000', '325000-350000', '375000-400000', '400000-425000', '425000-450000', '450000-475000', '475000-500000', '500000-and above']

df['AMT_INCOME_RANGE']=pd.cut(df['AMT_INCOME_TOTAL'],bins,labels=slot)
```

[240]: # bins for Credit amount

```

bins =
    ↪ [0,150000,200000,250000,300000,350000,400000,450000,500000,550000,600000,650000,700000,750000]
slots = ['0-150000', '150000-200000', '200000-250000', '250000-300000',
    ↪ '300000-350000', '350000-400000', '400000-450000',
    ↪ '450000-500000', '500000-550000', '550000-600000', '600000-650000', '650000-700000', '700000-750000',
    ↪ '750000-800000', '800000-850000', '850000-900000', '900000 and above']

df['AMT_CREDIT_RANGE']=pd.cut(df['AMT_CREDIT'],bins=bins,labels=slots)

```

```

[241]: # Dividing the dataset into two dataset of target=1(client with payment
    ↪ difficulties) and target=0(all other)

target0_df=df.loc[df["TARGET"]==0]
target1_df=df.loc[df["TARGET"]==1]

```

```

[242]: # Calculating Imbalance percentage

# Since the majority is target0 and minority is target1

round(len(target0_df)/len(target1_df),2)

```

[242]: 10.55

```

[243]: # Count plotting in logarithmic scale

def uniplot(df,col,title,hue =None):

    sns.set_style('darkgrid')
    sns.set_context('talk')
    plt.rcParams["axes.labelsize"] = 20
    plt.rcParams['axes.titlesize'] = 22
    plt.rcParams['axes.titlepad'] = 30

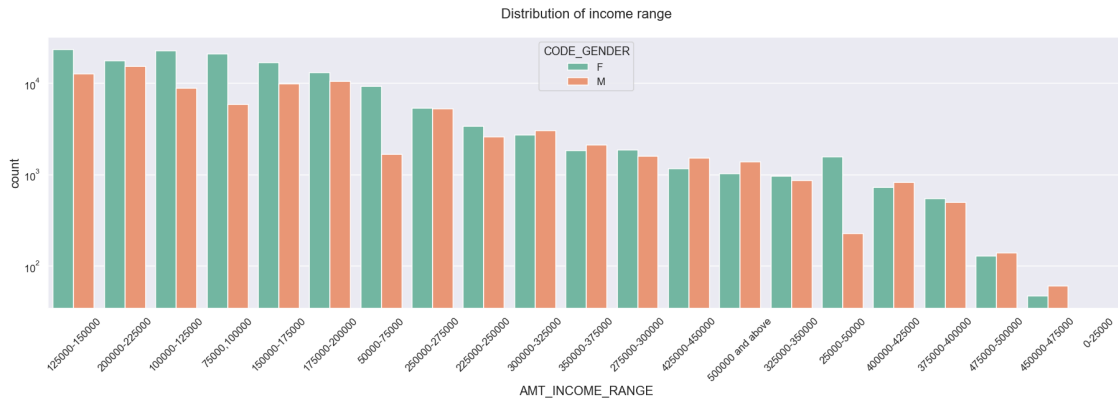
    temp = pd.Series(data = hue)
    fig, ax = plt.subplots()
    width = len(df[col].unique()) + 7 + 4*len(temp.unique())
    fig.set_size_inches(width , 8)
    plt.xticks(rotation=45)
    plt.yscale('log')
    plt.title(title)
    ax = sns.countplot(data = df, x= col, order=df[col].value_counts().
    ↪ index,hue = hue,palette='Set2')

    plt.show()

```

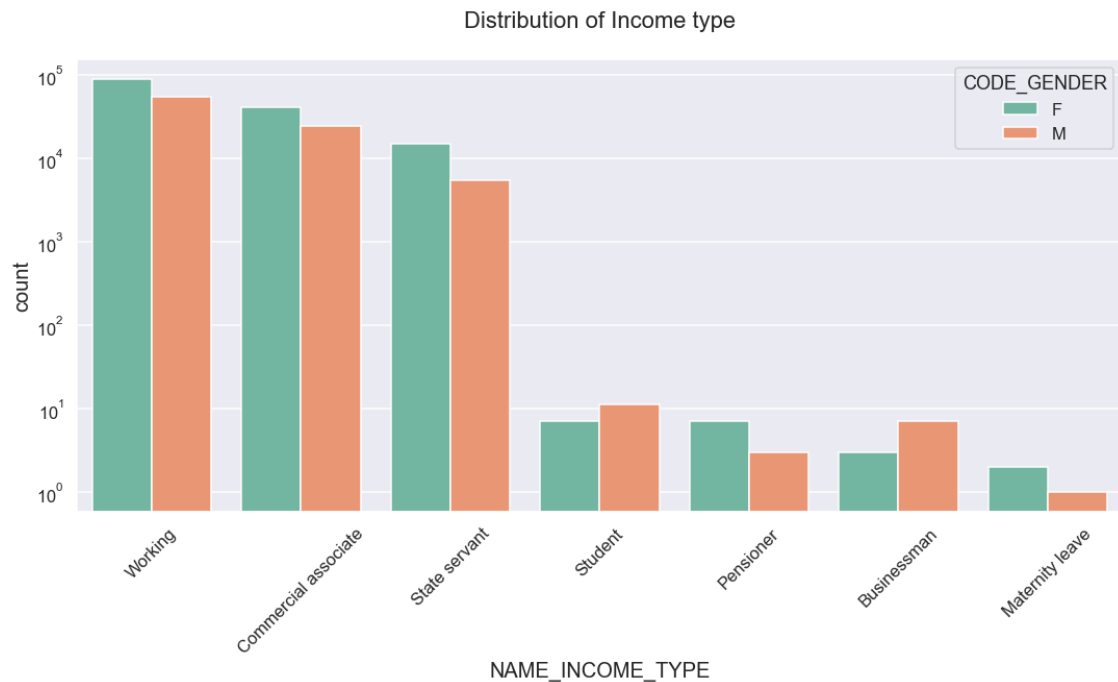

[244]: # Plotting for income range

```
uniplot(target0_df,col='AMT_INCOME_RANGE',title='Distribution of income_
↪range',hue='CODE_GENDER')
```



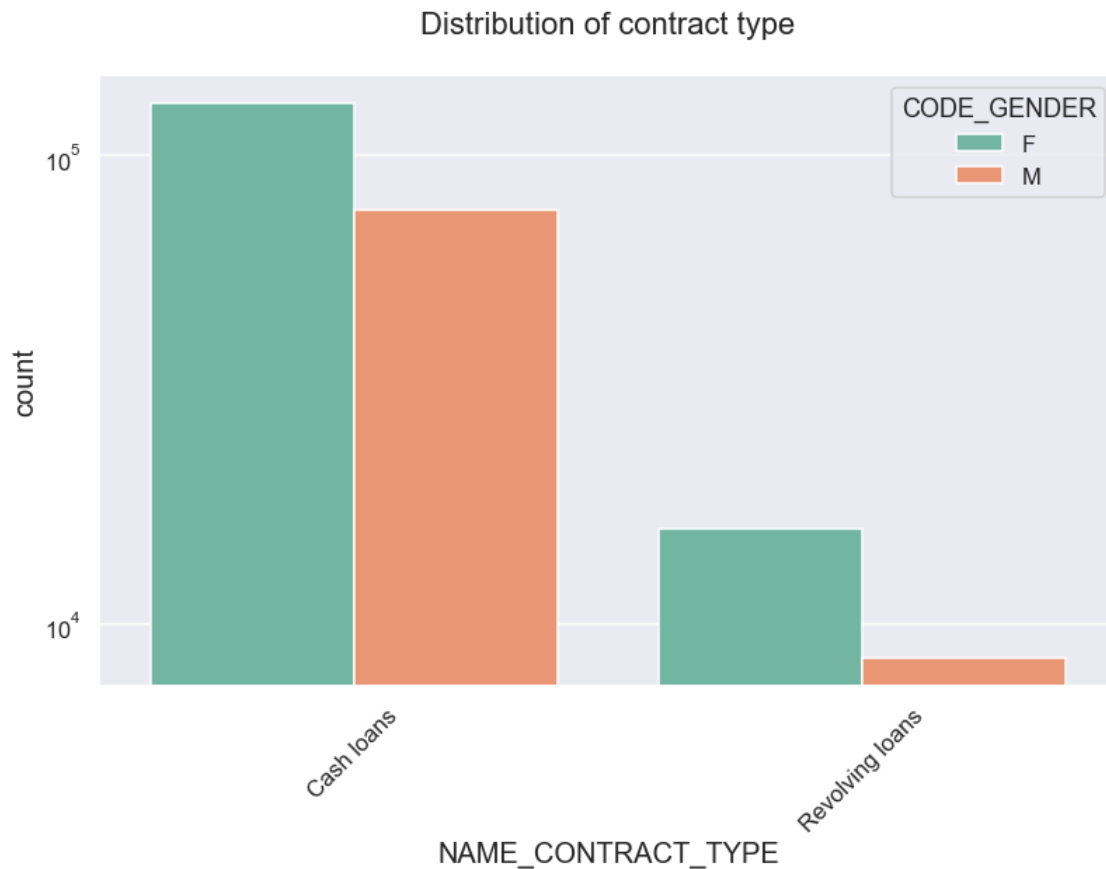
[245]: # Plotting for Income type

```
uniplot(target0_df,col='NAME_INCOME_TYPE',title='Distribution of Income_
↪type',hue='CODE_GENDER')
```



```
[246]: # Plotting for Contract type

uniplot(target0_df,col='NAME_CONTRACT_TYPE',title='Distribution of contract_
↳type',hue='CODE_GENDER')
```



```
[247]: # Plotting for Organization type in logarithmic scale

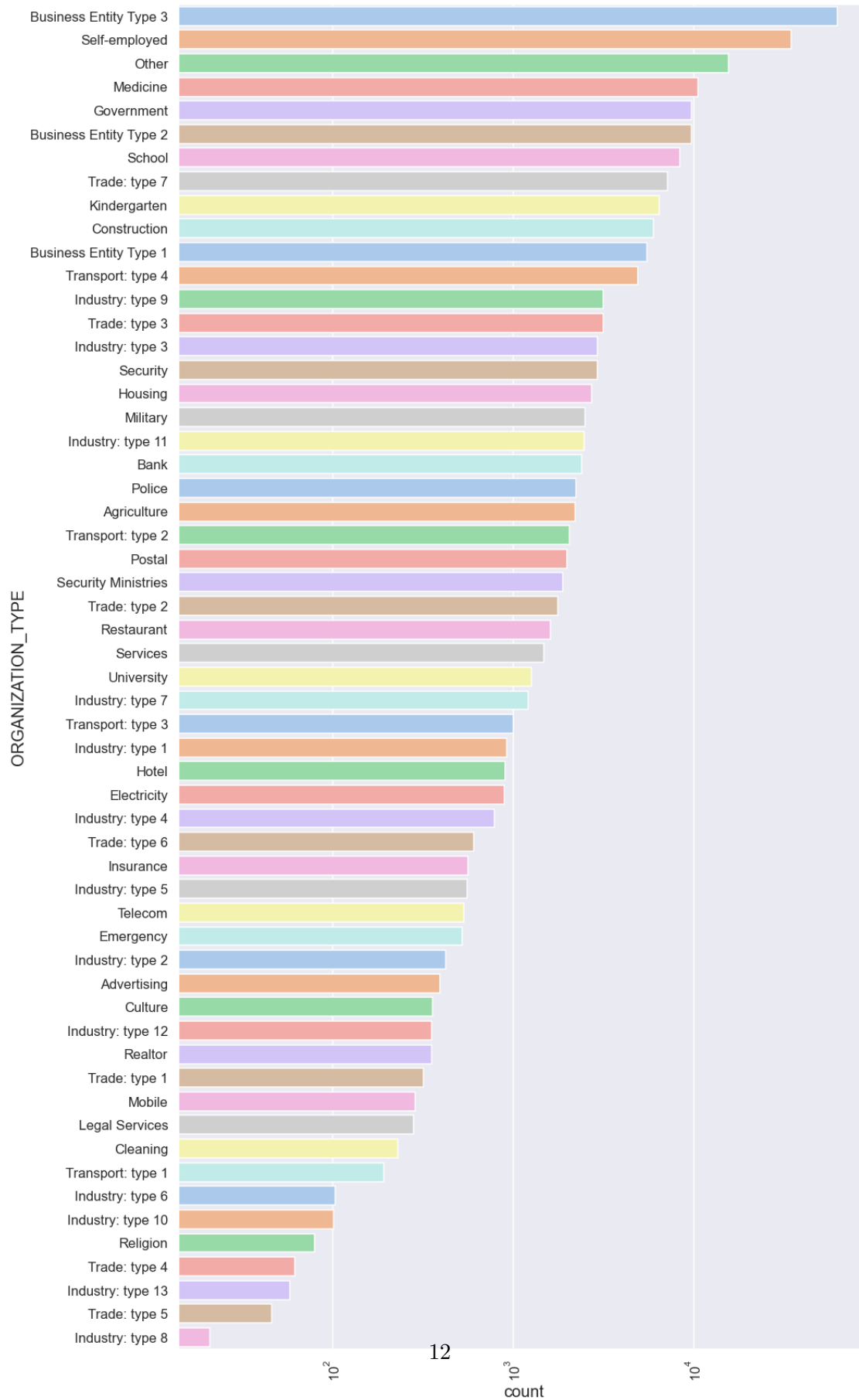
sns.set_style('darkgrid')
sns.set_context('talk')
plt.figure(figsize=(15,30))
plt.rcParams["axes.labelsize"] = 20
plt.rcParams['axes.titlesize'] = 22
plt.rcParams['axes.titlepad'] = 30

