Name- Hritik Mathur
Prn- 18070123057
Batch0- EA3

Bank_customer_segmentation

December 2, 2020

[3]:
```
!pip install jupyterthemes
```

```
Defaulting to user installation because normal site-packages is not writeable
Collecting jupyterthemes
  Downloading jupyterthemes-0.20.0-py2.py3-none-any.whl (7.0 MB)
Requirement already satisfied: jupyter-core in
c:\users\asus\appdata\roaming\python\python38\site-packages (from jupyterthemes)
(4.6.3)
Requirement already satisfied: ipython>=5.4.1 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from jupyterthemes)
(7.18.1)
Collecting lesscpy>=0.11.2
  Downloading lesscpy-0.14.0-py2.py3-none-any.whl (46 kB)
Requirement already satisfied: notebook>=5.6.0 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from jupyterthemes)
(6.1.3)
Requirement already satisfied: matplotlib>=1.4.3 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from jupyterthemes)
(3.3.1)
Requirement already satisfied: traitlets in c:\python38\lib\site-packages (from
jupyter-core->jupyterthemes) (5.0.0)
Requirement already satisfied: pywin32>=1.0; sys_platform == "win32" in
c:\python38\lib\site-packages (from jupyter-core->jupyterthemes) (228)
Requirement already satisfied: setuptools>=18.5 in c:\python38\lib\site-packages
(from ipython>=5.4.1->jupyterthemes) (47.1.0)
Requirement already satisfied: pygments in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
ipython>=5.4.1->jupyterthemes) (2.6.1)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
ipython>=5.4.1->jupyterthemes) (3.0.7)
Requirement already satisfied: jedi>=0.10 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
ipython>=5.4.1->jupyterthemes) (0.17.2)
Requirement already satisfied: backcall in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
ipython>=5.4.1->jupyterthemes) (0.2.0)
```

```
Requirement already satisfied: decorator in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
ipython>=5.4.1->jupyterthemes) (4.4.2)
Requirement already satisfied: colorama; sys_platform == "win32" in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
ipython>=5.4.1->jupyterthemes) (0.4.3)
Requirement already satisfied: pickleshare in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
ipython>=5.4.1->jupyterthemes) (0.7.5)
Collecting ply
  Downloading ply-3.11-py2.py3-none-any.whl (49 kB)
Requirement already satisfied: six in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
lesscpy>=0.11.2->jupyterthemes) (1.15.0)
Requirement already satisfied: tornado>=5.0 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (6.0.4)
Requirement already satisfied: ipython-genutils in c:\python38\lib\site-packages
(from notebook>=5.6.0->jupyterthemes) (0.2.0)
Requirement already satisfied: ipykernel in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (5.3.4)
Requirement already satisfied: prometheus-client in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (0.8.0)
Requirement already satisfied: jinja2 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (2.11.2)
Requirement already satisfied: jupyter-client>=5.3.4 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (6.1.7)
Requirement already satisfied: pyzmq>=17 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (19.0.2)
Requirement already satisfied: Send2Trash in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (1.5.0)
Requirement already satisfied: argon2-cffi in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (20.1.0)
Requirement already satisfied: terminado>=0.8.3 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (0.8.3)
Requirement already satisfied: nbformat in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (5.0.7)
Requirement already satisfied: nbconvert in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
notebook>=5.6.0->jupyterthemes) (5.6.1)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
matplotlib>=1.4.3->jupyterthemes) (2.4.7)
Requirement already satisfied: certifi>=2020.06.20 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
matplotlib>=1.4.3->jupyterthemes) (2020.6.20)
Requirement already satisfied: cycler>=0.10 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
matplotlib>=1.4.3->jupyterthemes) (0.10.0)
Requirement already satisfied: python-dateutil>=2.1 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
matplotlib>=1.4.3->jupyterthemes) (2.8.1)
Requirement already satisfied: numpy>=1.15 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
matplotlib>=1.4.3->jupyterthemes) (1.18.5)
Requirement already satisfied: kiwisolver>=1.0.1 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
matplotlib>=1.4.3->jupyterthemes) (1.2.0)
Requirement already satisfied: pillow>=6.2.0 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
matplotlib>=1.4.3->jupyterthemes) (7.2.0)
Requirement already satisfied: wcwidth in
c:\users\asus\appdata\roaming\python\python38\site-packages (from prompt-
toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0->ipython>=5.4.1->jupyterthemes) (0.2.5)
Requirement already satisfied: parso<0.8.0,>=0.7.0 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
jedi>=0.10->ipython>=5.4.1->jupyterthemes) (0.7.1)
Requirement already satisfied: MarkupSafe>=0.23 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
jinja2->notebook>=5.6.0->jupyterthemes) (1.1.1)
Requirement already satisfied: cffi>=1.0.0 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
argon2-cffi->notebook>=5.6.0->jupyterthemes) (1.14.2)
Requirement already satisfied: pywinpty>=0.5; os_name == "nt" in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
terminado>=0.8.3->notebook>=5.6.0->jupyterthemes) (0.5.7)
Requirement already satisfied: jsonschema!=2.5.0,>=2.4 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
nbformat->notebook>=5.6.0->jupyterthemes) (3.2.0)
Requirement already satisfied: bleach in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
nbconvert->notebook>=5.6.0->jupyterthemes) (3.1.5)
Requirement already satisfied: mistune<2,>=0.8.1 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
nbconvert->notebook>=5.6.0->jupyterthemes) (0.8.4)
Requirement already satisfied: testpath in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
nbconvert->notebook>=5.6.0->jupyterthemes) (0.4.4)
Requirement already satisfied: entrypoints>=0.2.2 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
nbconvert->notebook>=5.6.0->jupyterthemes) (0.3)
Requirement already satisfied: defusedxml in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
nbconvert->notebook>=5.6.0->jupyterthemes) (0.6.0)
Requirement already satisfied: pandocfilters>=1.4.1 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
nbconvert->notebook>=5.6.0->jupyterthemes) (1.4.2)
Requirement already satisfied: pycparser in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
cffi>=1.0.0->argon2-cffi->notebook>=5.6.0->jupyterthemes) (2.20)
Requirement already satisfied: pyrsistent>=0.14.0 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
jsonschema!=2.5.0,>=2.4->nbformat->notebook>=5.6.0->jupyterthemes) (0.16.0)
Requirement already satisfied: attrs>=17.4.0 in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
jsonschema!=2.5.0,>=2.4->nbformat->notebook>=5.6.0->jupyterthemes) (20.1.0)
Requirement already satisfied: packaging in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
bleach->nbconvert->notebook>=5.6.0->jupyterthemes) (20.4)
Requirement already satisfied: webencodings in
c:\users\asus\appdata\roaming\python\python38\site-packages (from
bleach->nbconvert->notebook>=5.6.0->jupyterthemes) (0.5.1)
Installing collected packages: ply, lesscpy, jupyterthemes
Successfully installed jupyterthemes-0.20.0 lesscpy-0.14.0 ply-3.11
```

[7]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler,normalize
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
from jupyterthemes import jtplot
jtplot.style(theme='monokai',context='notebook',ticks=True,grid=False)
```

[8]:
```
creditcard_df = pd.read_csv("Marketing_data.csv")
```

[9]:
```
creditcard_df
```

[9]:
```
     CUST_ID      BALANCE  BALANCE_FREQUENCY  PURCHASES  ONEOFF_PURCHASES  \
0     C10001    40.900749           0.818182      95.40              0.00
1     C10002  3202.467416           0.909091       0.00              0.00
2     C10003  2495.148862           1.000000     773.17            773.17
3     C10004  1666.670542           0.636364    1499.00           1499.00
4     C10005   817.714335           1.000000      16.00             16.00
...      ...          ...                ...        ...               ...
```

```
8945  C19186     28.493517               1.000000      291.12              0.00
8946  C19187     19.183215               1.000000      300.00              0.00
8947  C19188     23.398673               0.833333      144.40              0.00
8948  C19189     13.457564               0.833333        0.00              0.00
8949  C19190    372.708075               0.666667     1093.25           1093.25

      INSTALLMENTS_PURCHASES   CASH_ADVANCE  PURCHASES_FREQUENCY    \
0                      95.40      0.000000              0.166667
1                       0.00   6442.945483              0.000000
2                       0.00      0.000000              1.000000
3                       0.00    205.788017              0.083333
4                       0.00      0.000000              0.083333
...                      ...           ...                   ...
8945                  291.12      0.000000              1.000000
8946                  300.00      0.000000              1.000000
8947                  144.40      0.000000              0.833333
8948                    0.00     36.558778              0.000000
8949                    0.00    127.040008              0.666667

      ONEOFF_PURCHASES_FREQUENCY  PURCHASES_INSTALLMENTS_FREQUENCY     \
0                       0.000000                          0.083333
1                       0.000000                          0.000000
2                       1.000000                          0.000000
3                       0.083333                          0.000000
4                       0.083333                          0.000000
...                          ...                               ...
8945                    0.000000                          0.833333
8946                    0.000000                          0.833333
8947                    0.000000                          0.666667
8948                    0.000000                          0.000000
8949                    0.666667                          0.000000

      CASH_ADVANCE_FREQUENCY  CASH_ADVANCE_TRX   PURCHASES_TRX  CREDIT_LIMIT   \
0                   0.000000                 0               2        1000.0
1                   0.250000                 4               0        7000.0
2                   0.000000                 0              12        7500.0
3                   0.083333                 1               1        7500.0
4                   0.000000                 0               1        1200.0
...                      ...               ...             ...           ...
8945                0.000000                 0               6        1000.0
8946                0.000000                 0               6        1000.0
8947                0.000000                 0               5        1000.0
8948                0.166667                 2               0         500.0
8949                0.333333                 2              23        1200.0

        PAYMENTS   MINIMUM_PAYMENTS   PRC_FULL_PAYMENT  TENURE
0     201.802084         139.509787           0.000000      12
```

```
1      4103.032597     1072.340217        0.222222      12
2       622.066742      627.284787        0.000000      12
3         0.000000             NaN        0.000000      12
4       678.334763      244.791237        0.000000      12
...            ...             ...             ...     ...
8945    325.594462       48.886365        0.500000       6
8946    275.861322             NaN        0.000000       6
8947     81.270775       82.418369        0.250000       6
8948     52.549959       55.755628        0.250000       6
8949     63.165404       88.288956        0.000000       6

[8950 rows x 18 columns]
```

[10]:
```
creditcard_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8950 entries, 0 to 8949
Data columns (total 18 columns):
 #   Column                            Non-Null Count  Dtype
---  ------                            --------------  -----
 0   CUST_ID                           8950 non-null   object
 1   BALANCE                           8950 non-null   float64
 2   BALANCE_FREQUENCY                 8950 non-null   float64
 3   PURCHASES                         8950 non-null   float64
 4   ONEOFF_PURCHASES                  8950 non-null   float64
 5   INSTALLMENTS_PURCHASES            8950 non-null   float64
 6   CASH_ADVANCE                      8950 non-null   float64
 7   PURCHASES_FREQUENCY               8950 non-null   float64
 8   ONEOFF_PURCHASES_FREQUENCY        8950 non-null   float64
 9   PURCHASES_INSTALLMENTS_FREQUENCY  8950 non-null   float64
 10  CASH_ADVANCE_FREQUENCY            8950 non-null   float64
 11  CASH_ADVANCE_TRX                  8950 non-null   int64
 12  PURCHASES_TRX                     8950 non-null   int64
 13  CREDIT_LIMIT                      8949 non-null   float64
 14  PAYMENTS                          8950 non-null   float64
 15  MINIMUM_PAYMENTS                  8637 non-null   float64
 16  PRC_FULL_PAYMENT                  8950 non-null   float64
 17  TENURE                            8950 non-null   int64
dtypes: float64(14), int64(3), object(1)
memory usage: 1.2+ MB
```

[11]:
```
creditcard_df['BALANCE'].describe()
```

[11]:
```
count    8950.000000
mean     1564.474828
std      2081.531879
min         0.000000
25%       128.281915
```

```
50%          873.385231
75%         2054.140036
max        19043.138560
Name: BALANCE, dtype: float64
```

[12]:
```
creditcard_df.describe()
```

[12]:
```
            BALANCE  BALANCE_FREQUENCY     PURCHASES  ONEOFF_PURCHASES  \
count   8950.000000        8950.000000   8950.000000       8950.000000
mean    1564.474828           0.877271   1003.204834        592.437371
std     2081.531879           0.236904   2136.634782       1659.887917
min        0.000000           0.000000      0.000000          0.000000
25%      128.281915           0.888889     39.635000          0.000000
50%      873.385231           1.000000    361.280000         38.000000
75%     2054.140036           1.000000   1110.130000        577.405000
max    19043.138560           1.000000  49039.570000      40761.250000

       INSTALLMENTS_PURCHASES  CASH_ADVANCE  PURCHASES_FREQUENCY  \
count             8950.000000   8950.000000          8950.000000
mean               411.067645    978.871112             0.490351
std                904.338115   2097.163877             0.401371
min                  0.000000      0.000000             0.000000
25%                  0.000000      0.000000             0.083333
50%                 89.000000      0.000000             0.500000
75%                468.637500   1113.821139             0.916667
max              22500.000000  47137.211760             1.000000

       ONEOFF_PURCHASES_FREQUENCY  PURCHASES_INSTALLMENTS_FREQUENCY  \
count                 8950.000000                       8950.000000
mean                     0.202458                          0.364437
std                      0.298336                          0.397448
min                      0.000000                          0.000000
25%                      0.000000                          0.000000
50%                      0.083333                          0.166667
75%                      0.300000                          0.750000
max                      1.000000                          1.000000

       CASH_ADVANCE_FREQUENCY  CASH_ADVANCE_TRX  PURCHASES_TRX  CREDIT_LIMIT  \
count             8950.000000       8950.000000    8950.000000   8949.000000
mean                 0.135144          3.248827      14.709832   4494.449450
std                  0.200121          6.824647      24.857649   3638.815725
min                  0.000000          0.000000       0.000000     50.000000
25%                  0.000000          0.000000       1.000000   1600.000000
50%                  0.000000          0.000000       7.000000   3000.000000
75%                  0.222222          4.000000      17.000000   6500.000000
max                  1.500000        123.000000     358.000000  30000.000000

           PAYMENTS  MINIMUM_PAYMENTS  PRC_FULL_PAYMENT       TENURE
```

```
count    8950.000000        8637.000000       8950.000000   8950.000000
mean     1733.143852         864.206542          0.153715     11.517318
std      2895.063757        2372.446607          0.292499      1.338331
min         0.000000           0.019163          0.000000      6.000000
25%       383.276166         169.123707          0.000000     12.000000
50%       856.901546         312.343947          0.000000     12.000000
75%      1901.134317         825.485459          0.142857     12.000000
max     50721.483360       76406.207520          1.000000     12.000000
```

[13]:
```
creditcard_df[creditcard_df['ONEOFF_PURCHASES'] == 40761.25]
```

[13]:
```
    CUST_ID       BALANCE  BALANCE_FREQUENCY  PURCHASES  ONEOFF_PURCHASES  \
550  C10574  11547.52001                1.0   49039.57          40761.25

     INSTALLMENTS_PURCHASES  CASH_ADVANCE  PURCHASES_FREQUENCY  \
550                 8278.32    558.166886                  1.0

     ONEOFF_PURCHASES_FREQUENCY  PURCHASES_INSTALLMENTS_FREQUENCY  \
550                         1.0                          0.916667

     CASH_ADVANCE_FREQUENCY  CASH_ADVANCE_TRX  PURCHASES_TRX  CREDIT_LIMIT  \
550                0.083333                 1            101       22500.0

        PAYMENTS  MINIMUM_PAYMENTS  PRC_FULL_PAYMENT  TENURE
550  46930.59824       2974.069421              0.25      12
```

[14]:
```
creditcard_df['CASH_ADVANCE'].max()
```

[14]:
```
47137.211760000006
```

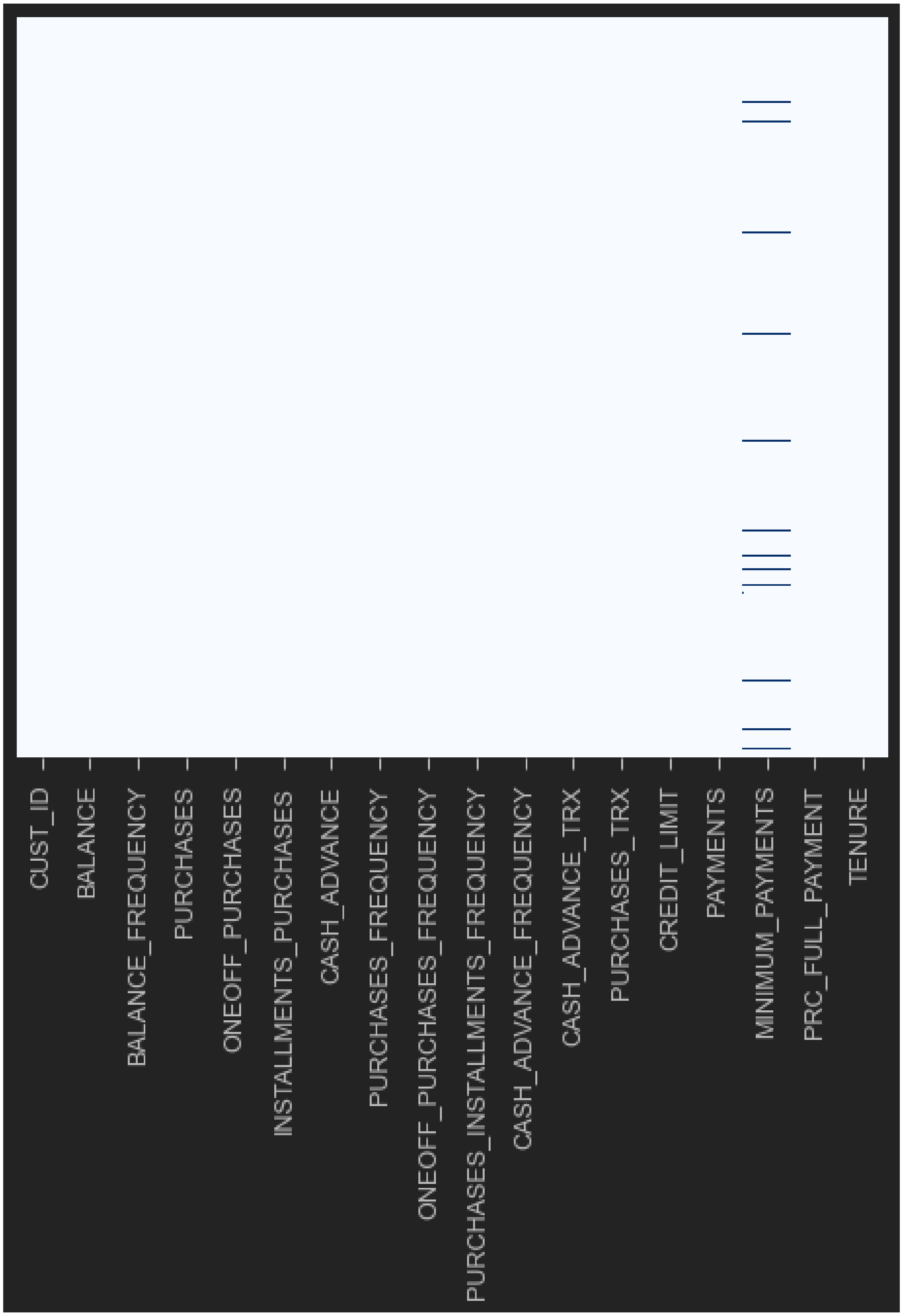
[15]:
```
creditcard_df[creditcard_df['CASH_ADVANCE'] == 47137.211760000006]
```

[15]:
```
     CUST_ID       BALANCE  BALANCE_FREQUENCY  PURCHASES  ONEOFF_PURCHASES  \
2159  C12226  10905.05381                1.0     431.93             133.5

      INSTALLMENTS_PURCHASES  CASH_ADVANCE  PURCHASES_FREQUENCY  \
2159                  298.43   47137.21176             0.583333

      ONEOFF_PURCHASES_FREQUENCY  PURCHASES_INSTALLMENTS_FREQUENCY  \
2159                        0.25                               0.5

      CASH_ADVANCE_FREQUENCY  CASH_ADVANCE_TRX  PURCHASES_TRX  CREDIT_LIMIT  \
2159                     1.0               123             21       19600.0

         PAYMENTS  MINIMUM_PAYMENTS  PRC_FULL_PAYMENT  TENURE
2159  39048.59762       5394.173671               0.0      12
```

1 VISUALIZATION

[16]:
```
sns.heatmap(creditcard_df.isnull(),yticklabels=False,cbar=False,cmap='Blues')
```

[16] :
```
<AxesSubplot:>
```


CUST_ID
BALANCE
BALANCE_FREQUENCY
PURCHASES
ONEOFF_PURCHASES
INSTALLMENTS_PURCHASES
CASH_ADVANCE
PURCHASES_FREQUENCY
ONEOFF_PURCHASES_FREQUENCY
PURCHASES_INSTALLMENTS_FREQUENCY
CASH_ADVANCE_FREQUENCY
CASH_ADVANCE_TRX
PURCHASES_TRX
CREDIT_LIMIT
PAYMENTS
MINIMUM_PAYMENTS
PRC_FULL_PAYMENT
TENURE

[17]:
```
creditcard_df.isnull().sum()
```

[17]:
```
CUST_ID                               0
BALANCE                               0
BALANCE_FREQUENCY                     0
PURCHASES                             0
ONEOFF_PURCHASES                      0
INSTALLMENTS_PURCHASES                0
CASH_ADVANCE                          0
PURCHASES_FREQUENCY                   0
ONEOFF_PURCHASES_FREQUENCY            0
PURCHASES_INSTALLMENTS_FREQUENCY      0
CASH_ADVANCE_FREQUENCY                0
CASH_ADVANCE_TRX                      0
PURCHASES_TRX                         0
CREDIT_LIMIT                          1
PAYMENTS                              0
MINIMUM_PAYMENTS                    313
PRC_FULL_PAYMENT                      0
TENURE                                0
dtype: int64
```

1.1 Fill up missing elements

[19]:
```
creditcard_df.loc[(creditcard_df['MINIMUM_PAYMENTS'].
↪isnull()==True),'MINIMUM_PAYMENTS']=creditcard_df['MINIMUM_PAYMENTS'].mean()
```

[20]:
```
creditcard_df.isnull().sum()
```

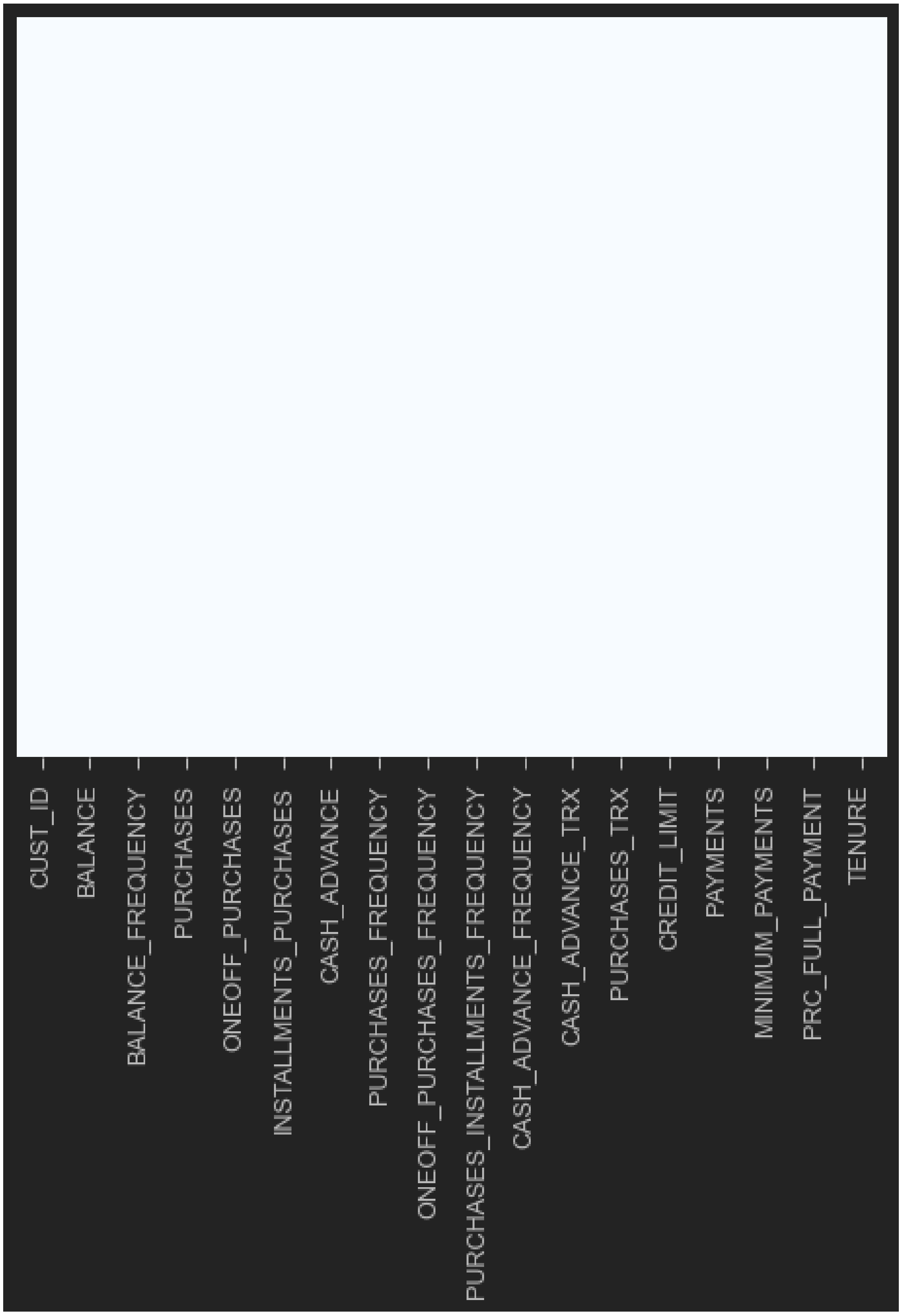
[20]:
```
CUST_ID                             0
BALANCE                             0
BALANCE_FREQUENCY                   0
PURCHASES                           0
ONEOFF_PURCHASES                    0
INSTALLMENTS_PURCHASES              0
CASH_ADVANCE                        0
PURCHASES_FREQUENCY                 0
ONEOFF_PURCHASES_FREQUENCY          0
PURCHASES_INSTALLMENTS_FREQUENCY    0
CASH_ADVANCE_FREQUENCY              0
CASH_ADVANCE_TRX                    0
PURCHASES_TRX                       0
CREDIT_LIMIT                        1
PAYMENTS                            0
MINIMUM_PAYMENTS                    0
PRC_FULL_PAYMENT                    0
TENURE                              0
```

```
dtype: int64
```

[21]:
```
creditcard_df.loc[(creditcard_df['CREDIT_LIMIT'].
↪isnull()==True),'CREDIT_LIMIT']=creditcard_df['CREDIT_LIMIT'].mean()
```

[23]:
```
sns.heatmap(creditcard_df.isnull(),yticklabels=False,cbar=False,cmap='Blues')
```

[23]:
```
<AxesSubplot:>
```


CUST_ID
BALANCE
BALANCE_FREQUENCY
PURCHASES
ONEOFF_PURCHASES
INSTALLMENTS_PURCHASES
CASH_ADVANCE
PURCHASES_FREQUENCY
ONEOFF_PURCHASES_FREQUENCY
PURCHASES_INSTALLMENTS_FREQUENCY
CASH_ADVANCE_FREQUENCY
CASH_ADVANCE_TRX
PURCHASES_TRX
CREDIT_LIMIT
PAYMENTS
MINIMUM_PAYMENTS
PRC_FULL_PAYMENT
TENURE

[24]:
```
#Check duplicated values
creditcard_df.duplicated().sum()
```

[24]:
```
0
```

[28]:
```
#Drop customer id column
creditcard_df.drop(['CUST_ID'], axis=1 ,inplace=True)
```

[29]:
```
creditcard_df
```

[29]:
```
          BALANCE  BALANCE_FREQUENCY  PURCHASES  ONEOFF_PURCHASES  \
0       40.900749           0.818182      95.40              0.00
1     3202.467416           0.909091       0.00              0.00
2     2495.148862           1.000000     773.17            773.17
3     1666.670542           0.636364    1499.00           1499.00
4      817.714335           1.000000      16.00             16.00
...           ...                ...        ...               ...
8945    28.493517           1.000000     291.12              0.00
8946    19.183215           1.000000     300.00              0.00
8947    23.398673           0.833333     144.40              0.00
8948    13.457564           0.833333       0.00              0.00
8949   372.708075           0.666667    1093.25           1093.25

      INSTALLMENTS_PURCHASES  CASH_ADVANCE  PURCHASES_FREQUENCY  \
0                      95.40      0.000000             0.166667
1                       0.00   6442.945483             0.000000
2                       0.00      0.000000             1.000000
3                       0.00    205.788017             0.083333
4                       0.00      0.000000             0.083333
...                      ...           ...                  ...
8945                  291.12      0.000000             1.000000
8946                  300.00      0.000000             1.000000
8947                  144.40      0.000000             0.833333
8948                    0.00     36.558778             0.000000
8949                    0.00    127.040008             0.666667

      ONEOFF_PURCHASES_FREQUENCY  PURCHASES_INSTALLMENTS_FREQUENCY  \
0                       0.000000                          0.083333
1                       0.000000                          0.000000
2                       1.000000                          0.000000
3                       0.083333                          0.000000
4                       0.083333                          0.000000
...                          ...                               ...
8945                    0.000000                          0.833333
8946                    0.000000                          0.833333
8947                    0.000000                          0.666667
8948                    0.000000                          0.000000
8949                    0.666667                          0.000000
```

```
      CASH_ADVANCE_FREQUENCY  CASH_ADVANCE_TRX  PURCHASES_TRX  CREDIT_LIMIT  \
0                   0.000000                 0              2        1000.0
1                   0.250000                 4              0        7000.0
2                   0.000000                 0             12        7500.0
3                   0.083333                 1              1        7500.0
4                   0.000000                 0              1        1200.0
...                      ...               ...            ...           ...
8945                0.000000                 0              6        1000.0
8946                0.000000                 0              6        1000.0
8947                0.000000                 0              5        1000.0
8948                0.166667                 2              0         500.0
8949                0.333333                 2             23        1200.0

         PAYMENTS  MINIMUM_PAYMENTS  PRC_FULL_PAYMENT  TENURE
0      201.802084        139.509787          0.000000      12
1     4103.032597       1072.340217          0.222222      12
2      622.066742        627.284787          0.000000      12
3        0.000000        864.206542          0.000000      12
4      678.334763        244.791237          0.000000      12
...           ...               ...               ...     ...
8945   325.594462         48.886365          0.500000       6
8946   275.861322        864.206542          0.000000       6
8947    81.270775         82.418369          0.250000       6
8948    52.549959         55.755628          0.250000       6
8949    63.165404         88.288956          0.000000       6

[8950 rows x 17 columns]
```

[33]:
```python
n=len(creditcard_df.columns)
print (n)
creditcard_df.columns
```

```
17
```

[33]:
```
Index(['BALANCE', 'BALANCE_FREQUENCY', 'PURCHASES', 'ONEOFF_PURCHASES',
       'INSTALLMENTS_PURCHASES', 'CASH_ADVANCE', 'PURCHASES_FREQUENCY',
       'ONEOFF_PURCHASES_FREQUENCY', 'PURCHASES_INSTALLMENTS_FREQUENCY',
       'CASH_ADVANCE_FREQUENCY', 'CASH_ADVANCE_TRX', 'PURCHASES_TRX',
       'CREDIT_LIMIT', 'PAYMENTS', 'MINIMUM_PAYMENTS', 'PRC_FULL_PAYMENT',
       'TENURE'],
      dtype='object')
```

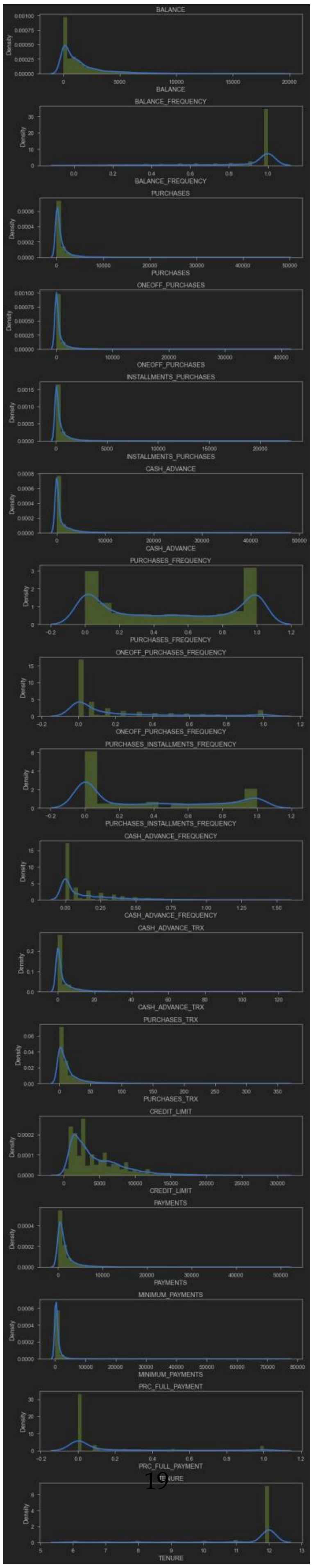
[39]:
```python
plt.figure(figsize = (10,50))
for i in range (len(creditcard_df.columns)):
    plt.subplot(17,1,i+1)
    sns.distplot(creditcard_df[creditcard_df.columns[i]],kde_kws={"color":"b"␣
 ↪,"lw":3 ,"label":"KDE"},hist_kws={'color':'g'})
```

```
    plt.title(creditcard_df.columns[i])

plt.tight_layout()
```

```
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
```

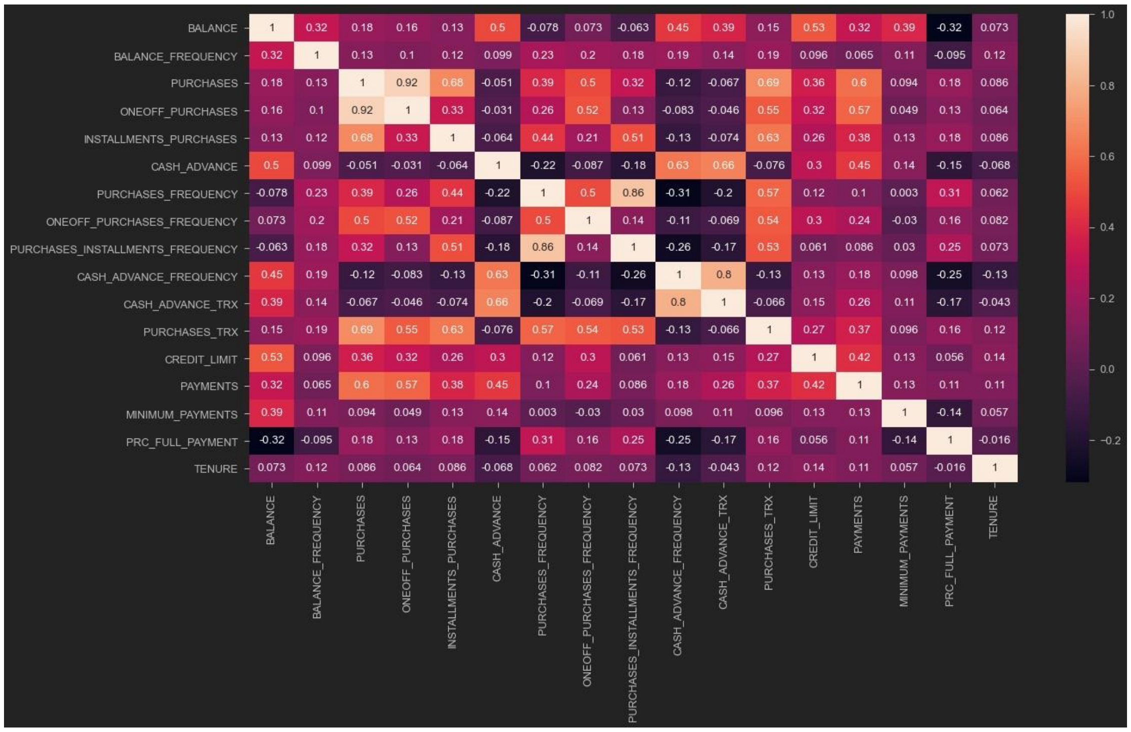
```
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
```

```
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\ASUS\AppData\Roaming\Python\Python38\site-
packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```


BALANCE
Density
BALANCE_FREQUENCY
PURCHASES
ONEOFF_PURCHASES
INSTALLMENTS_PURCHASES
CASH_ADVANCE
PURCHASES_FREQUENCY
ONEOFF_PURCHASES_FREQUENCY
PURCHASES_INSTALLMENTS_FREQUENCY
CASH_ADVANCE_FREQUENCY
CASH_ADVANCE_TRX
PURCHASES_TRX
CREDIT_LIMIT
PAYMENTS
MINIMUM_PAYMENTS
PRC_FULL_PAYMENT
TENURE

```
[44]: #correlation matrix
      correlation =creditcard_df.corr()
      #for size
      f,ax=plt.subplots(figsize=(20,10))
      sns.heatmap(correlation , annot=True)
```

```
[44]: <AxesSubplot:>
```



2 Find optimal no. of clusters using Elbow method

```
[45]: scaler = StandardScaler()
      creditcard_df_scaled =scaler.fit_transform(creditcard_df)
```

```
[47]: creditcard_df_scaled.shape
```

```
[47]: (8950, 17)
```

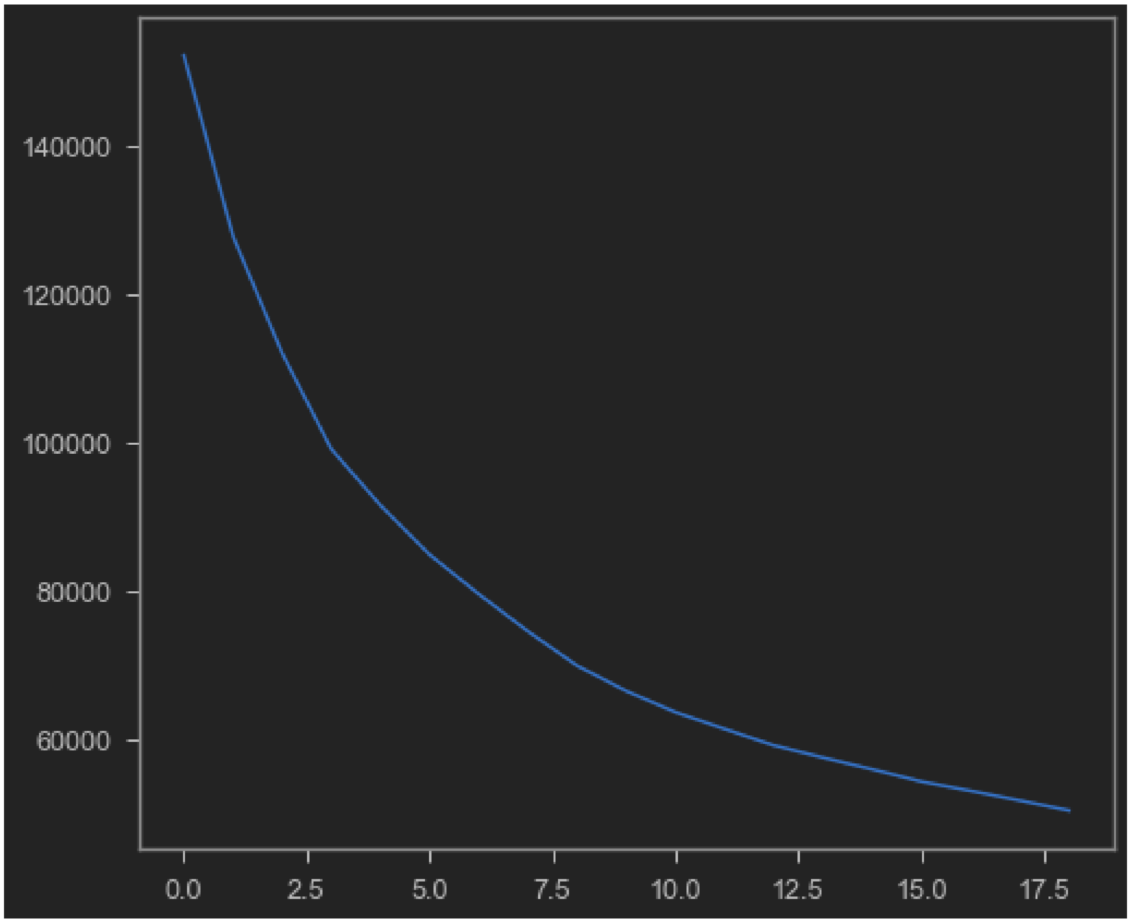
```
[48]: creditcard_df_scaled
```

```
[48]: array([[-0.73198937, -0.24943448, -0.42489974, ...,  -0.31096755,
              -0.52555097,  0.36067954],
            [ 0.78696085,  0.13432467, -0.46955188, ...,  0.08931021,
               0.2342269 ,  0.36067954],
[ 0.44713513,  0.51808382, -0.10766823, ..., -0.10166318,
 -0.52555097,  0.36067954],
...,
[-0.7403981 , -0.18547673, -0.40196519, ..., -0.33546549,
  0.32919999, -4.12276757],
[-0.74517423, -0.18547673, -0.46955188, ..., -0.34690648,
  0.32919999, -4.12276757],
[-0.57257511, -0.88903307,  0.04214581, ..., -0.33294642,
 -0.52555097, -4.12276757]])
```

2.1 WCSS(Within Cluster Sum of squares)

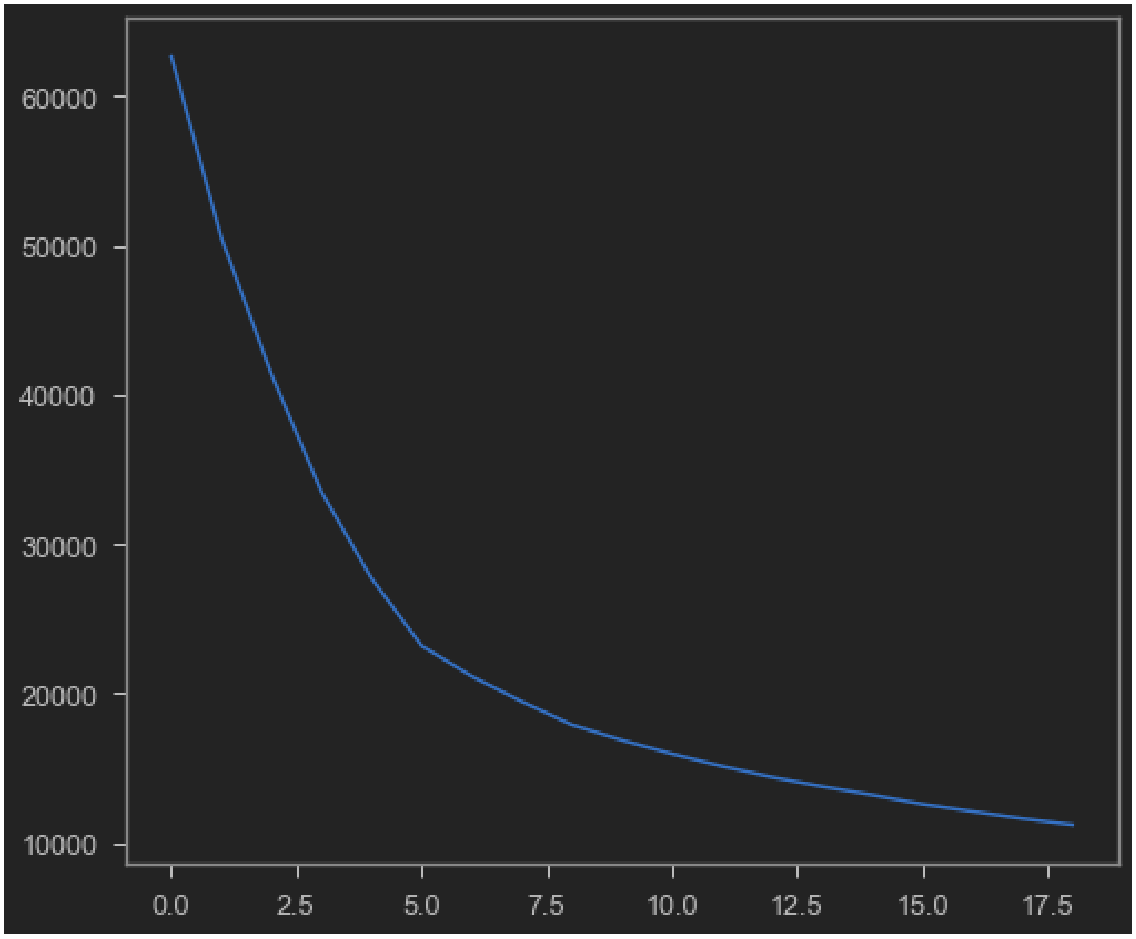
[57]:
```python
#For all columns
scores_1=[]
range_values= range(1,20)

for i in range_values:
    kmeans=KMeans(n_clusters=i)
    kmeans.fit(creditcard_df_scaled)
    scores_1.append(kmeans.inertia_)
#WCSS VS NO. OF  CLUSTERS
plt.plot(scores_1,'bx-')
#Optimal clusters= 7 or 8
```

[57]:
```
[<matplotlib.lines.Line2D at 0x274d2106220>]
```



[58]:
```
#For first seven columns
scores_2=[]
range_values= range(1,20)

for i  in range_values:
    kmeans=KMeans(n_clusters=i)
    kmeans.fit(creditcard_df_scaled[:,:7]) #All rows but 7 columns
    scores_2.append(kmeans.inertia_)
#WCSS VS NO. OF  CLUSTERS
plt.plot(scores_2,'bx-')
```

[58]:
```
[<matplotlib.lines.Line2D at 0x274ce7f0430>]
```



3 APPLY K-MEANS

[59]:
```
kmeans=KMeans(7)
kmeans.fit(creditcard_df_scaled)
labels =kmeans.labels_
```

[61]:
```
kmeans.cluster_centers_.shape #Centroid of different  clusters
```

[61]:
```
(7, 17)
```

[66]:
```
# Centroid for all columns
cluster_centers= pd.DataFrame(data=kmeans.cluster_centers_ ,␣
 ↪columns=[creditcard_df.columns])
cluster_centers
```

[66]:
```
    BALANCE BALANCE_FREQUENCY  PURCHASES ONEOFF_PURCHASES  \
0 -0.701543         -2.133587  -0.305982        -0.230503
1  0.524277          0.454804   1.798438         1.545694
2  1.671486          0.394844  -0.211058        -0.150129
```

```
3 -0.341141          -0.333100  -0.280989          -0.207459
4  0.018074           0.403907  -0.348728          -0.229219
5  1.923051           0.337717  11.212042          10.600367
6 -0.336978           0.355340   0.032553          -0.088290

  INSTALLMENTS_PURCHASES  CASH_ADVANCE  PURCHASES_FREQUENCY  \
0              -0.299905     -0.324030            -0.537637
1               1.412602     -0.240267             1.153143
2              -0.223304      1.991806            -0.451085
3              -0.282905      0.060232            -0.171997
4              -0.403334     -0.095952            -0.838344
5               7.033118      0.419625             1.046983
6               0.238997     -0.368884             0.977456

  ONEOFF_PURCHASES_FREQUENCY  PURCHASES_INSTALLMENTS_FREQUENCY  \
0                  -0.436587                         -0.427520
1                   1.794249                          0.966087
2                  -0.175268                         -0.409414
3                  -0.271368                         -0.207953
4                  -0.367101                         -0.763042
5                   1.915501                          0.981334
6                   0.199920                          0.895697

  CASH_ADVANCE_FREQUENCY  CASH_ADVANCE_TRX  PURCHASES_TRX  CREDIT_LIMIT  \
0              -0.523134         -0.377605      -0.414550     -0.175050
1              -0.353333         -0.263881       2.104041      1.113781
2               1.905279          1.915647      -0.253699      1.013354
3               0.291927         -0.007982      -0.379541     -0.560480
4               0.099250         -0.032095      -0.474034     -0.298306
5              -0.258912          0.061229       5.362438      3.044064
6              -0.470867         -0.359569       0.231858     -0.143487

   PAYMENTS  MINIMUM_PAYMENTS  PRC_FULL_PAYMENT    TENURE
0 -0.200441         -0.257489          0.288170  0.200378
1  0.971636          0.197651          0.472558  0.310304
2  0.813947          0.566762         -0.392488  0.073214
3 -0.389957         -0.207857          0.021533 -3.178761
4 -0.245820         -0.006715         -0.455446  0.272674
5  8.098975          1.120318          1.110132  0.310863
6 -0.179705         -0.080949          0.329358  0.271775
```

[68]:
```
cluster_centers=scaler.inverse_transform(cluster_centers)
cluster_centers= pd.DataFrame(data=cluster_centers , columns=[creditcard_df.
 ↪columns])
cluster_centers
#First cluster-Transactor
#Second-Revolvers
```

```
#Third-VIP
#Fourth-Low tenure
```

[68]:

```
       BALANCE  BALANCE_FREQUENCY     PURCHASES  ONEOFF_PURCHASES  \
0   104.272448           0.371844    349.469265        209.849357
1  2655.712374           0.985010   4845.596110       3157.972963
2  5043.530870           0.970806    552.275771        343.253161
3   854.418008           0.798362    402.868166        248.098605
4  1602.095244           0.972953    258.141752        211.981551
5  5567.142164           0.957273  24957.905000      18186.875667
6   863.082636           0.961447   1072.753769        445.894859

   INSTALLMENTS_PURCHASES CASH_ADVANCE PURCHASES_FREQUENCY\
0              139.867654   299.364326            0.274571
1             1688.465843   475.020604            0.953163
2              209.136974  5155.780925            0.309308
3              155.239781  1105.181375            0.421320
4               46.337896   777.655560            0.153883
5             6771.029333  1858.844605            0.910556
6              627.190000   205.304778            0.882651

   ONEOFF_PURCHASES_FREQUENCY  PURCHASES_INSTALLMENTS_FREQUENCY  \
0                    0.072215                          0.194530
1                    0.737717                          0.748385
2                    0.150172                          0.201726
3                    0.121503                          0.281791
4                    0.092944                          0.061185
5                    0.773889                          0.754444
6                    0.262098                          0.720410

   CASH_ADVANCE_FREQUENCY CASH_ADVANCE_TRX PURCHASES_TRX  CREDIT_LIMIT  \
0                0.030460         0.671953      4.405676   3857.545446
1                0.064439         1.448034     67.008427   8546.839888
2                0.516410        16.321710      8.403825   8181.444933
3                0.193562         3.194357      5.275862   2455.195062
4                0.155005         3.029803      2.927110   3409.090365
5                0.083333         3.666667    148.000000  15570.000000
6                0.040919         0.795033     20.472943   3972.386017

       PAYMENTS  MINIMUM_PAYMENTS  PRC_FULL_PAYMENT     TENURE
0   1152.887104        264.138375          0.238000  11.785476
1   4545.936049       1324.824872          0.291930  11.932584
2   4089.440839       2185.020685          0.038918  11.615298
3    604.255929        379.804248          0.160013   7.263323
4   1021.519339        848.556397          0.020504  11.882226
5  25178.882690       3475.059479          0.478409  11.933333
6   1212.915578        675.558052          0.250046  11.881023
```

[69]:
```
labels.shape
```

[69]:
```
(8950,)
```

[72]:
```
labels.max()
```

[72]:
```
6
```

[73]:
```
labels.min()
```

[73]:
```
0
```

[78]:
```
y_kmeans = kmeans.fit_predict(creditcard_df_scaled)
y_kmeans
```

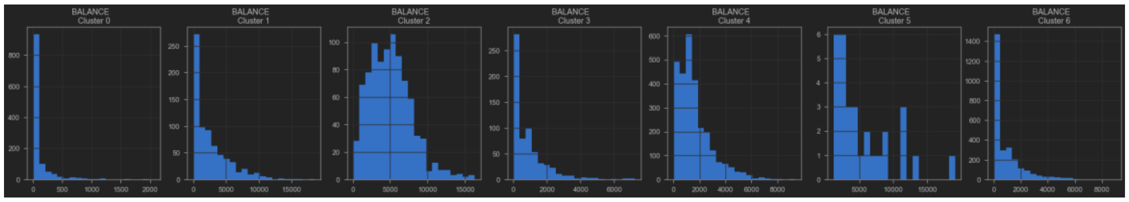
[78]:
```
array([2, 5, 1, ..., 1, 2, 3])
```

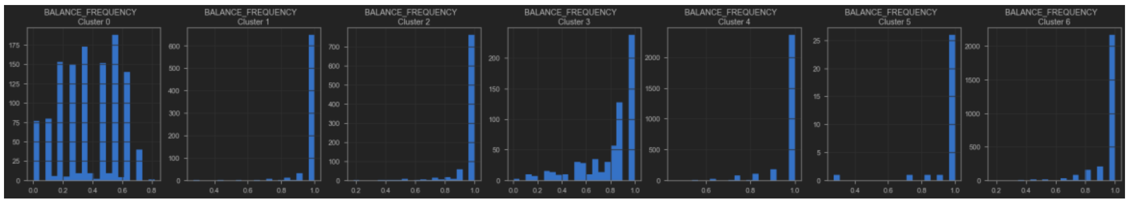
[80]:
```
#Add a new column name cluster
creditcard_df_cluster= pd.concat([creditcard_df,pd.DataFrame({"Cluster":
 ↪labels})],axis=1)
creditcard_df_cluster.head()
```

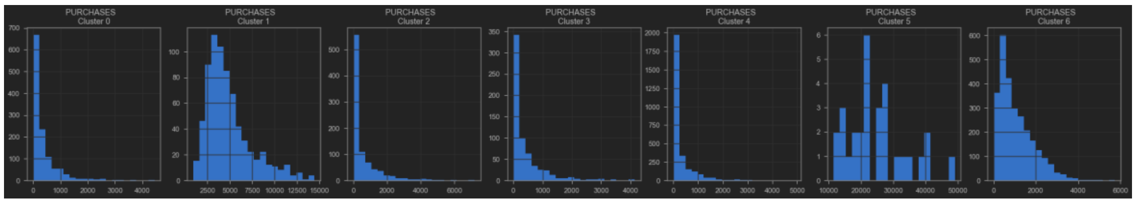
[80]:
```
       BALANCE  BALANCE_FREQUENCY  PURCHASES  ONEOFF_PURCHASES  \
0    40.900749           0.818182      95.40              0.00
1  3202.467416           0.909091       0.00              0.00
2  2495.148862           1.000000     773.17            773.17
3  1666.670542           0.636364    1499.00           1499.00
4   817.714335           1.000000      16.00             16.00

   INSTALLMENTS_PURCHASES  CASH_ADVANCE  PURCHASES_FREQUENCY  \
0                    95.4      0.000000             0.166667
1                     0.0   6442.945483             0.000000
2                     0.0      0.000000             1.000000
3                     0.0    205.788017             0.083333
4                     0.0      0.000000             0.083333

   ONEOFF_PURCHASES_FREQUENCY  PURCHASES_INSTALLMENTS_FREQUENCY  \
0                    0.000000  0.083333
1                    0.000000                          0.000000
2                    1.000000                          0.000000
3                    0.083333                          0.000000
4                    0.083333                          0.000000

   CASH_ADVANCE_FREQUENCY  CASH_ADVANCE_TRX  PURCHASES_TRX  CREDIT_LIMIT  \
0                0.000000                 0              2        1000.0
1                0.250000                 4              0        7000.0
2                0.000000                 0             12        7500.0
3                0.083333                 1              1        7500.0
4                0.000000                 0              1        1200.0

      PAYMENTS  MINIMUM_PAYMENTS  PRC_FULL_PAYMENT  TENURE  Cluster
```

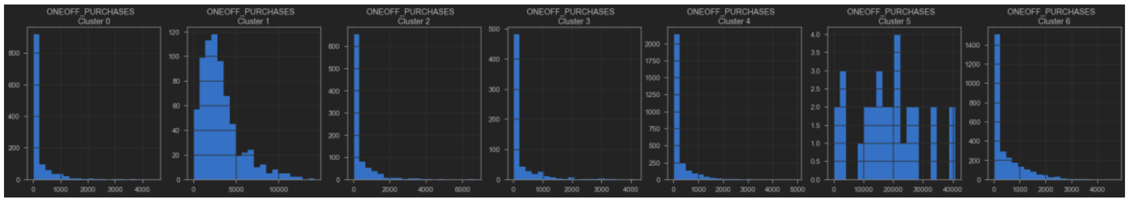
```
0    201.802084         139.509787          0.000000        12           4
1   4103.032597        1072.340217          0.222222        12           2
2    622.066742         627.284787          0.000000        12           6
3      0.000000         864.206542          0.000000        12           4
4    678.334763         244.791237          0.000000        12           4
```

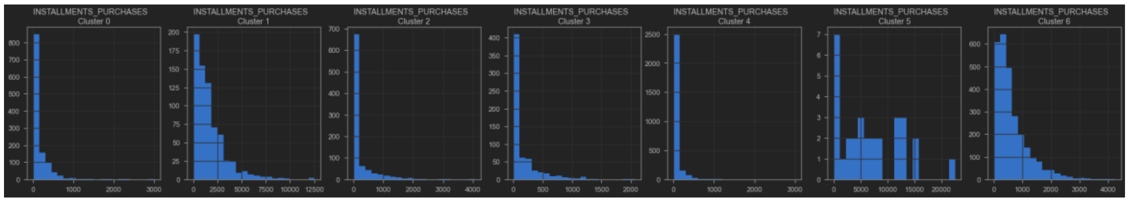
[82]:
```python
for i in creditcard_df.columns:
    plt.figure(figsize=(35,5))
    for j in range(7):
        plt.subplot(1,7,j+1)
        cluster=creditcard_df_cluster[creditcard_df_cluster['Cluster'] == j]
        cluster[i].hist(bins=20)
        plt.title('{}    \nCluster {}'.format(i,j))

plt.show()
```

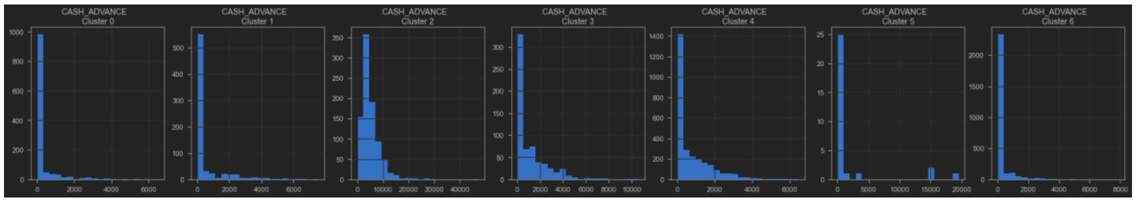


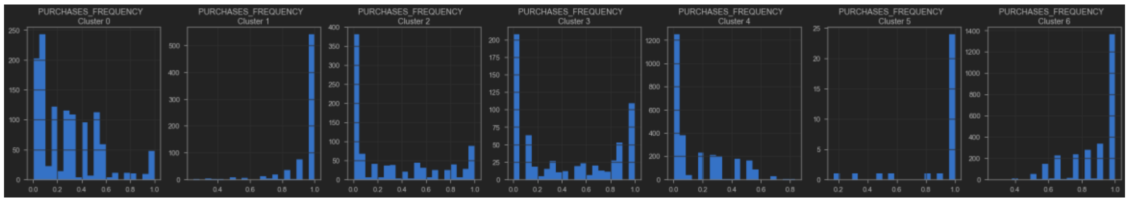


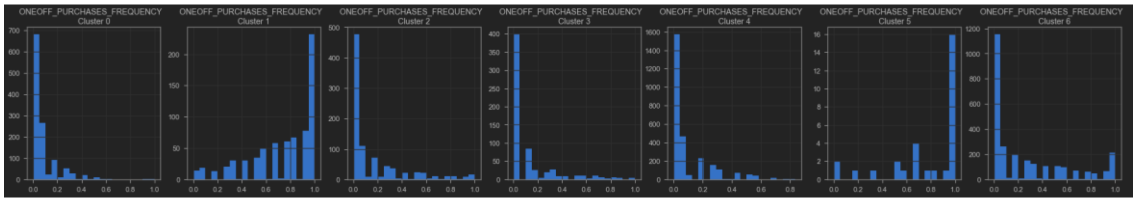


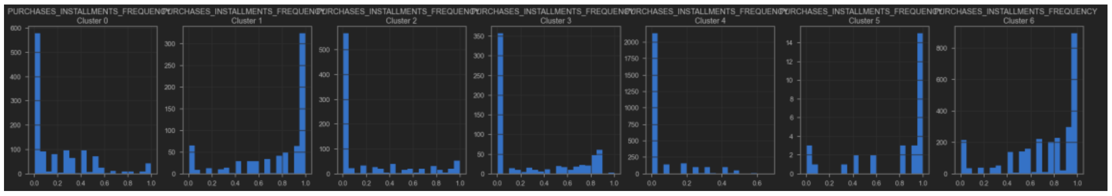


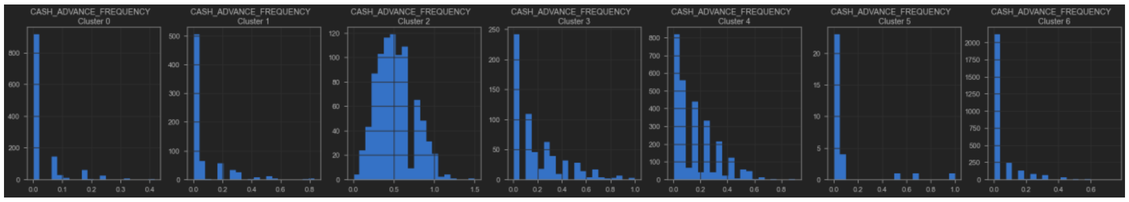

INSTALLMENTS_PURCHASES
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6

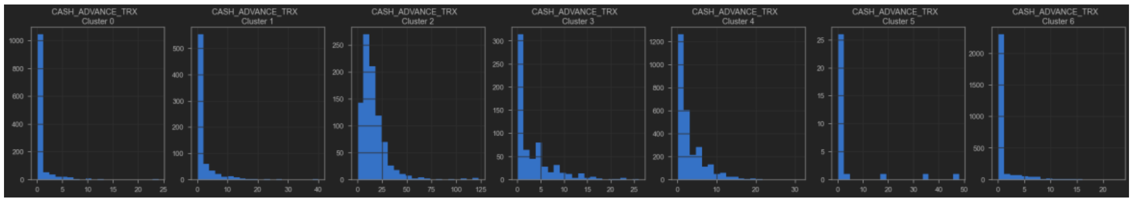

CASH_ADVANCE
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6

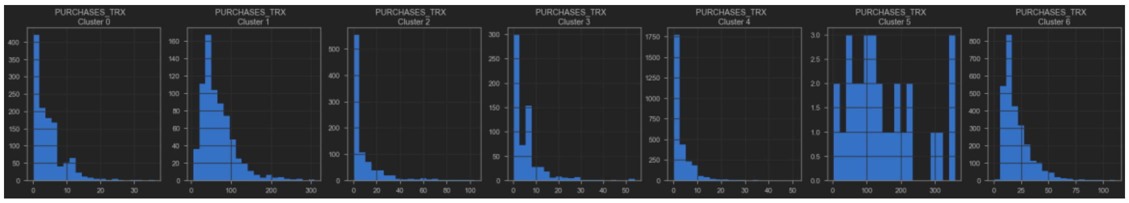

PURCHASES_FREQUENCY
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6

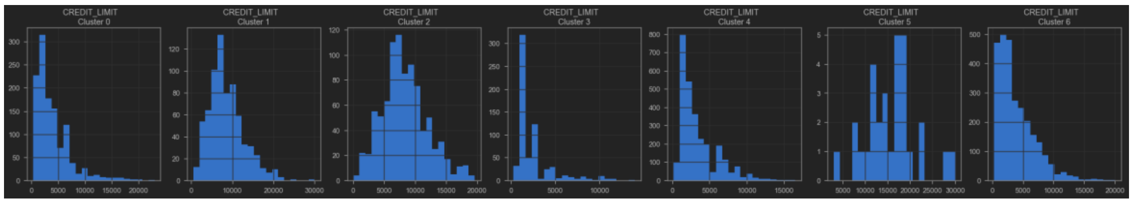

ONEOFF_PURCHASES_FREQUENCY
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6

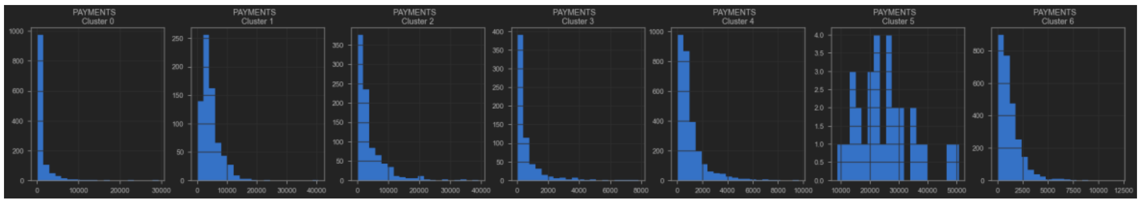

PURCHASES_INSTALLMENTS_FREQUENCY
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6

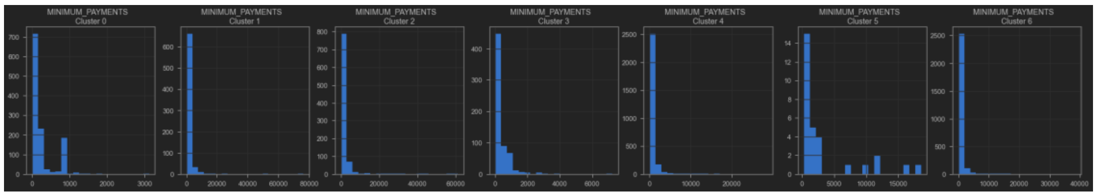

CASH_ADVANCE_FREQUENCY Cluster 0
CASH_ADVANCE_FREQUENCY Cluster 1
CASH_ADVANCE_FREQUENCY Cluster 2
CASH_ADVANCE_FREQUENCY Cluster 3
CASH_ADVANCE_FREQUENCY Cluster 4
CASH_ADVANCE_FREQUENCY Cluster 5
CASH_ADVANCE_FREQUENCY Cluster 6

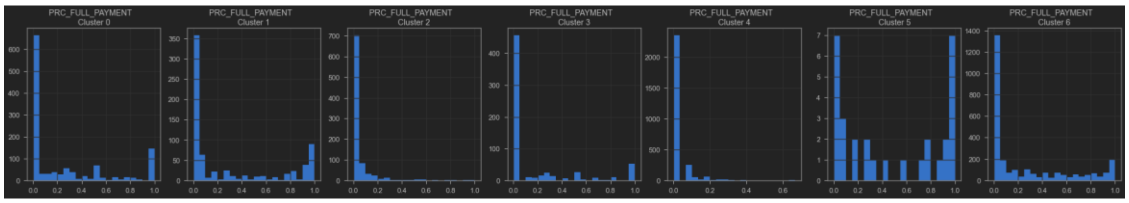

CASH_ADVANCE_TRX Cluster 0
CASH_ADVANCE_TRX Cluster 1
CASH_ADVANCE_TRX Cluster 2
CASH_ADVANCE_TRX Cluster 3
CASH_ADVANCE_TRX Cluster 4
CASH_ADVANCE_TRX Cluster 5
CASH_ADVANCE_TRX Cluster 6

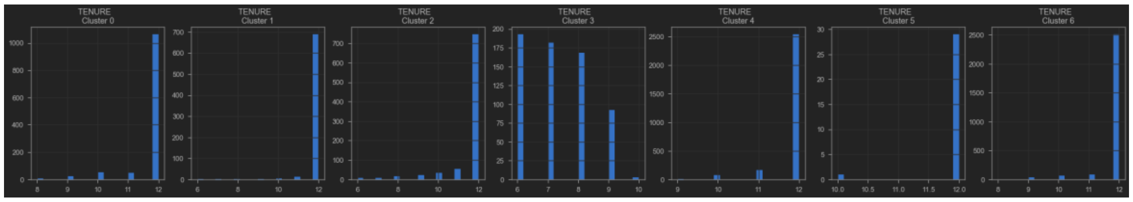

PURCHASES_TRX Cluster 0
PURCHASES_TRX Cluster 1
PURCHASES_TRX Cluster 2
PURCHASES_TRX Cluster 3
PURCHASES_TRX Cluster 4
PURCHASES_TRX Cluster 5
PURCHASES_TRX Cluster 6


CREDIT_LIMIT Cluster 0
CREDIT_LIMIT Cluster 1
CREDIT_LIMIT Cluster 2
CREDIT_LIMIT Cluster 3
CREDIT_LIMIT Cluster 4
CREDIT_LIMIT Cluster 5
CREDIT_LIMIT Cluster 6


PAYMENTS Cluster 0
PAYMENTS Cluster 1
PAYMENTS Cluster 2
PAYMENTS Cluster 3
PAYMENTS Cluster 4
PAYMENTS Cluster 5
PAYMENTS Cluster 6


MINIMUM_PAYMENTS Cluster 0
MINIMUM_PAYMENTS Cluster 1
MINIMUM_PAYMENTS Cluster 2
MINIMUM_PAYMENTS Cluster 3
MINIMUM_PAYMENTS Cluster 4
MINIMUM_PAYMENTS Cluster 5
MINIMUM_PAYMENTS Cluster 6





4 Apply PCA(Principal Component Analysis)

[83]:
```
pca = PCA(n_components=2)
principal_comp =pca.fit_transform(creditcard_df_scaled)
principal_comp
```

[83]:
```
array([[-1.68221972, -1.07645219],
       [-1.13828883,  2.50647801],
       [ 0.96968065, -0.38352858],
       ...,
       [-0.92620565, -1.81078787],
       [-2.33655278, -0.65796354],
       [-0.55642307, -0.40046237]])
```

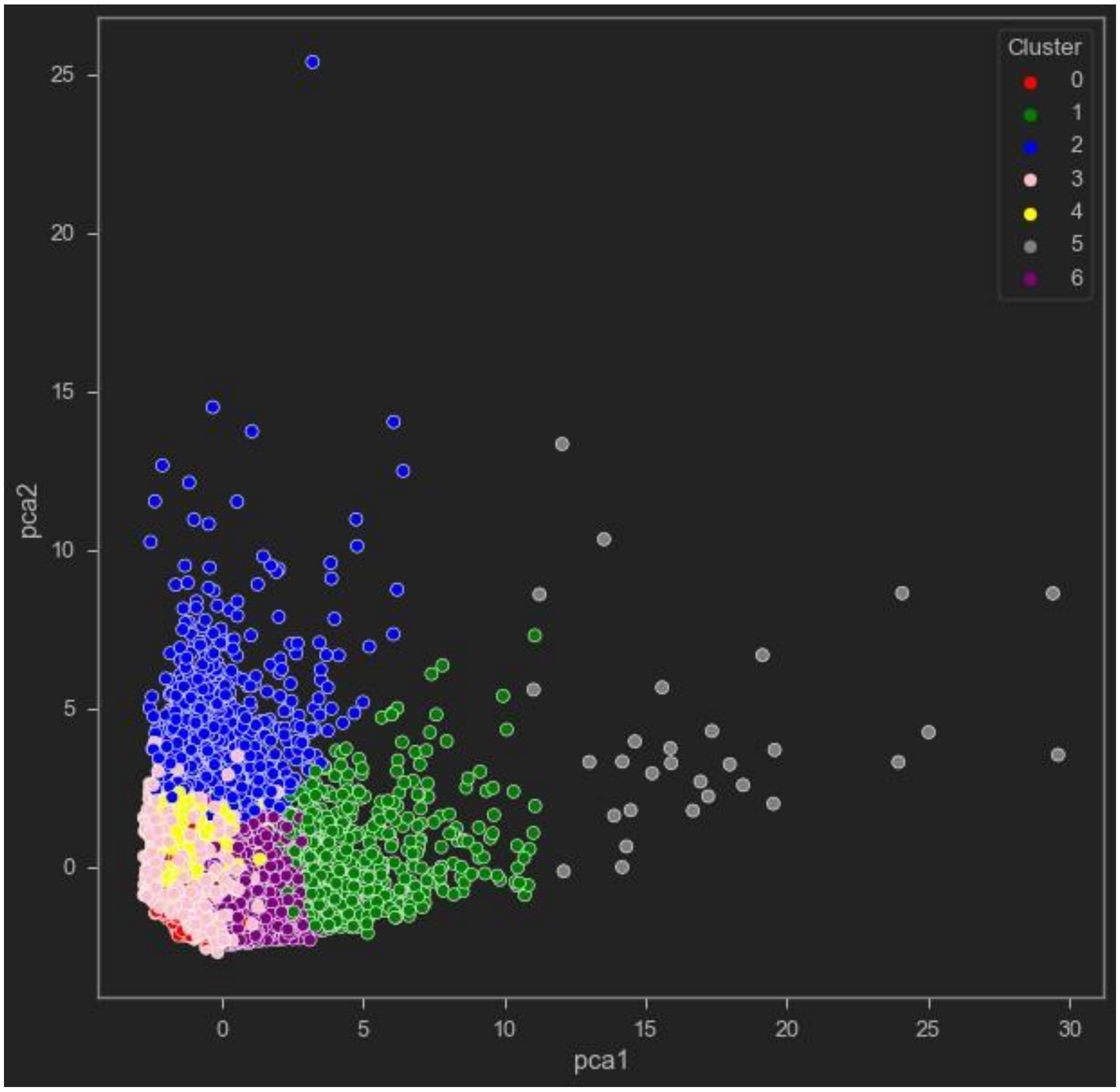
[84]:
```
pca_df=pd.DataFrame(data=principal_comp ,columns=['pca1','pca2'])
pca_df.head()
```

[84]:
```
       pca1      pca2
0 -1.682220 -1.076452
1 -1.138289  2.506478
2  0.969681 -0.383529
3 -0.873625  0.043165
4 -1.599435 -0.688583
```

[85]:
```
pca_df=pd.concat([pca_df,pd.DataFrame({'Cluster':labels})],axis=1)
pca_df.head()
```

[85]:
```
       pca1      pca2  Cluster
0 -1.682220 -1.076452        4
1 -1.138289  2.506478        2
2  0.969681 -0.383529        6
3 -0.873625  0.043165        4
4 -1.599435 -0.688583        4
```

[86]:
```
plt.figure(figsize=(10,10))
ax = sns.
 ↪scatterplot(x='pca1',y='pca2',hue='Cluster',data=pca_df,palette=['red','green','blue','pink
plt.show()
```



[ ]: