House Loan Data Analysis

Aurore Prevot

1) Importation of the dataset

```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
%matplotlib inline
```

```
data = pd.read_csv('loan_data (1).csv')
data.head()
```

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

5 rows × 122 columns

```
data.shape
```

```
(307511, 122)
```

```
data.iloc[:, :61].info()
```

```
 6   CNT_CHILDREN                 307511 non-null  int64

 7   AMT_INCOME_TOTAL             307511 non-null  float64
 8   AMT_CREDIT                   307511 non-null  float64
 9   AMT_ANNUITY                  307499 non-null  float64
 10  AMT_GOODS_PRICE              307233 non-null  float64
 11  NAME_TYPE_SUITE              306219 non-null  object
 12  NAME_INCOME_TYPE             307511 non-null  object
 13  NAME_EDUCATION_TYPE          307511 non-null  object
 14  NAME_FAMILY_STATUS           307511 non-null  object
 15  NAME_HOUSING_TYPE            307511 non-null  object
 16  REGION_POPULATION_RELATIVE   307511 non-null  float64
 17  DAYS_BIRTH                   307511 non-null  int64
```

```
 18  DAYS_EMPLOYED                307511 non-null  int64
 19  DAYS_REGISTRATION            307511 non-null  float64
 20  DAYS_ID_PUBLISH              307511 non-null  int64
 21  OWN_CAR_AGE                  104582 non-null  float64
 22  FLAG_MOBIL                   307511 non-null  int64
 23  FLAG_EMP_PHONE               307511 non-null  int64
 24  FLAG_WORK_PHONE              307511 non-null  int64
 25  FLAG_CONT_MOBILE             307511 non-null  int64
 26  FLAG_PHONE                   307511 non-null  int64
 27  FLAG_EMAIL                   307511 non-null  int64
 28  OCCUPATION_TYPE              211120 non-null  object
 29  CNT_FAM_MEMBERS              307509 non-null  float64
 30  REGION_RATING_CLIENT         307511 non-null  int64
 31  REGION_RATING_CLIENT_W_CITY  307511 non-null  int64
 32  WEEKDAY_APPR_PROCESS_START   307511 non-null  object
 33  HOUR_APPR_PROCESS_START      307511 non-null  int64
 34  REG_REGION_NOT_LIVE_REGION   307511 non-null  int64
 35  REG_REGION_NOT_WORK_REGION   307511 non-null  int64
 36  LIVE_REGION_NOT_WORK_REGION  307511 non-null  int64
 37  REG_CITY_NOT_LIVE_CITY       307511 non-null  int64
 38  REG_CITY_NOT_WORK_CITY       307511 non-null  int64
 39  LIVE_CITY_NOT_WORK_CITY      307511 non-null  int64
 40  ORGANIZATION_TYPE            307511 non-null  object
 41  EXT_SOURCE_1                 134133 non-null  float64
 42  EXT_SOURCE_2                 306851 non-null  float64
 43  EXT_SOURCE_3                 246546 non-null  float64
 44  APARTMENTS_AVG               151450 non-null  float64
 45  BASEMENTAREA_AVG             127568 non-null  float64
 46  YEARS_BEGINEXPLUATATION_AVG  157504 non-null  float64
 47  YEARS_BUILD_AVG              103023 non-null  float64
 48  COMMONAREA_AVG               92646 non-null   float64
 49  ELEVATORS_AVG                143620 non-null  float64
 50  ENTRANCES_AVG                152683 non-null  float64
 51  FLOORSMAX_AVG                154491 non-null  float64
 52  FLOORSMIN_AVG                98869 non-null   float64
 53  LANDAREA_AVG                 124921 non-null  float64
 54  LIVINGAPARTMENTS_AVG         97312 non-null   float64
 55  LIVINGAREA_AVG               153161 non-null  float64
 56  NONLIVINGAPARTMENTS_AVG      93997 non-null   float64
 57  NONLIVINGAREA_AVG            137829 non-null  float64
 58  APARTMENTS_MODE              151450 non-null  float64
 59  BASEMENTAREA_MODE            127568 non-null  float64
 60  YEARS_BEGINEXPLUATATION_MODE 157504 non-null  float64
dtypes: float64(28), int64(21), object(12)
memory usage: 143.1+ MB
```

```
data.iloc[:, 61:].info()
```

```
 6   LANDAREA_MODE                124921 non-null  float64
 7   LIVINGAPARTMENTS_MODE        97312 non-null   float64

 8   LIVINGAREA_MODE              153161 non-null  float64
 9   NONLIVINGAPARTMENTS_MODE     93997 non-null   float64
 10  NONLIVINGAREA_MODE           137829 non-null  float64
 11  APARTMENTS_MEDI              151450 non-null  float64
 12  BASEMENTAREA_MEDI            127568 non-null  float64
 13  YEARS_BEGINEXPLUATATION_MEDI 157504 non-null  float64
 14  YEARS_BUILD_MEDI             103023 non-null  float64
 15  COMMONAREA_MEDI              92646 non-null   float64
 16  ELEVATORS_MEDI               143620 non-null  float64
 17  ENTRANCES_MEDI               152683 non-null  float64
 18  FLOORSMAX_MEDI               154491 non-null  float64
```

```
 19  FLOORSMIN_MEDI                98869 non-null   float64
 20  LANDAREA_MEDI                 124921 non-null  float64
 21  LIVINGAPARTMENTS_MEDI         97312 non-null   float64
 22  LIVINGAREA_MEDI               153161 non-null  float64
 23  NONLIVINGAPARTMENTS_MEDI      93997 non-null   float64
 24  NONLIVINGAREA_MEDI            137829 non-null  float64
 25  FONDKAPREMONT_MODE            97216 non-null   object
 26  HOUSETYPE_MODE                153214 non-null  object
 27  TOTALAREA_MODE                159080 non-null  float64
 28  WALLSMATERIAL_MODE            151170 non-null  object
 29  EMERGENCYSTATE_MODE           161756 non-null  object
 30  OBS_30_CNT_SOCIAL_CIRCLE      306490 non-null  float64
 31  DEF_30_CNT_SOCIAL_CIRCLE      306490 non-null  float64
 32  OBS_60_CNT_SOCIAL_CIRCLE      306490 non-null  float64
 33  DEF_60_CNT_SOCIAL_CIRCLE      306490 non-null  float64
 34  DAYS_LAST_PHONE_CHANGE        307510 non-null  float64
 35  FLAG_DOCUMENT_2               307511 non-null  int64
 36  FLAG_DOCUMENT_3               307511 non-null  int64
 37  FLAG_DOCUMENT_4               307511 non-null  int64
 38  FLAG_DOCUMENT_5               307511 non-null  int64
 39  FLAG_DOCUMENT_6               307511 non-null  int64
 40  FLAG_DOCUMENT_7               307511 non-null  int64
 41  FLAG_DOCUMENT_8               307511 non-null  int64
 42  FLAG_DOCUMENT_9               307511 non-null  int64
 43  FLAG_DOCUMENT_10              307511 non-null  int64
 44  FLAG_DOCUMENT_11              307511 non-null  int64
 45  FLAG_DOCUMENT_12              307511 non-null  int64
 46  FLAG_DOCUMENT_13              307511 non-null  int64
 47  FLAG_DOCUMENT_14              307511 non-null  int64
 48  FLAG_DOCUMENT_15              307511 non-null  int64
 49  FLAG_DOCUMENT_16              307511 non-null  int64
 50  FLAG_DOCUMENT_17              307511 non-null  int64
 51  FLAG_DOCUMENT_18              307511 non-null  int64
 52  FLAG_DOCUMENT_19              307511 non-null  int64
 53  FLAG_DOCUMENT_20              307511 non-null  int64
 54  FLAG_DOCUMENT_21              307511 non-null  int64
 55  AMT_REQ_CREDIT_BUREAU_HOUR    265992 non-null  float64
 56  AMT_REQ_CREDIT_BUREAU_DAY     265992 non-null  float64
 57  AMT_REQ_CREDIT_BUREAU_WEEK    265992 non-null  float64
 58  AMT_REQ_CREDIT_BUREAU_MON     265992 non-null  float64
 59  AMT_REQ_CREDIT_BUREAU_QRT     265992 non-null  float64
 60  AMT_REQ_CREDIT_BUREAU_YEAR    265992 non-null  float64
dtypes: float64(37), int64(20), object(4)
memory usage: 143.1+ MB
```

2) Checking for null values and Treatment of null values

```
data.iloc[:, :60].isnull().sum()
```

```
CODE_GENDER                       0
FLAG_OWN_CAR                      0
FLAG_OWN_REALTY                   0
CNT_CHILDREN                      0
AMT_INCOME_TOTAL                  0
AMT_CREDIT                        0
AMT_ANNUITY                      12
AMT_GOODS_PRICE                 278
NAME_TYPE_SUITE                1292
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
OWN_CAR_AGE                    202929
FLAG_MOBIL                          0
FLAG_EMP_PHONE                      0
FLAG_WORK_PHONE                     0
FLAG_CONT_MOBILE                    0
FLAG_PHONE                          0
FLAG_EMAIL                          0
OCCUPATION_TYPE                 96391
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
EXT_SOURCE_1                   173378
EXT_SOURCE_2                      660
EXT_SOURCE_3                    60965
APARTMENTS_AVG                 156061
BASEMENTAREA_AVG               179943
YEARS_BEGINEXPLUATATION_AVG    150007
YEARS_BUILD_AVG                204488
COMMONAREA_AVG                 214865
ELEVATORS_AVG                  163891
ENTRANCES_AVG                  154828
FLOORSMAX_AVG                  153020
FLOORSMIN_AVG                  208642
LANDAREA_AVG                   182590
LIVINGAPARTMENTS_AVG           210199
LIVINGAREA_AVG                 154350
NONLIVINGAPARTMENTS_AVG        213514
NONLIVINGAREA_AVG              169682
APARTMENTS_MODE                156061
BASEMENTAREA_MODE              179943
dtype: int64
```

```
data.iloc[:, 61:120].isnull().sum()
```

```
ELEVATORS_MODE                 163891
ENTRANCES_MODE                 154828
FLOORSMAX_MODE                 153020
FLOORSMIN_MODE                 208642
LANDAREA_MODE                  182590
LIVINGAPARTMENTS_MODE          210199
LIVINGAREA_MODE                154350
NONLIVINGAPARTMENTS_MODE       213514
NONLIVINGAREA_MODE             169682
APARTMENTS_MEDI                156061
BASEMENTAREA_MEDI              179943
YEARS_BEGINEXPLUATATION_MEDI   150007
```

```
YEARS_BUILD_MEDI               204488
COMMONAREA_MEDI                214865
ELEVATORS_MEDI                 163891
ENTRANCES_MEDI                 154828
FLOORSMAX_MEDI                 153020
FLOORSMIN_MEDI                 208642
LANDAREA_MEDI                  182590
LIVINGAPARTMENTS_MEDI          210199
LIVINGAREA_MEDI                154350
NONLIVINGAPARTMENTS_MEDI       213514
NONLIVINGAREA_MEDI             169682
FONDKAPREMONT_MODE             210295
HOUSETYPE_MODE                 154297
TOTALAREA_MODE                 148431
WALLSMATERIAL_MODE             156341
EMERGENCYSTATE_MODE            145755
OBS_30_CNT_SOCIAL_CIRCLE         1021
DEF_30_CNT_SOCIAL_CIRCLE         1021
OBS_60_CNT_SOCIAL_CIRCLE         1021
DEF_60_CNT_SOCIAL_CIRCLE         1021
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
AMT_REQ_CREDIT_BUREAU_HOUR      41519
AMT_REQ_CREDIT_BUREAU_DAY       41519
AMT_REQ_CREDIT_BUREAU_WEEK      41519
AMT_REQ_CREDIT_BUREAU_MON       41519
dtype: int64
```

```
column_names = data.columns
column_names
```

```
Index(['SK_ID_CURR', 'TARGET', 'NAME_CONTRACT_TYPE', 'CODE_GENDER',
       'FLAG_OWN_CAR', 'FLAG_OWN_REALTY', 'CNT_CHILDREN', 'AMT_INCOME_TOTAL',
       'AMT_CREDIT', 'AMT_ANNUITY',
       ...
       'FLAG_DOCUMENT_18', 'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_HOUR',
       'AMT_REQ_CREDIT_BUREAU_DAY', 'AMT_REQ_CREDIT_BUREAU_WEEK',
       'AMT_REQ_CREDIT_BUREAU_MON', 'AMT_REQ_CREDIT_BUREAU_QRT',
       'AMT_REQ_CREDIT_BUREAU_YEAR'],
      dtype='object', length=122)
```

```
def num_column_with_null(dataframe):
  column_name = dataframe.columns
  columns_with_null = []
  for column in column_name:
    if dataframe[column].isnull().sum() !=0:
      columns_with_null.append(column)
  return len(columns_with_null), columns_with_null


num_column_with_null(data)
```

```
       'BASEMENTAREA_AVG',
       'YEARS_BEGINEXPLUATATION_AVG',
       'YEARS_BUILD_AVG',
       'COMMONAREA_AVG',
       'ELEVATORS_AVG',
       'ENTRANCES_AVG',
       'FLOORSMAX_AVG',
       'FLOORSMIN_AVG',
       'LANDAREA_AVG',
       'LIVINGAPARTMENTS_AVG',
       'LIVINGAREA_AVG',
       'NONLIVINGAPARTMENTS_AVG',
       'NONLIVINGAREA_AVG',
       'APARTMENTS_MODE',
       'BASEMENTAREA_MODE',
       'YEARS_BEGINEXPLUATATION_MODE',
       'YEARS_BUILD_MODE',
       'COMMONAREA_MODE',
       'ELEVATORS_MODE',
       'ENTRANCES_MODE',
       'FLOORSMAX_MODE',
       'FLOORSMIN_MODE',
       'LANDAREA_MODE',
       'LIVINGAPARTMENTS_MODE',
       'LIVINGAREA_MODE',
       'NONLIVINGAPARTMENTS_MODE',
       'NONLIVINGAREA_MODE',
       'APARTMENTS_MEDI',
       'BASEMENTAREA_MEDI',
       'YEARS_BEGINEXPLUATATION_MEDI',
       'YEARS_BUILD_MEDI',
       'COMMONAREA_MEDI',
       'ELEVATORS_MEDI',
       'ENTRANCES_MEDI',
       'FLOORSMAX_MEDI',
       'FLOORSMIN_MEDI',
       'LANDAREA_MEDI',
       'LIVINGAPARTMENTS_MEDI',
       'LIVINGAREA_MEDI',
       'NONLIVINGAPARTMENTS_MEDI',
       'NONLIVINGAREA_MEDI',
       'FONDKAPREMONT_MODE',
       'HOUSETYPE_MODE',
       'TOTALAREA_MODE',
       'WALLSMATERIAL_MODE',
       'EMERGENCYSTATE_MODE',
       'OBS_30_CNT_SOCIAL_CIRCLE',
       'DEF_30_CNT_SOCIAL_CIRCLE',
       'OBS_60_CNT_SOCIAL_CIRCLE',
       'DEF_60_CNT_SOCIAL_CIRCLE',
```

```
       'DAYS_LAST_PHONE_CHANGE',
       'AMT_REQ_CREDIT_BUREAU_HOUR',
       'AMT_REQ_CREDIT_BUREAU_DAY',
       'AMT_REQ_CREDIT_BUREAU_WEEK',
       'AMT_REQ_CREDIT_BUREAU_MON',
       'AMT_REQ_CREDIT_BUREAU_QRT',
       'AMT_REQ_CREDIT_BUREAU_YEAR'])
```

Observation: *Only 67 columns have null values.*

```
col_with_50_null_value = []
for column in num_column_with_null(data)[1]:
  if data[column].isnull().sum()/data.shape[0] >= 0.5:
    col_with_50_null_value.append(column)

print(len(col_with_50_null_value))
col_with_50_null_value
```

```
    41
    ['OWN_CAR_AGE',
     'EXT_SOURCE_1',
     'APARTMENTS_AVG',
     'BASEMENTAREA_AVG',
     'YEARS_BUILD_AVG',
     'COMMONAREA_AVG',
     'ELEVATORS_AVG',
     'ENTRANCES_AVG',
     'FLOORSMIN_AVG',
     'LANDAREA_AVG',
     'LIVINGAPARTMENTS_AVG',
     'LIVINGAREA_AVG',
     'NONLIVINGAPARTMENTS_AVG',
     'NONLIVINGAREA_AVG',
     'APARTMENTS_MODE',
     'BASEMENTAREA_MODE',
     'YEARS_BUILD_MODE',
     'COMMONAREA_MODE',
     'ELEVATORS_MODE',
     'ENTRANCES_MODE',
     'FLOORSMIN_MODE',
     'LANDAREA_MODE',
     'LIVINGAPARTMENTS_MODE',
     'LIVINGAREA_MODE',
     'NONLIVINGAPARTMENTS_MODE',
     'NONLIVINGAREA_MODE',
     'APARTMENTS_MEDI',
     'BASEMENTAREA_MEDI',
     'YEARS_BUILD_MEDI',
     'COMMONAREA_MEDI',
     'ELEVATORS_MEDI',
     'ENTRANCES_MEDI',
     'FLOORSMIN_MEDI',
     'LANDAREA_MEDI',
     'LIVINGAPARTMENTS_MEDI',
     'LIVINGAREA_MEDI',
     'NONLIVINGAPARTMENTS_MEDI',
     'NONLIVINGAREA_MEDI',
     'FONDKAPREMONT_MODE',
```

```
 'HOUSETYPE_MODE',
 'WALLSMATERIAL_MODE']
```

```
data['TARGET'].value_counts(normalize=True)
```

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

Observation: *41 columns have more than 50% of missing values so they will be dropped.*

```
data_no_null = data.drop(col_with_50_null_value, axis=1)
data_no_null.shape
```

```
(307511, 81)
```

```
data_no_null['TARGET'].value_counts(normalize=True)
```

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

Observation: *Dropping the 41 columns didn't change the distribution of the target column.*

```
num_column_with_null(data_no_null)
```

```
(26,
 ['AMT_ANNUITY',
  'AMT_GOODS_PRICE',
  'NAME_TYPE_SUITE',
  'OCCUPATION_TYPE',
  'CNT_FAM_MEMBERS',
  'EXT_SOURCE_2',
  'EXT_SOURCE_3',
  'YEARS_BEGINEXPLUATATION_AVG',
  'FLOORSMAX_AVG',
  'YEARS_BEGINEXPLUATATION_MODE',
  'FLOORSMAX_MODE',
  'YEARS_BEGINEXPLUATATION_MEDI',
  'FLOORSMAX_MEDI',
  'TOTALAREA_MODE',
  'EMERGENCYSTATE_MODE',
  'OBS_30_CNT_SOCIAL_CIRCLE',
  'DEF_30_CNT_SOCIAL_CIRCLE',
  'OBS_60_CNT_SOCIAL_CIRCLE',
  'DEF_60_CNT_SOCIAL_CIRCLE',
  'DAYS_LAST_PHONE_CHANGE',
  'AMT_REQ_CREDIT_BUREAU_HOUR',
  'AMT_REQ_CREDIT_BUREAU_DAY',
  'AMT_REQ_CREDIT_BUREAU_WEEK',
  'AMT_REQ_CREDIT_BUREAU_MON',
  'AMT_REQ_CREDIT_BUREAU_QRT',
  'AMT_REQ_CREDIT_BUREAU_YEAR'])
```

```
for col in num_column_with_null(data_no_null)[1]:
```

```
if data_no_null[col].dtypes == "float64":
  print(f"{col} has {data_no_null[col].isnull().sum()} null values.")
  print(f"Value counts:\n{data_no_null[col].value_counts()}\n\n")
```

```
16.0        23
17.0        14
18.0         6

19.0         3
24.0         1
23.0         1
27.0         1
22.0         1
Name: AMT_REQ_CREDIT_BUREAU_MON, dtype: int64


AMT_REQ_CREDIT_BUREAU_QRT has 41519 null values.
Value counts:
0.0      215417
1.0       33862
2.0       14412
3.0        1717
4.0         476
5.0          64
6.0          28
8.0           7
7.0           7
261.0         1
19.0          1
Name: AMT_REQ_CREDIT_BUREAU_QRT, dtype: int64


AMT_REQ_CREDIT_BUREAU_YEAR has 41519 null values.
Value counts:
0.0     71801
1.0     63405
2.0     50192
3.0     33628
4.0     20714
5.0     12052
6.0      6967
7.0      3869
8.0      2127
9.0      1096
11.0       31
12.0       30
10.0       22
13.0       19
14.0       10
17.0        7
15.0        6
19.0        4
18.0        4
16.0        3
25.0        1
23.0        1
22.0        1
21.0        1
20.0        1
Name: AMT_REQ_CREDIT_BUREAU_YEAR, dtype: int64
```

Observations: *From the results above we will perform the actions below for the numerical features. Missing values will be replaced by:*

- *the mean for 'AMT_ANNUITY', 'AMT_GOODS_PRICE', 'EXT_SOURCE_2', 'EXT_SOURCE_3', 'YEARS_BEGINEXPLUATATION_AVG', 'FLOORSMAX_AVG', 'YEARS_BEGINEXPLUATATION_MODE', 'FLOORSMAX_MODE', 'YEARS_BEGINEXPLUATATION_MEDI', 'FLOORSMAX_MEDI', 'TOTALAREA_MODE', 'DAYS_LAST_PHONE_CHANGE'.*
- *the mode for 'CNT_FAM_MEMBERS', 'OBS_30_CNT_SOCIAL_CIRCLE', 'DEF_30_CNT_SOCIAL_CIRCLE', 'OBS_60_CNT_SOCIAL_CIRCLE', 'DEF_60_CNT_SOCIAL_CIRCLE', 'AMT_REQ_CREDIT_BUREAU_HOUR', 'AMT_REQ_CREDIT_BUREAU_DAY', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'AMT_REQ_CREDIT_BUREAU_MON', 'AMT_REQ_CREDIT_BUREAU_QRT'.*
- *the median for 'AMT_REQ_CREDIT_BUREAU_YEAR'.*

```
data_no_null['AMT_ANNUITY'].mean()
```

```
27108.573909183444
```

```
data_no_null['CNT_FAM_MEMBERS'].mode()[0]
```

```
2.0
```

```
data_no_null['AMT_REQ_CREDIT_BUREAU_YEAR'].median()
```

```
1.0
```

```
mean_col = ['AMT_ANNUITY', 'AMT_GOODS_PRICE', 'EXT_SOURCE_2', 'EXT_SOURCE_3', 'YEARS_BEGINEXPLU
for col in mean_col:
  data_no_null[col].fillna(data_no_null[col].mean(), inplace=True)

mode_col = ['CNT_FAM_MEMBERS', 'OBS_30_CNT_SOCIAL_CIRCLE', 'DEF_30_CNT_SOCIAL_CIRCLE', 'OBS_60_
for col in mode_col:
  data_no_null[col].fillna(data_no_null[col].mode()[0], inplace=True)

data_no_null['AMT_REQ_CREDIT_BUREAU_YEAR'].fillna(data_no_null['AMT_REQ_CREDIT_BUREAU_YEAR'].me

num_column_with_null(data_no_null)
```

```
(3, ['NAME_TYPE_SUITE', 'OCCUPATION_TYPE', 'EMERGENCYSTATE_MODE'])
```

```
print(f"'NAME_TYPE_SUITE' has {data_no_null['NAME_TYPE_SUITE'].isnull().sum()} null values.")
print(f"Value counts: \n{data_no_null['NAME_TYPE_SUITE'].value_counts()}\n\n")

print(f"'OCCUPATION_TYPE' has {data_no_null['OCCUPATION_TYPE'].isnull().sum()} null values.")
print(f"Value counts: \n{data_no_null['OCCUPATION_TYPE'].value_counts()}\n\n")

print(f"'EMERGENCYSTATE_MODE' has {data_no_null['EMERGENCYSTATE_MODE'].isnull().sum()} null val
print(f"Value counts: \n{data_no_null['EMERGENCYSTATE_MODE'].value_counts()}")
```

```
'NAME_TYPE_SUITE' has 1292 null values.
Value counts:
```

```
Unaccompanied      248526
Family              40149
Spouse, partner     11370
Children             3267
Other_B              1770
Other_A               866
Group of people       271
Name: NAME_TYPE_SUITE, dtype: int64


'OCCUPATION_TYPE' has 96391 null values.
Value counts:
Laborers                 55186
Sales staff              32102
Core staff               27570
Managers                 21371
Drivers                  18603
High skill tech staff    11380
Accountants               9813
Medicine staff            8537
Security staff            6721
Cooking staff             5946
Cleaning staff            4653
Private service staff     2652
Low-skill Laborers        2093
Waiters/barmen staff      1348
Secretaries               1305
Realty agents              751
HR staff                   563
IT staff                   526
Name: OCCUPATION_TYPE, dtype: int64


'EMERGENCYSTATE_MODE' has 145755 null values.
Value counts:
No     159428
Yes      2328
Name: EMERGENCYSTATE_MODE, dtype: int64
```

Observation: *For categorical features the missing values will be replace with the mode.*

```
for col in num_column_with_null(data_no_null)[1]:
  data_no_null[col].fillna(data_no_null[col].mode()[0], inplace=True)

num_column_with_null(data_no_null)
```

```
(0, [])
```

Observation: *All the null values have been filled.*

3) Percentage of values in the target column

```
data_no_null['TARGET'].value_counts(normalize=True)
```

```
0    0.919271
```

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

```
data_no_null['TARGET'].value_counts()
```

```
0    282686
1     24825
Name: TARGET, dtype: int64
```

```
data_no_null['TARGET'].value_counts()[0]
```

```
282686
```

```
data_no_null['TARGET'].value_counts()[1]/data_no_null['TARGET'].value_counts()[0] * 100
```

```
8.781828601345662
```

Observations:

- *92% of loaner are good payers and 8% are default payers.*
- *The data is highly imbalanced.*

4) Balance the data

```
! pip install imbalanced-learn
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public
Requirement already satisfied: imbalanced-learn in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: scipy>=0.19.1 in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: numpy>=1.13.3 in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: scikit-learn>=0.24 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.7/dist-packa
```

```
import imblearn
print(imblearn.__version__)
```

```
0.8.1
```

```
label = data_no_null.pop('TARGET')
label.head()
```

```
0    1
1    0
2    0
3    0
4    0
Name: TARGET, dtype: int64
```

```
features = data_no_null
features.head()
```

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

5 rows × 80 columns

```
label.value_counts()
```

```
0    282686
1     24825
Name: TARGET, dtype: int64
```

```
features.info()
```

```
 25  FLAG_EMAIL                    307511 non-null  int64
 26  OCCUPATION_TYPE               307511 non-null  object
 27  CNT_FAM_MEMBERS               307511 non-null  float64
 28  REGION_RATING_CLIENT          307511 non-null  int64
 29  REGION_RATING_CLIENT_W_CITY   307511 non-null  int64
 30  WEEKDAY_APPR_PROCESS_START    307511 non-null  object
 31  HOUR_APPR_PROCESS_START       307511 non-null  int64
 32  REG_REGION_NOT_LIVE_REGION    307511 non-null  int64
 33  REG_REGION_NOT_WORK_REGION    307511 non-null  int64
 34  LIVE_REGION_NOT_WORK_REGION   307511 non-null  int64
 35  REG_CITY_NOT_LIVE_CITY        307511 non-null  int64
 36  REG_CITY_NOT_WORK_CITY        307511 non-null  int64
 37  LIVE_CITY_NOT_WORK_CITY       307511 non-null  int64
 38  ORGANIZATION_TYPE             307511 non-null  object
 39  EXT_SOURCE_2                  307511 non-null  float64
 40  EXT_SOURCE_3                  307511 non-null  float64
 41  YEARS_BEGINEXPLUATATION_AVG   307511 non-null  float64
 42  FLOORSMAX_AVG                 307511 non-null  float64
 43  YEARS_BEGINEXPLUATATION_MODE  307511 non-null  float64
 44  FLOORSMAX_MODE                307511 non-null  float64
 45  YEARS_BEGINEXPLUATATION_MEDI  307511 non-null  float64
 46  FLOORSMAX_MEDI                307511 non-null  float64
 47  TOTALAREA_MODE                307511 non-null  float64
 48  EMERGENCYSTATE_MODE           307511 non-null  object
 49  OBS_30_CNT_SOCIAL_CIRCLE      307511 non-null  float64
 50  DEF_30_CNT_SOCIAL_CIRCLE      307511 non-null  float64
 51  OBS_60_CNT_SOCIAL_CIRCLE      307511 non-null  float64
 52  DEF_60_CNT_SOCIAL_CIRCLE      307511 non-null  float64
 53  DAYS_LAST_PHONE_CHANGE        307511 non-null  float64
 54  FLAG_DOCUMENT_2               307511 non-null  int64
 55  FLAG_DOCUMENT_3               307511 non-null  int64
 56  FLAG_DOCUMENT_4               307511 non-null  int64
 57  FLAG_DOCUMENT_5               307511 non-null  int64
 58  FLAG_DOCUMENT_6               307511 non-null  int64
```

```
 59  FLAG_DOCUMENT_7               307511 non-null  int64
 60  FLAG_DOCUMENT_8               307511 non-null  int64
 61  FLAG_DOCUMENT_9               307511 non-null  int64
 62  FLAG_DOCUMENT_10              307511 non-null  int64
 63  FLAG_DOCUMENT_11              307511 non-null  int64
 64  FLAG_DOCUMENT_12              307511 non-null  int64
 65  FLAG_DOCUMENT_13              307511 non-null  int64
 66  FLAG_DOCUMENT_14              307511 non-null  int64
 67  FLAG_DOCUMENT_15              307511 non-null  int64
 68  FLAG_DOCUMENT_16              307511 non-null  int64
 69  FLAG_DOCUMENT_17              307511 non-null  int64
 70  FLAG_DOCUMENT_18              307511 non-null  int64
 71  FLAG_DOCUMENT_19              307511 non-null  int64
 72  FLAG_DOCUMENT_20              307511 non-null  int64
 73  FLAG_DOCUMENT_21              307511 non-null  int64
 74  AMT_REQ_CREDIT_BUREAU_HOUR    307511 non-null  float64
 75  AMT_REQ_CREDIT_BUREAU_DAY     307511 non-null  float64
 76  AMT_REQ_CREDIT_BUREAU_WEEK    307511 non-null  float64
 77  AMT_REQ_CREDIT_BUREAU_MON     307511 non-null  float64
 78  AMT_REQ_CREDIT_BUREAU_QRT     307511 non-null  float64
 79  AMT_REQ_CREDIT_BUREAU_YEAR    307511 non-null  float64
dtypes: float64(27), int64(40), object(13)
memory usage: 187.7+ MB
```

```
cat_list_indices = [features.columns.get_loc(col)for col in features.select_dtypes(include=['ob
cat_list_indices
```

```
[1, 2, 3, 4, 10, 11, 12, 13, 14, 26, 30, 38, 48]
```

```
from imblearn.over_sampling import SMOTENC

sm = SMOTENC(categorical_features=cat_list_indices)

features_bal, label_bal = sm.fit_resample(features, label)

label_bal.value_counts()
```

```
1    282686
0    282686
Name: TARGET, dtype: int64
```

Observation: *The dataset is now balanced.*

5) Plotting the data

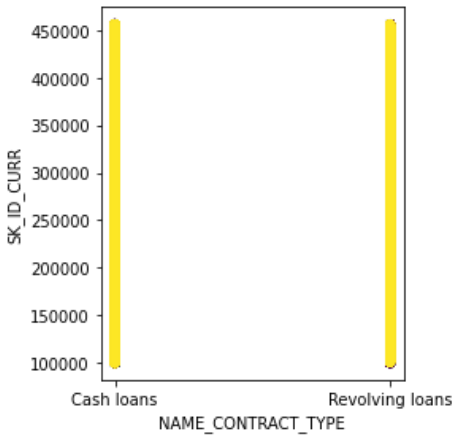
```
import seaborn as sns
```

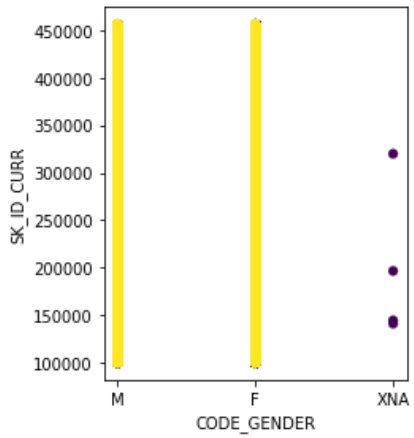
```
num_col_list = features_bal._get_numeric_data().columns.tolist()
len(num_col_list)
```

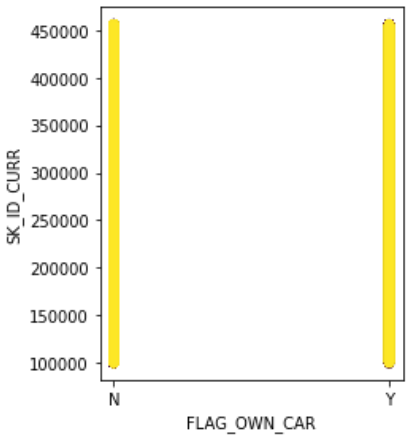
```
67
```

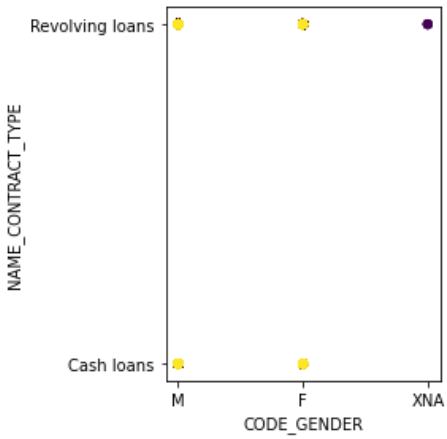
Observation: *As there are 80 features in the data, we will plot the first 10 against the following 5.*

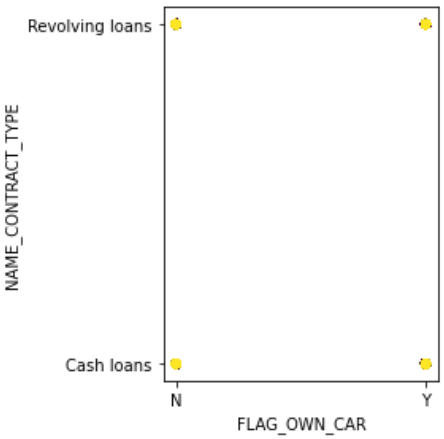
```
for i in range(10):
    plt.figure(figsize=(20,4))
    plt.subplot(151)
    plt.scatter(y=features_bal.iloc[:, i], x=features_bal.iloc[:, i+1], c=label_bal, linewidths
    plt.ylabel(features_bal.columns.tolist()[i])
    plt.xlabel(features_bal.columns.tolist()[i+1])
    plt.subplot(152)
    plt.scatter(y=features_bal.iloc[:, i], x=features_bal.iloc[:, i+2], c=label_bal, linewidths
    plt.ylabel(features_bal.columns.tolist()[i])
    plt.xlabel(features_bal.columns.tolist()[i+2])
    plt.subplot(153)
    plt.scatter(y=features_bal.iloc[:, i], x=features_bal.iloc[:, i+3], c=label_bal, linewidths
    plt.ylabel(features_bal.columns.tolist()[i])
    plt.xlabel(features_bal.columns.tolist()[i+3])
    plt.subplot(154)
    plt.scatter(y=features_bal.iloc[:, i], x=features_bal.iloc[:, i+4], c=label_bal, linewidths
    plt.ylabel(features_bal.columns.tolist()[i])
    plt.xlabel(features_bal.columns.tolist()[i+4])
    plt.subplot(155)
    plt.scatter(y=features_bal.iloc[:, i], x=features_bal.iloc[:, i+5], c=label_bal, linewidths
    plt.ylabel(features_bal.columns.tolist()[i])
    plt.xlabel(features_bal.columns.tolist()[i+5])
    plt.tight_layout()
    plt.show()
```

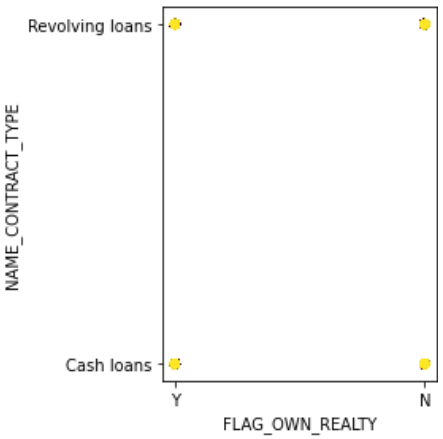

SK_ID_CURR
450000
400000
350000
300000
250000
200000
150000
100000
Cash loans
Revolving loans
NAME_CONTRACT_TYPE

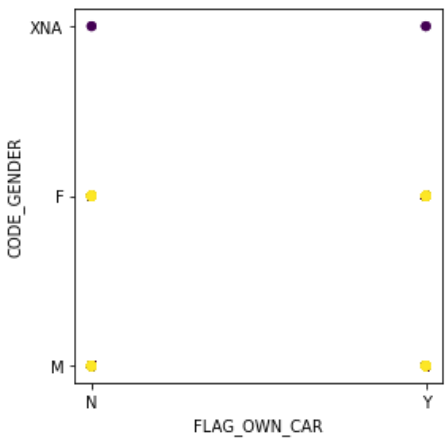

SK_ID_CURR
450000
400000
350000
300000
250000
200000
150000
100000
M
F
XNA
CODE_GENDER

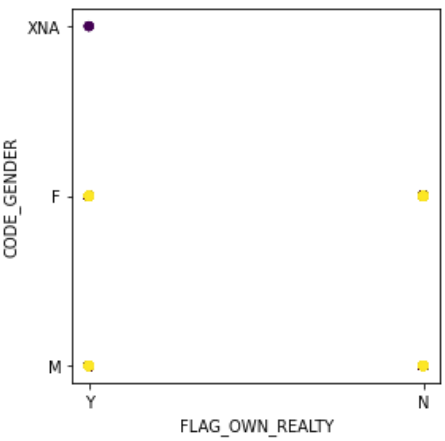

SK_ID_CURR
450000
400000
350000
300000
250000
200000
150000
100000
N
Y
FLAG_OWN_CAR

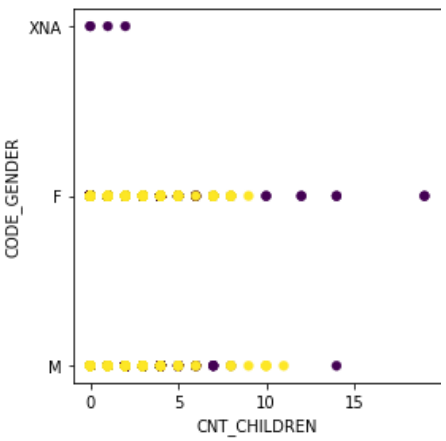

NAME_CONTRACT_TYPE
Revolving loans
Cash loans
M
F
XNA
CODE_GENDER

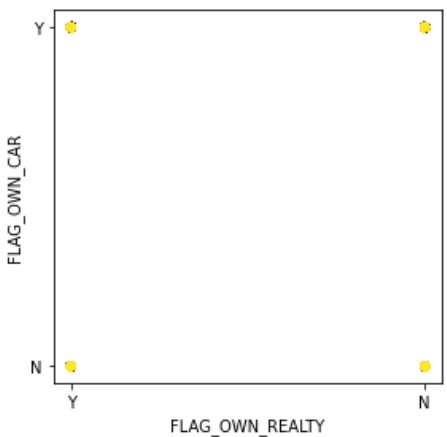

NAME_CONTRACT_TYPE
Revolving loans
Cash loans
N
Y
FLAG_OWN_CAR

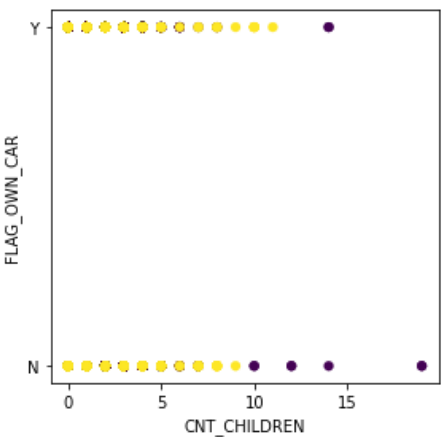

NAME_CONTRACT_TYPE
Revolving loans
Cash loans
Y
N
FLAG_OWN_REALTY

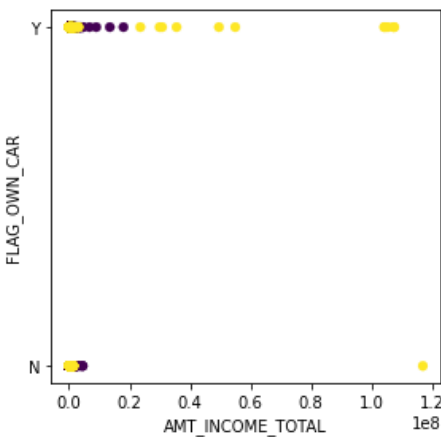

CODE_GENDER
XNA
F
M
N
Y
FLAG_OWN_CAR


CODE_GENDER
XNA
F
M
Y
N
FLAG_OWN_REALTY


CODE_GENDER
XNA
F
M
0
5
10
15
CNT_CHILDREN


FLAG_OWN_CAR
Y
N
Y
N
FLAG_OWN_REALTY

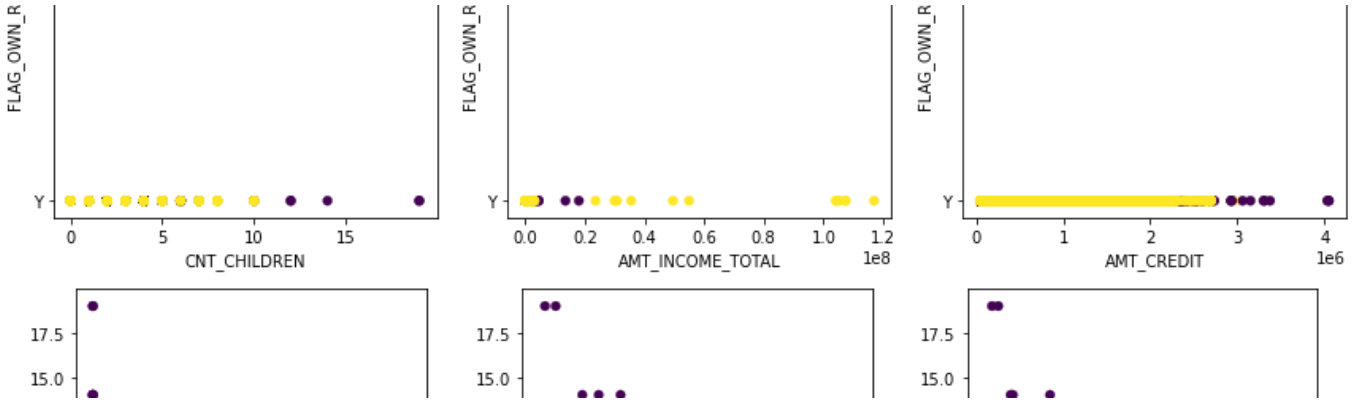

FLAG_OWN_CAR
Y
N
0
5
10
15
CNT_CHILDREN


FLAG_OWN_CAR
Y
N
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1e8
AMT_INCOME_TOTAL


N
EALTY

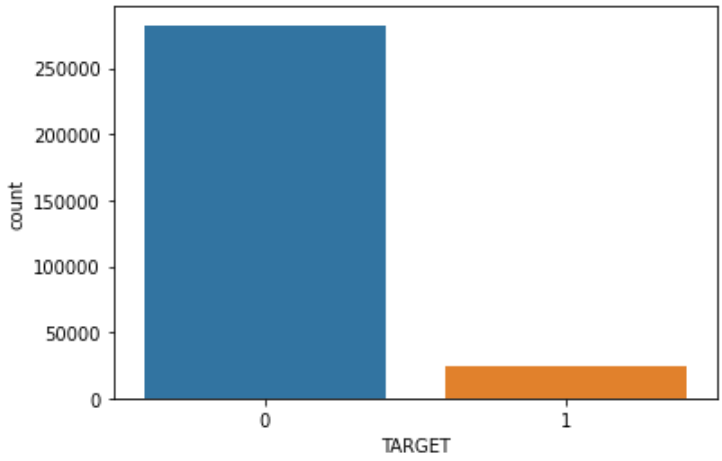

N
EALTY


N
EALTY



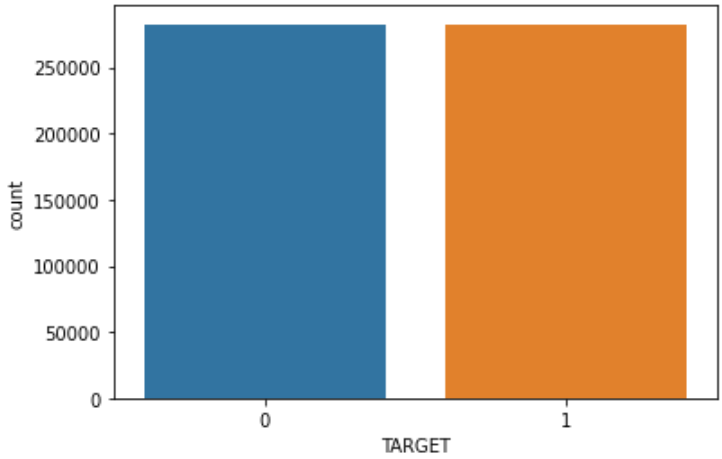

```
sns.countplot(x=label)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0c8d776310>
```




```
sns.countplot(x=label_bal)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0caa2a2a50>
```



6) Encoding the required columns for the model

```
cat_list = features.select_dtypes(include=['object']).columns.tolist()
cat_list
```

```
['NAME_CONTRACT_TYPE',
```

```
      'CODE_GENDER',
      'FLAG_OWN_CAR',
      'FLAG_OWN_REALTY',
      'NAME_TYPE_SUITE',
      'NAME_INCOME_TYPE',
      'NAME_EDUCATION_TYPE',
      'NAME_FAMILY_STATUS',
      'NAME_HOUSING_TYPE',
      'OCCUPATION_TYPE',
      'WEEKDAY_APPR_PROCESS_START',
      'ORGANIZATION_TYPE',
      'EMERGENCYSTATE_MODE']


for col in cat_list:
  print(f"Value counts for {col}:\n{features_bal[col].value_counts()}\n")
```

```
Trade: type 7              12683
School                     12288
Construction               12262
Kindergarten               10416
Business Entity Type 1      9284
Transport: type 4           8234
Trade: type 3               5294
Industry: type 3            5152
Security                    4846
Industry: type 9            4610
Housing                     4104
Agriculture                 3988
Industry: type 11           3905
Bank                        3260
Military                    3251
Transport: type 2           3070
Postal                      3066
Restaurant                  2943
Police                      2885
Trade: type 2               2624
Security Ministries         2386
Transport: type 3           2143
Services                    2047
Industry: type 7            1798
University                  1682
Industry: type 1            1598
Industry: type 4            1327
Hotel                       1288
Electricity                 1198
Telecom                      779
Industry: type 5             754
Emergency                    754
Trade: type 6                734
Insurance                    715
Industry: type 2             620
Advertising                  609
Realtor                      571
Culture                      479
Trade: type 1                453
Cleaning                     449
Mobile                       448
Industry: type 12            432
Legal Services               424
Transport: type 1            236
Industry: type 6             147
Industry: type 10            147
```

```
Industry: type 13            104
Religion                     102
Trade: type 4                 78
Trade: type 5                 67
Industry: type 8              36
Name: ORGANIZATION_TYPE, dtype: int64

Value counts for EMERGENCYSTATE_MODE:
No     563034
Yes      2338
Name: EMERGENCYSTATE_MODE, dtype: int64
```

```
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()

for col in cat_list:
  features_bal[col] = le.fit_transform(features_bal[col])

features_bal.info()
```

```
 24  FLAG_PHONE                    565372 non-null  int64
 25  FLAG_EMAIL                    565372 non-null  int64
 26  OCCUPATION_TYPE               565372 non-null  int64
 27  CNT_FAM_MEMBERS               565372 non-null  float64
 28  REGION_RATING_CLIENT          565372 non-null  int64
 29  REGION_RATING_CLIENT_W_CITY   565372 non-null  int64
 30  WEEKDAY_APPR_PROCESS_START    565372 non-null  int64
 31  HOUR_APPR_PROCESS_START       565372 non-null  int64
 32  REG_REGION_NOT_LIVE_REGION    565372 non-null  int64
 33  REG_REGION_NOT_WORK_REGION    565372 non-null  int64
 34  LIVE_REGION_NOT_WORK_REGION   565372 non-null  int64
 35  REG_CITY_NOT_LIVE_CITY        565372 non-null  int64
 36  REG_CITY_NOT_WORK_CITY        565372 non-null  int64
 37  LIVE_CITY_NOT_WORK_CITY       565372 non-null  int64
 38  ORGANIZATION_TYPE             565372 non-null  int64
 39  EXT_SOURCE_2                  565372 non-null  float64
 40  EXT_SOURCE_3                  565372 non-null  float64
 41  YEARS_BEGINEXPLUATATION_AVG   565372 non-null  float64
 42  FLOORSMAX_AVG                 565372 non-null  float64
 43  YEARS_BEGINEXPLUATATION_MODE  565372 non-null  float64
 44  FLOORSMAX_MODE                565372 non-null  float64
 45  YEARS_BEGINEXPLUATATION_MEDI  565372 non-null  float64
 46  FLOORSMAX_MEDI                565372 non-null  float64
 47  TOTALAREA_MODE                565372 non-null  float64
 48  EMERGENCYSTATE_MODE           565372 non-null  int64
 49  OBS_30_CNT_SOCIAL_CIRCLE      565372 non-null  float64
 50  DEF_30_CNT_SOCIAL_CIRCLE      565372 non-null  float64
 51  OBS_60_CNT_SOCIAL_CIRCLE      565372 non-null  float64
 52  DEF_60_CNT_SOCIAL_CIRCLE      565372 non-null  float64
 53  DAYS_LAST_PHONE_CHANGE        565372 non-null  float64
 54  FLAG_DOCUMENT_2               565372 non-null  int64
 55  FLAG_DOCUMENT_3               565372 non-null  int64
 56  FLAG_DOCUMENT_4               565372 non-null  int64
 57  FLAG_DOCUMENT_5               565372 non-null  int64
 58  FLAG_DOCUMENT_6               565372 non-null  int64
 59  FLAG_DOCUMENT_7               565372 non-null  int64
 60  FLAG_DOCUMENT_8               565372 non-null  int64
 61  FLAG_DOCUMENT_9               565372 non-null  int64
 62  FLAG_DOCUMENT_10              565372 non-null  int64
 63  FLAG_DOCUMENT_11              565372 non-null  int64
```

```
 64  FLAG_DOCUMENT_12              565372 non-null  int64
 65  FLAG_DOCUMENT_13              565372 non-null  int64
 66  FLAG_DOCUMENT_14              565372 non-null  int64
 67  FLAG_DOCUMENT_15              565372 non-null  int64
 68  FLAG_DOCUMENT_16              565372 non-null  int64
 69  FLAG_DOCUMENT_17              565372 non-null  int64
 70  FLAG_DOCUMENT_18              565372 non-null  int64
 71  FLAG_DOCUMENT_19              565372 non-null  int64
 72  FLAG_DOCUMENT_20              565372 non-null  int64
 73  FLAG_DOCUMENT_21              565372 non-null  int64
 74  AMT_REQ_CREDIT_BUREAU_HOUR    565372 non-null  float64
 75  AMT_REQ_CREDIT_BUREAU_DAY     565372 non-null  float64
 76  AMT_REQ_CREDIT_BUREAU_WEEK    565372 non-null  float64
 77  AMT_REQ_CREDIT_BUREAU_MON     565372 non-null  float64
 78  AMT_REQ_CREDIT_BUREAU_QRT     565372 non-null  float64
 79  AMT_REQ_CREDIT_BUREAU_YEAR    565372 non-null  float64
dtypes: float64(27), int64(53)
memory usage: 345.1 MB
```

Observations: *All the features are numerical.*

7) Modeling with Sensitivity as metrix

Scaling the features

```
from sklearn.preprocessing import RobustScaler
rbFeatures=RobustScaler()
features_final = rbFeatures.fit_transform(features_bal)
```

Splitting the dataset

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(features_final,
                                                 label_bal,
                                                 test_size=0.2,
                                                 random_state=10)
```

Creation of the model

```
import tensorflow as tf

model = tf.keras.models.Sequential()

model.add(tf.keras.layers.Dense( units =10  , activation= 'leaky_relu' , input_shape= (features
model.add(tf.keras.layers.Dense( units =10 , activation= 'leaky_relu' ))
model.add(tf.keras.layers.Dense( units =10 , activation= 'leaky_relu' ))

model.add(tf.keras.layers.Dense( units = 1, activation= 'sigmoid' ))
```

```
model.compile(optimizer="adam" , loss="binary_crossentropy" , metrics=[tf.keras.metrics.Recall(

history = model.fit(X_train,y_train, epochs=500 , validation_data=(X_test,y_test))
```

```
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2324 - recall: 0.859
Epoch 473/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2321 - recall: 0.858
Epoch 474/500
14135/14135 [==============================] - 39s 3ms/step - loss: 0.2321 - recall: 0.859
Epoch 475/500
14135/14135 [==============================] - 37s 3ms/step - loss: 0.2320 - recall: 0.859
Epoch 476/500
14135/14135 [==============================] - 37s 3ms/step - loss: 0.2319 - recall: 0.859
Epoch 477/500
14135/14135 [==============================] - 39s 3ms/step - loss: 0.2320 - recall: 0.859
Epoch 478/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2323 - recall: 0.859
Epoch 479/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2319 - recall: 0.859
Epoch 480/500
14135/14135 [==============================] - 40s 3ms/step - loss: 0.2320 - recall: 0.859
Epoch 481/500
14135/14135 [==============================] - 37s 3ms/step - loss: 0.2321 - recall: 0.859
Epoch 482/500
14135/14135 [==============================] - 37s 3ms/step - loss: 0.2319 - recall: 0.859
Epoch 483/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2318 - recall: 0.859
Epoch 484/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2316 - recall: 0.859
Epoch 485/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2319 - recall: 0.859
Epoch 486/500
14135/14135 [==============================] - 37s 3ms/step - loss: 0.2318 - recall: 0.859
Epoch 487/500
14135/14135 [==============================] - 40s 3ms/step - loss: 0.2321 - recall: 0.859
Epoch 488/500
14135/14135 [==============================] - 39s 3ms/step - loss: 0.2319 - recall: 0.859
Epoch 489/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2321 - recall: 0.859
Epoch 490/500
14135/14135 [==============================] - 40s 3ms/step - loss: 0.2319 - recall: 0.859
Epoch 491/500
14135/14135 [==============================] - 39s 3ms/step - loss: 0.2319 - recall: 0.859
Epoch 492/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2319 - recall: 0.859
Epoch 493/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2317 - recall: 0.859
Epoch 494/500
14135/14135 [==============================] - 40s 3ms/step - loss: 0.2318 - recall: 0.859

Epoch 495/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2318 - recall: 0.859
Epoch 496/500
14135/14135 [==============================] - 38s 3ms/step - loss: 0.2316 - recall: 0.859
Epoch 497/500
14135/14135 [==============================] - 39s 3ms/step - loss: 0.2318 - recall: 0.859
Epoch 498/500
14135/14135 [==============================] - 37s 3ms/step - loss: 0.2318 - recall: 0.859
Epoch 499/500
14135/14135 [==============================] - 39s 3ms/step - loss: 0.2317 - recall: 0.859
Epoch 500/500
14135/14135 [==============================] - 40s 3ms/step - loss: 0.2317 - recall: 0.859
```

8) Calculation of the area under the ROC curve

```
y_pred = model.predict(X_test)
y_pred.flatten()
```

```
array([0.99414647, 0.9998138 , 0.7776313 , ..., 0.04916841, 0.80299544,
       0.1899837 ], dtype=float32)
```

```
from sklearn.metrics import roc_curve, auc
fpr, tpr, _ = roc_curve(y_test, y_pred)
auc = auc(fpr, tpr)
```

```
plt.plot([0, 1], [0, 1])
plt.plot(fpr, tpr, label=f'ROC with area={round(auc, 2)}')
plt.xlabel('False positive rate')
plt.ylabel('True positive rate')
plt.title('ROC curve')
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f0c891cbf90>
```