plt.title("Distribution of Organization type for target - 0")

plt.xticks(rotation=90)
plt.xscale('log')
```

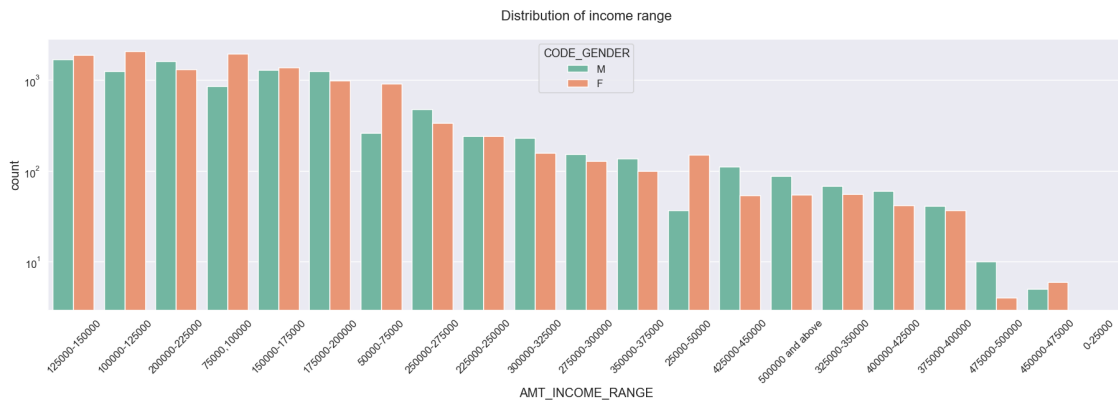
```
sns.  
    ↳countplot(data=target0_df,y='ORGANIZATION_TYPE',order=target0_df['ORGANIZATION_TYPE'].  
    ↳value_counts().index,palette='pastel')  
  
plt.show()
```

Distribution of Organization type for target - 0



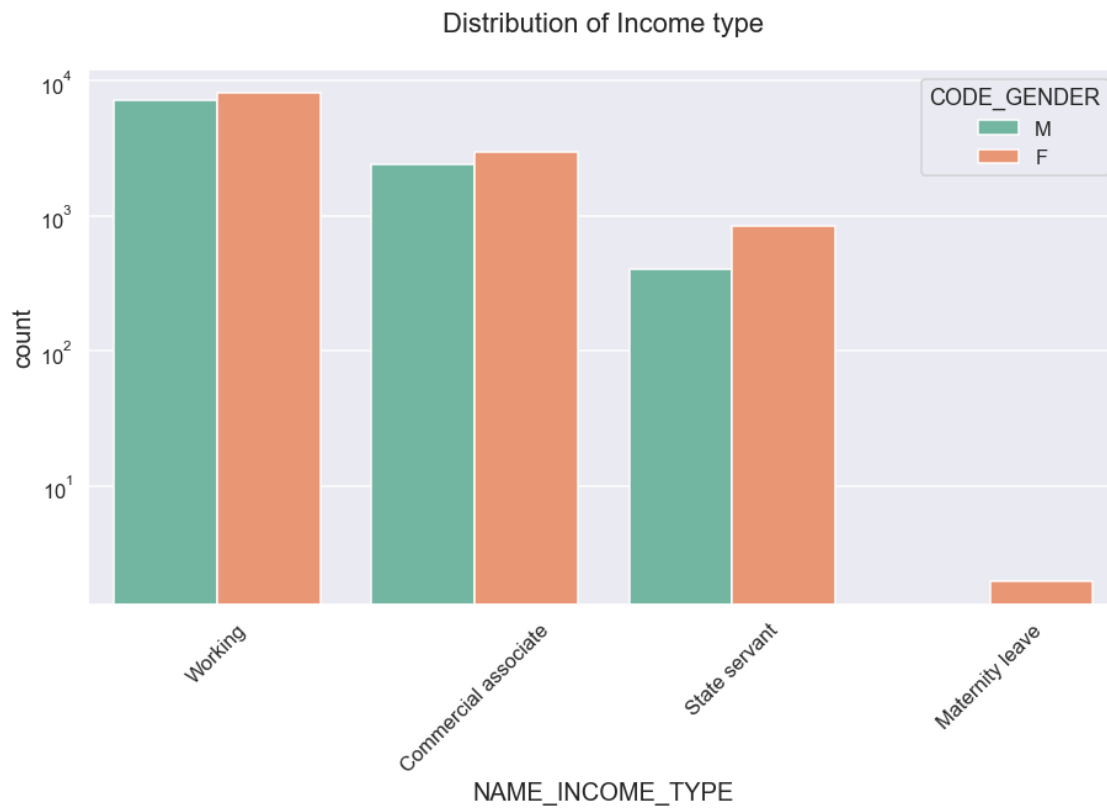
```
[248]: # Plotting for income range
```

```
unipLOT(target1_df,col='AMT_INCOME_RANGE',title='Distribution of income_  
↪range',hue='CODE_GENDER')
```



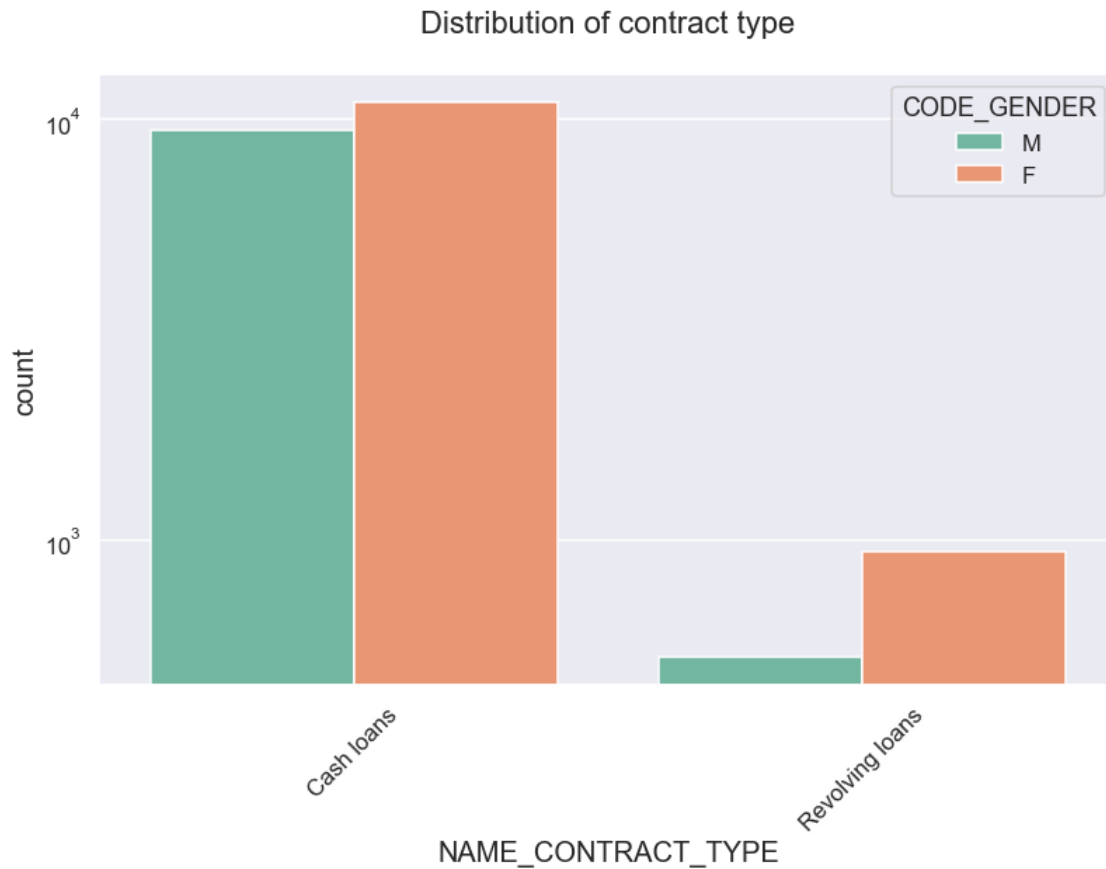
```
[249]: # Plotting for Income type
```

```
unipLOT(target1_df,col='NAME_INCOME_TYPE',title='Distribution of Income_  
↪type',hue='CODE_GENDER')
```



```
[250]: # Plotting for Contract type

uniplot(target1_df,col='NAME_CONTRACT_TYPE',title='Distribution of contract_
↳type',hue='CODE_GENDER')
```



```
[251]: # Plotting for Organization type

sns.set_style('darkgrid')
sns.set_context('talk')
plt.figure(figsize=(15,30))
plt.rcParams["axes.labelsize"] = 20
plt.rcParams['axes.titlesize'] = 22
plt.rcParams['axes.titlepad'] = 30

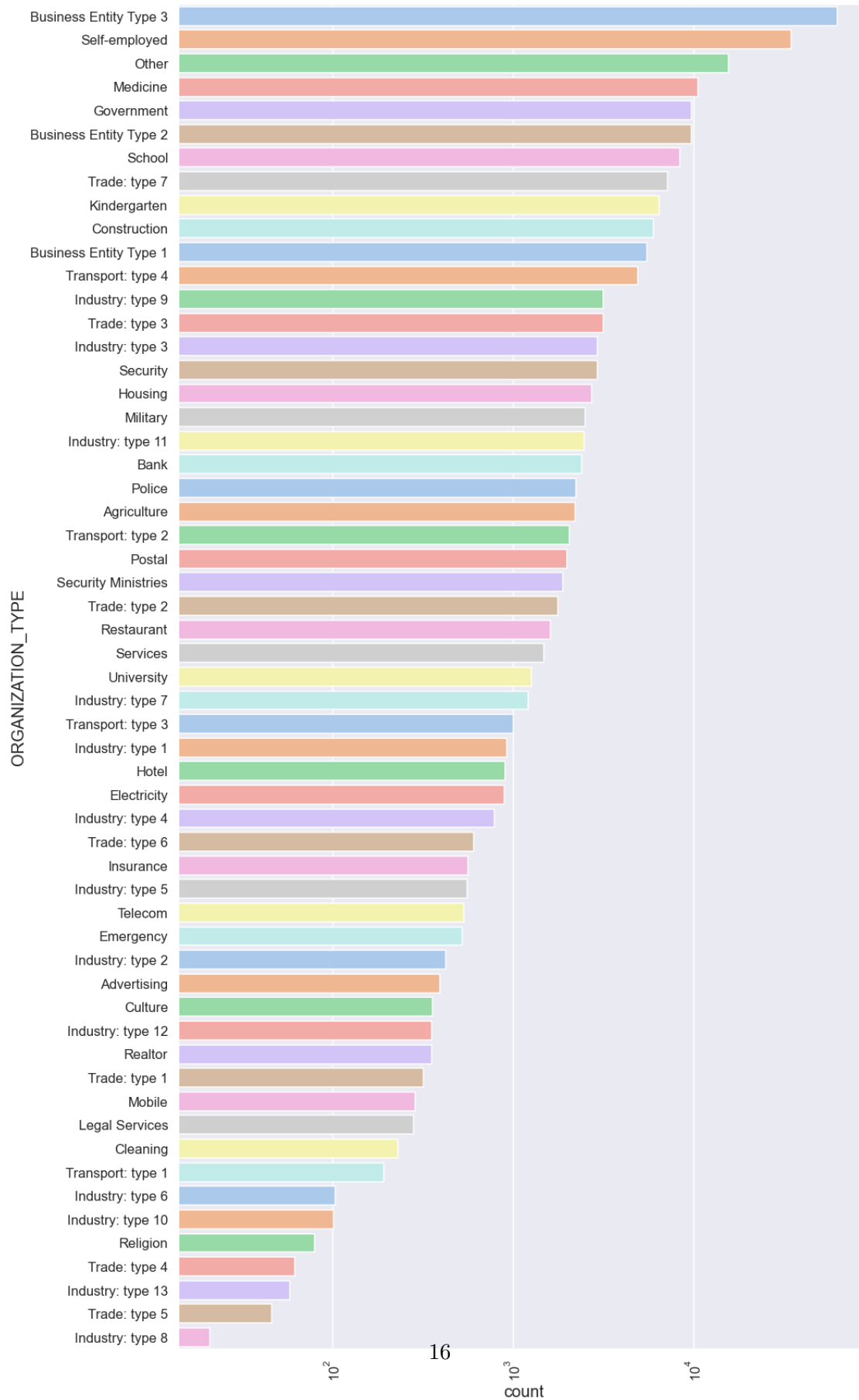
plt.title("Distribution of Organization type for target - 1")

plt.xticks(rotation=90)
plt.xscale('log')

sns.
    ↪ countplot(data=target0_df,y='ORGANIZATION_TYPE',order=target0_df['ORGANIZATION_TYPE'].
    ↪ value_counts().index,palette='pastel')

plt.show()
```

Distribution of Organization type for target - 1




```
[252]: # Finding some correlation for numerical columns for both target 0 and 1
```

