ljmveoung

February 8, 2025

[1]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectKBest, f_classif
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report,␣
 ↪confusion_matrix
import warnings
warnings.filterwarnings('ignore')
```

[2]:
```python
train = pd.read_csv('train.csv')
test = pd.read_csv('test.csv')
```

[3]:
```python
train.head(10)
```

[3]:
```
  customer_id  age     profession relationship_status qualification  \
0   CST_28349   42     Management             Married       Masters
1   CST_25974   42     Management             Engaged           NaN
2   CST_29163   59    Blue-collar             Engaged           NaN
3   CST_14319   45     Management             Married       Masters
4    CST_1387   36         Admin.             Engaged   High School
5   CST_39915   25        Student   In a Relationship     Bachelors
6    CST_6435   30     Management             Married       Masters
7   CST_31199   38     Technician             Married           NaN
8    CST_8780   35    Blue-collar             Engaged     Bachelors
9   CST_17052   56  Self-employed             Engaged     Bachelors

   annual_income  annual_expenses  bank_balance has_home_loan has_car_loan  \
0   1.554848e+05     1.223666e+05      25796.98           yes           No
1   7.354741e+06     4.066426e+06    1144097.20           yes           No
2   4.154800e+05     3.217619e+05      71040.17           yes           No
3   0.000000e+00     0.000000e+00          0.00           yes           No
4   2.202652e+05     1.206680e+05      41127.00           yes           No
5   2.421387e+05     5.357482e+04      38477.30            no           No
```

```
6   7.077756e+06     3.084730e+06    1346083.92          yes          No
7   4.643544e+06     3.137080e+06     763261.91          yes          No
8   5.058187e+05     4.142157e+05      93334.84          yes          No
9   2.336427e+06     7.535675e+05     424512.25          yes          No

   …  credit_score  customer_category  prev_call_duration(min)  \
0  …         868.0                  C                     8.45
1  …         729.0                  B                     3.37
2  …         792.0                  C                     2.27
3  …         787.0                  B                     4.12
4  …         752.0                  C                     2.68
5  …           NaN                  C                     2.57
6  …         693.0                  C                     5.07
7  …         783.0                  C                     1.65
8  …         766.0                  C                     1.28
9  …         742.0                  D                     2.58

  n_prev_campaign_calls  days_since_last_contact  total_contacts_so_far  \
0                     1                      220                      2
1                     2                      184                      2
2                     1                       -1                      0
3                     2                       -1                      0
4                     8                       -1                      0
5                     1                       -1                      0
6                     2                       -1                      0
7                     1                      102                      6
8                     8                       -1                      0
9                     4                       -1                      0

   prev_outcome  targeted_mode_of_communication targeted_month is_target
0     Succesful                      Cell phone            Jan         0
1       Failure                             Sms            Nov         0
2    No Outcome                      Cell phone            Feb         0
3    No Outcome                      Cell phone            Jul         0
4    No Outcome                           Email            May         0
5    No Outcome                      Cell phone            Jun         0
6    No Outcome                           Email            May         0
7     Succesful                      Cell phone            Feb         1
8    No Outcome                           Email            Jun         0
9    No Outcome                      Cell phone            Jul         0

[10 rows x 22 columns]
```

[4]:
```
train.tail(10)
```

[4]:
```
      customer_id  age   profession relationship_status qualification  \
15274   CST_35441   37          NaN       Widow/Widower       Masters
```

15275	CST_29221	53	Technician	In a Relationship	Masters
15276	CST_8288	37	Housemaid	Engaged	Bachelors
15277	CST_15288	54	Technician	In a Relationship	NaN
15278	CST_28227	26	Admin.	Engaged	Bachelors
15279	CST_8335	38	Blue-collar	Engaged	High School
15280	CST_2412	35	Services	NaN	Bachelors
15281	CST_41912	28	Management	In a Relationship	Masters
15282	CST_16073	27	Services	Single	Masters
15283	CST_32369	38	Admin.	Married	Bachelors

	annual_income	annual_expenses	bank_balance	has_home_loan \
15274	-8.875558e+02	-3.848436e+02	-138.76	no
15275	-1.313695e+05	-7.633842e+04	-24094.53	yes
15276	4.015309e+06	3.242833e+06	677830.58	yes
15277	4.769018e+05	1.469559e+05	75258.29	yes
15278	-9.576156e+04	-7.783913e+04	-17353.30	yes
15279	-8.075779e+05	-3.946453e+05	-135608.60	yes
15280	1.728594e+06	6.037924e+05	302202.73	yes
15281	7.740389e+04	5.712646e+04	13097.41	no
15282	9.028109e+06	3.885062e+06	1545276.69	yes
15283	2.853072e+05	2.173621e+05	49666.02	yes

	has_car_loan	…	credit_score	customer_category \
15274	No	…	506.0	A
15275	Yes	…	753.0	A
15276	No	…	720.0	A
15277	No	…	848.0	A
15278	No	…	802.0	C
15279	No	…	741.0	C
15280	No	…	721.0	A
15281	No	…	836.0	D
15282	No	…	822.0	D
15283	No	…	775.0	B

	prev_call_duration(min)	n_prev_campaign_calls	days_since_last_contact \
15274	1.68	1	349
15275	4.82	1	-1
15276	2.37	3	-1
15277	11.23	3	-1
15278	1.32	1	225
15279	3.23	4	-1
15280	6.17	1	-1
15281	11.27	2	146
15282	11.03	1	-1
15283	3.92	2	335

	total_contacts_so_far	prev_outcome	targeted_mode_of_communication \

```
15274                        1       Failure                      Cell phone
15275                        0    No Outcome                      Cell phone
15276                        0    No Outcome                           Email
15277                        0    No Outcome                      Cell phone
15278                        2     Succesful                      Cell phone
15279                        0    No Outcome                           Email
15280                        0    No Outcome                           Email
15281                        2       Failure                      Cell phone
15282                        0    No Outcome                      Cell phone
15283                        2       Failure                      Cell phone

      targeted_month is_target
15274            May         0
15275            Feb         0
15276            Jun         0
15277            Jul         1
15278            Jan         0
15279            Jun         0
15280            May         0
15281            Oct         1
15282            Jul         0
15283            Apr         0

[10 rows x 22 columns]
```

[5]:
```
test.tail(10)
```

[5]:
```
     customer_id  age   profession relationship_status qualification  \
6771   CST_22015   48  Blue-collar                 NaN     Bachelors
6772   CST_13000   47   Technician             Married   High School
6773   CST_40832   44       Admin.             Engaged     Bachelors
6774   CST_17904   35     Services             Married     Bachelors
6775   CST_11045   52   Technician             Married     Bachelors
6776   CST_44903   70       Admin.             Engaged     Bachelors
6777   CST_19157   45     Services             Engaged     Bachelors
6778   CST_27311   33   Technician             Married     Bachelors
6779   CST_30414   35          NaN            Divorced     Bachelors
6780    CST_6423   44  Blue-collar             Married   High School

      annual_income  annual_expenses  bank_balance has_home_loan has_car_loan  \
6771   9.840030e+05    711061.778364     172547.00           yes           No
6772   5.700040e+04     45724.176795       9088.06            no           No
6773   1.312330e+06    767887.892199     245357.69           yes           No
6774   0.000000e+00         0.000000          0.00            no           No
6775   9.212208e+05    212677.567414     148190.05            no           No
6776   7.215614e+05    254776.417537     126201.34            no           No
6777   0.000000e+00         0.000000          0.00            no           No
```

```
6778   2.455013e+05    126267.698255      39488.47           yes           No
6779   1.313423e+06    668085.473155     218675.54            no           No
6780  -2.817674e+05   -144400.243155     -53689.68           yes          NaN

      …  n_defaults  credit_score  customer_category  \
6771  …           0         729.0                  C
6772  …           0         682.0                  C
6773  …           0         716.0                  A
6774  …           0         790.0                  D
6775  …           0         687.0                  B
6776  …           0         867.0                  D
6777  …           0         784.0                  B
6778  …           0         841.0                  A
6779  …           0         818.0                  A
6780  …           0           NaN                  B

     prev_call_duration(min)  n_prev_campaign_calls  days_since_last_contact  \
6771                    1.55                      3                       -1
6772                    2.78                      3                       -1
6773                    3.35                      1                      103
6774                   14.05                      2                       -1
6775                    1.45                      2                       -1
6776                    3.95                      2                       -1
6777                    3.27                      2                       -1
6778                    1.27                      1                      170
6779                    3.95                      1                       -1
6780                    0.98                      4                       -1

      total_contacts_so_far  prev_outcome targeted_mode_of_communication  \
6771                      0    No Outcome                     Cell phone
6772                      0    No Outcome                            Sms
6773                      1       Failure                     Cell phone
6774                      0    No Outcome                     Cell phone
6775                      0    No Outcome                          Email
6776                      0    No Outcome                            Sms
6777                      0    No Outcome                     Cell phone
6778                      3       Failure                     Cell phone
6779                      0    No Outcome                     Cell phone
6780                      0    No Outcome                          Email

     targeted_month
6771            Aug
6772            Jul
6773            Aug
6774            Jul
6775            Jun
6776            Sep
```

```
6777            Aug
6778            Nov
6779            Feb
6780            May

[10 rows x 21 columns]
```

[6]:
```
train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15284 entries, 0 to 15283
Data columns (total 22 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   customer_id                     15284 non-null  object
 1   age                             15284 non-null  int64
 2   profession                      13847 non-null  object
 3   relationship_status             13829 non-null  object
 4   qualification                   13866 non-null  object
 5   annual_income                   15284 non-null  float64
 6   annual_expenses                 15284 non-null  float64
 7   bank_balance                    14972 non-null  float64
 8   has_home_loan                   14974 non-null  object
 9   has_car_loan                    14995 non-null  object
 10  total_loans                     14084 non-null  float64
 11  n_defaults                      15284 non-null  int64
 12  credit_score                    14089 non-null  float64
 13  customer_category               15284 non-null  object
 14  prev_call_duration(min)         15284 non-null  float64
 15  n_prev_campaign_calls           15284 non-null  int64
 16  days_since_last_contact         15284 non-null  int64
 17  total_contacts_so_far           15284 non-null  int64
 18  prev_outcome                    15284 non-null  object
 19  targeted_mode_of_communication  15284 non-null  object
 20  targeted_month                  15284 non-null  object
 21  is_target                       15284 non-null  int64
dtypes: float64(6), int64(6), object(10)
memory usage: 2.6+ MB
```

[7]:
```
test.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6781 entries, 0 to 6780
Data columns (total 21 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   customer_id                     6781 non-null   object
 1   age                             6781 non-null   int64
```

```
 2   profession                        6126 non-null   object
 3   relationship_status               6135 non-null   object
 4   qualification                     6090 non-null   object
 5   annual_income                     6781 non-null   float64
 6   annual_expenses                   6781 non-null   float64
 7   bank_balance                      6656 non-null   float64
 8   has_home_loan                     6638 non-null   object
 9   has_car_loan                      6660 non-null   object
 10  total_loans                       6254 non-null   float64
 11  n_defaults                        6781 non-null   int64
 12  credit_score                      6245 non-null   float64
 13  customer_category                 6781 non-null   object
 14  prev_call_duration(min)           6781 non-null   float64
 15  n_prev_campaign_calls             6781 non-null   int64
 16  days_since_last_contact           6781 non-null   int64
 17  total_contacts_so_far             6781 non-null   int64
 18  prev_outcome                      6781 non-null   object
 19  targeted_mode_of_communication    6781 non-null   object
 20  targeted_month                    6781 non-null   object
dtypes: float64(6), int64(5), object(10)
memory usage: 1.1+ MB
```

[8]:
```
train.shape
```

[8]:
```
(15284, 22)
```

[9]:
```
test.shape
```

[9]:
```
(6781, 21)
```

[10]:
```
train.columns
```

[10]:
```
Index(['customer_id', 'age', 'profession', 'relationship_status',
       'qualification', 'annual_income', 'annual_expenses', 'bank_balance',
       'has_home_loan', 'has_car_loan', 'total_loans', 'n_defaults',
       'credit_score', 'customer_category', 'prev_call_duration(min)',
       'n_prev_campaign_calls', 'days_since_last_contact',
       'total_contacts_so_far', 'prev_outcome',
       'targeted_mode_of_communication', 'targeted_month', 'is_target'],
      dtype='object')
```

[11]:
```
test.columns
```

[11]:
```
Index(['customer_id', 'age', 'profession', 'relationship_status',
       'qualification', 'annual_income', 'annual_expenses', 'bank_balance',
       'has_home_loan', 'has_car_loan', 'total_loans', 'n_defaults',
       'credit_score', 'customer_category', 'prev_call_duration(min)',
```

```
       'n_prev_campaign_calls', 'days_since_last_contact',
       'total_contacts_so_far', 'prev_outcome',
       'targeted_mode_of_communication', 'targeted_month'],
      dtype='object')
```

[12]:
```
train.dtypes
```

[12]:
```
customer_id                        object
age                                 int64
profession                         object
relationship_status                object
qualification                      object
annual_income                     float64
annual_expenses                   float64
bank_balance                      float64
has_home_loan                      object
has_car_loan                       object
total_loans                       float64
n_defaults                          int64
credit_score                      float64
customer_category                  object
prev_call_duration(min)           float64
n_prev_campaign_calls               int64
days_since_last_contact             int64
total_contacts_so_far               int64
prev_outcome                       object
targeted_mode_of_communication     object
targeted_month                     object
is_target                           int64
dtype: object
```

[13]:
```
# Duplicates
train.duplicated().sum()
```

[13]:
```
0
```

[14]:
```
test.duplicated().sum()
```

[14]:
```
0
```

[15]:
```
# Null Values
train.isnull().sum()
```

[15]:
```
customer_id                          0
age                                  0
profession                        1437
relationship_status               1455
```

```
qualification                    1418
annual_income                       0
annual_expenses                     0
bank_balance                      312
has_home_loan                     310
has_car_loan                      289
total_loans                      1200
n_defaults                          0
credit_score                     1195
customer_category                   0
prev_call_duration(min)             0
n_prev_campaign_calls               0
days_since_last_contact             0
total_contacts_so_far               0
prev_outcome                        0
targeted_mode_of_communication      0
targeted_month                      0
is_target                           0
dtype: int64
```

[16]:
```
test.isnull().sum()
```

[16]:
```
customer_id                       0
age                               0
profession                      655
relationship_status             646
qualification                   691
annual_income                     0
annual_expenses                   0
bank_balance                    125
has_home_loan                   143
has_car_loan                    121
total_loans                     527
n_defaults                        0
credit_score                    536
customer_category                 0
prev_call_duration(min)           0
n_prev_campaign_calls             0
days_since_last_contact           0
total_contacts_so_far             0
prev_outcome                      0
targeted_mode_of_communication    0
targeted_month                    0
dtype: int64
```

[98]:
```
num_col = train.select_dtypes(include=('number')).columns
cat_col = train.select_dtypes(include=('object')).columns
```

```python
num_col= num_col.drop('is_target')

for col in num_col:
    train[col] = train[col].fillna(train[col].mean())
    test[col] = test[col].fillna(test[col].mean())

for col in cat_col:
    train[col] = train[col].fillna(train[col].mode().iloc[0])
    test[col] = test[col].fillna(test[col].mode().iloc[0])
```

[100]:
```python
train.isnull().sum()
```

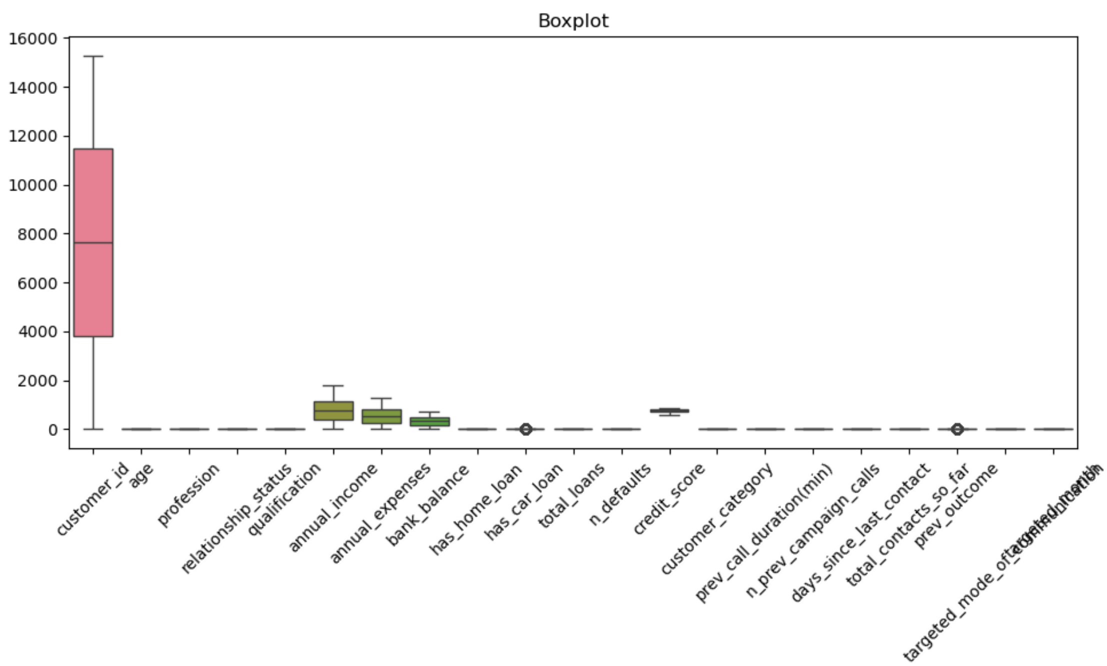
[100]:
```
customer_id                     0
age                             0
profession                      0
relationship_status             0
qualification                   0
annual_income                   0
annual_expenses                 0
bank_balance                    0
has_home_loan                   0
has_car_loan                    0
total_loans                     0
n_defaults                      0
credit_score                    0
customer_category               0
prev_call_duration(min)         0
n_prev_campaign_calls           0
days_since_last_contact         0
total_contacts_so_far           0
prev_outcome                    0
targeted_mode_of_communication  0
targeted_month                  0
is_target                       0
dtype: int64
```

[102]:
```python
test.isnull().sum()
```

[102]:
```
customer_id                     0
age                             0
profession                      0
relationship_status             0
qualification                   0
annual_income                   0
annual_expenses                 0
bank_balance                    0
has_home_loan                   0
```

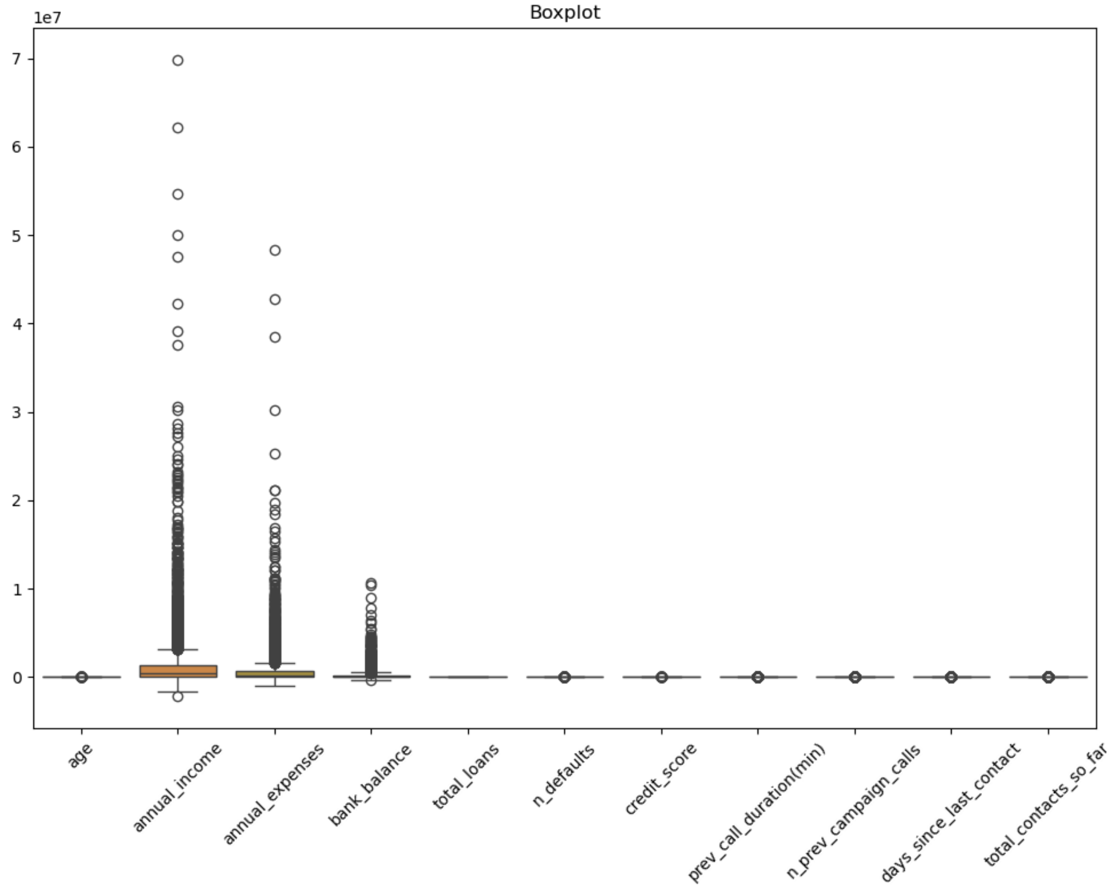
```
has_car_loan                     0
total_loans                      0
n_defaults                       0
credit_score                     0
customer_category                0
prev_call_duration(min)          0
n_prev_campaign_calls            0
days_since_last_contact          0
total_contacts_so_far            0
prev_outcome                     0
targeted_mode_of_communication   0
targeted_month                   0
dtype: int64
```

[104]:
```python
# Boxplot for outlier
plt.figure(figsize=(10,6))
sns.boxplot(data = train[num_col])
plt.title('Boxplot')
plt.xticks(rotation = 45)
plt.tight_layout()
plt.show()
```



[21]:
```python
# Boxplot for outlier
plt.figure(figsize=(10,8))
sns.boxplot(data = test[num_col])
```

```
plt.title('Boxplot')
plt.xticks(rotation =45)
plt.tight_layout()
plt.show()
```



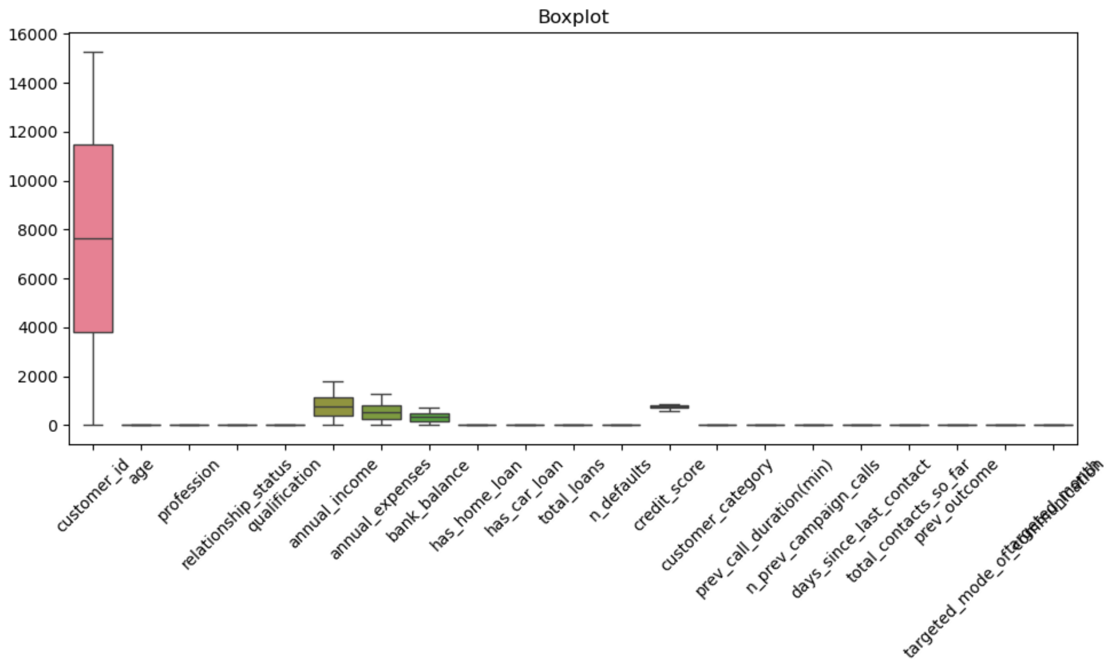
[106]:
```
def fix_outliers(df, num_col):
    for col in num_col:
        Q1 = df[col].quantile(0.25)
        Q3 = df[col].quantile(0.75)
        IQR = Q3 - Q1
        lower_bound = Q1 - 1.5 * IQR
        upper_bound = Q3 + 1.5 * IQR
        df[col] = df[col].clip(lower = lower_bound, upper=upper_bound)
    return df
```

[108]:
```
train = fix_outliers(train, num_col)
test = fix_outliers(test, num_col)
```

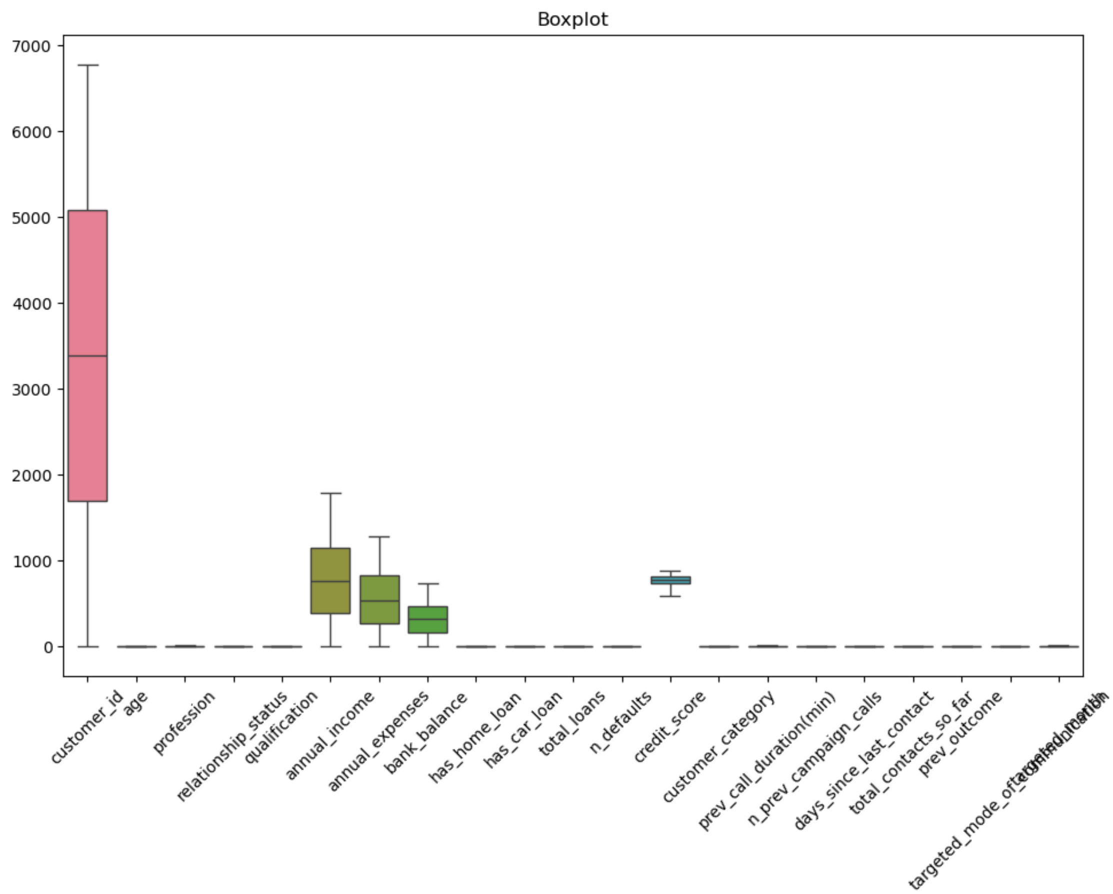
[110]:
```
train.shape
```

[110]: (15284, 22)

[112]:
```
# Boxplot for outlier
plt.figure(figsize=(10,6))
sns.boxplot(data = train[num_col])
plt.title('Boxplot')
plt.xticks(rotation = 45)
plt.tight_layout()
plt.show()
```



[114]:
```
# Boxplot for outlier
plt.figure(figsize=(10,8))
sns.boxplot(data = test[num_col])
plt.title('Boxplot')
plt.xticks(rotation =45)
plt.tight_layout()
plt.show()
```



[27]:
```
train[num_col].skew()
```

[27]:
```
age                        0.566979
annual_income              1.092445
annual_expenses            1.106242
bank_balance               1.045332
total_loans                0.185931
n_defaults                 0.000000
credit_score              -0.268971
prev_call_duration(min)    1.048147
n_prev_campaign_calls      1.090912
days_since_last_contact    0.000000
total_contacts_so_far      0.000000
dtype: float64
```

[28]:
```
train['age'] = np.sqrt(train['age'])
train['annual_income'] = np.sqrt(train['annual_income'])
train['annual_expenses'] = np.sqrt(train['annual_expenses'])
train['bank_balance'] = np.sqrt(train['bank_balance'])
```

```
train['n_prev_campaign_calls']  = np.sqrt(train['n_prev_campaign_calls'])
```

[29]:
```
train[num_col].skew()
```

[29]:
```
age                        0.305251
annual_income              0.387044
annual_expenses            0.414538
bank_balance               0.331747
total_loans                0.185931
n_defaults                 0.000000
credit_score              -0.268971
prev_call_duration(min)    1.048147
n_prev_campaign_calls      0.723456
days_since_last_contact    0.000000
total_contacts_so_far      0.000000
dtype: float64
```

[30]:
```
test[num_col].skew()
```

[30]:
```
age                        0.571660
annual_income              1.101398
annual_expenses            1.120099
bank_balance               1.054872
total_loans                0.169646
n_defaults                 0.000000
credit_score              -0.206382
prev_call_duration(min)    1.015489
n_prev_campaign_calls      1.121671
days_since_last_contact    0.000000
total_contacts_so_far      0.000000
dtype: float64
```

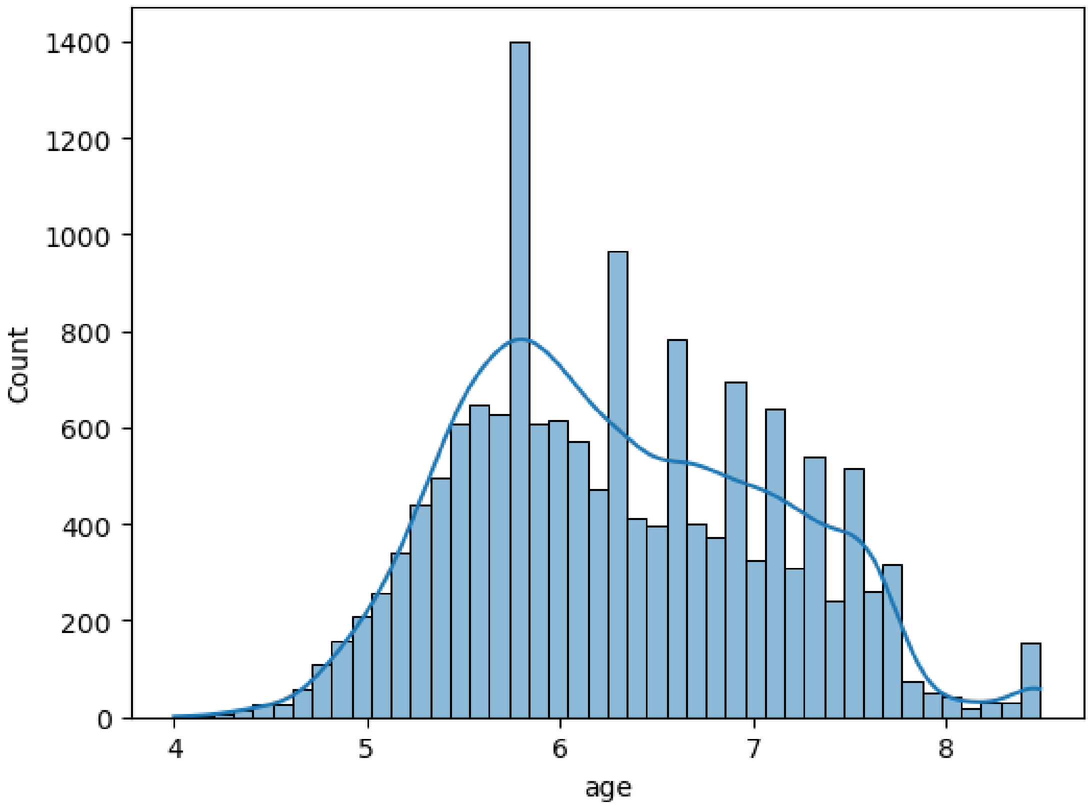
[31]:
```
test['age'] = np.sqrt(test['age'])
test['annual_income'] = np.sqrt(test['annual_income'])
test['annual_expenses'] = np.sqrt(test['annual_expenses'])
test['bank_balance'] = np.sqrt(test['bank_balance'])
test['n_prev_campaign_calls']  = np.sqrt(test['n_prev_campaign_calls'])
```

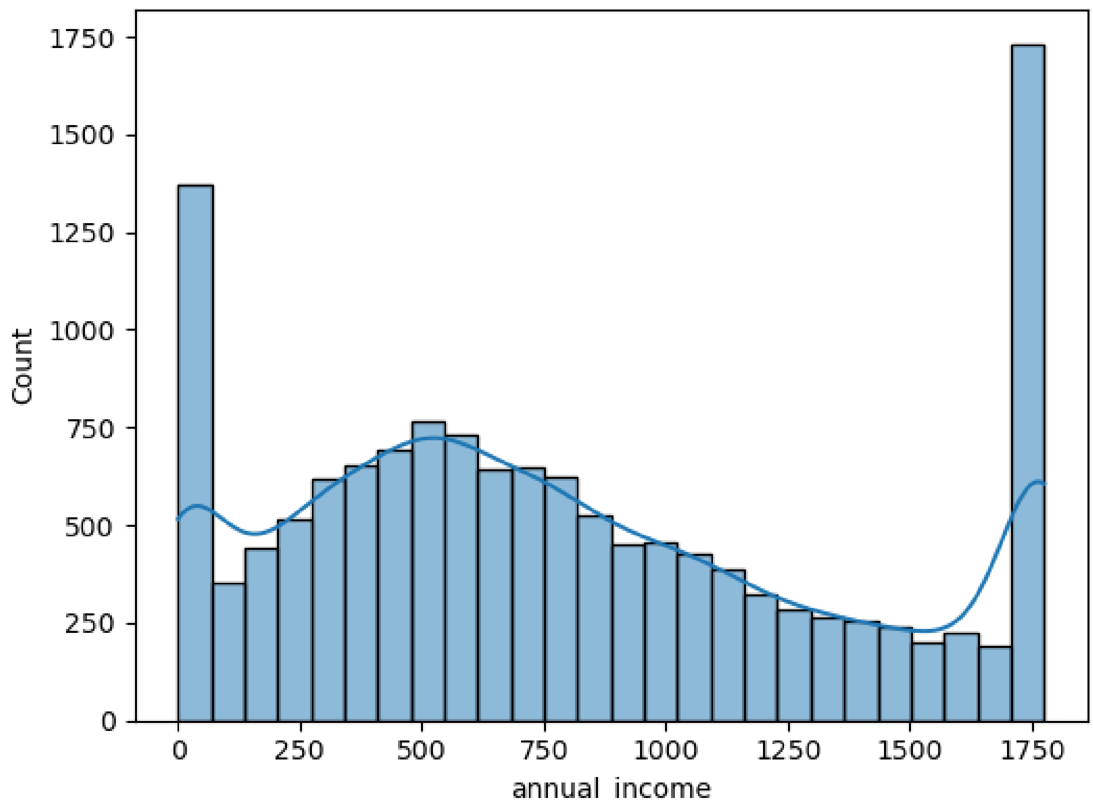
[32]:
```
test[num_col].skew()
```

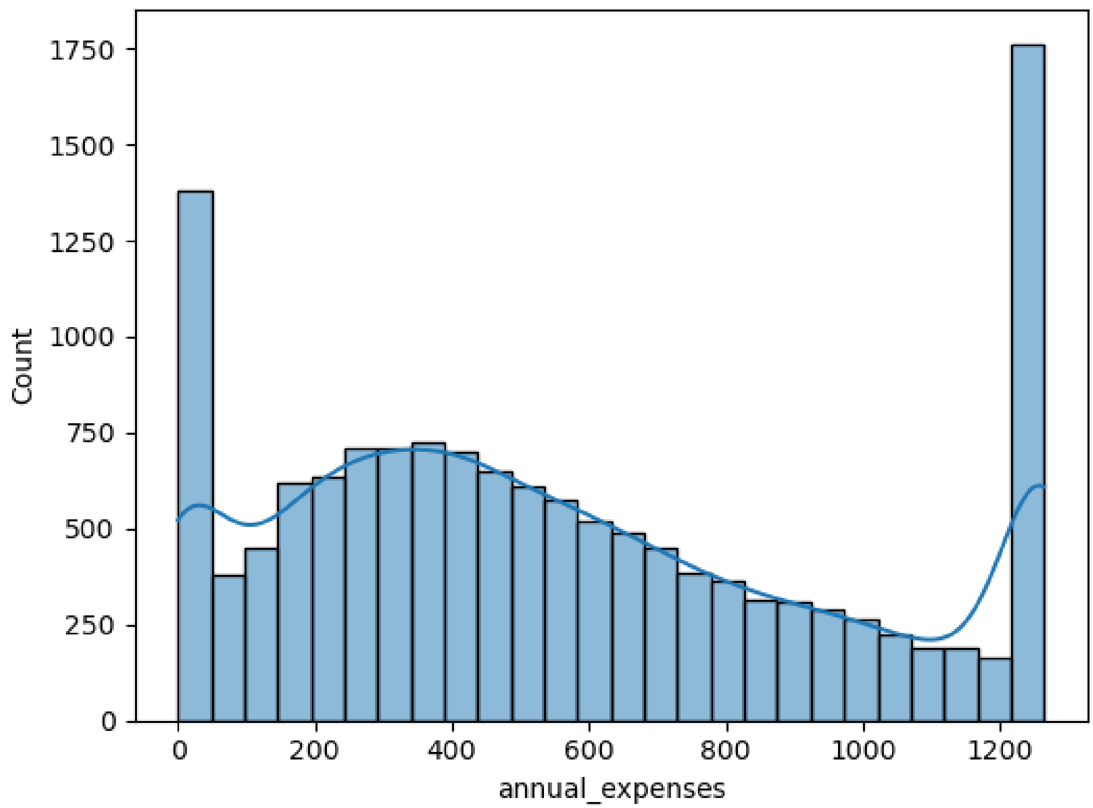
[32]:
```
age                        0.305854
annual_income              0.390806
annual_expenses            0.427500
bank_balance               0.338905
total_loans                0.169646
n_defaults                 0.000000
credit_score              -0.206382
```

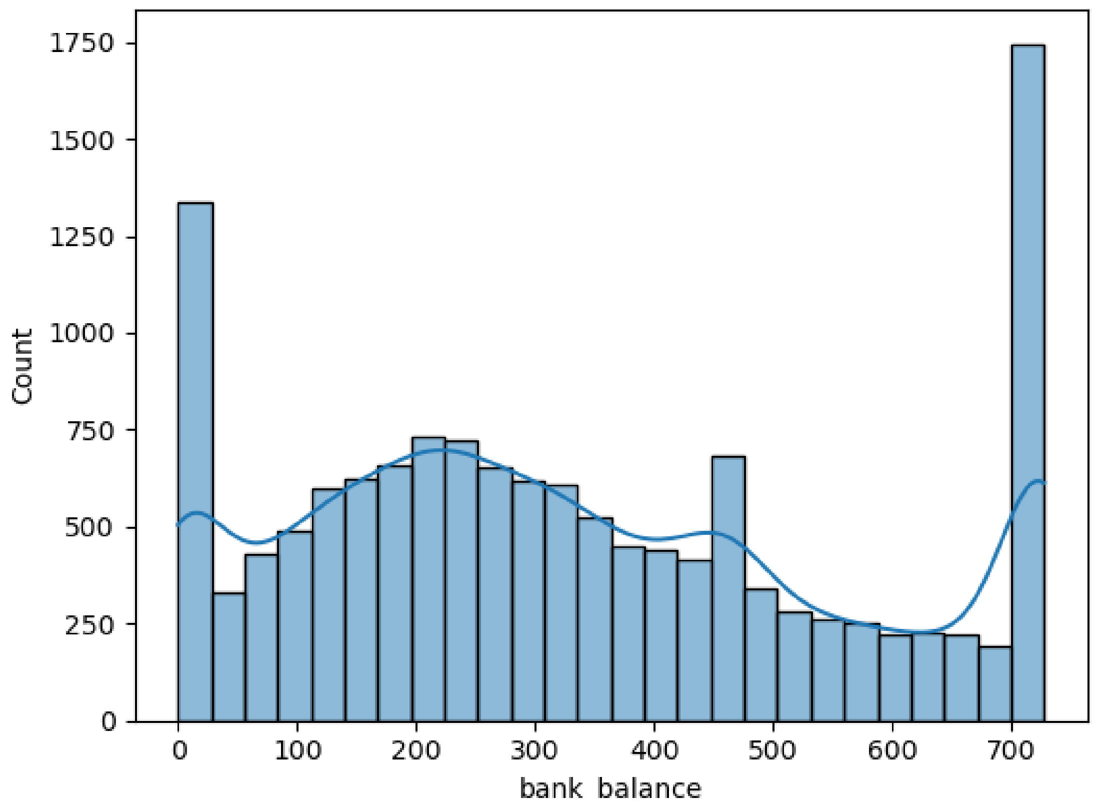
```
prev_call_duration(min)    1.015489
n_prev_campaign_calls      0.750028
days_since_last_contact    0.000000
total_contacts_so_far      0.000000
dtype: float64
```

[33]:
```python
for col in num_col:
    sns.histplot(data = train[col], kde =True)
    plt.show()
```

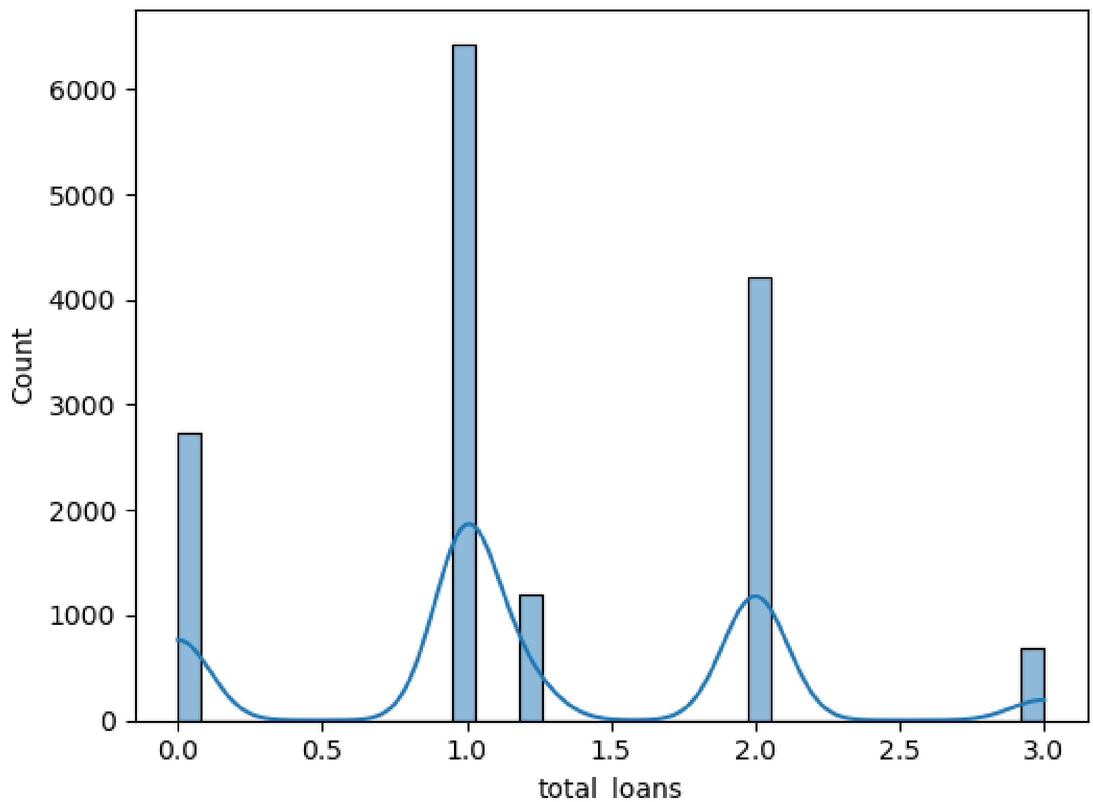


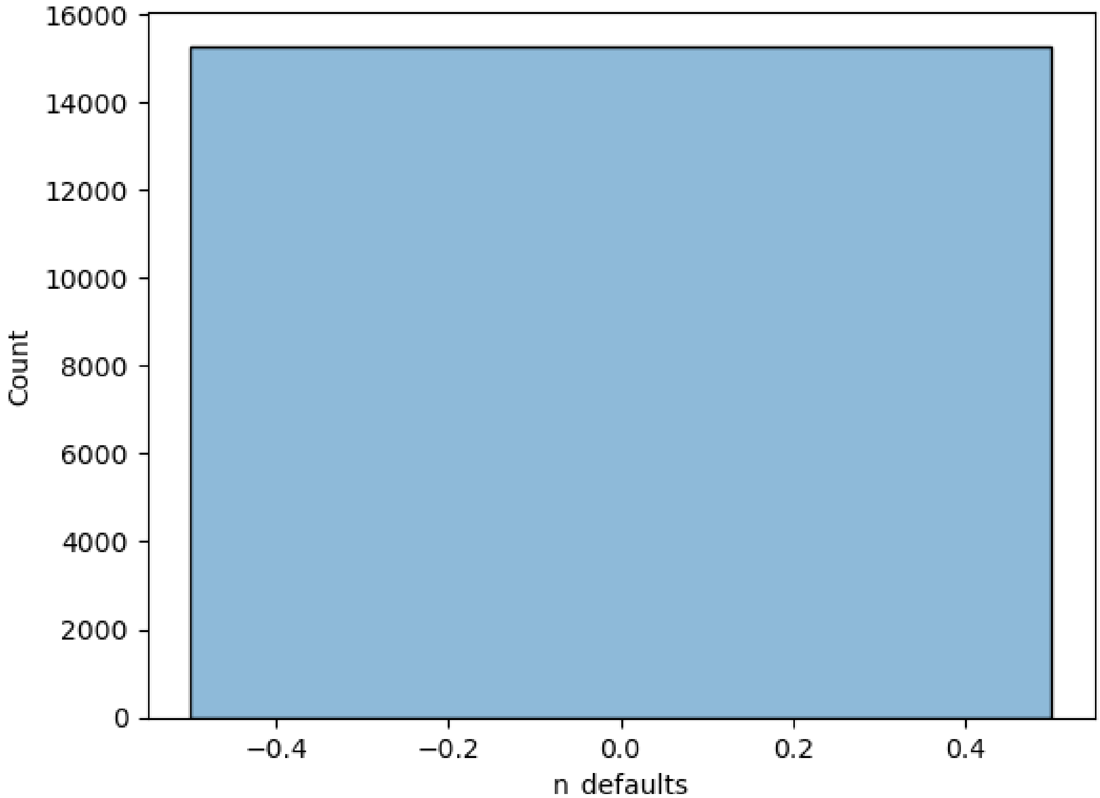

Count
annual_income
0
250
500
750
1000
1250
1500
1750

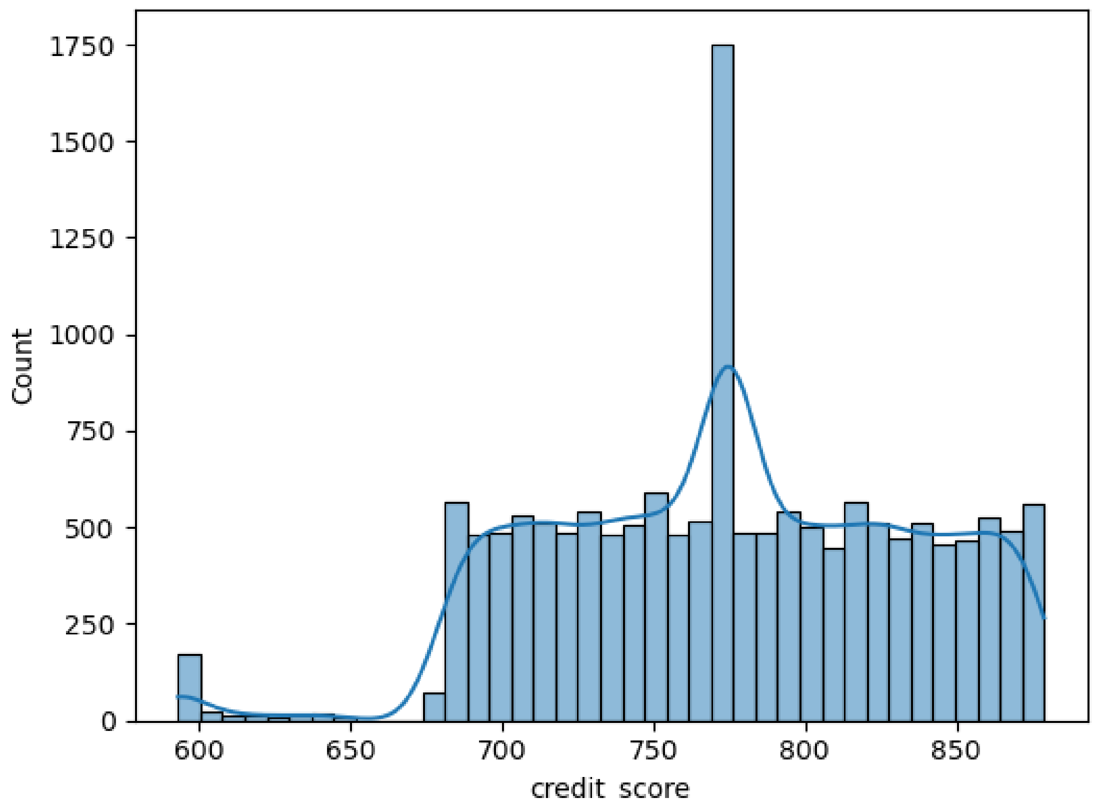

1750
1500
1250
1000
750
500
250
0
Count
0
200
400
600
800
1000
1200
annual_expenses



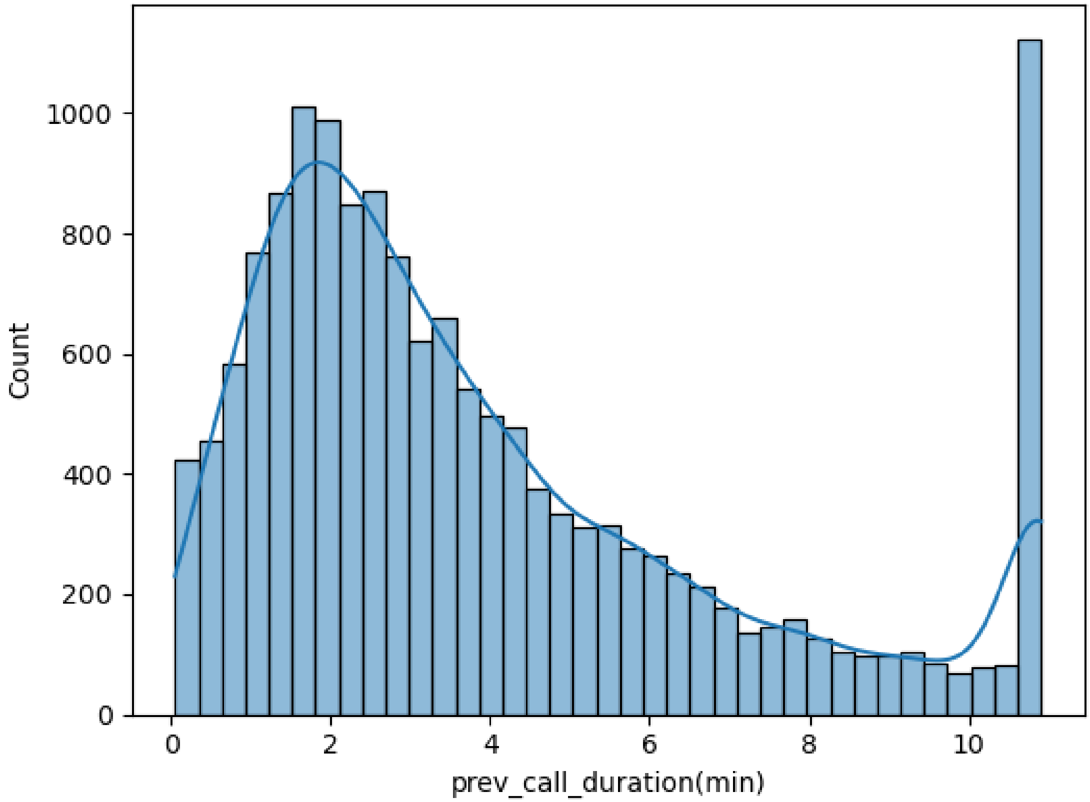
1750
1500
1250
1000
750
500
250
0
Count
0
100
200
300
400
500
600
700
bank_balance


6000
5000
4000
3000
2000
1000
0
Count
0.0
0.5
1.0
1.5
2.0
2.5
3.0
total_loans

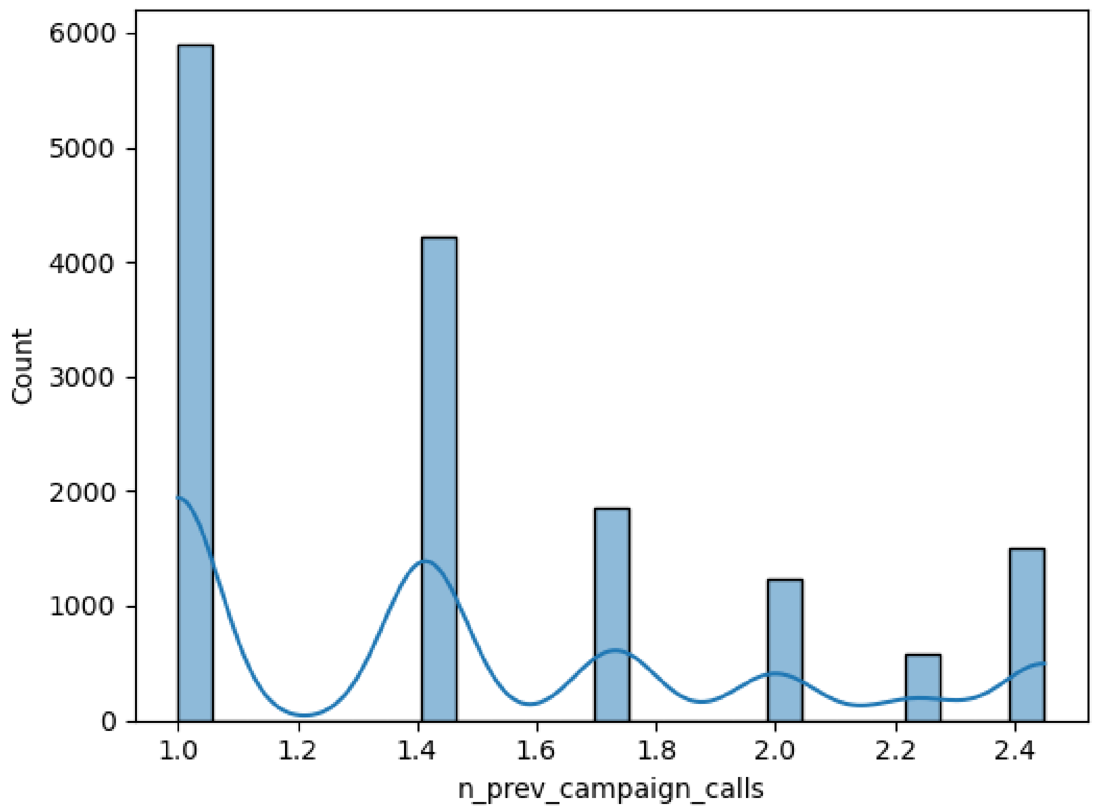

16000
14000
12000
10000
8000
6000
4000
2000
0
Count
−0.4
−0.2
0.0
0.2
0.4
n_defaults

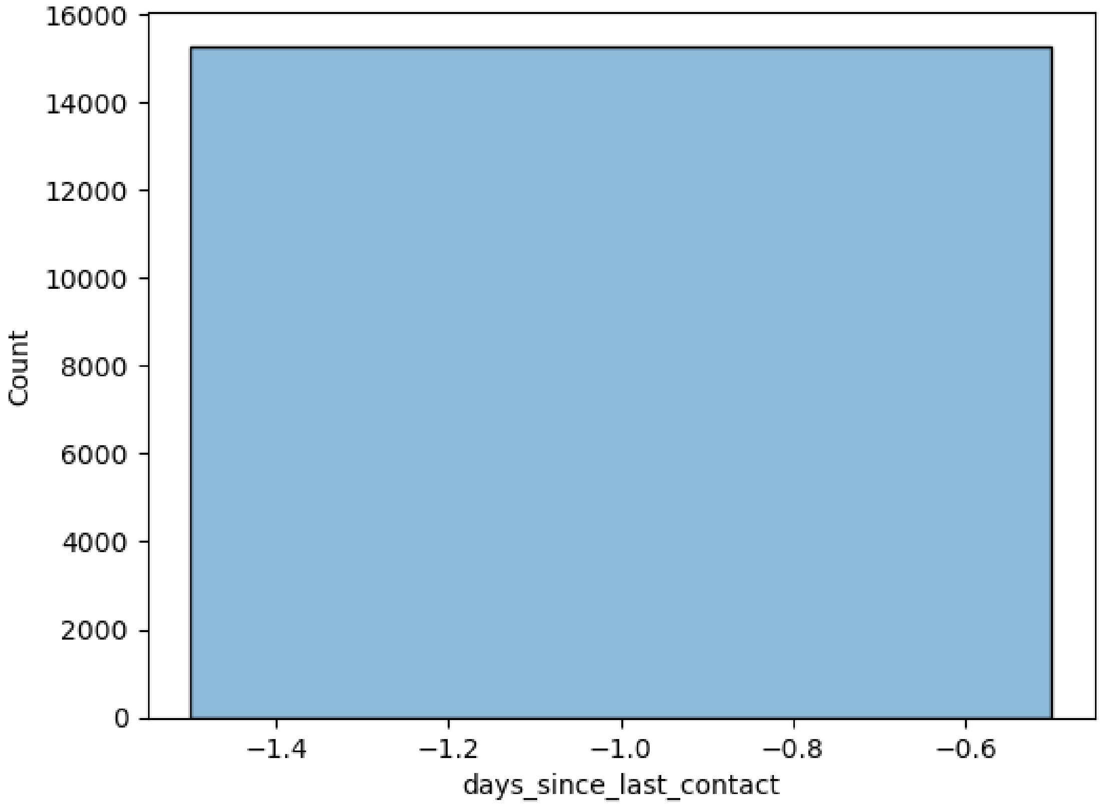


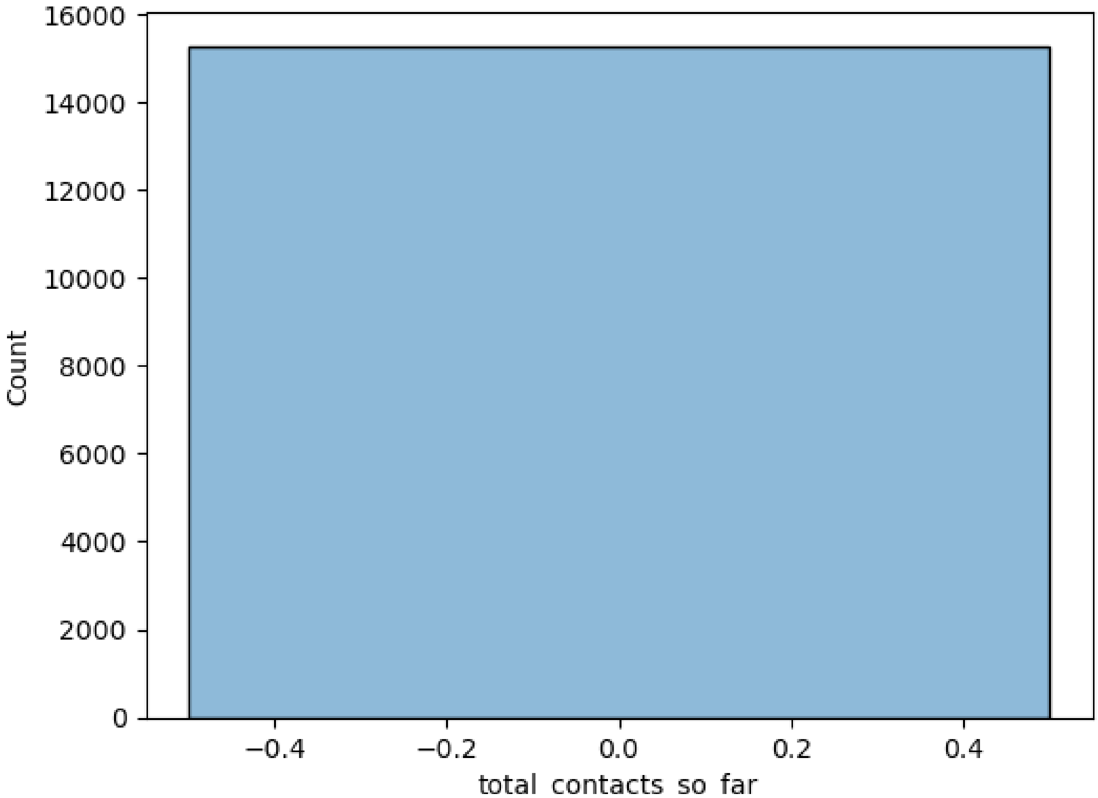
1750
1500
1250
1000
750
500
250
0
Count
600
650
700
750
800
850
credit_score



1000
800
600
400
200
0
Count
0
2
4
6
8
10
prev_call_duration(min)


6000
5000
4000
3000
2000
1000
0
Count
1.0
1.2
1.4
1.6
1.8
2.0
2.2
2.4
n_prev_campaign_calls

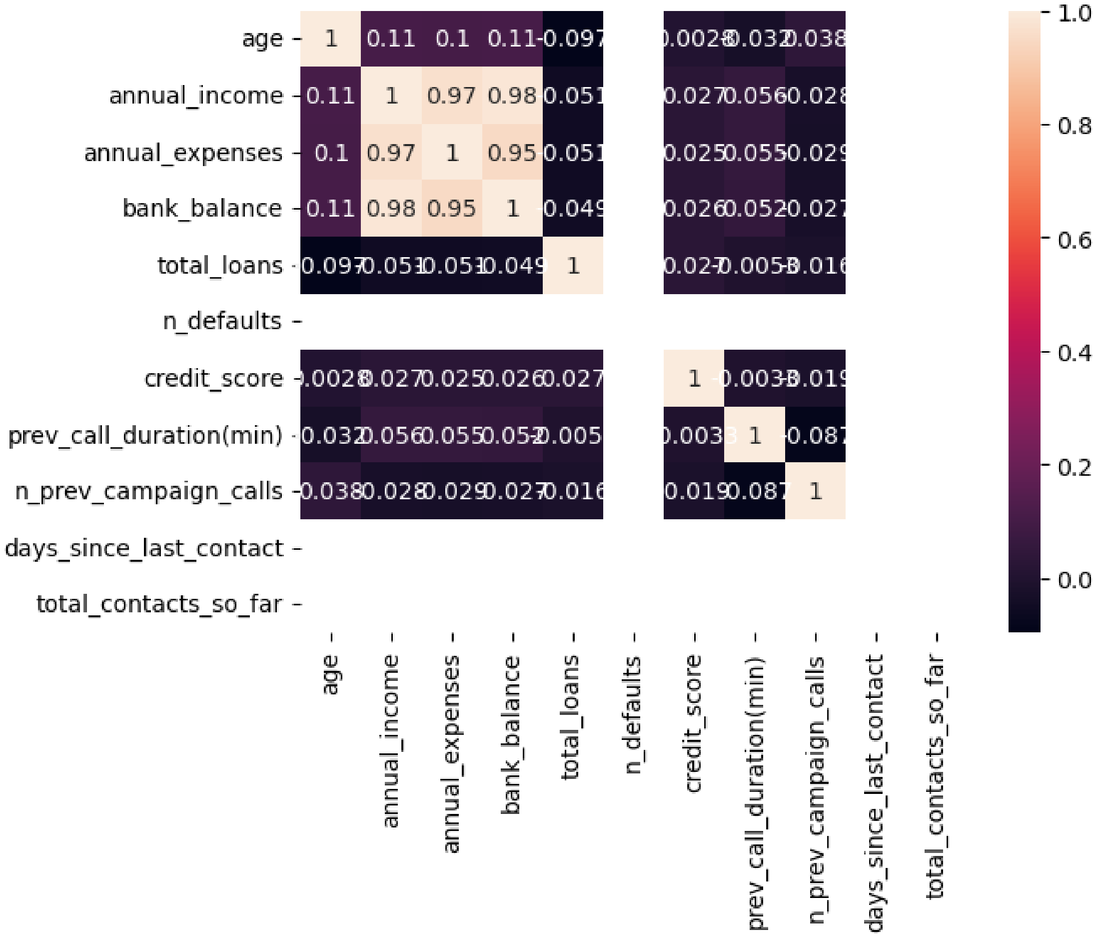

16000
14000
12000
10000
8000
6000
4000
2000
0
Count
−1.4
−1.2
−1.0
−0.8
−0.6
days_since_last_contact



[34]:
```
corr = train[num_col].corr()
```

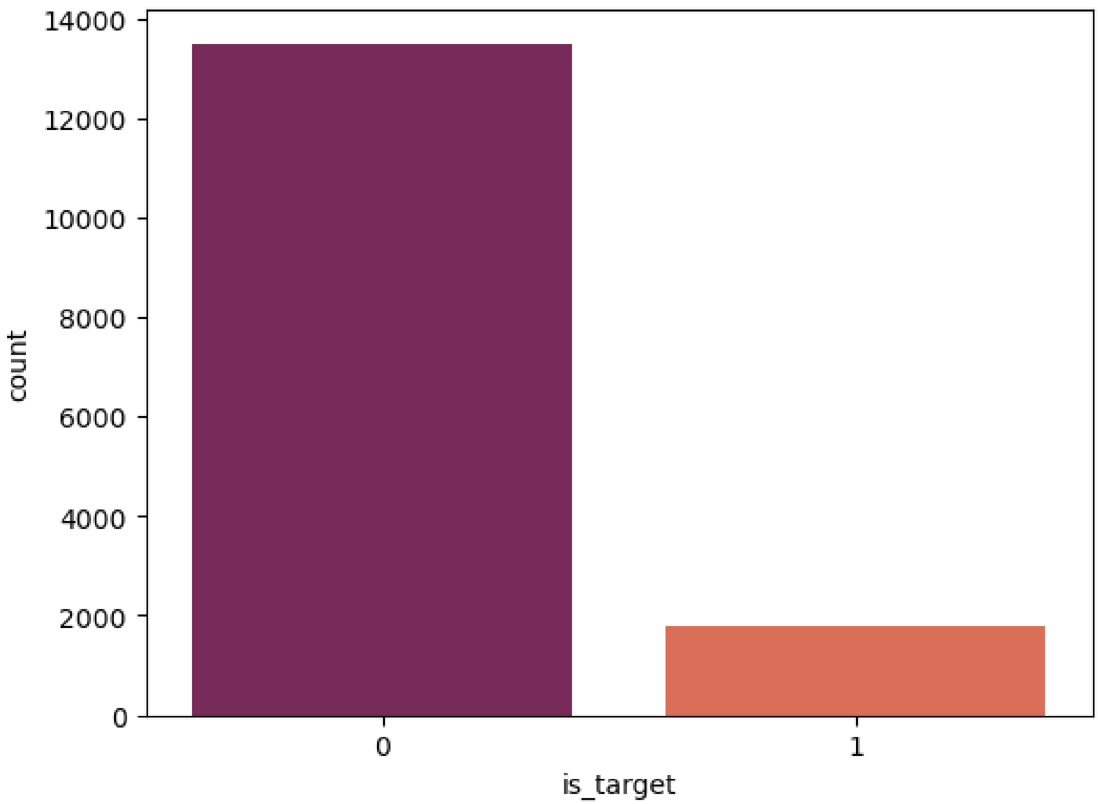
[35]:
```
sns.heatmap(corr, annot =True)
```

[35]: <Axes: >

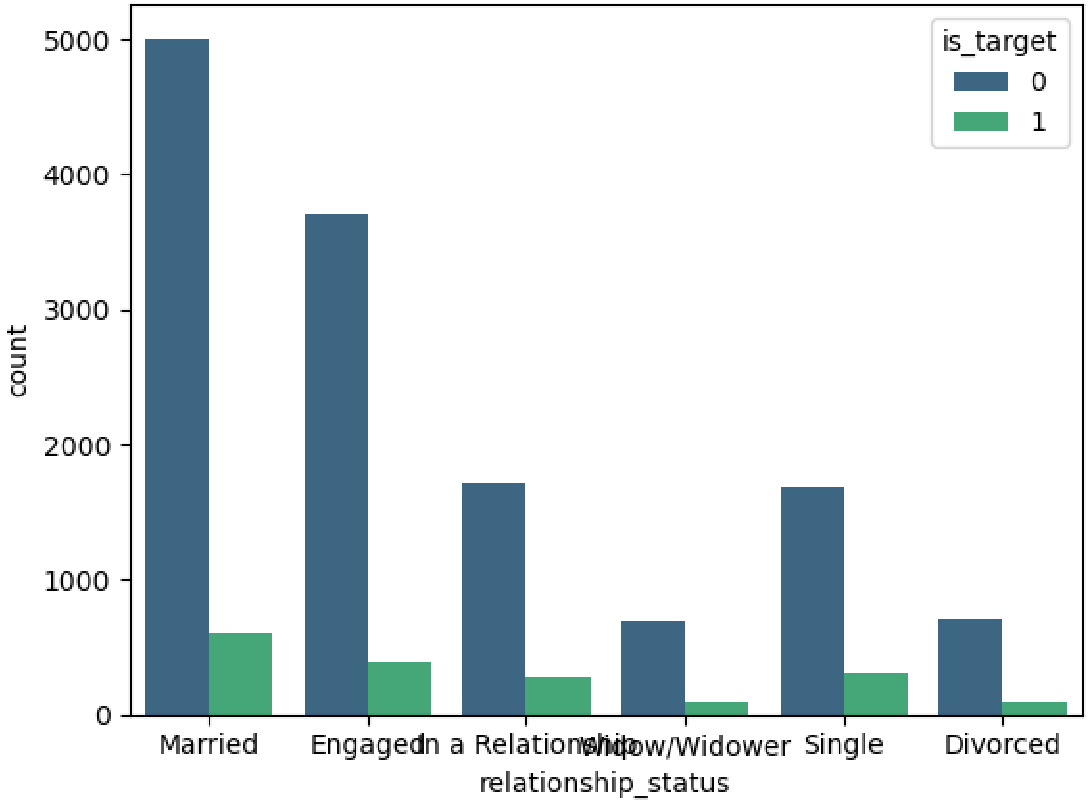


[36]:

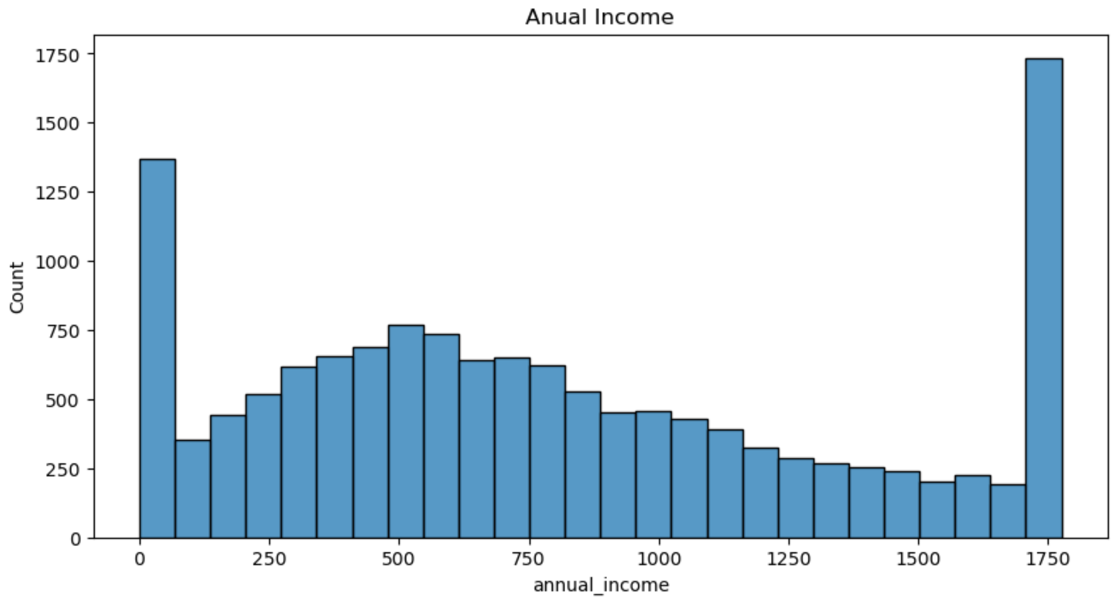
```
sns.countplot(data = train, x= 'is_target', palette = 'rocket')
plt.show()
```



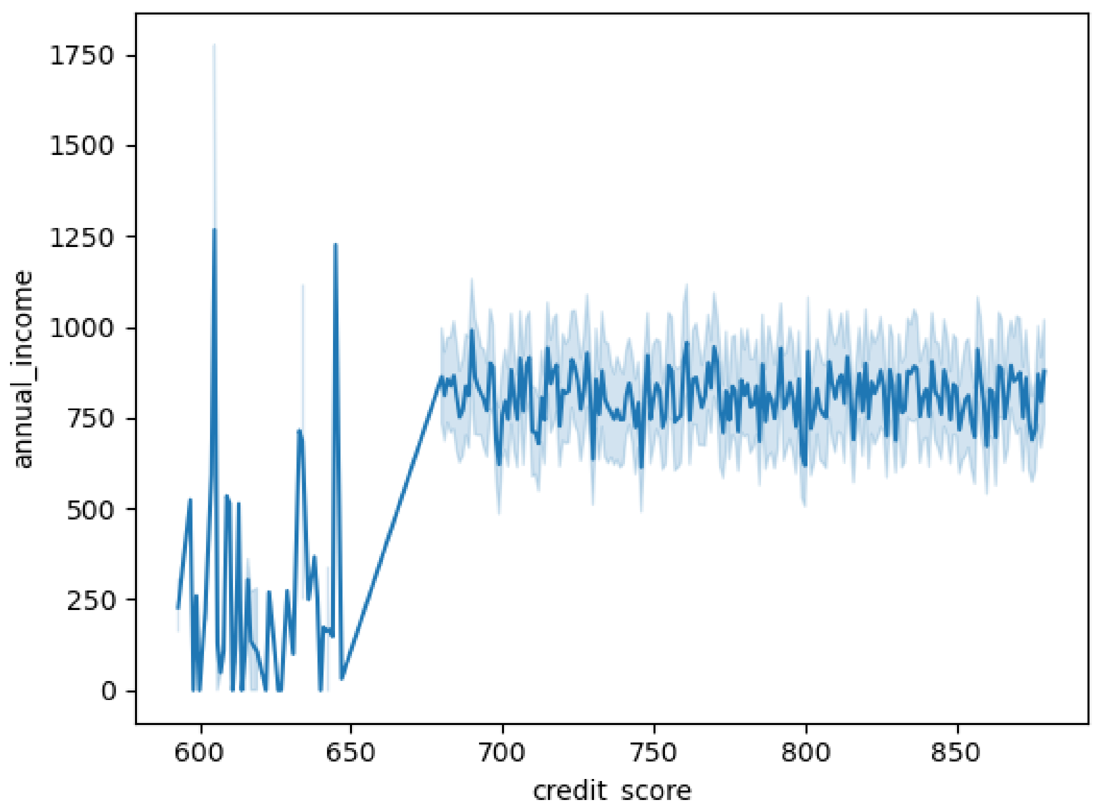
[37]:
```
sns.countplot(data=train, x= 'relationship_status', hue = 'is_target', palette␣
↪= 'viridis')
plt.show()
```



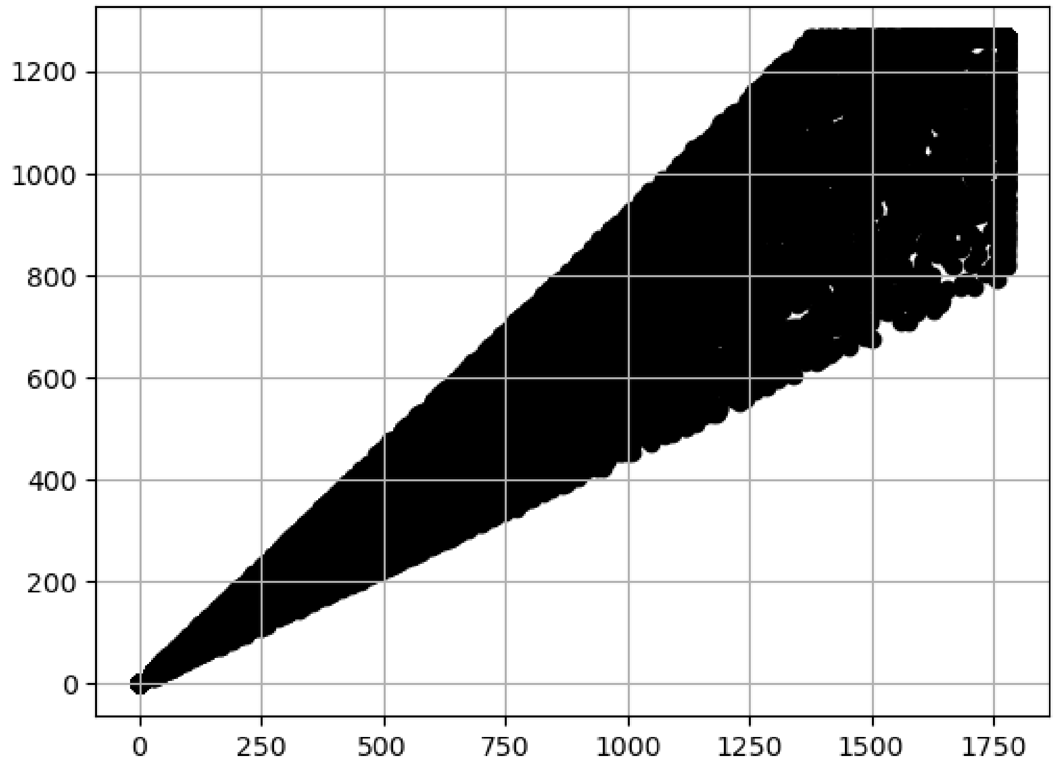
[38]:
```
#Visualizations
#Histogram
plt.figure(figsize= (10,5))
sns.histplot(x=train['annual_income'])
plt.title("Anual Income")
plt.show()
```



[39]:
```
sns.lineplot(data=train, x='credit_score', y='annual_income')
plt.show()
```



[40]:
```
plt.scatter(train['annual_income'], train['annual_expenses'], color = 'black')
plt.grid(True)
```



[41]:
```
# Encoding
encoder = LabelEncoder()

for col in cat_col:
    train[col] = encoder.fit_transform(train[col])
    test[col] = encoder.fit_transform(test[col])
```

[42]:
```
train.head()
```

[42]:
```
   customer_id       age  profession  relationship_status  qualification  \
0         6861  6.480741           4                    3              2
1         5988  6.480741           4                    1              0
2         7170  7.681146           1                    1              0
3         1627  6.708204           4                    3              2
4         1451  6.000000           0                    1              1

   annual_income  annual_expenses  bank_balance  has_home_loan  has_car_loan  \
0     394.315551       349.809437    160.614383              1             0
```

```
1    1775.680157     1264.587651    727.080736             1             0
2     644.577352      567.240604    266.533619             1             0
3       0.000000        0.000000      0.000000             1             0
4     469.324206      347.372930    202.797929             1             0

   … credit_score  customer_category  prev_call_duration(min)  \
0  …        868.0                  2                     8.45
1  …        729.0                  1                     3.37
2  …        792.0                  2                     2.27
3  …        787.0                  1                     4.12
4  …        752.0                  2                     2.68

   n_prev_campaign_calls  days_since_last_contact  total_contacts_so_far  \
0               1.000000                       -1                      0
1               1.414214                       -1                      0
2               1.000000                       -1                      0
3               1.414214                       -1                      0
4               2.449490                       -1                      0

   prev_outcome  targeted_mode_of_communication  targeted_month  is_target
0             3                               0               4          0
1             0                               2               9          0
2             2                               0               3          0
3             2                               0               5          0
4             2                               1               8          0

[5 rows x 22 columns]
```

[86]:
```
test.head()
```

[86]:
```
   customer_id       age  profession  relationship_status  qualification  \
0         2395  5.567764           4                    0              0
1         2948  7.000000          10                    5              1
2         1575  5.385165           9                    3              2
3         1364  5.656854           0                    2              0
4         5606  8.485281           1                    3              2

   annual_income  annual_expenses  bank_balance  has_home_loan  has_car_loan  \
0     461.442008       257.410456    202.277062              0             1
1     843.154354       737.146033    470.884615              0             0
2       0.000000         0.000000      0.000000              0             0
3    1521.592443      1286.793663    677.796703              1             0
4    1788.745342      1286.793663    732.400816              0             0

   … n_defaults  credit_score  customer_category  prev_call_duration(min)  \
0  …          0         634.0                  0                     2.58
1  …          0         738.0                  3                     3.27
```

```
2  …               0       742.0                       1                    1.85
3  …               0       793.0                       3                    1.28
4  …               0       808.0                       2                   10.53

   n_prev_campaign_calls  days_since_last_contact  total_contacts_so_far  \
0               1.000000                       -1                      0
1               1.000000                       -1                      0
2               2.236068                       -1                      0
3               2.449490                       -1                      0
4               1.000000                       -1                      0

   prev_outcome  targeted_mode_of_communication  targeted_month
0             2                               0               9
1             2                               0               4
2             2                               0               1
3             2                               2               5
4             3                               0               8

[5 rows x 21 columns]
```

[116]:

```python
# Splitting x and y
X = train.drop(columns=['is_target'])
y= train['is_target']
X
```

[116]:

```
       customer_id       age  profession  relationship_status  qualification  \
0             6861  6.480741           4                    3              2
1             5988  6.480741           4                    1              0
2             7170  7.681146           1                    1              0
3             1627  6.708204           4                    3              2
4             1451  6.000000           0                    1              1
…              …         …          …                    …              …
15279        14677  6.164414           1                    1              1
15280         5283  5.916080           7                    3              0
15281        12019  5.291503           4                    2              2
15282         2288  5.196152           7                    4              2
15283         8368  6.164414           0                    3              0

       annual_income  annual_expenses  bank_balance  has_home_loan  \
0         394.315551       349.809437    160.614383              1
1        1775.680157      1264.587651    727.080736              1
2         644.577352       567.240604    266.533619              1
3           0.000000         0.000000      0.000000              1
4         469.324206       347.372930    202.797929              1
…               …               …             …              …
15279     802.388174       564.225552    334.907862              1
15280    1314.760203       777.040784    549.729688              1
```

```
15281     278.215539       239.011417    114.443916              0
15282    1775.680157      1264.587651    727.080736              1
15283     534.141524       466.221035    222.858744              1

       has_car_loan  …  n_defaults  credit_score  customer_category  \
0                 0  …           0         868.0                  2
1                 0  …           0         729.0                  1
2                 0  …           0         792.0                  2
3                 0  …           0         787.0                  1
4                 0  …           0         752.0                  2
…               …  …           …           …                  …
15279             0  …           0         741.0                  2
15280             0  …           0         721.0                  0
15281             0  …           0         836.0                  3
15282             0  …           0         822.0                  3
15283             0  …           0         775.0                  1

       prev_call_duration(min)  n_prev_campaign_calls  \
0                        8.450               1.000000
1                        3.370               1.414214
2                        2.270               1.000000
3                        4.120               1.414214
4                        2.680               2.449490
…                          …                    …
15279                    3.230               2.000000
15280                    6.170               1.000000
15281                   10.905               1.414214
15282                   10.905               1.000000
15283                    3.920               1.414214

       days_since_last_contact  total_contacts_so_far  prev_outcome  \
0                           -1                      0             2
1                           -1                      0             2
2                           -1                      0             2
3                           -1                      0             2
4                           -1                      0             2
…                          …                    …           …
15279                       -1                      0             2
15280                       -1                      0             2
15281                       -1                      0             2
15282                       -1                      0             2
15283                       -1                      0             2

       targeted_mode_of_communication  targeted_month
0                                   0               4
1                                   2               9
2                                   0               3
```

```
3                                    0               5
4                                    1               8
…                                 …             …
15279                                1               6
15280                                1               8
15281                                0              10
15282                                0               5
15283                                0               0

[15284 rows x 21 columns]
```

[118]:
```
selector = SelectKBest(score_func = f_classif, k=10)
X_selected = selector.fit_transform(X,y)
selected_features = X.columns[selector.get_support()].tolist()
selected_features
```

[118]:
```
['customer_id',
 'qualification',
 'annual_income',
 'annual_expenses',
 'bank_balance',
 'has_home_loan',
 'total_loans',
 'prev_call_duration(min)',
 'n_prev_campaign_calls',
 'targeted_mode_of_communication']
```

[120]:
```
# Train test split
X_train, X_test, y_train, y_test = train_test_split(X_selected,y, test_size=0.
 ↪2, random_state = 42)
```

[124]:
```
# Feature scaling
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.fit_transform(X_test)
# Testing data
scaled_X_test = scaler.fit_transform(test[selected_features])
```

[126]:
```
model = RandomForestClassifier(random_state=42)
model.fit(X_train,y_train)
```

[126]:
```
RandomForestClassifier(random_state=42)
```

[130]:
```
y_pred = model.predict(X_test)
acc = accuracy_score(y_test, y_pred)
cr = classification_report(y_test, y_pred)
cf = confusion_matrix(y_test,y_pred)
```

```python
print("accuracy: ", acc)
print(cr)
print(cf)
```

```
accuracy:  0.8979391560353287
              precision    recall  f1-score   support

           0       0.92      0.96      0.94      2711
           1       0.57      0.39      0.46       346

    accuracy                           0.90      3057
   macro avg       0.75      0.68      0.70      3057
weighted avg       0.89      0.90      0.89      3057

[[2611  100]
 [ 212  134]]
```

[ ]:

```python

```