Credit EDA ASSIGNMENT

Submitted by Snehil Tiwari

Business Objective

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.

In other words, the company wants to understand the driving factors (or driver variables) behind loan default, i.e. the variables which are strong indicators of default. The company can utilise this knowledge for its portfolio and risk assessment.
To develop your understanding of the domain, you are advised to independently research a little about risk analytics - understanding the types of variables and their significance should be enough.

APPLICATION DATASET ANALYSIS

Introduction

In this notebook, we have mainly focused on analyzing **application_data.csv** i.e. data about current application of loan.

Approach of Application Data Analysis

- Importing Module, libraries and suppressing warning messages
- Reading the Dataset into Pandas Dataframe
- We have divided the features into small segments and analyzed segment-wise using a smaller dataframe containing only relevant categories.
- Data Cleaning, Missing Data Handling, Type casting are done segment-wise.
- Plots and percentage wise Defaulter calculation are done segment-wise as well.

Reading the Dataset into Pandas Dataframe

Reading The Dataset

In [226]:
```
application_dataset = pd.read_csv("application_data.csv")
prev_appl_dataset = pd.read_csv("previous_application.csv")
```

In [227]:
```
# Reading Application Dataset
application_dataset.head()
```

Out[227]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDI
0	100002	1	Cash loans	M	N	Y	0	202500.0	406597.
1	100003	0	Cash loans	F	N	N	0	270000.0	1293502.
2	100004	0	Revolving loans	M	Y	Y	0	67500.0	135000.
3	100006	0	Cash loans	F	N	Y	0	135000.0	312682.
4	100007	0	Cash loans	M	N	Y	0	121500.0	513000.

5 rows × 122 columns

In [228]:
```
# Reading Previous Application Dataset
prev_appl_dataset.head()
```

Out[228]:

	SK_ID_PREV	SK_ID_CURR	NAME_CONTRACT_TYPE	AMT_ANNUITY	AMT_APPLICATION	AMT_CREDIT	AMT_DOWN_PAYMENT	AMT_GOODS_PRICE	WEEKD
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

Comment:

prev_ap_df contains 37 features and 1670214 rows (Out of which 15 features are float64, 6 features are integer, 16 features are object datatype)

application_df contains 121 features, 1 target variable, and 307511 rows (Out of which 65 features are float64, 41 features are integer, 16 features are object datatype)

Checking Data Imbalance and Ratio

Comment: This data is highly imbalanced as number of defaulter is very less in total population.

Data Imbalance Ratio

Defaulter : Non-Defaulter = 8 : 92 = 2 : 23

Checking Data Imbalance

In [234]:
```
plt.title("Payment Status of Current Application")
sns.countplot(application_dataset['TARGET'])
plt.show()
```

Data Imbalance Ratio

In [235]:
```
non_default  = application_dataset[application_dataset["TARGET"] == 0]
default  = application_dataset[application_dataset["TARGET"] == 1]

print("No. of defaulters: ", default.shape[0])
print("No. of non-defaulters: ", non_default.shape[0])
```

```
No. of defaulters:  24825
No. of non-defaulters:  282686
```

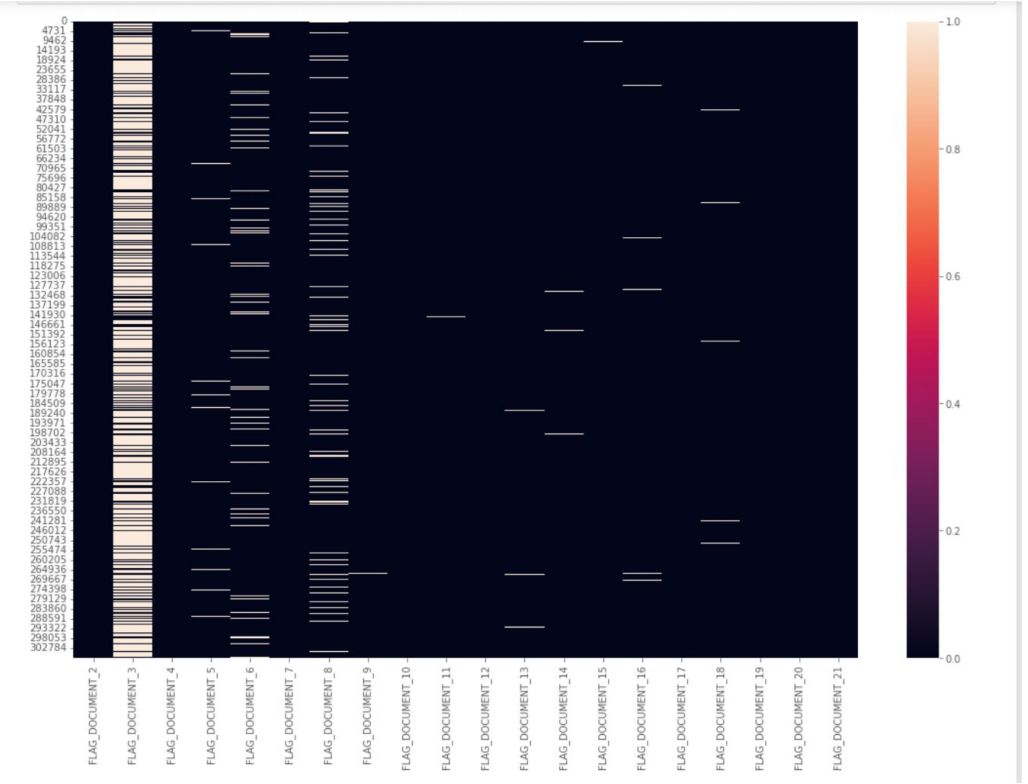
Applicant Documents Submission

Here we are analyzing 'FLAG_DOCUMENT_2','FLAG_DOCUMENT_3',...,'FLAG_DOCUMENT_21' columns. Our goal to understand whether trend of document submission and identify impact on TARGET variable(if any).

Comment:

- The heatmap suggests that all of the documents except Document 3 were not provided by applicants in majority of the cases.
- Hence we can assume all the document (except document 3) will not contribute towards analyzing the data. Hence all this columns can be dropped.

<u>Checking both defaulter and non-defaulter entries to identify importance of Document 3, using equal ylim to for better visibilty</u>



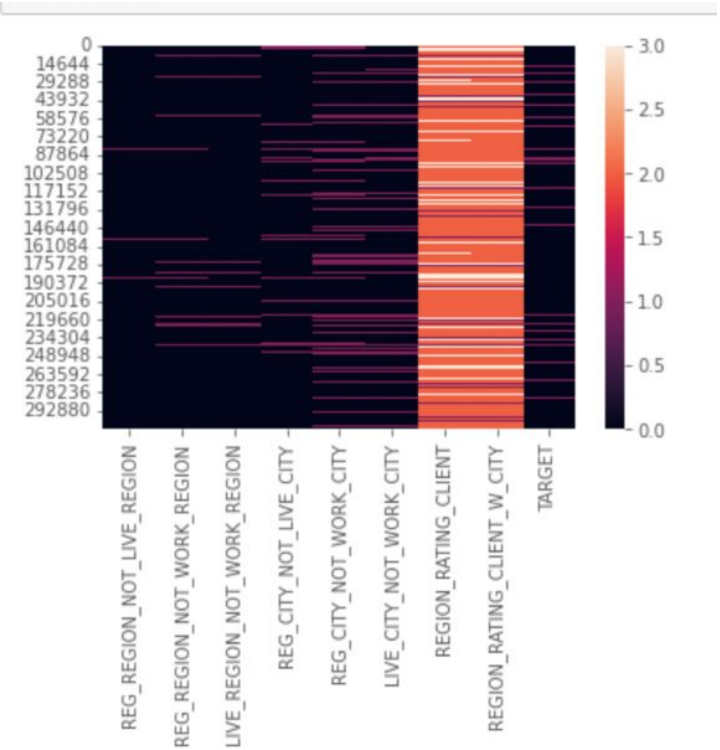
Comment:

- FLAG_DOCUMENT_3 is showing similar trend for both non-defaulters and defaulters.
- Hence, this column can be dropped.

Housing Information of Applicant

Comment: All of the features have very high (47-70%) missing data percentage. Hen
Comment:

- Most of the applicants live in House/Apartment
- Applicants living with their parents or in rented apartment have higher rate of



Region Related Data

Comment:

- All the features are labeled as 0 and 1 REG_REGION_NOT_LIVE_REGION mostly contains 0, hence can be removed REG_REGION_NOT_WORK_REGION, LIVE_REGION_NOT_WORK_REGION columns are identical, hence one of them ca be removed REG_CITY_NOT_WORK_CITY, LIVE_CITY_NOT_WORK_CITY columns are identical, hence one of them can be removed

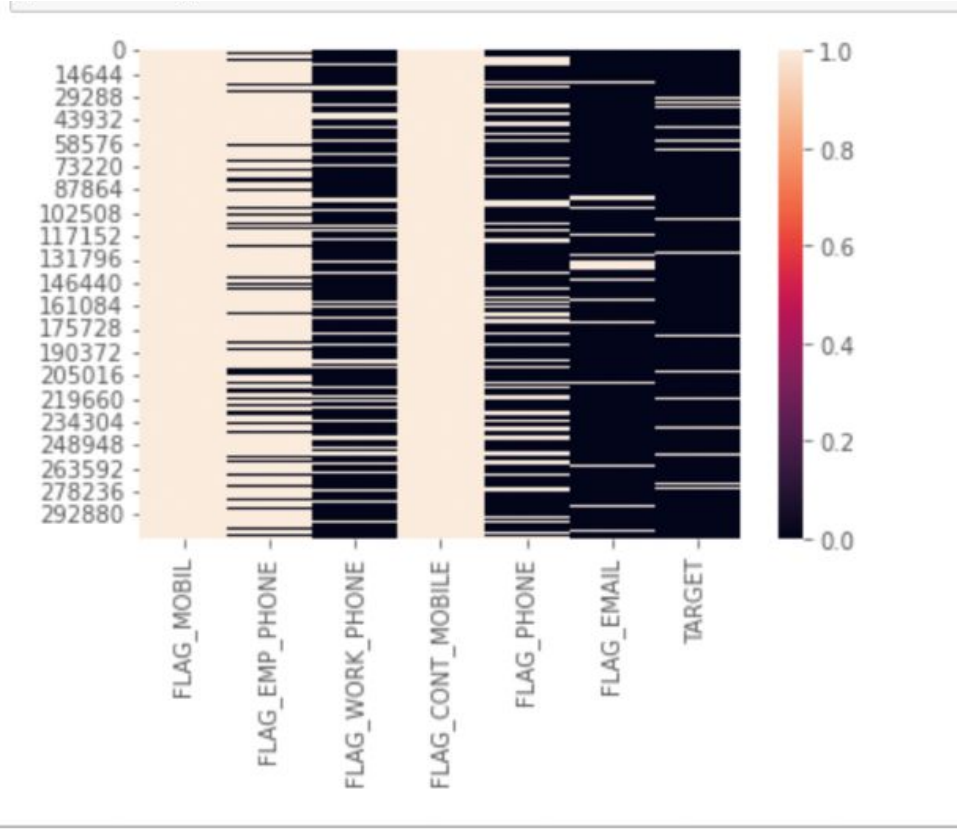
Analysis for the ‘Target variable’ in the Application Dataset

Region Related Data

Comment:

Defaulter rate is highest when REG_REGION_NOT_WORK_REGION=0 i.e. permanent address and working address is same Highest Applicants have Region rating of 2

Contact Related Information



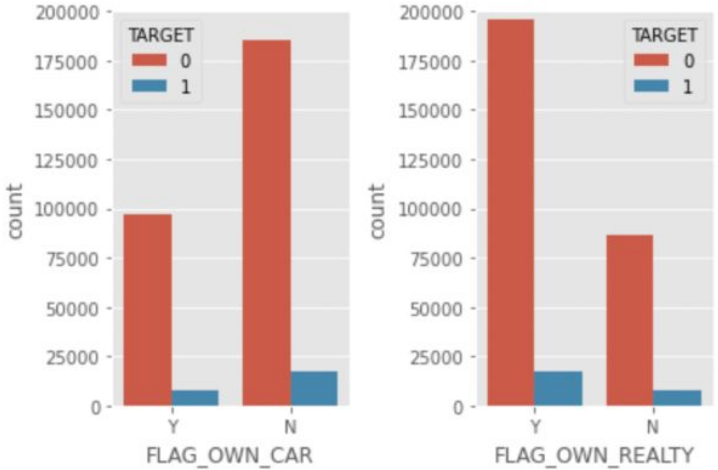
Comment:

All the features in contact_df are categorical (0 and 1)

As there is no similarity of patterns of TARGET value with the features,we are assuming the feature are not useful for analysis.

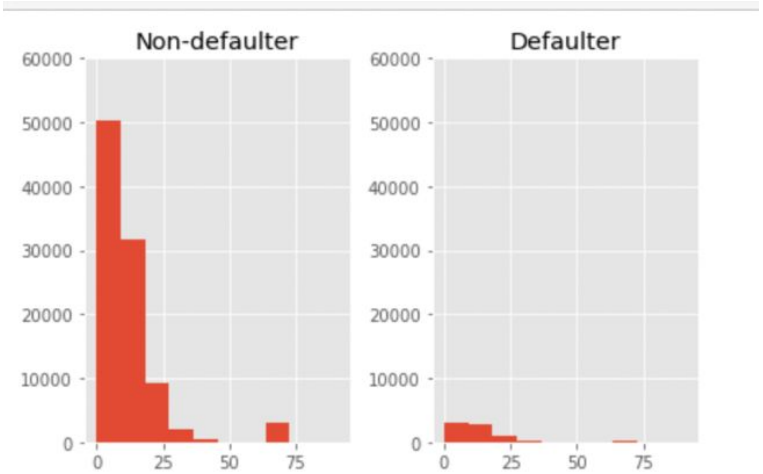
Hence, all of the features can be removed

Asset Details



Comment:

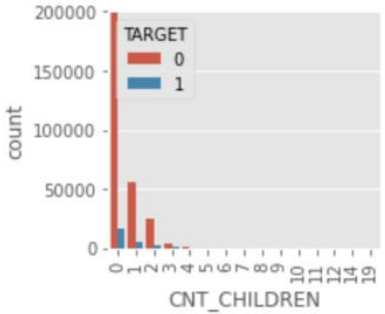
Most of the applicants own realty Most of the applicants do not own cars People not owning reality and car and have a slightly higher default rate than the people who own reality and car

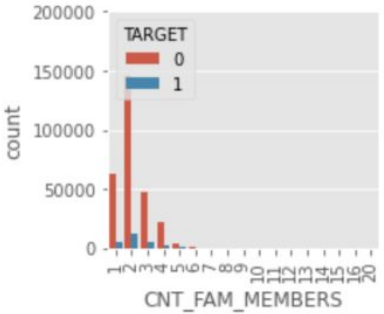


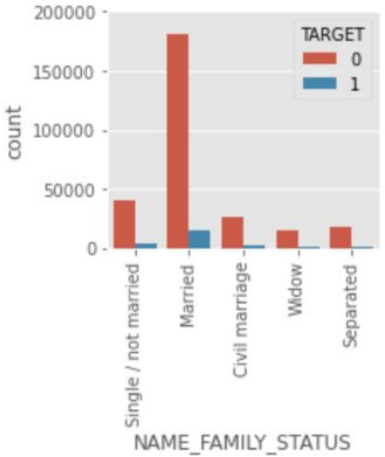
Comment:

Defaulter or not, most applicants have car age between 0-25 years. Since for both target value, trend is similar, this feature can dropped.

Family Related Info



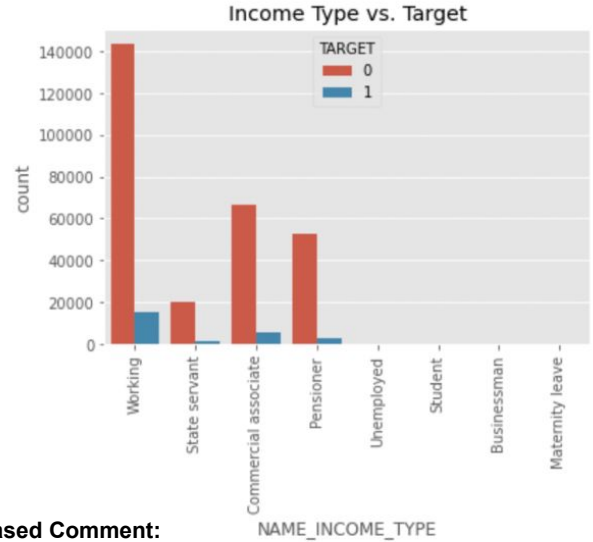




Comment:

- Default rate is highest for Civil Marriage and Single applicants
- Most of the applicants are married (and/or) no children (and/or) 2 family members.
- Applicants with relatively more number of children (and/or) family members have higher default percentage.
- For some of the cases where count children/family members is high, and the default rate is very high or very low.
- This cases cannot be taken as a conclusion as number of applicants having a large family is very low.

Occupation & Education Feature Observation



Occupation based Comment:

Most of the applicants are working.

Applicants on Maternity Leave and Unemployed has highest percentage of Defaulter

Businessman have lowest (0) percentage of Defaulter

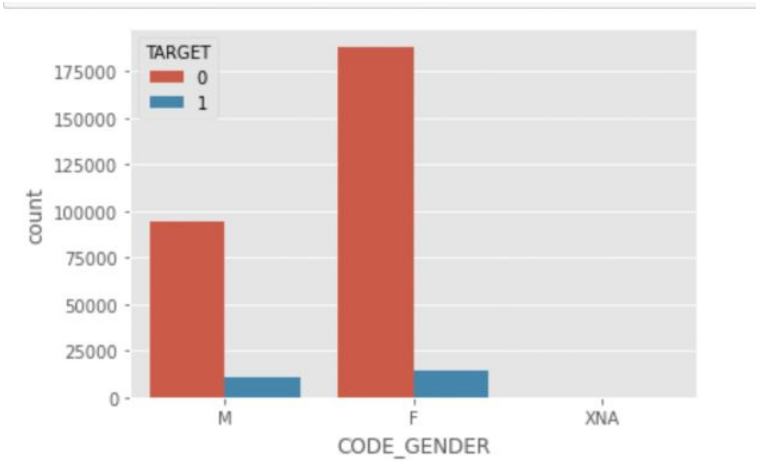
However, applicants of income type('Unemployed', 'Student', 'Businessman', 'Maternity leave') are very few in the dataset to contribute in the analysis.

	Value	Percentage of Defaulter
3	Lower secondary	10.930537
0	Secondary / secondary special	8.939929
2	Incomplete higher	8.484966
1	Higher education	5.355187
4	Academic degree	1.829268

Education based Comment:

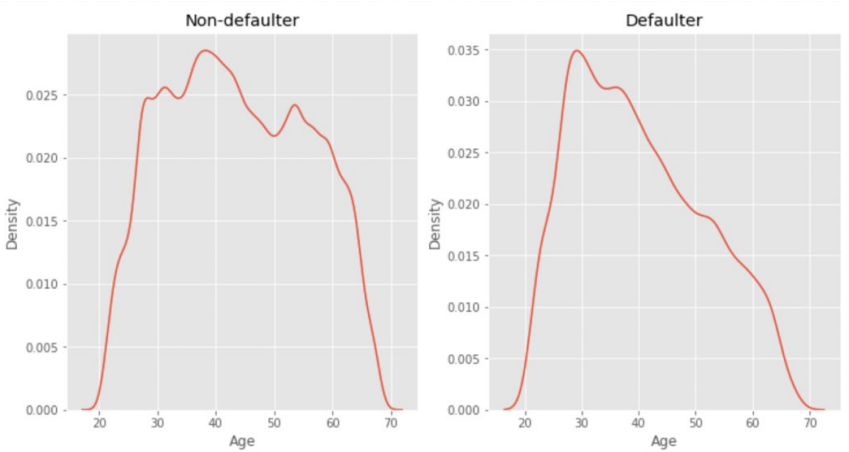
Applicants having "Lower secondary" education have highest percentage of Defaulter.

Applicant Gender and Age Feature



Comment:

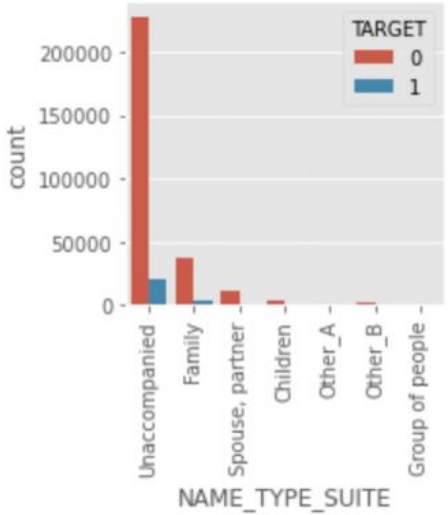
1. Female applicants are more than male applicants
2. Defaulter percentage is higher for male applicants

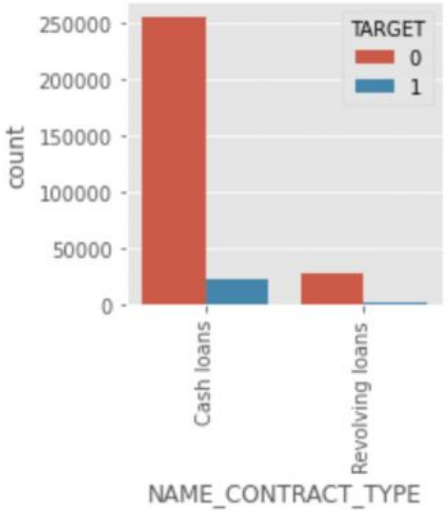


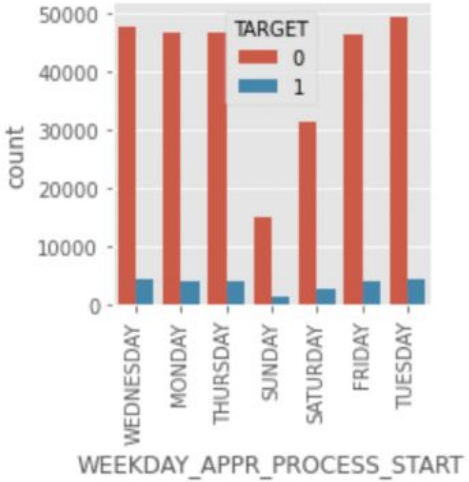
Comment:

1. People of age 30 have higher default rate.
2. Default cases are less for applicants more than 40 years old.

Applicant Insights with Target Column







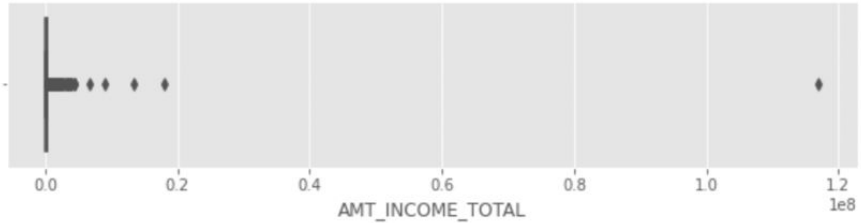
Comment:

1. Most of the applicants are unaccompanied
2. While applying for loan Number Cash loans is quite higher than Revolving Loans
3. All weekdays have similar number of applicants than weekend(Saturday and Sunday)

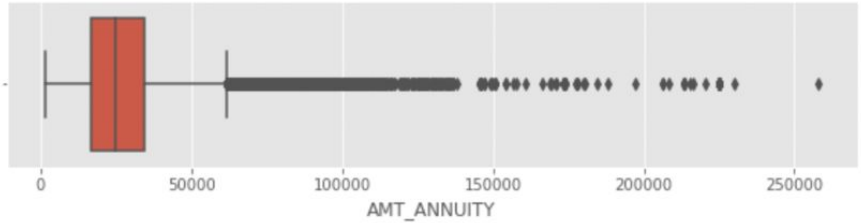
OUTLIERS

Income and Annuity

```
plt.figure(figsize=(10,2))
sns.boxplot(application_dataset['AMT_INCOME_TOTAL'])
plt.show()
```




```
plt.figure(figsize=(10,2))
sns.boxplot(application_dataset['AMT_ANNUITY'])
plt.show()
```



Boxplot is showing the outliers for income and annuity, there are few entries having very large annuity and income than others.

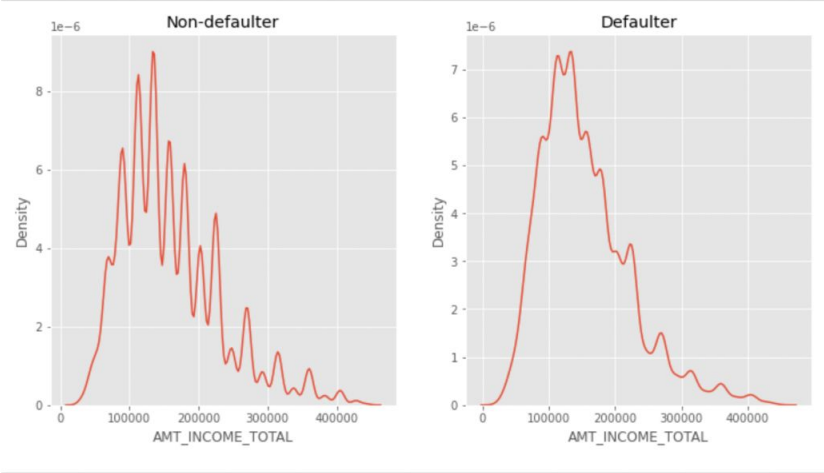
Considering this entries will mislead the average income of the entire population and further analysis.

Hence, excluding values outside 99 percentile for AMT_ANNUITY and AMT_INCOME_TOTAL

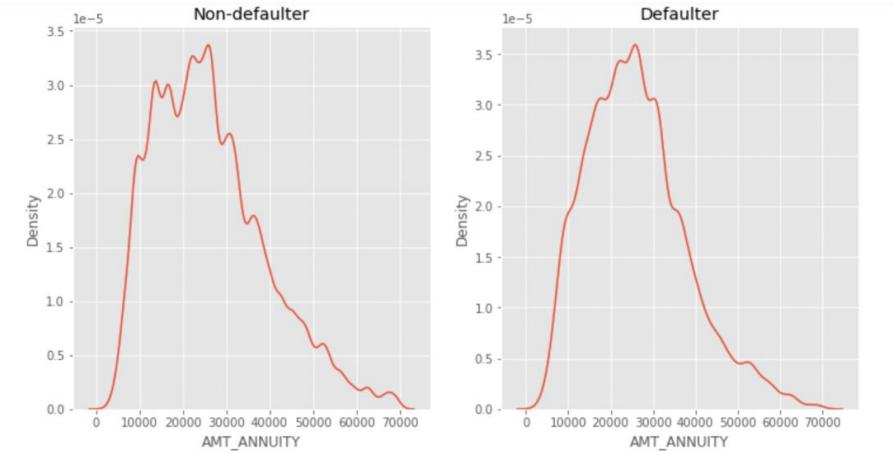
Excluding values outside 99 percentile for AMT_ANNUITY and AMT_INCOME_TOTAL

Comment:

For Defaulters and non-defaulters AMT_ANNUITY distribution is similar.

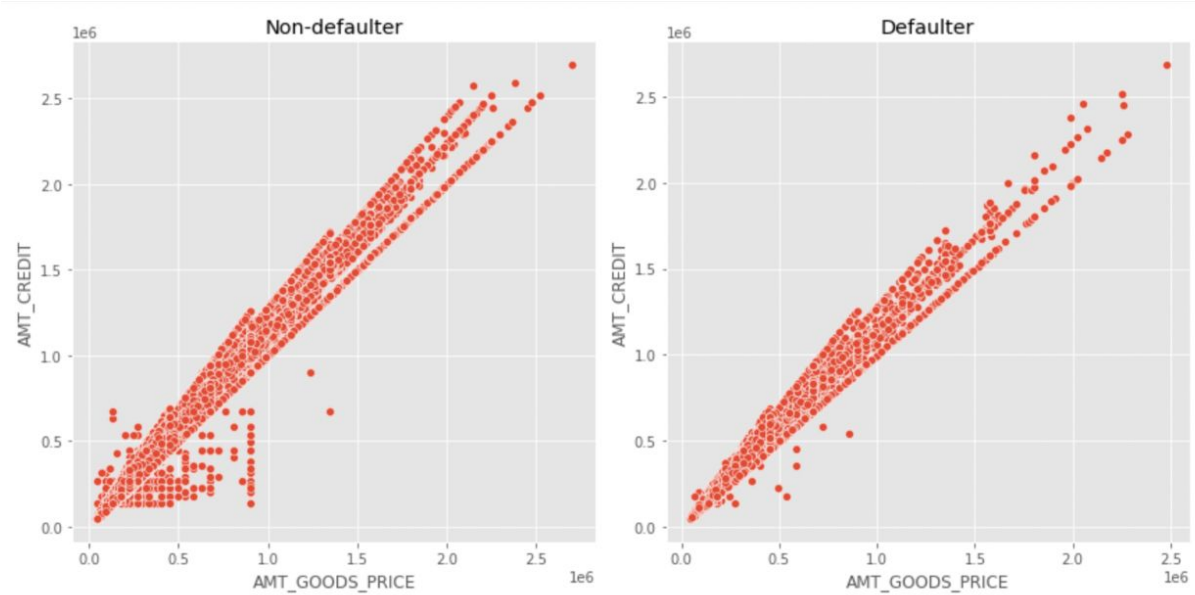


AMT_INCOME_TOTAL



AMT_ANNUITY

AMT_CREDIT and AMT_GOODS_PRICE Observation

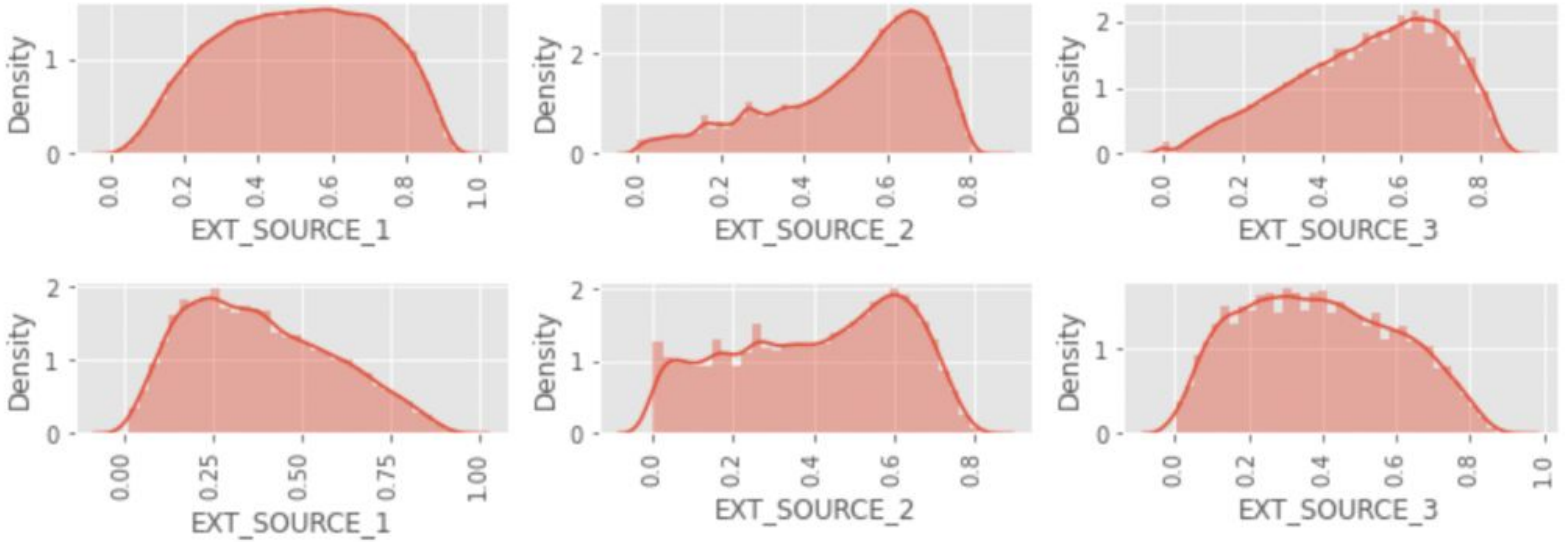


Comment:

AMT_CREDIT and AMT_GOODS_PRICE have linear relation.

For lower range of AMT_CREDIT and AMT_GOODS_PRICE, amount of defaulters is less than that of non-defaulters

EXT_SOURCE Observations



Comment:

'EXT_SOURCE_1' and 'EXT_SOURCE_3' have very different distribution for defaulters and non-defaulters.

Top 10 Correlation

Top 10 correlation for Defaulters

Listing the correlations in pair sorted in descending order

Out[299]:

SK_ID_CURR	SK_ID_CURR	1.000000
OBS_60_CNT_SOCIAL_CIRCLE	OBS_30_CNT_SOCIAL_CIRCLE	0.998269
BASEMENTAREA_AVG	BASEMENTAREA_MEDI	0.998250
COMMONAREA_MEDI	COMMONAREA_AVG	0.998107
YEARS_BUILD_MEDI	YEARS_BUILD_AVG	0.998100
NONLIVINGAPARTMENTS_AVG	NONLIVINGAPARTMENTS_MEDI	0.998075
FLOORSMIN_MEDI	FLOORSMIN_AVG	0.997825
LIVINGAPARTMENTS_MEDI	LIVINGAPARTMENTS_AVG	0.997668
FLOORSMAX_MEDI	FLOORSMAX_AVG	0.997187
NONLIVINGAPARTMENTS_MODE	NONLIVINGAPARTMENTS_MEDI	0.997032
ENTRANCES_MEDI	ENTRANCES_AVG	0.996700

dtype: float64

Top 10 correlation for Non-defaulters

Listing the correlations in pair sorted in descending order

SK_ID_CURR	SK_ID_CURR	1.000000
YEARS_BUILD_AVG	YEARS_BUILD_MEDI	0.998522
OBS_60_CNT_SOCIAL_CIRCLE	OBS_30_CNT_SOCIAL_CIRCLE	0.998508
FLOORSMIN_MEDI	FLOORSMIN_AVG	0.997202
FLOORSMAX_AVG	FLOORSMAX_MEDI	0.997018
ENTRANCES_AVG	ENTRANCES_MEDI	0.996899
ELEVATORS_AVG	ELEVATORS_MEDI	0.996161
COMMONAREA_AVG	COMMONAREA_MEDI	0.995857
LIVINGAREA_MEDI	LIVINGAREA_AVG	0.995568
APARTMENTS_AVG	APARTMENTS_MEDI	0.995163
BASEMENTAREA_MEDI	BASEMENTAREA_AVG	0.994081

dtype: float64

Top 5 important columns

Family Info: (Important driving features : 'CNT_FAM_MEMBERS', 'CNT_CHILDREN') i. Most of the applicants are married (and/or) no children (and/or) 2 family members. ii. Applicants with relatively more number of children (and/or) family members have higher default percentage. (For some of the cases where count children/family members is high, and the default rate is very high or very low. This cases cannot be considered for analysis as number of applicants having a large family is very low.)

Education and Occupation Info: (Important driving features :'NAME_INCOME_TYPE', 'OCCUPATION_TYPE') Most of the applicants are working. Applicants on Maternity Leave and Unemployed has highest percentage of Defaulter Businessman have lowest (0) percentage of Defaulter However applicants of income type('Unemployed', 'Student', 'Businessman', 'Maternity leave') are very few in the dataset to contribute in the analysis.

CODE_GENDER: Female applicants are more than male applicants Defauter percentage is higher for male applicants XNA values can be replaced with "Unknown"

DAYS_BIRTH: A derived column 'Age' from this gave useful information. People of age 25-35 have higher default rate Default cases are less for applicants more than 40 years old.

'EXT_SOURCE_1' and 'EXT_SOURCE_3' have very different distribution for defaulters and non-defaulters. This can be important features.

Summary: Analyses of Application Dataset

1. This data is highly imbalanced as number of defaulter is very less in total population.

2. 'CNT_FAM_MEMBERS', 'CNT_CHILDREN','NAME_INCOME_TYPE', 'OCCUPATION_TYPE',CODE_GENDER, 'EXT_SOURCE_1' and 'EXT_SOURCE_3' are some of the important driving factors.
3. **Documents** : Considered features 'FLAG_DOCUMENT_2','FLAG_DOCUMENT_3',...,'FLAG_DOCUMENT_21' for this segment. Majority of the applicants did not submit any documents apart from DOCUMENT_3. FLAG_DOCUMENT_3 has similar impact on defaulters and non-defaulters. Hence, these columns can be dropped.
4. **Housing**: All of the features considered have very high (47-70%) missing data percentage. Hence, all these features can be dropped.
5. Plot of 'NAME_HOUSING_TYPE' vs 'TARGET' shows that
    a. Most of the applicants live in House/Apartment
    b. Applicants living with their parents or in rented apartment have higher rate of default.
6. **Social Circle Info**: The features show similar trend for defaulters and non-defaulters, can be dropped.
7. **Regional Info**: Defaulter rate is highest when REG_REGION_NOT_WORK_REGION=0 i.e. permanent address and working address is same
8. **Contact Info** : Considered 'FLAG_MOBIL','FLAG_EMP_PHONE' etc for this segment. No impact on Target, features can be dropped.
9. **Asset Info** :
    - Most of the applicants own realty
    - Most of the applicants do not own cars
    - People not owning reality and car and have a slightly higher default rate than the people who own reality and car

PREVIOUS APPLICATION DATASET ANALYSIS

Now, Lets Analyse PREVIOUS APPLICATION Dataset

Introduction In this notebook, we have mainly focused on analyzing previous_application.csv i.e. data about previous application of an applicant.

Approach For the Exploratory data analysis, mentioned steps have been followed.

--> Import Modules

--> Read the dataset

--> Data Cleaning

Missing value handling Type Casting Fixing Rows and Columns - removing unnecessary rows/columns (through missing value handling and correlation) Handling Outliers --> **Univariate Analysis** --> **Bivariate and Multivariate Analysis**

Data Cleaning

Function_name : missingdata_percentage
Usage : Returns % of missing values for all features in a DataFrame
Arguments : dataframe
Returns : a dataframe containing categories having missing values and % of missing values in those categories

Observation:

1. There are 16 features in prev_app_df that have missing values.
2. Permanently dropping the features (RATE_INTEREST_PRIMARY and RATE_INTEREST_PRIVILEGED) as 99% data is missing.
3. Dropping rows containing missing values for the features(AMT_CREDIT and PRODUCT_COMBINATION) for very low % of missing data.
4. Dropping entries would not cause impact the analysis as percentage of missing value is very low (~2%).
5. Filling missing value as 'Unaccompanied' as most common value in previous application dataset(Imputing missing values)

Extracting the numeric features from previous application data

In [314]:
```
prev_num_df = pd.DataFrame()

for col in numeric_features:
    prev_num_df[col] = prev_appl_dataset[col]

prev_num_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1669867 entries, 0 to 1670213
Data columns (total 19 columns):
 #   Column                       Non-Null Count    Dtype
---  ------                       --------------    -----
 0   SK_ID_PREV                   1669867 non-null  int64
 1   SK_ID_CURR                   1669867 non-null  int64
 2   AMT_ANNUITY                  1297978 non-null  float64
 3   AMT_APPLICATION              1669867 non-null  float64
 4   AMT_CREDIT                   1669867 non-null  float64
 5   AMT_DOWN_PAYMENT             774370 non-null   float64
 6   AMT_GOODS_PRICE              1284699 non-null  float64
 7   HOUR_APPR_PROCESS_START      1669867 non-null  int64
 8   NFLAG_LAST_APPL_IN_DAY       1669867 non-null  int64
 9   RATE_DOWN_PAYMENT            774370 non-null   float64
 10  DAYS_DECISION                1669867 non-null  int64
 11  SELLERPLACE_AREA             1669867 non-null  int64
 12  CNT_PAYMENT                  1297983 non-null  float64
 13  DAYS_FIRST_DRAWING           997149 non-null   float64
 14  DAYS_FIRST_DUE               997149 non-null   float64
 15  DAYS_LAST_DUE_1ST_VERSION    997149 non-null   float64
 16  DAYS_LAST_DUE                997149 non-null   float64
 17  DAYS_TERMINATION             997149 non-null   float64
 18  NFLAG_INSURED_ON_APPROVAL    997149 non-null   float64
dtypes: float64(13), int64(6)
memory usage: 254.8 MB
```

Checking correlation between numeric features of previous application data

Comment:

'DAYS_LAST_DUE' and 'DAYS_TERMINATION' are highly correlated 'DAYS_FIRST_DRAWING' and 'DAYS_LAST_DUE_1st_VERSION' have high negative correlation 'AMT_ANNUITY','AMT_APPLICATION','AMT_CREDIT','AMT_GOODS_PRICE' are highly correlated The features can be removed before modelling this data, as they would cause collinearity **'DAYS_TERMINATION','DAYS_LAST_DUE_1st_VERSION','AMT_APPLICATION','AMT_CREDIT','AMT_GOODS_PRICE'** For EDA purpose we are not removing them.

'SK_ID_PREV' column is not required for analysis.

MERGE DATASET ANALYSIS

Merging only required columns of application_data with previous_application_data

Checking on the numeric data

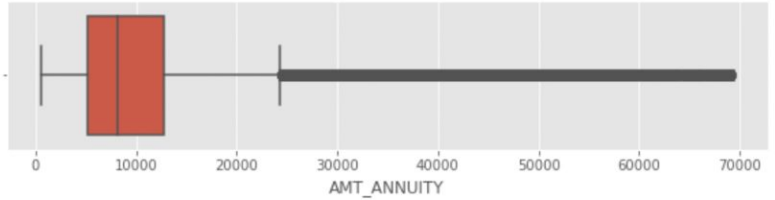
	SK_ID_CURR	TARGET	AMT_ANNUITY	AMT_APPLICATION	AMT_CREDIT	AMT_DOWN_PAYMENT	AMT_GOODS_PRICE	HOUR_APPR_PROCESS_STAF
count	1.429841e+06	1.429841e+06	1.106482e+06	1.413387e+06	1.413387e+06	6.641610e+05	1.094176e+06	1.413387e+(
mean	2.784721e+05	8.621658e-02	1.583720e+04	1.752825e+05	1.963976e+05	6.655317e+03	2.264512e+05	1.247888e+(
std	1.028026e+05	2.806837e-01	1.472491e+04	2.936432e+05	3.195033e+05	2.062030e+04	3.159376e+05	3.331533e+(
min	1.000020e+05	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	-9.000000e-01	0.000000e+00	0.000000e+(
25%	1.893620e+05	0.000000e+00	6.257880e+03	1.975500e+04	2.491200e+04	0.000000e+00	4.982850e+04	1.000000e+(
50%	2.789590e+05	0.000000e+00	1.122376e+04	7.090200e+04	8.064000e+04	1.791000e+03	1.102455e+05	1.200000e+(
75%	3.675110e+05	0.000000e+00	2.042111e+04	1.800000e+05	2.156400e+05	7.695000e+03	2.295000e+05	1.500000e+(
max	4.562550e+05	1.000000e+00	4.180581e+05	5.850000e+06	4.509688e+06	3.060045e+06	5.850000e+06	2.300000e+(

Comment: Not dropping the rest of columns with missing values, will use them for further analysis.

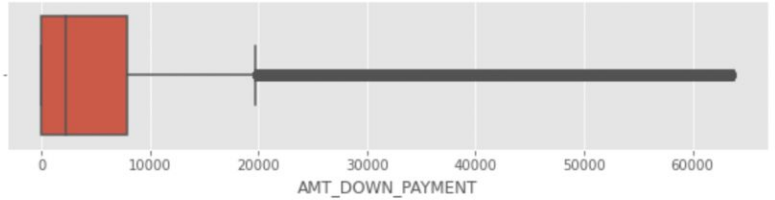
Handling Outliers

Excluding values outside 99 percentile for AMT_ANNUITY and AMT_DOWN_PAYMENT. Checking the features after updating.

```
In [325]: plt.figure(figsize=(10,2))
          sns.boxplot(prev_ap_merged['AMT_ANNUITY'])
          plt.show()
```




```
In [326]: plt.figure(figsize=(10,2))
          sns.boxplot(prev_ap_merged['AMT_DOWN_PAYMENT'])
          plt.show()
```



Checking Data Imbalance in Previous Application Data

Observation:

The applicants whose previous loans were approved are more likely to pay current loan in time, than the applicants whose previous loans were rejected.

7% of the previously approved loan applicants that defaulted in current loan

90 % of the previously refused loan applicants that were able to pay current loan

Checking Payment Status with Target column

Comment: This data is highly imbalanced as number of defaulter is very less in total population.

Merged Dataset: Univariate, Bivariate and Multivariate Analysis

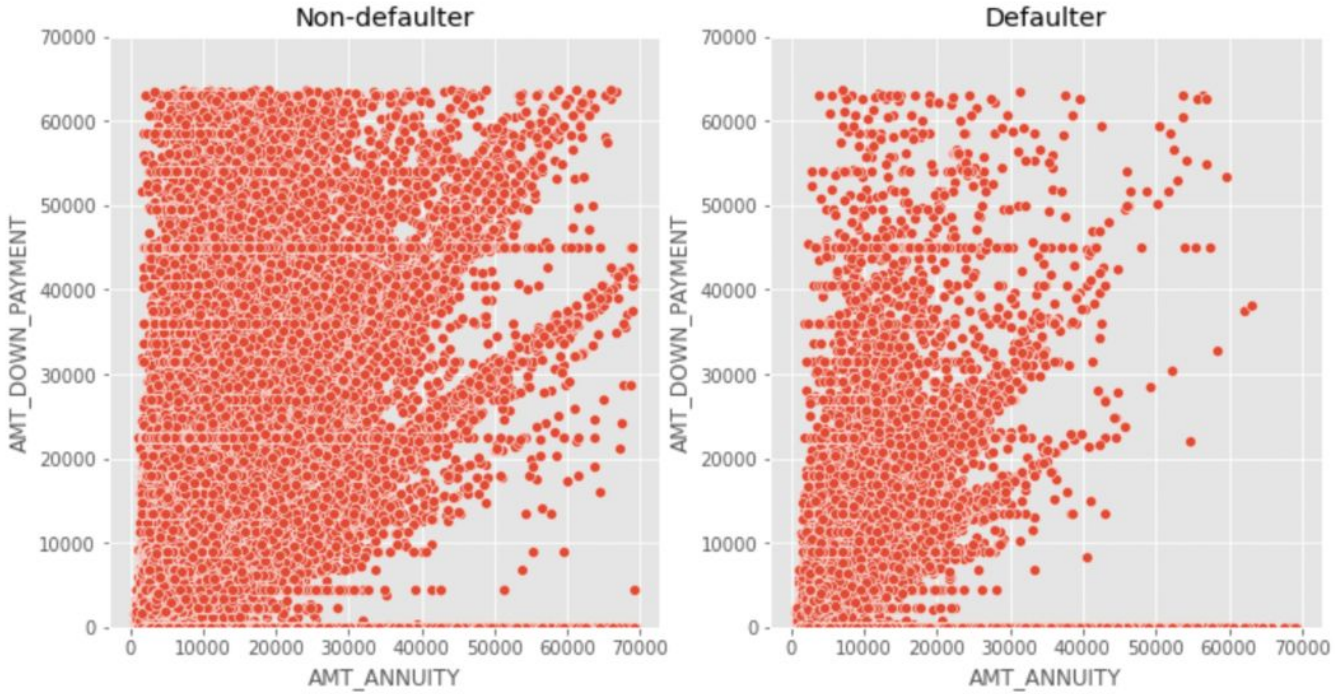
```
In [332]: print(prev_ap_merged.FLAG_LAST_APPL_PER_CONTRACT.value_counts())
          print(prev_ap_merged.NFLAG_LAST_APPL_IN_DAY.value_counts())

          Y    628193
          Name: FLAG_LAST_APPL_PER_CONTRACT, dtype: int64
          1.0    627967
          0.0       226
          Name: NFLAG_LAST_APPL_IN_DAY, dtype: int64
```

Comment:

'FLAG_LAST_APPL_PER_CONTRACT' can be dropped for having fixed value in all entries.

'NFLAG_LAST_APPL_IN_DAY' can be dropped for having highly imbalance data.

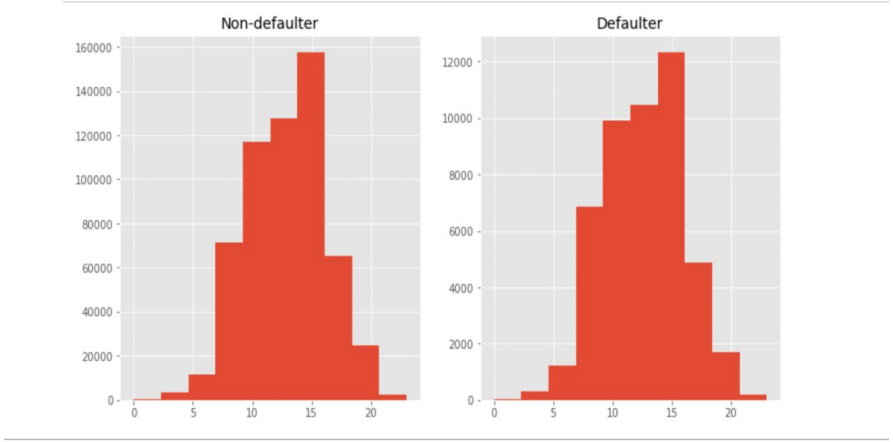
Analysis of Numeric Features of Previous Application Data

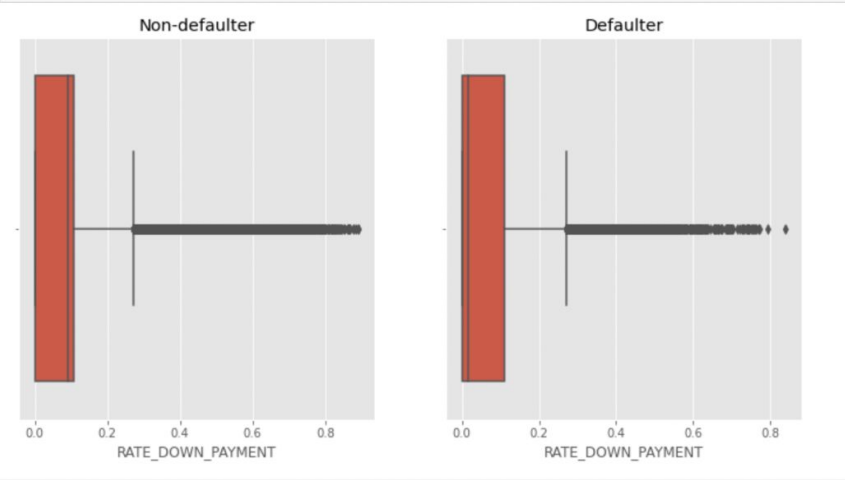


Comment:

Number of defaulters are less for larger amount of annuity of previous application. For higher down payment, defaulter cases are less.

Analysing Impact of HOUR_APPR_PROCESS_START and RATE_DOWN_PAYMENT with Target Column





Comment:

Most of the loans are applied around 15:00 hours.

This feature is does not have visible impact on TARGET variable

Comment:

1. The non-defaulter for down payment is a continuous value. The non-defaulters are equal for almost all rates.
2. In case of defaulter, we can see few outliers near and above 0.8 rate for down payment. That means, there are few exceptional values for defaulters with high rate of interest.

Checking the quantile values for exact percentile

In [337]:
```
# Percentile of RATE_DOWN_PAYMENT for non-defaulters

prev_ap_merged[prev_ap_merged["TARGET"] == 0]['RATE_DOWN_PAYMENT'].quantile([0.5, 0.7, 0.9, 0.95, 0.99])
```

Out[337]:
```
0.50    0.091255
0.70    0.108909
0.90    0.211895
0.95    0.282832
0.99    0.498074
Name: RATE_DOWN_PAYMENT, dtype: float64
```

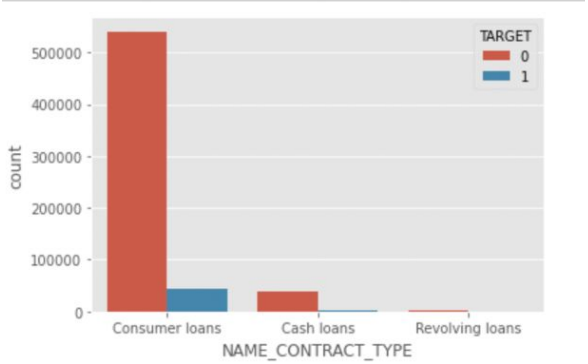
In [338]:
```
# Percentile of RATE_DOWN_PAYMENT for defaulters

prev_ap_merged[prev_ap_merged["TARGET"] == 1]['RATE_DOWN_PAYMENT'].quantile([0.5, 0.7, 0.9, 0.95, 0.99])
```

Out[338]:
```
0.50    0.017238
0.70    0.104260
0.90    0.199685
0.95    0.232643
0.99    0.454294
Name: RATE_DOWN_PAYMENT, dtype: float64
```

Comment:

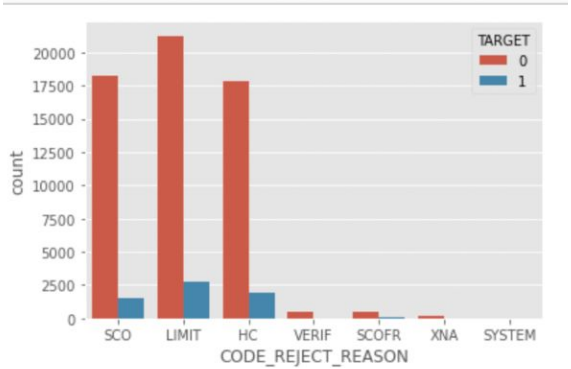
For those who had lower rate of down payment in previous application, cases of default are higher.

Analysis of Categorical Features of Previous Application Data



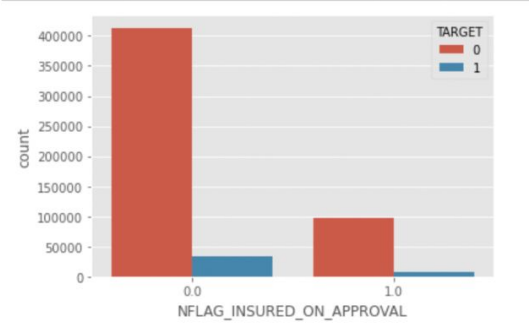
Comment:

Highest number of loans are applied for Consumer Loans



Comment:

As seen in the above plot, 'SCO', 'LIMIT' and 'HC' are the most common reason of rejection.



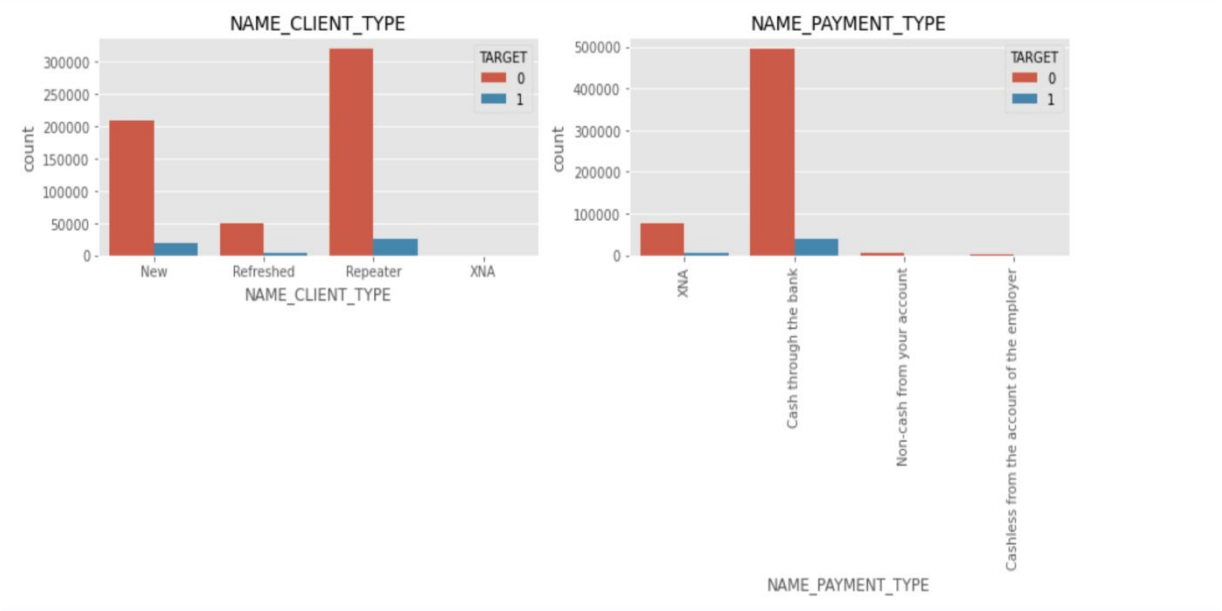
Comment:

Most of the people did not request insurance during previous loan application.

Analysis of NAME_CLIENT_TYPE and NAME_PAYMENT_TYPE with TARGET Column

Comment:

Most of the applicants are repeater. 'Cash through the bank' is the most frequently used payment method



Analysing NAME_GOODS_CATEGORY with Target Column

We calculated the percentage of defaulter in total applicants having the same label of the category.For example: In 100 applicants, whose 'NAME_GOODS_CATEGORY' is Insurance We found that approx. 10 people have payment difficulty.

Comment:

Highest percentage of default cases are for the applicants who previously applied for Insurance and Vehicles

[ ]:

	Value	Percentage of Defaulter
23	Insurance	10.526316
0	Vehicles	10.25741
14	Jewelry	9.124951
17	Auto Accessories	9.029763
3	Mobile	8.615336
15	Office Appliances	8.307692
8	Computers	8.074335
20	Weapon	8.064516
21	Direct Sales	8.024691
5	Audio/Video	7.698706
7	Photo / Cinema Equipment	7.455
18	Sport and Leisure	7.35415
2	Consumer Electronics	7.066548
4	Construction Materials	6.97832
9	XNA	6.885879
24	Additional Service	6.730769
6	Gardening	6.723063
11	Homewares	6.706444
19	Medicine	6.196747
25	Education	5.882353
1	Furniture	5.860781
10	Clothing and Accessories	5.807427
13	Other	5.765921
12	Medical Supplies	5.56419
16	Tourism	4.444444
22	Fitness	4.268293
26	Animals	0.0

Conclusion of analysis of more Categorial Features with Target column

1. There are feature columns in the dataset that are highly correlated to each other. Which means both will have similar impact on the target value. Those features can be removed before feeding this data to a model to avoid collinearity.
2. Feature columns with 50% or more missing data can be dropped.
3. Following columns should be converted to integer. DAYS_FIRST_DRAWING float64 DAYS_FIRST_DUE float64 DAYS_LAST_DUE_1ST_VERSION float64 DAYS_LAST_DUE float64 DAYS_TERMINATION float64
4. This categorical column has only 0 and 1 and hence can be converted into integer column. NFLAG_INSURED_ON_APPROVAL float64.
5. This dataset is highly imbalanced
6. The applicants whose previous loans were approved are more likely to pay current loan in time, than the applicants whose previous loans were rejected. **NAME_CONTRACT_STATUS** is an important feature.
7. 7% of the previously approved loan applicants that defaulted in current loan
8. 90 % of the previously refused loan applicants that were able to pay current loan
9. 'SCO', 'LIMIT' and 'HC' are the most common reason of rejection.
10. Most of the people did not request insurance during previous loan application.
11. For "Cards" defaulter percentage is highest (17%). **'NAME_PORTFOLIO'** is an important feature for analyzing 'TARGET' variable.
12. 15% loan application defaulted for AP+ (Cash Loan). **'CHANNEL_TYPE'** is an important feature for analyzing 'TARGET' variable.
13. Highest percentage (17%) of default cases is for 'Card Street'. **'PRODUCT_COMBINATION'** is an important driving factor.

Thank You!