```
target0_corr=target0_df.iloc[0:,2:]
target1_corr=target1_df.iloc[0:,2:]

target0=target0_corr.corr(method='spearman')
target1=target1_corr.corr(method='spearman')
```

```
[253]: # Correlation for target 0
```

```
target0
```

```
[253]:
```

	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	\
CNT_CHILDREN	1.000000	-0.021950	-0.023652	
AMT_INCOME_TOTAL	-0.021950	1.000000	0.403876	
AMT_CREDIT	-0.023652	0.403876	1.000000	
AMT_ANNUITY	-0.010795	0.472204	0.826689	
REGION_POPULATION_RELATIVE	-0.030579	0.110074	0.060706	
DAYS_BIRTH	0.266534	-0.054666	-0.169030	
DAYS_EMPLOYED	0.030948	-0.060868	-0.104251	
DAYS_REGISTRATION	0.155518	0.040559	-0.015318	
DAYS_ID_PUBLISH	-0.119164	-0.036702	-0.038197	
HOURL_APPR_PROCESS_START	-0.030162	0.073503	0.036923	
REG_REGION_NOT_LIVE_REGION	-0.022813	0.077634	0.015118	
REG_REGION_NOT_WORK_REGION	-0.015475	0.159962	0.041693	
LIVE_REGION_NOT_WORK_REGION	-0.005576	0.148281	0.045175	
REG_CITY_NOT_LIVE_CITY	0.002344	-0.001023	-0.040616	
REG_CITY_NOT_WORK_CITY	0.007487	-0.013856	-0.037000	
LIVE_CITY_NOT_WORK_CITY	0.013295	-0.004758	-0.011194	

	AMT_ANNUITY	REGION_POPULATION_RELATIVE	\
CNT_CHILDREN	-0.010795	-0.030579	
AMT_INCOME_TOTAL	0.472204	0.110074	
AMT_CREDIT	0.826689	0.060706	
AMT_ANNUITY	1.000000	0.064328	
REGION_POPULATION_RELATIVE	0.064328	1.000000	
DAYS_BIRTH	-0.100287	-0.041663	
DAYS_EMPLOYED	-0.074643	0.000900	
DAYS_REGISTRATION	0.010712	-0.042400	
DAYS_ID_PUBLISH	-0.027354	-0.010299	
HOURL_APPR_PROCESS_START	0.032953	0.133213	
REG_REGION_NOT_LIVE_REGION	0.033435	-0.025292	
REG_REGION_NOT_WORK_REGION	0.070841	0.032446	
LIVE_REGION_NOT_WORK_REGION	0.069051	0.056814	

REG_CITY_NOT_LIVE_CITY	-0.019954	-0.049779
REG_CITY_NOT_WORK_CITY	-0.024085	-0.034808
LIVE_CITY_NOT_WORK_CITY	-0.008087	-0.007332

	DAYS_BIRTH	DAYS_EMPLOYED	DAYS_REGISTRATION \
CNT_CHILDREN	0.266534	0.030948	0.155518
AMT_INCOME_TOTAL	-0.054666	-0.060868	0.040559
AMT_CREDIT	-0.169030	-0.104251	-0.015318
AMT_ANNUITY	-0.100287	-0.074643	0.010712
REGION_POPULATION_RELATIVE	-0.041663	0.000900	-0.042400
DAYS_BIRTH	1.000000	0.307787	0.265449
DAYS_EMPLOYED	0.307787	1.000000	0.126708
DAYS_REGISTRATION	0.265449	0.126708	1.000000
DAYS_ID_PUBLISH	0.083331	0.106823	0.036788
HOURL_APPR_PROCESS_START	0.051299	0.026444	-0.029553
REG_REGION_NOT_LIVE_REGION	0.058627	0.065435	0.017715
REG_REGION_NOT_WORK_REGION	0.038104	0.086966	0.015092
LIVE_REGION_NOT_WORK_REGION	0.012789	0.063533	0.007716
REG_CITY_NOT_LIVE_CITY	0.167477	0.118224	0.038064
REG_CITY_NOT_WORK_CITY	0.111539	0.125954	0.047339
LIVE_CITY_NOT_WORK_CITY	0.029007	0.069567	0.027231

	DAYS_ID_PUBLISH	HOURL_APPR_PROCESS_START \
CNT_CHILDREN	-0.119164	-0.030162
AMT_INCOME_TOTAL	-0.036702	0.073503
AMT_CREDIT	-0.038197	0.036923
AMT_ANNUITY	-0.027354	0.032953
REGION_POPULATION_RELATIVE	-0.010299	0.133213
DAYS_BIRTH	0.083331	0.051299
DAYS_EMPLOYED	0.106823	0.026444
DAYS_REGISTRATION	0.036788	-0.029553
DAYS_ID_PUBLISH	1.000000	0.008538
HOURL_APPR_PROCESS_START	0.008538	1.000000
REG_REGION_NOT_LIVE_REGION	0.027302	0.051744
REG_REGION_NOT_WORK_REGION	0.020823	0.067352
LIVE_REGION_NOT_WORK_REGION	0.008525	0.053813
REG_CITY_NOT_LIVE_CITY	0.054875	0.011287
REG_CITY_NOT_WORK_CITY	0.033427	-0.005971
LIVE_CITY_NOT_WORK_CITY	0.001476	-0.010720

	REG_REGION_NOT_LIVE_REGION \
CNT_CHILDREN	-0.022813
AMT_INCOME_TOTAL	0.077634
AMT_CREDIT	0.015118
AMT_ANNUITY	0.033435
REGION_POPULATION_RELATIVE	-0.025292
DAYS_BIRTH	0.058627

DAYS_EMPLOYED	0.065435
DAYS_REGISTRATION	0.017715
DAYS_ID_PUBLISH	0.027302
HOURL_APPR_PROCESS_START	0.051744
REG_REGION_NOT_LIVE_REGION	1.000000
REG_REGION_NOT_WORK_REGION	0.461596
LIVE_REGION_NOT_WORK_REGION	0.090193
REG_CITY_NOT_LIVE_CITY	0.342321
REG_CITY_NOT_WORK_CITY	0.142429
LIVE_CITY_NOT_WORK_CITY	0.003479

REG_REGION_NOT_WORK_REGION \

CNT_CHILDREN	-0.015475
AMT_INCOME_TOTAL	0.159962
AMT_CREDIT	0.041693
AMT_ANNUITY	0.070841
REGION_POPULATION_RELATIVE	0.032446
DAYS_BIRTH	0.038104
DAYS_EMPLOYED	0.086966
DAYS_REGISTRATION	0.015092
DAYS_ID_PUBLISH	0.020823
HOURL_APPR_PROCESS_START	0.067352
REG_REGION_NOT_LIVE_REGION	0.461596
REG_REGION_NOT_WORK_REGION	1.000000
LIVE_REGION_NOT_WORK_REGION	0.860421
REG_CITY_NOT_LIVE_CITY	0.148476
REG_CITY_NOT_WORK_CITY	0.220372
LIVE_CITY_NOT_WORK_CITY	0.178472

LIVE_REGION_NOT_WORK_REGION \

CNT_CHILDREN	-0.005576
AMT_INCOME_TOTAL	0.148281
AMT_CREDIT	0.045175
AMT_ANNUITY	0.069051
REGION_POPULATION_RELATIVE	0.056814
DAYS_BIRTH	0.012789
DAYS_EMPLOYED	0.063533
DAYS_REGISTRATION	0.007716
DAYS_ID_PUBLISH	0.008525
HOURL_APPR_PROCESS_START	0.053813
REG_REGION_NOT_LIVE_REGION	0.090193
REG_REGION_NOT_WORK_REGION	0.860421
LIVE_REGION_NOT_WORK_REGION	1.000000
REG_CITY_NOT_LIVE_CITY	0.015010
REG_CITY_NOT_WORK_CITY	0.167753
LIVE_CITY_NOT_WORK_CITY	0.220865

	REG_CITY_NOT_LIVE_CITY	REG_CITY_NOT_WORK_CITY \
CNT_CHILDREN	0.002344	0.007487
AMT_INCOME_TOTAL	-0.001023	-0.013856
AMT_CREDIT	-0.040616	-0.037000
AMT_ANNUITY	-0.019954	-0.024085
REGION_POPULATION_RELATIVE	-0.049779	-0.034808
DAYS_BIRTH	0.167477	0.111539
DAYS_EMPLOYED	0.118224	0.125954
DAYS_REGISTRATION	0.038064	0.047339
DAYS_ID_PUBLISH	0.054875	0.033427
HOURL_APPR_PROCESS_START	0.011287	-0.005971
REG_REGION_NOT_LIVE_REGION	0.342321	0.142429
REG_REGION_NOT_WORK_REGION	0.148476	0.220372
LIVE_REGION_NOT_WORK_REGION	0.015010	0.167753
REG_CITY_NOT_LIVE_CITY	1.000000	0.442640
REG_CITY_NOT_WORK_CITY	0.442640	1.000000
LIVE_CITY_NOT_WORK_CITY	0.011782	0.820828

	LIVE_CITY_NOT_WORK_CITY
CNT_CHILDREN	0.013295
AMT_INCOME_TOTAL	-0.004758
AMT_CREDIT	-0.011194
AMT_ANNUITY	-0.008087
REGION_POPULATION_RELATIVE	-0.007332
DAYS_BIRTH	0.029007
DAYS_EMPLOYED	0.069567
DAYS_REGISTRATION	0.027231
DAYS_ID_PUBLISH	0.001476
HOURL_APPR_PROCESS_START	-0.010720
REG_REGION_NOT_LIVE_REGION	0.003479
REG_REGION_NOT_WORK_REGION	0.178472
LIVE_REGION_NOT_WORK_REGION	0.220865
REG_CITY_NOT_LIVE_CITY	0.011782
REG_CITY_NOT_WORK_CITY	0.820828
LIVE_CITY_NOT_WORK_CITY	1.000000

[254]: # Correlation for target 1

target1

	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT \
CNT_CHILDREN	1.000000	-0.039123	0.000427
AMT_INCOME_TOTAL	-0.039123	1.000000	0.364559
AMT_CREDIT	0.000427	0.364559	1.000000
AMT_ANNUITY	0.015133	0.428947	0.812093
REGION_POPULATION_RELATIVE	-0.029682	0.058005	0.043545
DAYS_BIRTH	0.175025	-0.103026	-0.200718

DAYS_EMPLOYED	0.006823	-0.053798	-0.107605
DAYS_REGISTRATION	0.110854	0.011378	-0.021973
DAYS_ID_PUBLISH	-0.091042	-0.051113	-0.065143
HOURL_APPR_PROCESS_START	-0.040338	0.078779	0.024616
REG_REGION_NOT_LIVE_REGION	-0.035213	0.075615	0.015043
REG_REGION_NOT_WORK_REGION	-0.040853	0.156374	0.032536
LIVE_REGION_NOT_WORK_REGION	-0.027993	0.145982	0.034861
REG_CITY_NOT_LIVE_CITY	-0.016072	-0.003813	-0.030974
REG_CITY_NOT_WORK_CITY	-0.005444	-0.006241	-0.032882
LIVE_CITY_NOT_WORK_CITY	0.009557	0.004230	-0.012465

	AMT_ANNUITY	REGION_POPULATION_RELATIVE	\
CNT_CHILDREN	0.015133	-0.029682	
AMT_INCOME_TOTAL	0.428947	0.058005	
AMT_CREDIT	0.812093	0.043545	
AMT_ANNUITY	1.000000	0.028666	
REGION_POPULATION_RELATIVE	0.028666	1.000000	
DAYS_BIRTH	-0.100200	-0.044444	
DAYS_EMPLOYED	-0.060193	-0.015246	
DAYS_REGISTRATION	0.019762	-0.033490	
DAYS_ID_PUBLISH	-0.044128	-0.017779	
HOURL_APPR_PROCESS_START	0.021129	0.109400	
REG_REGION_NOT_LIVE_REGION	0.029646	-0.032702	
REG_REGION_NOT_WORK_REGION	0.060363	-0.008160	
LIVE_REGION_NOT_WORK_REGION	0.059724	0.012602	
REG_CITY_NOT_LIVE_CITY	-0.011744	-0.057239	
REG_CITY_NOT_WORK_CITY	-0.015938	-0.044761	
LIVE_CITY_NOT_WORK_CITY	-0.003012	-0.014753	

	DAYS_BIRTH	DAYS_EMPLOYED	DAYS_REGISTRATION	\
CNT_CHILDREN	0.175025	0.006823	0.110854	
AMT_INCOME_TOTAL	-0.103026	-0.053798	0.011378	
AMT_CREDIT	-0.200718	-0.107605	-0.021973	
AMT_ANNUITY	-0.100200	-0.060193	0.019762	
REGION_POPULATION_RELATIVE	-0.044444	-0.015246	-0.033490	
DAYS_BIRTH	1.000000	0.256870	0.192350	
DAYS_EMPLOYED	0.256870	1.000000	0.086286	
DAYS_REGISTRATION	0.192350	0.086286	1.000000	
DAYS_ID_PUBLISH	0.146246	0.104244	0.061563	
HOURL_APPR_PROCESS_START	0.041994	0.010328	-0.044753	
REG_REGION_NOT_LIVE_REGION	0.046320	0.069566	0.006362	
REG_REGION_NOT_WORK_REGION	0.022208	0.082264	0.000896	
LIVE_REGION_NOT_WORK_REGION	0.000356	0.056081	-0.001416	
REG_CITY_NOT_LIVE_CITY	0.145884	0.118869	0.015831	
REG_CITY_NOT_WORK_CITY	0.096181	0.139863	0.039204	
LIVE_CITY_NOT_WORK_CITY	0.009633	0.069316	0.026105	

	DAYS_ID_PUBLISH	hour_APPR_PROCESS_START \
CNT_CHILDREN	-0.091042	-0.040338
AMT_INCOME_TOTAL	-0.051113	0.078779
AMT_CREDIT	-0.065143	0.024616
AMT_ANNUITY	-0.044128	0.021129
REGION_POPULATION_RELATIVE	-0.017779	0.109400
DAYS_BIRTH	0.146246	0.041994
DAYS_EMPLOYED	0.104244	0.010328
DAYS_REGISTRATION	0.061563	-0.044753
DAYS_ID_PUBLISH	1.000000	0.012709
hour_APPR_PROCESS_START	0.012709	1.000000
REG_REGION_NOT_LIVE_REGION	0.024860	0.050953
REG_REGION_NOT_WORK_REGION	0.013162	0.063877
LIVE_REGION_NOT_WORK_REGION	0.002567	0.050300
REG_CITY_NOT_LIVE_CITY	0.048184	0.003947
REG_CITY_NOT_WORK_CITY	0.015838	0.004775
LIVE_CITY_NOT_WORK_CITY	-0.015598	0.002319

	REG_REGION_NOT_LIVE_REGION \
CNT_CHILDREN	-0.035213
AMT_INCOME_TOTAL	0.075615
AMT_CREDIT	0.015043
AMT_ANNUITY	0.029646
REGION_POPULATION_RELATIVE	-0.032702
DAYS_BIRTH	0.046320
DAYS_EMPLOYED	0.069566
DAYS_REGISTRATION	0.006362
DAYS_ID_PUBLISH	0.024860
hour_APPR_PROCESS_START	0.050953
REG_REGION_NOT_LIVE_REGION	1.000000
REG_REGION_NOT_WORK_REGION	0.506747
LIVE_REGION_NOT_WORK_REGION	0.068368
REG_CITY_NOT_LIVE_CITY	0.322030
REG_CITY_NOT_WORK_CITY	0.150968
LIVE_CITY_NOT_WORK_CITY	-0.013946

	REG_REGION_NOT_WORK_REGION \
CNT_CHILDREN	-0.040853
AMT_INCOME_TOTAL	0.156374
AMT_CREDIT	0.032536
AMT_ANNUITY	0.060363
REGION_POPULATION_RELATIVE	-0.008160
DAYS_BIRTH	0.022208
DAYS_EMPLOYED	0.082264
DAYS_REGISTRATION	0.000896
DAYS_ID_PUBLISH	0.013162
hour_APPR_PROCESS_START	0.063877

REG_REGION_NOT_LIVE_REGION	0.506747
REG_REGION_NOT_WORK_REGION	1.000000
LIVE_REGION_NOT_WORK_REGION	0.846872
REG_CITY_NOT_LIVE_CITY	0.141416
REG_CITY_NOT_WORK_CITY	0.224370
LIVE_CITY_NOT_WORK_CITY	0.181231

	LIVE_REGION_NOT_WORK_REGION \
CNT_CHILDREN	-0.027993
AMT_INCOME_TOTAL	0.145982
AMT_CREDIT	0.034861
AMT_ANNUITY	0.059724
REGION_POPULATION_RELATIVE	0.012602
DAYS_BIRTH	0.000356
DAYS_EMPLOYED	0.056081
DAYS_REGISTRATION	-0.001416
DAYS_ID_PUBLISH	0.002567
hour_appr_process_start	0.050300
REG_REGION_NOT_LIVE_REGION	0.068368
REG_REGION_NOT_WORK_REGION	0.846872
LIVE_REGION_NOT_WORK_REGION	1.000000
REG_CITY_NOT_LIVE_CITY	-0.006978
REG_CITY_NOT_WORK_CITY	0.167717
LIVE_CITY_NOT_WORK_CITY	0.233975

	REG_CITY_NOT_LIVE_CITY	REG_CITY_NOT_WORK_CITY \
CNT_CHILDREN	-0.016072	-0.005444
AMT_INCOME_TOTAL	-0.003813	-0.006241
AMT_CREDIT	-0.030974	-0.032882
AMT_ANNUITY	-0.011744	-0.015938
REGION_POPULATION_RELATIVE	-0.057239	-0.044761
DAYS_BIRTH	0.145884	0.096181
DAYS_EMPLOYED	0.118869	0.139863
DAYS_REGISTRATION	0.015831	0.039204
DAYS_ID_PUBLISH	0.048184	0.015838
hour_appr_process_start	0.003947	0.004775
REG_REGION_NOT_LIVE_REGION	0.322030	0.150968
REG_REGION_NOT_WORK_REGION	0.141416	0.224370
LIVE_REGION_NOT_WORK_REGION	-0.006978	0.167717
REG_CITY_NOT_LIVE_CITY	1.000000	0.478266
REG_CITY_NOT_WORK_CITY	0.478266	1.000000
LIVE_CITY_NOT_WORK_CITY	-0.029432	0.768247

	LIVE_CITY_NOT_WORK_CITY
CNT_CHILDREN	0.009557
AMT_INCOME_TOTAL	0.004230
AMT_CREDIT	-0.012465

AMT_ANNUITY	-0.003012
REGION_POPULATION_RELATIVE	-0.014753
DAYS_BIRTH	0.009633
DAYS_EMPLOYED	0.069316
DAYS_REGISTRATION	0.026105
DAYS_ID_PUBLISH	-0.015598
HOURL_APPR_PROCESS_START	0.002319
REG_REGION_NOT_LIVE_REGION	-0.013946
REG_REGION_NOT_WORK_REGION	0.181231
LIVE_REGION_NOT_WORK_REGION	0.233975
REG_CITY_NOT_LIVE_CITY	-0.029432
REG_CITY_NOT_WORK_CITY	0.768247
LIVE_CITY_NOT_WORK_CITY	1.000000

```
[255]: # Now, plotting the above correlation with heat map as it is the best choice to
      ↪ visualize
```

```
# figure size
```

```
def targets_corr(data,title):
    plt.figure(figsize=(15, 10))
    plt.rcParams['axes.titlesize'] = 25
    plt.rcParams['axes.titlepad'] = 70
```

```
# heatmap with a color map of choice
```

```
    sns.heatmap(data, cmap=sns.cubehelix_palette(as_cmap=True),,annot=False)
    plt.title(title)
    plt.yticks(rotation=0)
    plt.show()
```

```
Input In [255]
```

```
sns.heatmap(data, cmap=sns.cubehelix_palette(as_cmap=True),,annot=False)
```

```
SyntaxError: invalid syntax
```

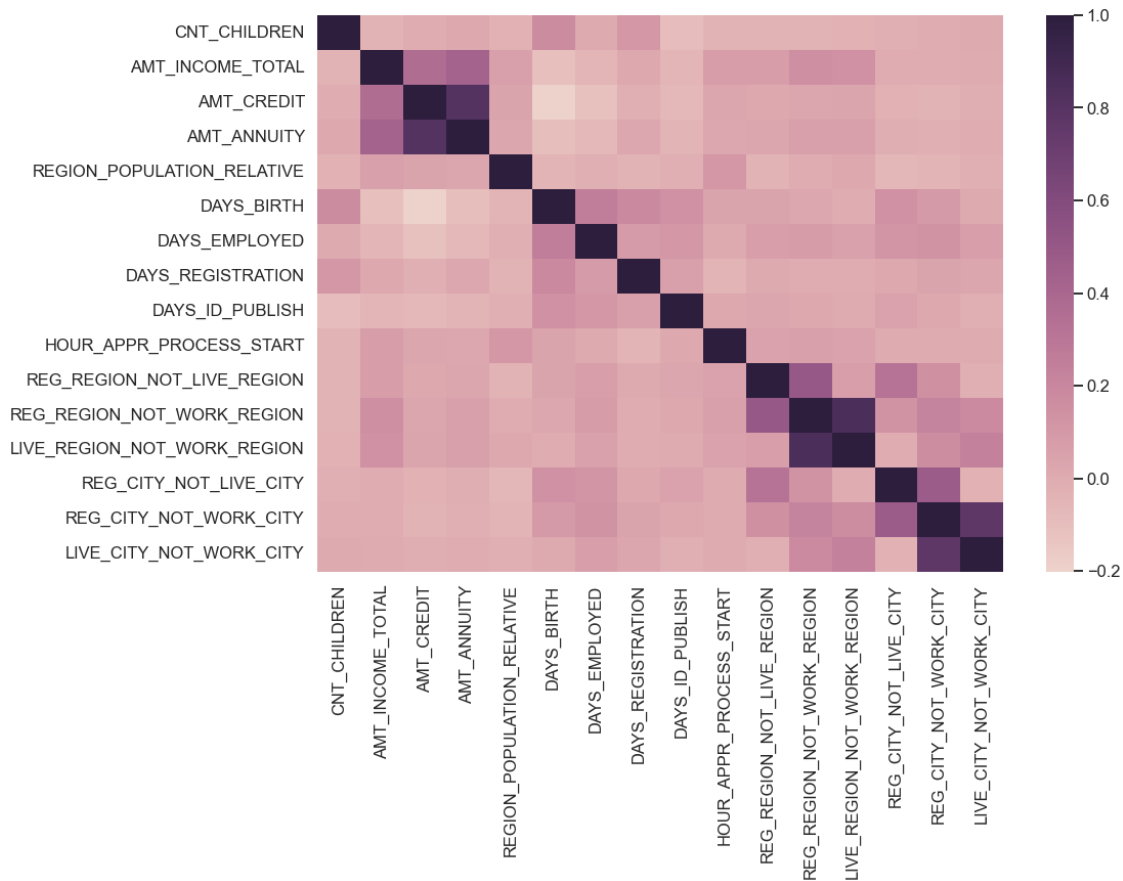
```
[ ]: # For Target 0
```

```
targets_corr(data=target0,title='Correlation for target 0')
```

```
[256]: # For Target 1
```

```
targets_corr(data=target1,title='Correlation for target 1')
```


Correlation for target 1



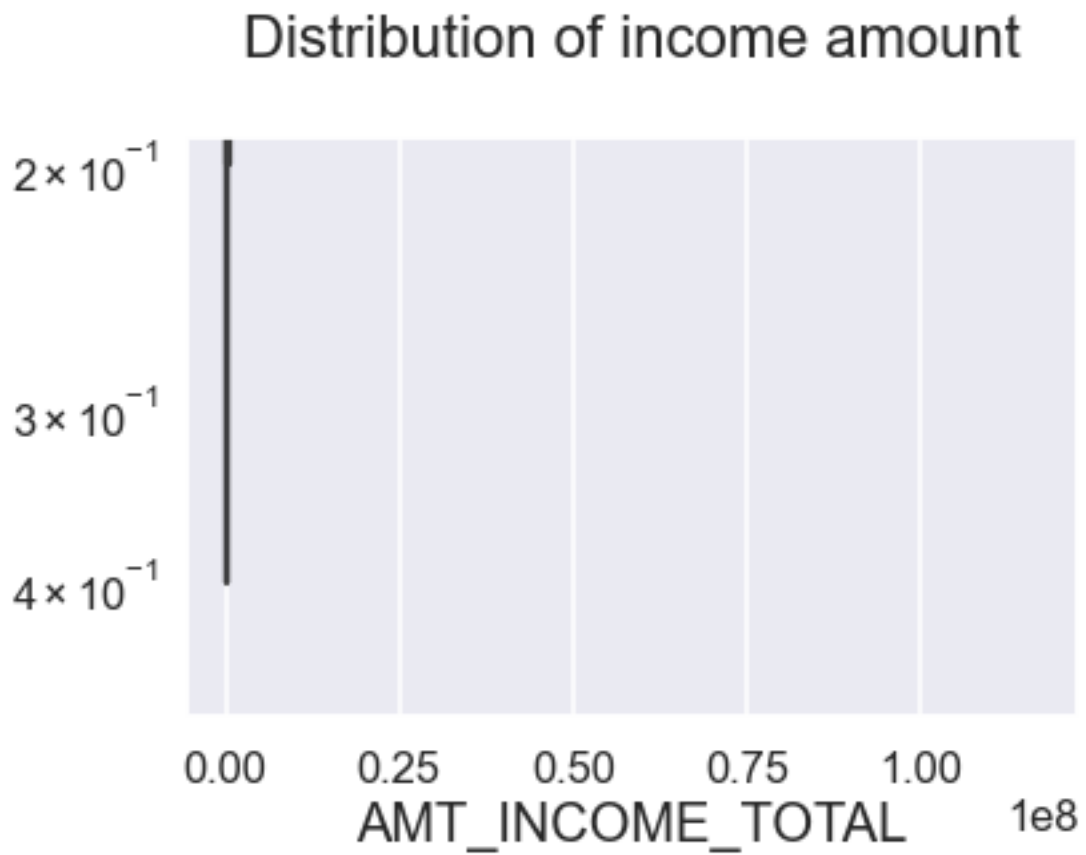
[257]: *# Box plotting for univariate variables analysis in logarithmic scale*

```
def univariate_numerical(data,col,title):
    sns.set_style('darkgrid')
    sns.set_context('talk')
    plt.rcParams["axes.labelsize"] = 20
    plt.rcParams['axes.titlesize'] = 22
    plt.rcParams['axes.titlepad'] = 30

    plt.title(title)
    plt.yscale('log')
    sns.boxplot(data =target1_df, x=col,orient='v')
    plt.show()
```

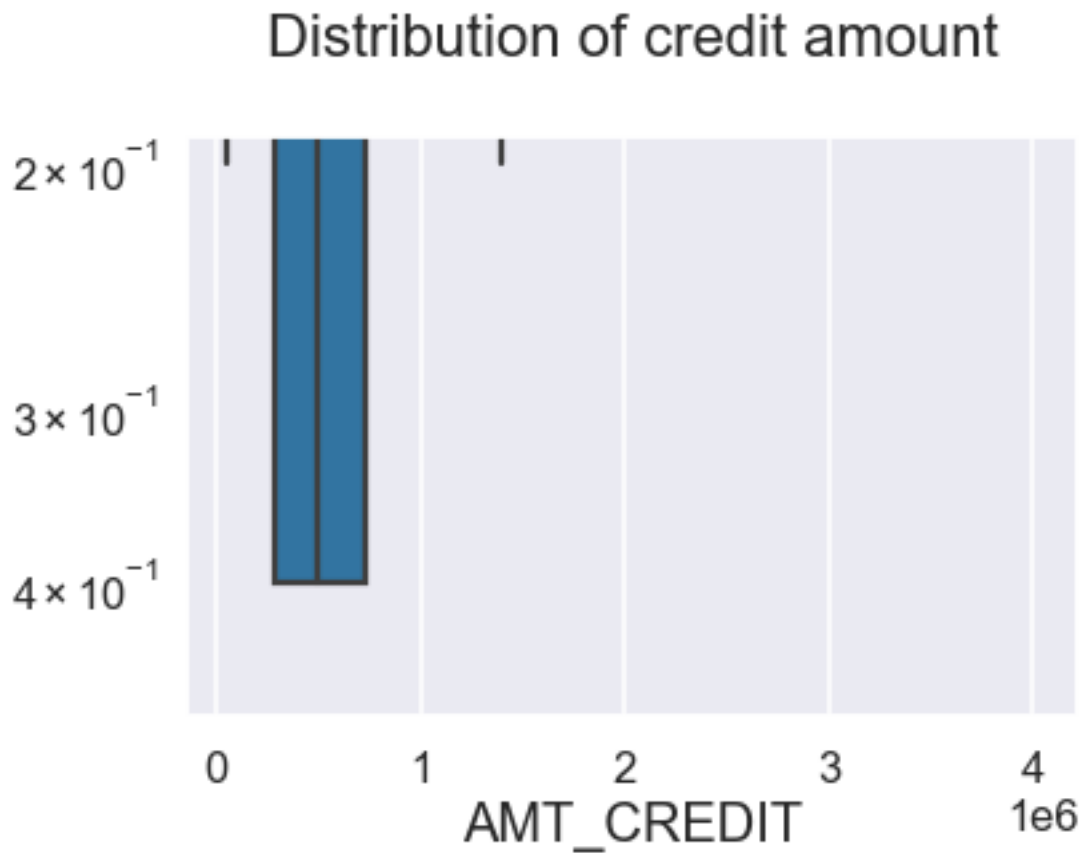
[258]: *# Distribution of income amount*

```
univariate_numerical(data=target0_df,col='AMT_INCOME_TOTAL',title='Distribution of income amount')
```



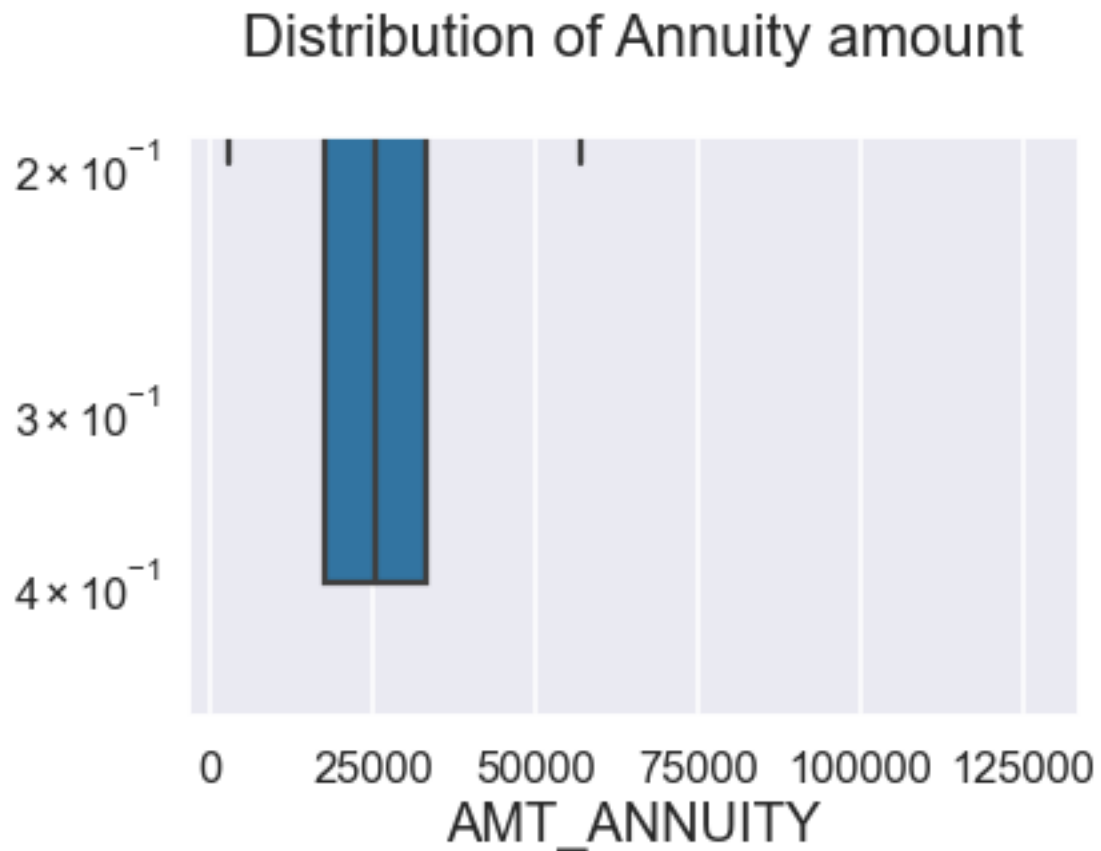
[259]: *# Disrtibution of credit amount*

```
univariate_numerical(data=target0_df,col='AMT_CREDIT',title='Distribution of credit amount')
```

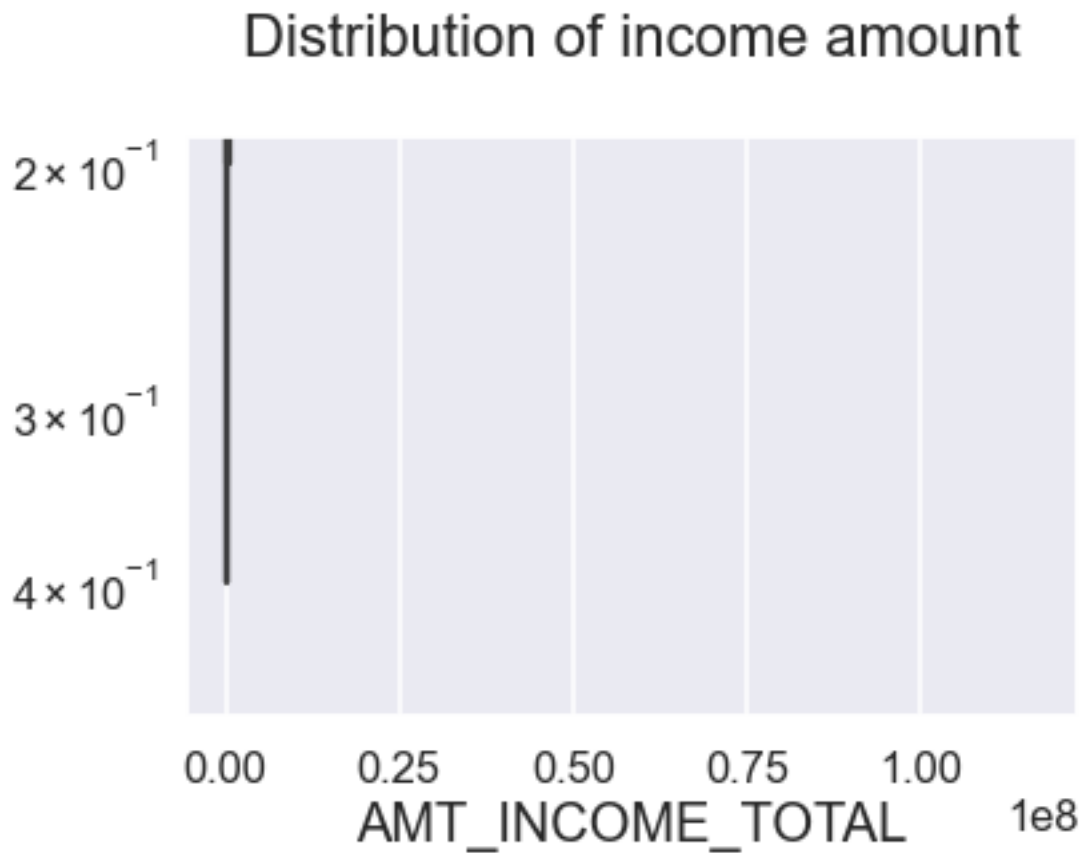


```
[260]: # Distribution of annuity amount

univariate_numerical(data=target0_df,col='AMT_ANNUITY',title='Distribution of_
↳Annuity amount')
```



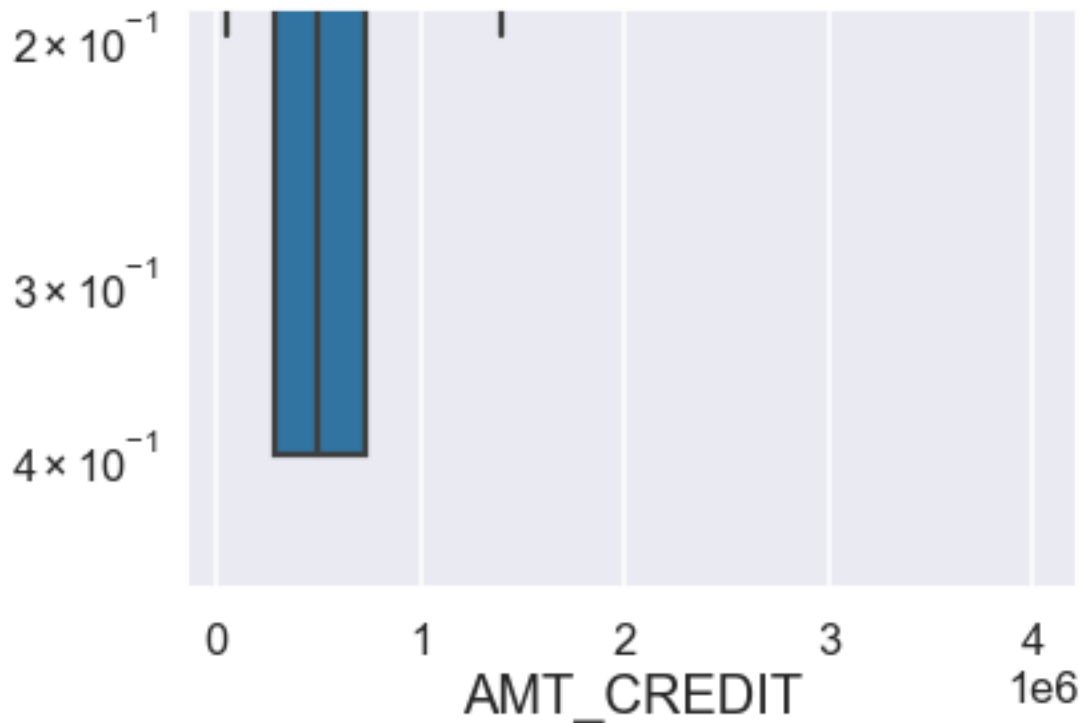
```
[261]: # Distribution of income amount  
  
univariate_numerical(data=target1_df,col='AMT_INCOME_TOTAL',title='Distribution  
of income amount')
```



```
[262]: # Distribution of credit amount

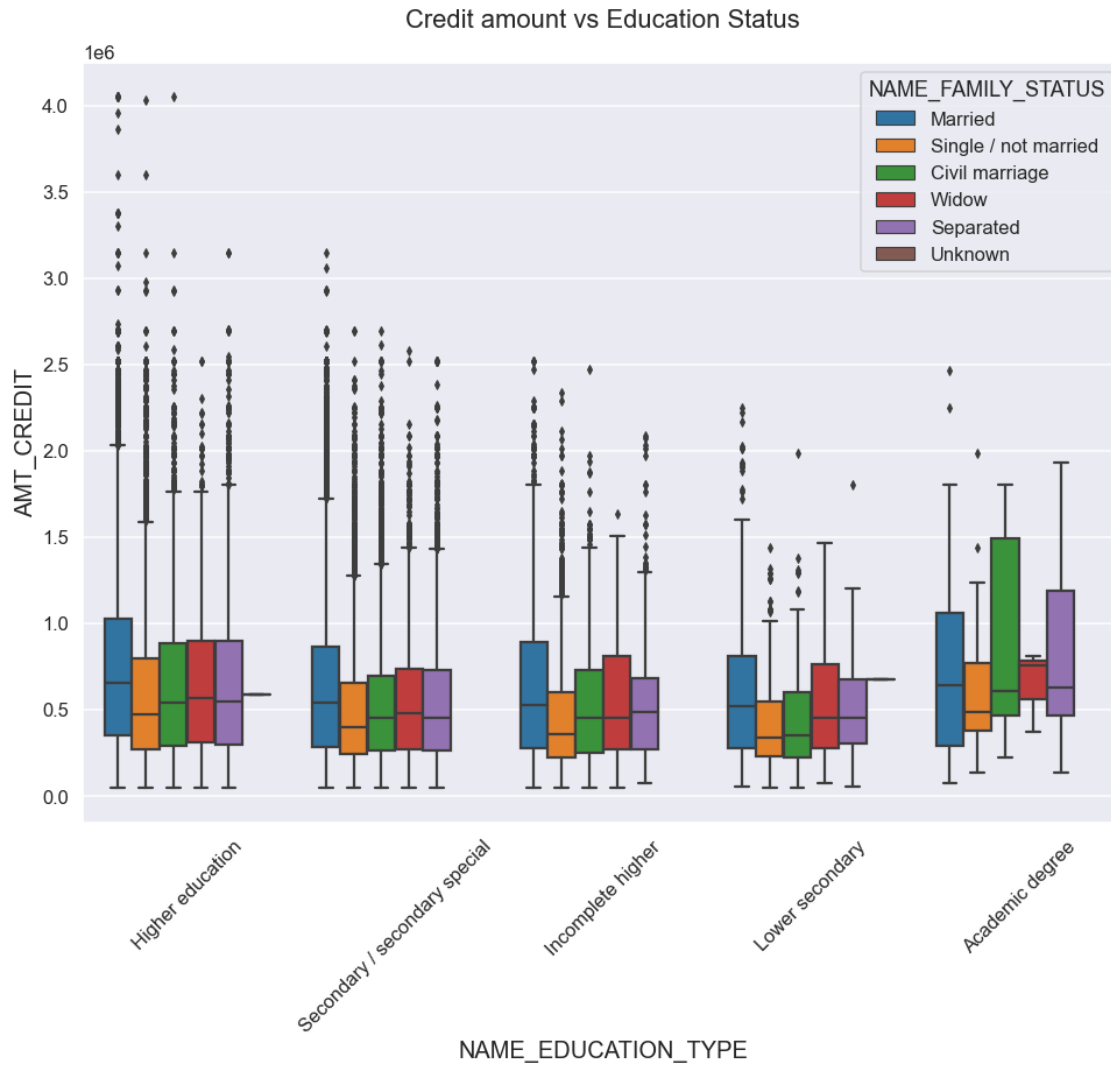
univariate_numerical(data=target1_df,col='AMT_CREDIT',title='Distribution of_
↳credit amount')
```

Distribution of credit amount



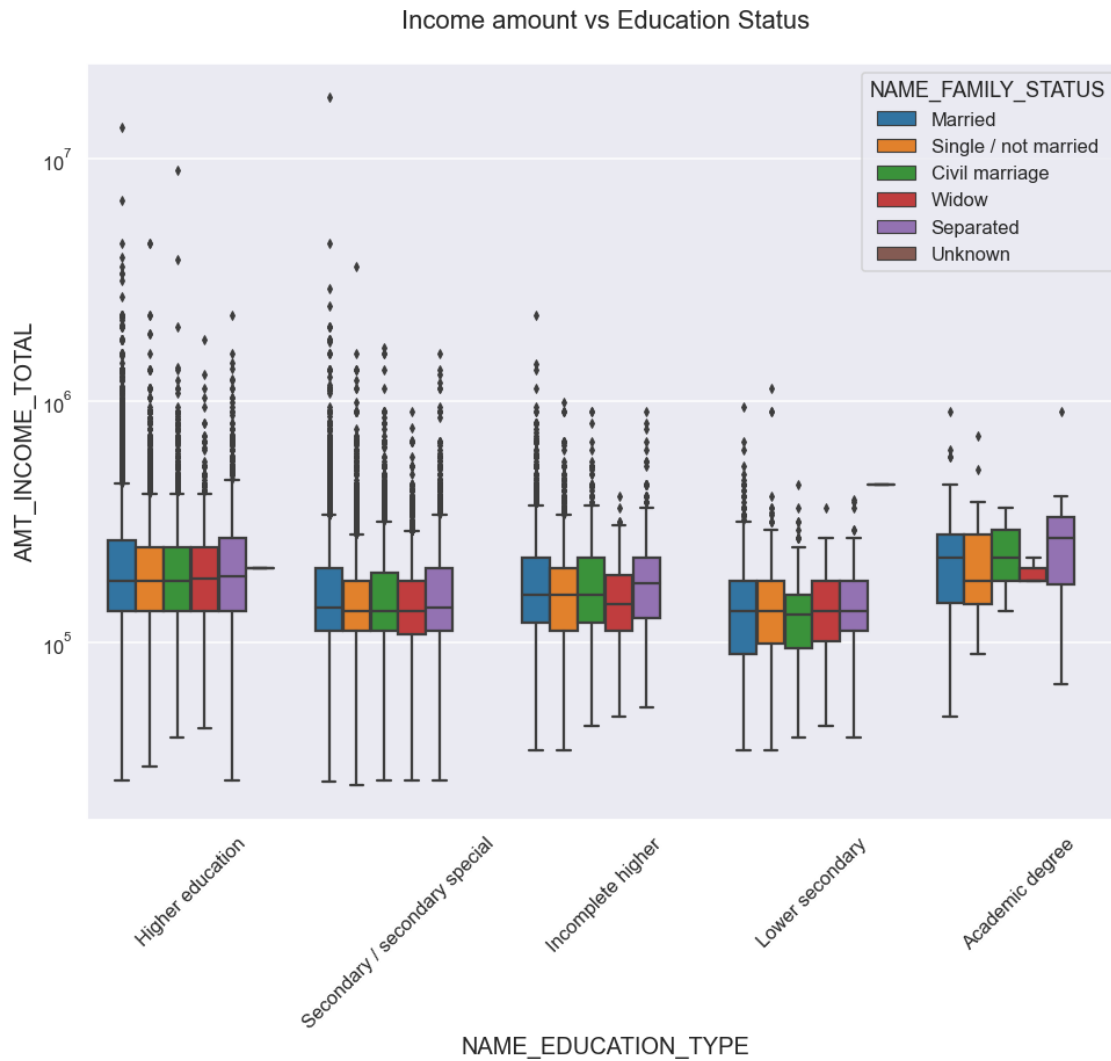
```
[263]: # Box plotting for Credit amount

plt.figure(figsize=(16,12))
plt.xticks(rotation=45)
sns.boxplot(data =target0_df, x='NAME_EDUCATION_TYPE',y='AMT_CREDIT', hue=
    ↪ 'NAME_FAMILY_STATUS',orient='v')
plt.title('Credit amount vs Education Status')
plt.show()
```



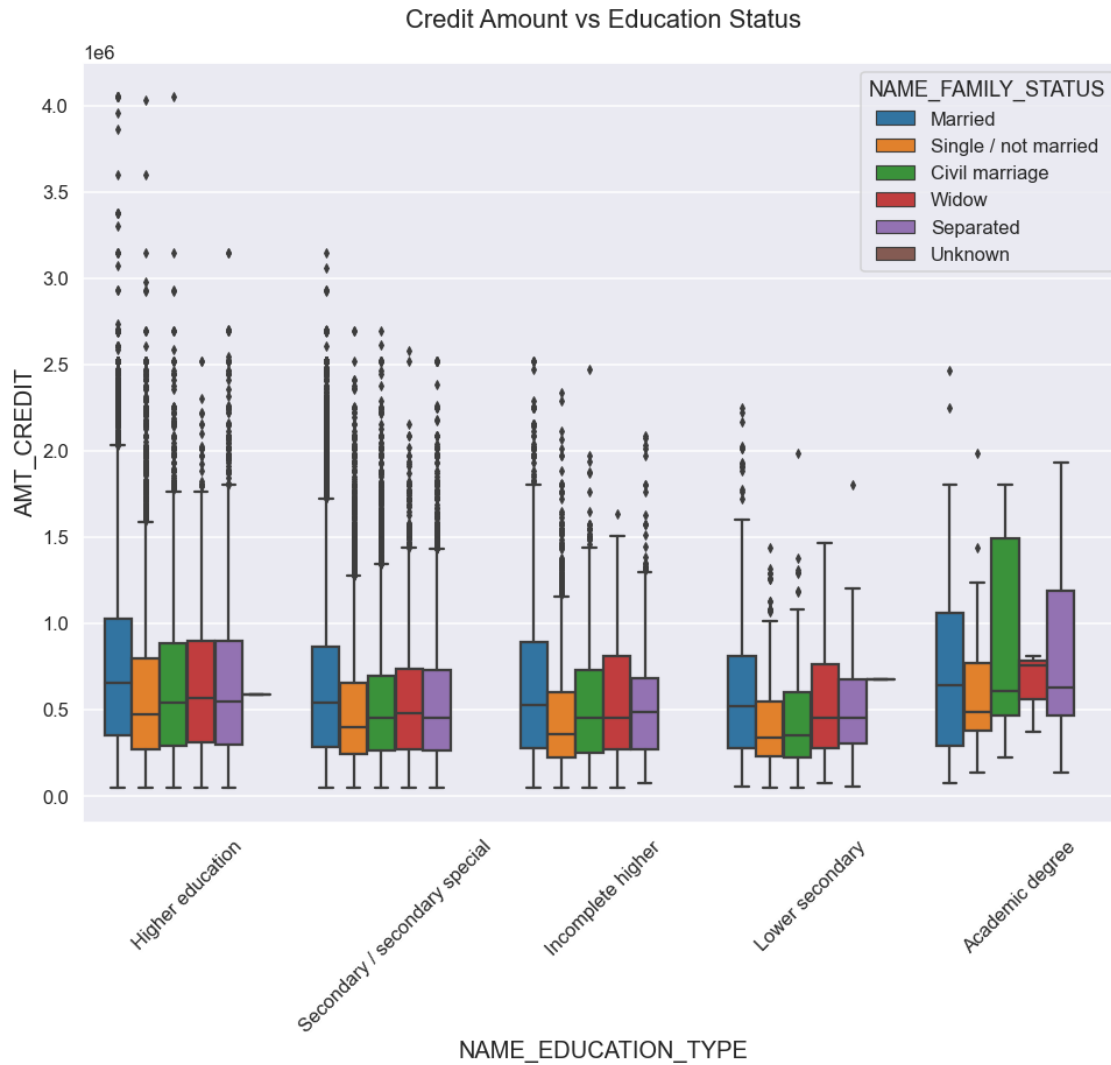
[264]: *# Box plotting for Income amount in logarithmic scale*

```
plt.figure(figsize=(16,12))
plt.xticks(rotation=45)
plt.yscale('log')
sns.boxplot(data =target0_df, x='NAME_EDUCATION_TYPE',y='AMT_INCOME_TOTAL', hue_
    ↪='NAME_FAMILY_STATUS',orient='v')
plt.title('Income amount vs Education Status')
plt.show()
```



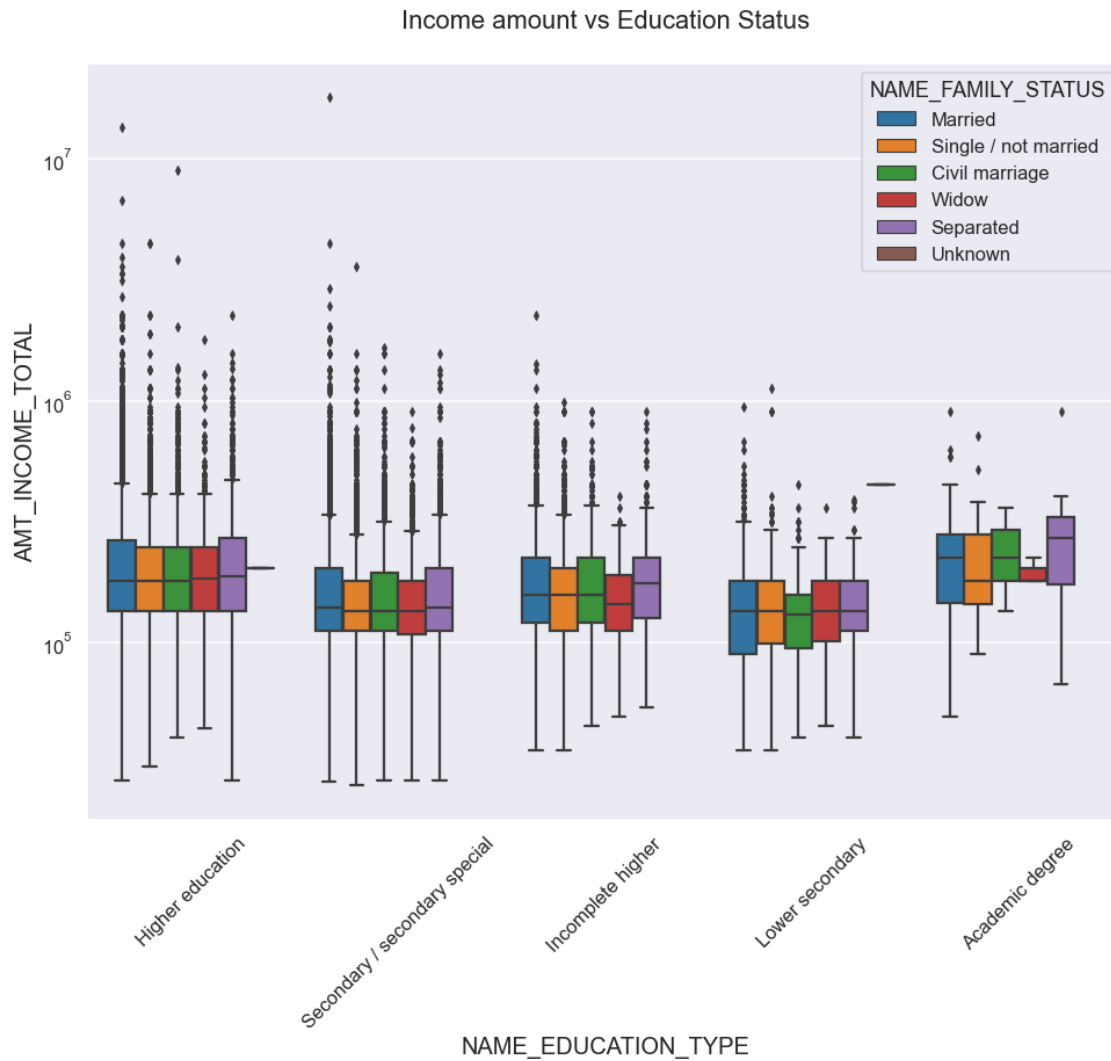
```
[265]: # Box plotting for credit amount

plt.figure(figsize=(16,12))
plt.xticks(rotation=45)
sns.boxplot(data =target0_df, x='NAME_EDUCATION_TYPE',y='AMT_CREDIT', hue=
    ↪='NAME_FAMILY_STATUS',orient='v')
plt.title('Credit Amount vs Education Status')
plt.show()
```

```
[266]: # Box plotting for Income amount in logarithmic scale

plt.figure(figsize=(16,12))
plt.xticks(rotation=45)
plt.yscale('log')
sns.boxplot(data =target0_df, x='NAME_EDUCATION_TYPE',y='AMT_INCOME_TOTAL', hue_
    ↪='NAME_FAMILY_STATUS',orient='v')
plt.title('Income amount vs Education Status')
plt.show()
```



```
[267]: # Reading the dataset of previous application
```

```
df1=pd.read_csv("previous_application.csv")
```

```
[268]: # Cleaning the missing data
```

```
# listing the null values columns having more than 30%
```

```
emptycol1=df1.isnull().sum()
```

```
emptycol1=emptycol1[emptycol1.values>(0.3*len(emptycol1))]
```

```
len(emptycol1)
```

```
[268]: 15
```

[269]: *# Removing those 15 columns*

```
emptycol1 = list(emptycol1[emptycol1.values>=0.3].index)
df1.drop(labels=emptycol1,axis=1,inplace=True)

df1.shape
```

[269]: (1670214, 22)

[270]: *# Removing the column values of 'XNA' and 'XAP'*

```
df1=df1.drop(df1[df1['NAME_CASH_LOAN_PURPOSE']=='XNA'].index)
df1=df1.drop(df1[df1['NAME_CASH_LOAN_PURPOSE']=='XNA'].index)
df1=df1.drop(df1[df1['NAME_CASH_LOAN_PURPOSE']=='XAP'].index)

df1.shape
```

[270]: (69635, 22)

[271]: *# Now merging the Application dataset with previous appliaction dataset*

```
new_df=pd.merge(left=df,right=df1,how='inner',on='SK_ID_CURR',suffixes='_x')
```

[272]: *# Renaming the column names after merging*

```
new_df1 = new_df.rename({'NAME_CONTRACT_TYPE_' :  
    ↳ 'NAME_CONTRACT_TYPE', 'AMT_CREDIT_' : 'AMT_CREDIT', 'AMT_ANNUITY_' : 'AMT_ANNUITY',  
    ↳ 'WEEKDAY_APPR_PROCESS_START_' :  
    ↳ 'WEEKDAY_APPR_PROCESS_START',  
    ↳ 'HOUR_APPR_PROCESS_START_' :  
    ↳ 'HOUR_APPR_PROCESS_START', 'NAME_CONTRACT_TYPEx' : 'NAME_CONTRACT_TYPE_PREV',  
    ↳ 'AMT_CREDITx' : 'AMT_CREDIT_PREV', 'AMT_ANNUITYx' :  
    ↳ 'AMT_ANNUITY_PREV',  
    ↳ 'WEEKDAY_APPR_PROCESS_STARTx' :  
    ↳ 'WEEKDAY_APPR_PROCESS_START_PREV',  
    ↳ 'HOUR_APPR_PROCESS_STARTx' :  
    ↳ 'HOUR_APPR_PROCESS_START_PREV'}, axis=1)
```

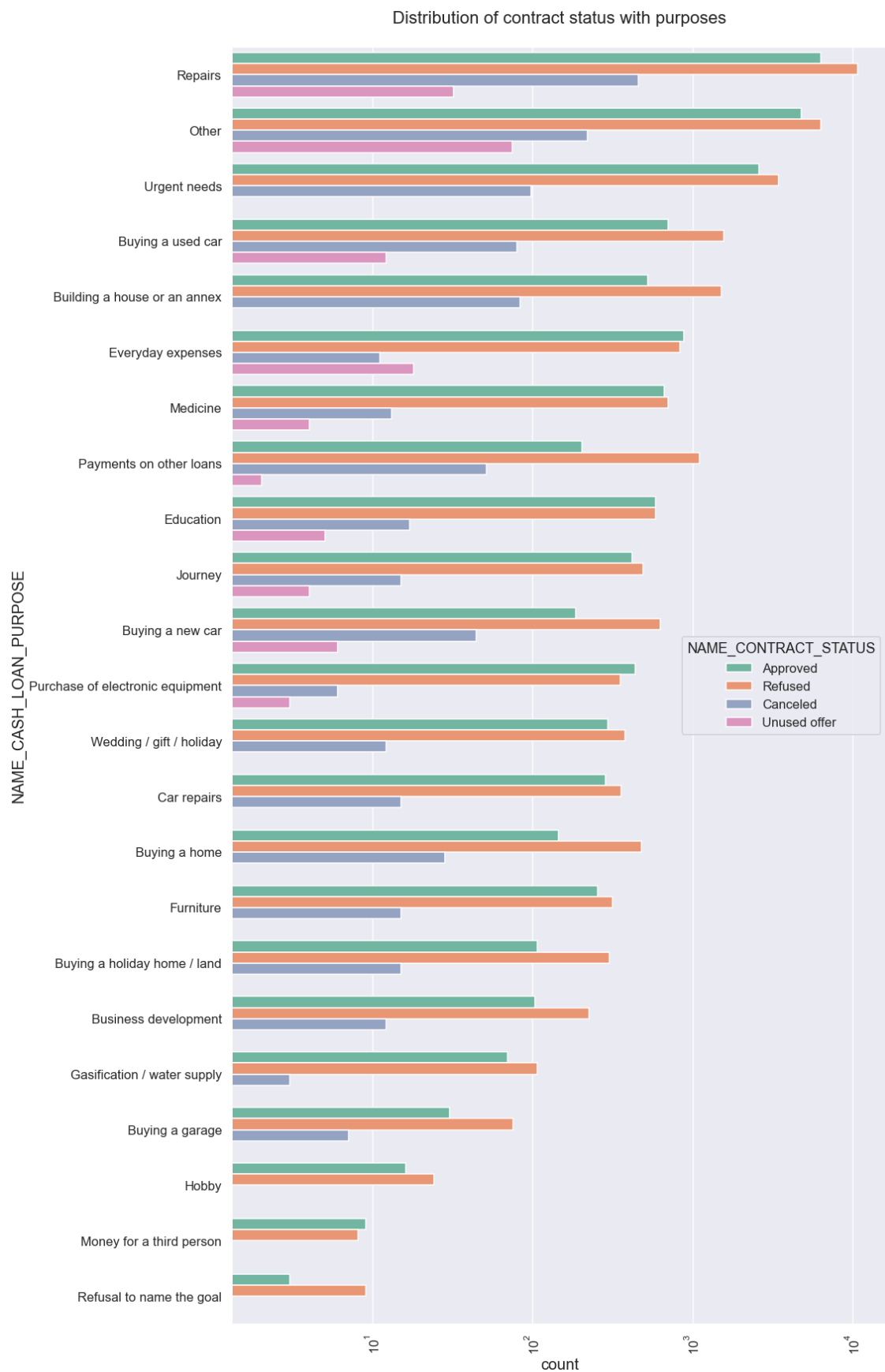
[273]: *# Removing unwanted columns for analysis*

```
new_df1.drop(['SK_ID_CURR', 'WEEKDAY_APPR_PROCESS_START',  
    ↳ 'HOUR_APPR_PROCESS_START', 'REG_REGION_NOT_LIVE_REGION',  
    ↳ 'REG_REGION_NOT_WORK_REGION', 'LIVE_REGION_NOT_WORK_REGION',  
    ↳ 'REG_CITY_NOT_LIVE_CITY',  
    ↳ 'REG_CITY_NOT_WORK_CITY',  
    ↳ 'LIVE_CITY_NOT_WORK_CITY', 'WEEKDAY_APPR_PROCESS_START_PREV',
```

```
        'HOUR_APPR_PROCESS_START_PREV',  
        ↪ 'FLAG_LAST_APPL_PER_CONTRACT', 'NFLAG_LAST_APPL_IN_DAY'], axis=1, inplace=True)
```

```
[274]: # Distribution of contract status in logarithmic scale
```

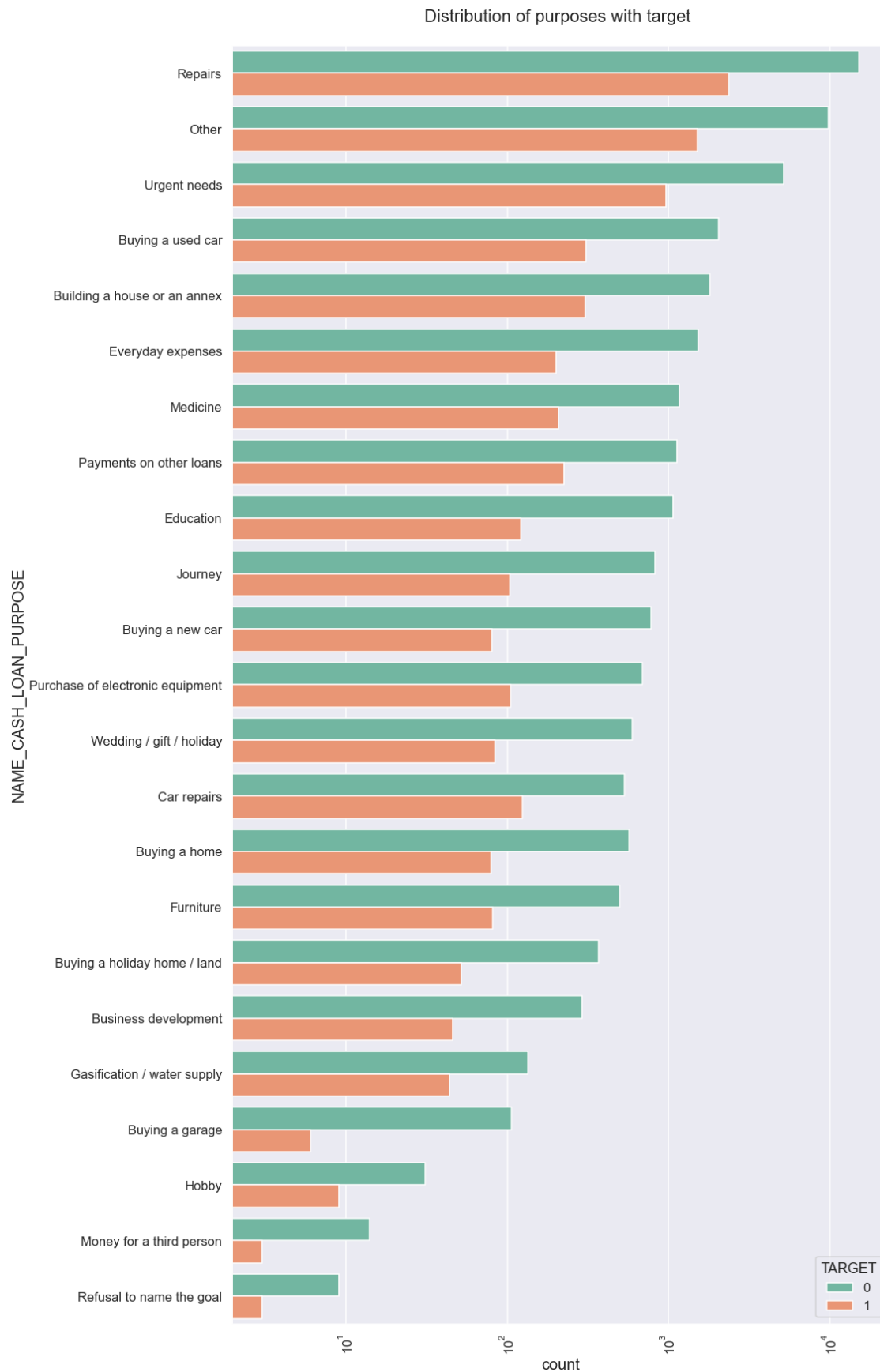
```
sns.set_style('darkgrid')  
sns.set_context('talk')  
  
plt.figure(figsize=(15,30))  
plt.rcParams["axes.labelsize"] = 20  
plt.rcParams['axes.titlesize'] = 22  
plt.rcParams['axes.titlepad'] = 30  
plt.xticks(rotation=90)  
plt.xscale('log')  
plt.title('Distribution of contract status with purposes')  
ax = sns.countplot(data = new_df1, y= 'NAME_CASH_LOAN_PURPOSE',  
                  order=new_df1['NAME_CASH_LOAN_PURPOSE'].value_counts().  
                  ↪ index, hue = 'NAME_CONTRACT_STATUS', palette='Set2')
```



```
[275]: # Distribution of contract status

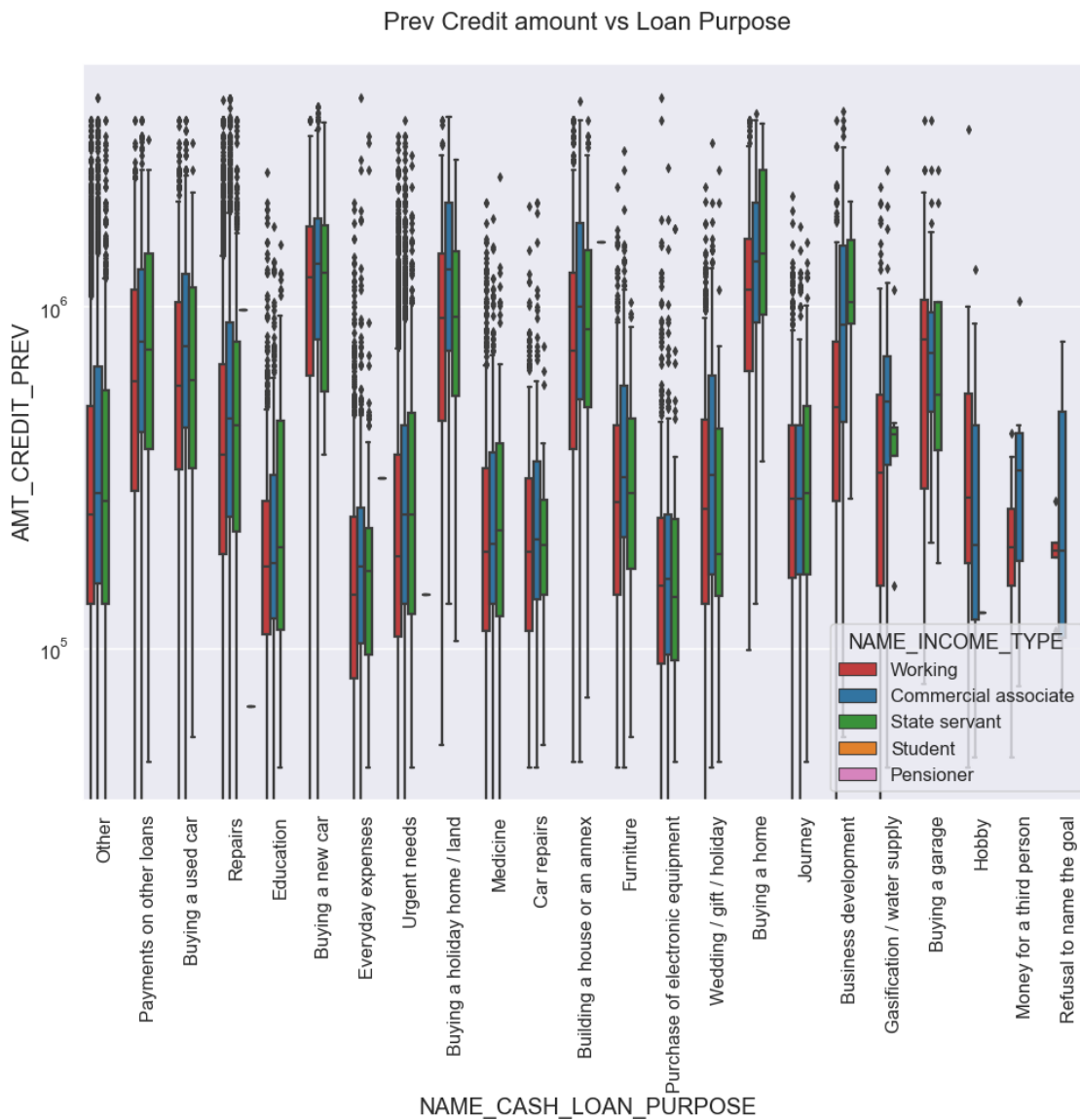
sns.set_style('darkgrid')
sns.set_context('talk')

plt.figure(figsize=(15,30))
plt.rcParams["axes.labelsize"] = 20
plt.rcParams['axes.titlesize'] = 22
plt.rcParams['axes.titlepad'] = 30
plt.xticks(rotation=90)
plt.xscale('log')
plt.title('Distribution of purposes with target ')
ax = sns.countplot(data = new_df1, y= 'NAME_CASH_LOAN_PURPOSE',
                   order=new_df1['NAME_CASH_LOAN_PURPOSE'].value_counts().
    ↪ index, hue = 'TARGET', palette='Set2')
```



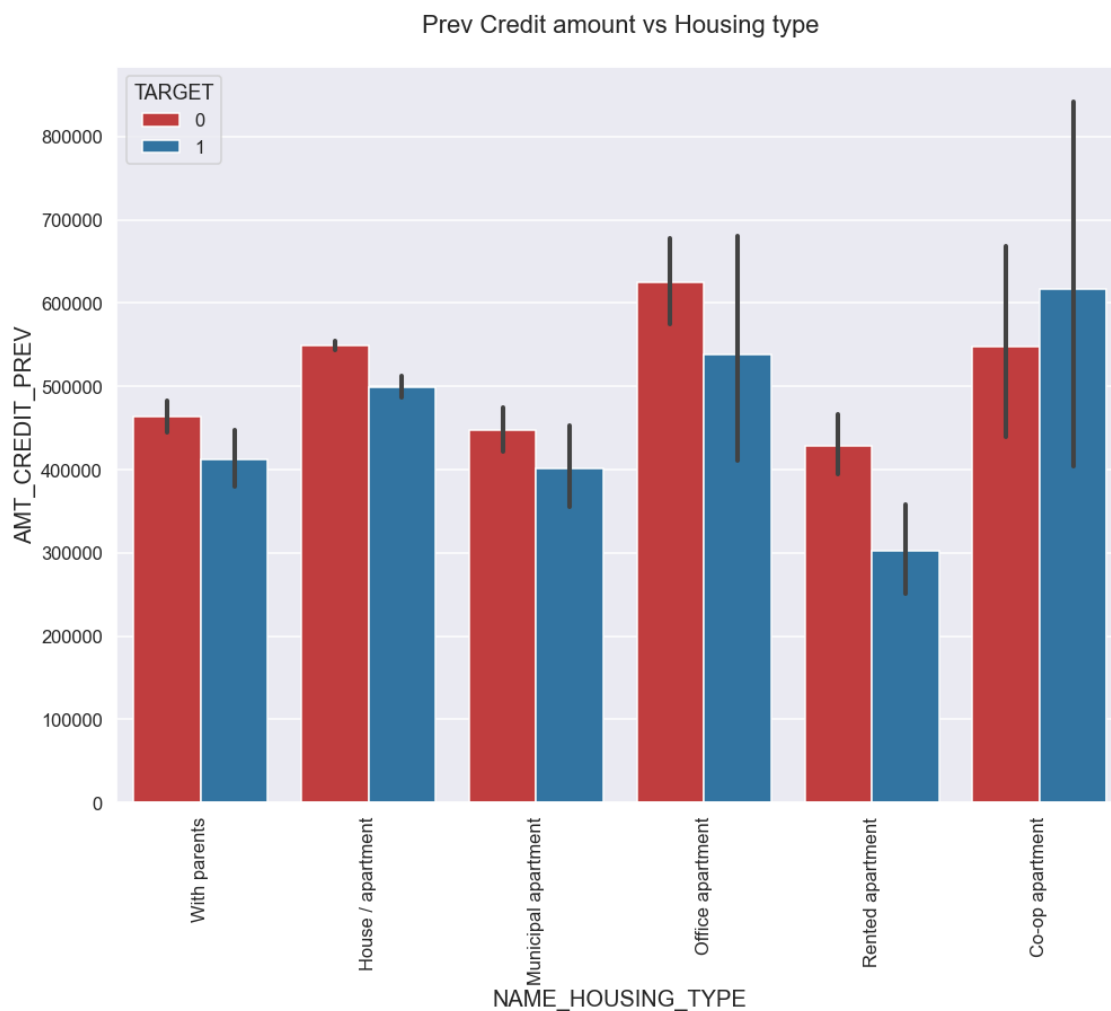
```
[276]: # Box plotting for Credit amount in logarithmic scale

plt.figure(figsize=(16,12))
plt.xticks(rotation=90)
plt.yscale('log')
sns.boxplot(data =new_df1,
            x='NAME_CASH_LOAN_PURPOSE',hue='NAME_INCOME_TYPE',y='AMT_CREDIT_PREV',orient='v',palette=
            ['tab:red', 'tab:blue','tab:green','tab:orange','tab:pink'])
plt.title('Prev Credit amount vs Loan Purpose')
plt.show()
```




```
[277]: # Box plotting for Credit amount prev vs Housing type in logarithmic scale

plt.figure(figsize=(16,12))
plt.xticks(rotation=90)
sns.barplot(data =new_df1,
            y='AMT_CREDIT_PREV',hue='TARGET',x='NAME_HOUSING_TYPE',palette = ['tab:red',
            'tab:blue'])
plt.title('Prev Credit amount vs Housing type')
plt.show()
```



CONCLUSION 1. 'Student', 'pensioner' and 'Businessman' should be given higher priority for giving loans by the banks compared to the clients with housing 'type other than 'Co-op apartment' for successful payments.

2. Income type 'Working' should be given less priority by the banks due to large number of

unsuccessful payments compared to others.

3. Higher number of unsuccessful payments occurred in loan type 'Repair' in the past. So Banks should keep this in mind while considering loan types 'Repair'.
4. Housing type 'With parents' should be given the higher priority by the banks because of their successful payments on time as per data.

[]:

[]: