Bank

April 21, 2021

1 Importing Required Libraries

[1]:
```
# Basic Libraries for Data organization, Statistical operations and Plotting
import numpy as np
import pandas as pd
import seaborn as sns
%matplotlib inline
from matplotlib import pyplot as plt
# For loading .arff files
from scipy.io import arff
# Library for performing k-NN and MICE imputations
import fancyimpute
# Library to perform Expectation-Maximization (EM) imputation
import impyute as impy
# To perform mean imputation
from sklearn.impute import SimpleImputer
#To perform kFold Cross Validation
from sklearn.model_selection import KFold
from sklearn.model_selection import train_test_split
# Formatted counter of class labels
from collections import Counter
# Ordered Dictionary
from collections import OrderedDict
# Library imbalanced-learn to deal with the data imbalance. To use SMOTE␣
↪oversampling
from imblearn.over_sampling import SMOTE
from sklearn.preprocessing import StandardScaler
# Impoting classification models
from sklearn.linear_model import LogisticRegression
from statsmodels.stats.outliers_influence import variance_inflation_factor
from statsmodels.tools.tools import add_constant


from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
from sklearn.metrics import classification_report
```

```
from sklearn.metrics import confusion_matrix
from sklearn.metrics import roc_curve
from sklearn.metrics import precision_recall_curve
```

2 Importing Data

[2]:
```
def import_data(i):
  df1.train=arff.loadarff('C:/Users/rick7/Desktop/Bankruptcy Prediction/
 ↪'+str(i)+'year.arff')
  df1.train=pd.DataFrame(df1.train[0])
  return df1.train

df1=pd.DataFrame()
df1=import_data(3)
```

```
<ipython-input-2-997bd113e94a>:2: UserWarning: Pandas doesn't allow columns to
be created via a new attribute name - see https://pandas.pydata.org/pandas-
docs/stable/indexing.html#attribute-access
  df1.train=arff.loadarff('C:/Users/rick7/Desktop/Bankruptcy
Prediction/'+str(i)+'year.arff')
```

[3]:
```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10503 entries, 0 to 10502
Data columns (total 65 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Attr1   10503 non-null  float64
 1   Attr2   10503 non-null  float64
 2   Attr3   10503 non-null  float64
 3   Attr4   10485 non-null  float64
 4   Attr5   10478 non-null  float64
 5   Attr6   10503 non-null  float64
 6   Attr7   10503 non-null  float64
 7   Attr8   10489 non-null  float64
 8   Attr9   10500 non-null  float64
 9   Attr10  10503 non-null  float64
 10  Attr11  10503 non-null  float64
 11  Attr12  10485 non-null  float64
 12  Attr13  10460 non-null  float64
 13  Attr14  10503 non-null  float64
 14  Attr15  10495 non-null  float64
 15  Attr16  10489 non-null  float64
 16  Attr17  10489 non-null  float64
 17  Attr18  10503 non-null  float64
```

```
 18  Attr19  10460 non-null  float64
 19  Attr20  10460 non-null  float64
 20  Attr21  9696 non-null   float64
 21  Attr22  10503 non-null  float64
 22  Attr23  10460 non-null  float64
 23  Attr24  10276 non-null  float64
 24  Attr25  10503 non-null  float64
 25  Attr26  10489 non-null  float64
 26  Attr27  9788 non-null   float64
 27  Attr28  10275 non-null  float64
 28  Attr29  10503 non-null  float64
 29  Attr30  10460 non-null  float64
 30  Attr31  10460 non-null  float64
 31  Attr32  10402 non-null  float64
 32  Attr33  10485 non-null  float64
 33  Attr34  10489 non-null  float64
 34  Attr35  10503 non-null  float64
 35  Attr36  10503 non-null  float64
 36  Attr37  5767 non-null   float64
 37  Attr38  10503 non-null  float64
 38  Attr39  10460 non-null  float64
 39  Attr40  10485 non-null  float64
 40  Attr41  10301 non-null  float64
 41  Attr42  10460 non-null  float64
 42  Attr43  10460 non-null  float64
 43  Attr44  10460 non-null  float64
 44  Attr45  9912 non-null   float64
 45  Attr46  10485 non-null  float64
 46  Attr47  10417 non-null  float64
 47  Attr48  10503 non-null  float64
 48  Attr49  10460 non-null  float64
 49  Attr50  10489 non-null  float64
 50  Attr51  10503 non-null  float64
 51  Attr52  10417 non-null  float64
 52  Attr53  10275 non-null  float64
 53  Attr54  10275 non-null  float64
 54  Attr55  10503 non-null  float64
 55  Attr56  10460 non-null  float64
 56  Attr57  10503 non-null  float64
 57  Attr58  10474 non-null  float64
 58  Attr59  10503 non-null  float64
 59  Attr60  9911 non-null   float64
 60  Attr61  10486 non-null  float64
 61  Attr62  10460 non-null  float64
 62  Attr63  10485 non-null  float64
 63  Attr64  10275 non-null  float64
 64  class   10503 non-null  object
dtypes: float64(64), object(1)
```

```
memory usage: 5.2+ MB
```

[4]:
```
df1.head()
```

[4]:
```
     Attr1    Attr2    Attr3   Attr4    Attr5    Attr6     Attr7   Attr8  \
0  0.174190  0.41299  0.14371  1.3480 -28.9820  0.60383  0.219460  1.1225
1  0.146240  0.46038  0.28230  1.6294   2.5952  0.00000  0.171850  1.1721
2  0.000595  0.22612  0.48839  3.1599  84.8740  0.19114  0.004572  2.9881
3  0.024526  0.43236  0.27546  1.7833 -10.1050  0.56944  0.024526  1.3057
4  0.188290  0.41504  0.34231  1.9279 -58.2740  0.00000  0.233580  1.4094

    Attr9   Attr10  …    Attr56    Attr57   Attr58    Attr59  Attr60  \
0  1.1961  0.46359  …  0.163960  0.375740  0.83604  0.000007  9.7145
1  1.6018  0.53962  …  0.027516  0.271000  0.90108  0.000000  5.9882
2  1.0077  0.67566  …  0.007639  0.000881  0.99236  0.000000  6.7742
3  1.0509  0.56453  …  0.048398  0.043445  0.95160  0.142980  4.2286
4  1.3393  0.58496  …  0.176480  0.321880  0.82635  0.073039  2.5912

   Attr61   Attr62  Attr63  Attr64  class
0  6.2813   84.291  4.3303  4.0341   b'0'
1  4.1103  102.190  3.5716  5.9500   b'0'
2  3.7922   64.846  5.6287  4.4581   b'0'
3  5.0528   98.783  3.6950  3.4844   b'0'
4  7.0756  100.540  3.6303  4.6375   b'0'

[5 rows x 65 columns]
```

[4]:
```
df1_train,df1_test=train_test_split(df1,test_size=0.
 ↪2,random_state=42,stratify=df1['class'])
```

[5]:
```
df1_test.iloc[:,:-1].head()
```

[5]:
```
          Attr1    Attr2     Attr3   Attr4    Attr5     Attr6     Attr7  \
2717   0.051705  0.55183  0.035779  1.1049 -49.0120  0.118170  0.066662
9358  -0.002134  0.39575  0.181490  1.5821  11.1380 -0.016031  0.000683
3760  -0.000167  0.60663  0.093930  1.1658   2.5715 -0.014167  0.001476
10299 -0.246520  0.74705 -0.080064  0.8821 -72.8190  0.000000 -0.246520
608    0.002794  0.20194  0.173990  1.8616   7.8195  0.000000  0.007824

         Attr8   Attr9   Attr10  …    Attr55    Attr56    Attr57   Attr58  \
2717   0.81217  2.0320  0.44817  …   122.810  0.007153  0.115370  0.96984
9358   1.38680  1.0049  0.54880  …  6959.000  0.004836 -0.003889  0.99516
3760   0.64845  3.8371  0.39337  …    21.887 -0.011544 -0.000425  0.99962
10299  0.33838  1.3161  0.25279  …  -251.000 -0.259330 -0.975220  0.94278
608    3.95200  1.4376  0.79806  …   934.000  0.012829  0.003501  0.99460

        Attr59   Attr60   Attr61   Attr62  Attr63   Attr64
```

```
2717   0.47057   7.2783  22.2740   61.239  5.9603   3.2601
9358   0.15293   7.3759   5.1190  108.420  3.3665   2.0717
3760   0.10223  59.4790   6.5103   53.880  6.7743  11.2970
10299  0.00000   4.0771   5.4775  188.340  1.9380   3.2850
608    0.00000   9.9063   6.9273   51.271  7.1190   2.3036

[5 rows x 64 columns]
```

[6]:
```python
def response_to_int(df1_train):
 df1_train.iloc[:,-1]=pd.to_numeric(df1_train.iloc[:,-1])
 df1_train.info()


response_to_int(df1_test)
response_to_int(df1_train)
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2101 entries, 2717 to 9008
Data columns (total 65 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Attr1   2101 non-null   float64
 1   Attr2   2101 non-null   float64
 2   Attr3   2101 non-null   float64
 3   Attr4   2096 non-null   float64
 4   Attr5   2095 non-null   float64
 5   Attr6   2101 non-null   float64
 6   Attr7   2101 non-null   float64
 7   Attr8   2097 non-null   float64
 8   Attr9   2101 non-null   float64
 9   Attr10  2101 non-null   float64
 10  Attr11  2101 non-null   float64
 11  Attr12  2096 non-null   float64
 12  Attr13  2093 non-null   float64
 13  Attr14  2101 non-null   float64
 14  Attr15  2100 non-null   float64
 15  Attr16  2097 non-null   float64
 16  Attr17  2097 non-null   float64
 17  Attr18  2101 non-null   float64
 18  Attr19  2093 non-null   float64
 19  Attr20  2093 non-null   float64
 20  Attr21  1952 non-null   float64
 21  Attr22  2101 non-null   float64
 22  Attr23  2093 non-null   float64
 23  Attr24  2065 non-null   float64
 24  Attr25  2101 non-null   float64
 25  Attr26  2097 non-null   float64
```

```
 26  Attr27  1950 non-null   float64
 27  Attr28  2050 non-null   float64
 28  Attr29  2101 non-null   float64
 29  Attr30  2093 non-null   float64
 30  Attr31  2093 non-null   float64
 31  Attr32  2081 non-null   float64
 32  Attr33  2096 non-null   float64
 33  Attr34  2097 non-null   float64
 34  Attr35  2101 non-null   float64
 35  Attr36  2101 non-null   float64
 36  Attr37  1136 non-null   float64
 37  Attr38  2101 non-null   float64
 38  Attr39  2093 non-null   float64
 39  Attr40  2096 non-null   float64
 40  Attr41  2060 non-null   float64
 41  Attr42  2093 non-null   float64
 42  Attr43  2093 non-null   float64
 43  Attr44  2093 non-null   float64
 44  Attr45  1985 non-null   float64
 45  Attr46  2096 non-null   float64
 46  Attr47  2082 non-null   float64
 47  Attr48  2101 non-null   float64
 48  Attr49  2093 non-null   float64
 49  Attr50  2097 non-null   float64
 50  Attr51  2101 non-null   float64
 51  Attr52  2083 non-null   float64
 52  Attr53  2050 non-null   float64
 53  Attr54  2050 non-null   float64
 54  Attr55  2101 non-null   float64
 55  Attr56  2093 non-null   float64
 56  Attr57  2101 non-null   float64
 57  Attr58  2095 non-null   float64
 58  Attr59  2101 non-null   float64
 59  Attr60  1985 non-null   float64
 60  Attr61  2098 non-null   float64
 61  Attr62  2093 non-null   float64
 62  Attr63  2096 non-null   float64
 63  Attr64  2050 non-null   float64
 64  class   2101 non-null   int64
dtypes: float64(64), int64(1)
memory usage: 1.1 MB
<class 'pandas.core.frame.DataFrame'>
Int64Index: 8402 entries, 1805 to 5744
Data columns (total 65 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Attr1   8402 non-null   float64
 1   Attr2   8402 non-null   float64
```

```
2   Attr3   8402 non-null   float64
3   Attr4   8389 non-null   float64
4   Attr5   8383 non-null   float64
5   Attr6   8402 non-null   float64
6   Attr7   8402 non-null   float64
7   Attr8   8392 non-null   float64
8   Attr9   8399 non-null   float64
9   Attr10  8402 non-null   float64
10  Attr11  8402 non-null   float64
11  Attr12  8389 non-null   float64
12  Attr13  8367 non-null   float64
13  Attr14  8402 non-null   float64
14  Attr15  8395 non-null   float64
15  Attr16  8392 non-null   float64
16  Attr17  8392 non-null   float64
17  Attr18  8402 non-null   float64
18  Attr19  8367 non-null   float64
19  Attr20  8367 non-null   float64
20  Attr21  7744 non-null   float64
21  Attr22  8402 non-null   float64
22  Attr23  8367 non-null   float64
23  Attr24  8211 non-null   float64
24  Attr25  8402 non-null   float64
25  Attr26  8392 non-null   float64
26  Attr27  7838 non-null   float64
27  Attr28  8225 non-null   float64
28  Attr29  8402 non-null   float64
29  Attr30  8367 non-null   float64
30  Attr31  8367 non-null   float64
31  Attr32  8321 non-null   float64
32  Attr33  8389 non-null   float64
33  Attr34  8392 non-null   float64
34  Attr35  8402 non-null   float64
35  Attr36  8402 non-null   float64
36  Attr37  4631 non-null   float64
37  Attr38  8402 non-null   float64
38  Attr39  8367 non-null   float64
39  Attr40  8389 non-null   float64
40  Attr41  8241 non-null   float64
41  Attr42  8367 non-null   float64
42  Attr43  8367 non-null   float64
43  Attr44  8367 non-null   float64
44  Attr45  7927 non-null   float64
45  Attr46  8389 non-null   float64
46  Attr47  8335 non-null   float64
47  Attr48  8402 non-null   float64
48  Attr49  8367 non-null   float64
49  Attr50  8392 non-null   float64
```

```
 50  Attr51  8402 non-null   float64
 51  Attr52  8334 non-null   float64
 52  Attr53  8225 non-null   float64
 53  Attr54  8225 non-null   float64
 54  Attr55  8402 non-null   float64
 55  Attr56  8367 non-null   float64
 56  Attr57  8402 non-null   float64
 57  Attr58  8379 non-null   float64
 58  Attr59  8402 non-null   float64
 59  Attr60  7926 non-null   float64
 60  Attr61  8388 non-null   float64
 61  Attr62  8367 non-null   float64
 62  Attr63  8389 non-null   float64
 63  Attr64  8225 non-null   float64
 64  class   8402 non-null   int64
dtypes: float64(64), int64(1)
memory usage: 4.2 MB
```

```
P:\Anaconda\lib\site-packages\pandas\core\indexing.py:1745:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  isetter(ilocs[0], value)
```

[7]:
```
sum(df1_train['class']==0)
```

[7]: 8006

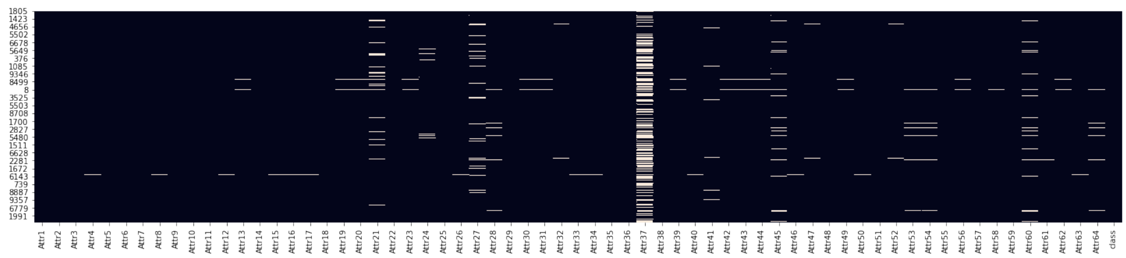
[8]:
```
print('Train DataSet Length=', len(df1_train), '\tCleaned Length=',␣
 ↪len(df1_train.dropna(axis=0, how='any')),
'\t Number of Missing Observations =', len(df1_train)-len(df1_train.
 ↪dropna(axis=0, how='any')))

print('Test DataSet Length=', len(df1_test), '\tCleaned Length=', len(df1_test.
 ↪dropna(axis=0, how='any')),
'\t Number of Missing Observations =', len(df1_test)-len(df1_test.
 ↪dropna(axis=0, how='any')))
```

```
Train DataSet Length= 8402      Cleaned Length= 3891     Number of Missing
Observations = 4511
Test DataSet Length= 2101       Cleaned Length= 994      Number of Missing
Observations = 1107
```

[9]:
```
def drop_response(df1_train) :
 response=df1_train.iloc[:,-1]
```

```
df1_train=df1_train.drop(labels='class',axis=1)
```

[9]:
```
def sparasity_plot(df1_train):
 fig, ax = plt.subplots(figsize=(25,5))
 sns.heatmap(df1_train.isnull(), cbar=False,ax=ax)

sparasity_plot(df1_train)
```



[12]:
```
df1_train.describe()
```

[12]:
```
            Attr1        Attr2        Attr3         Attr4         Attr5  \
count  8402.000000  8402.000000  8402.000000   8389.000000  8.383000e+03
mean      0.054847     0.588978     0.129347     11.308973 -1.764028e+03
std       0.705906     4.913137     4.905252    585.295212  1.323300e+05
min     -17.692000     0.000000  -445.880000      0.002238 -1.190300e+07
25%       0.000691     0.255532     0.015534      1.036500 -5.222450e+01
50%       0.042716     0.463860     0.197595      1.606600  1.778200e+00
75%       0.123448     0.689250     0.420228      2.953800  5.606300e+01
max      52.652000   446.880000    17.708000  53433.000000  6.837700e+05

            Attr6        Attr7         Attr8       Attr9       Attr10  …  \
count  8402.000000  8402.000000   8392.000000  8399.00000  8402.000000  …
mean     -0.087670     0.067418     16.054715     1.82838     0.397535  …
std       5.471911     0.709017    675.340779     8.40540     4.915815  …
min    -486.720000   -17.692000     -2.081800    -1.21570  -445.860000  …
25%       0.000000     0.002069      0.432820     1.00920     0.297382  …
50%       0.000000     0.050569      1.109750     1.19210     0.515275  …
75%       0.072553     0.140813      2.824825     2.05715     0.724808  …
max      45.533000    52.652000  53432.000000   740.44000    11.837000  …

           Attr56       Attr57       Attr58       Attr59        Attr60  \
count  8367.000000  8402.000000  8379.000000  8402.000000  7.926000e+03
mean     -0.029447     0.032885     3.414699     1.660515  6.390950e+02
std       5.936958    20.490525   198.503181    86.322725  4.137726e+04
min    -529.350000 -1667.300000  -198.690000  -172.070000  0.000000e+00
```

```
25%        0.004887     0.007037      0.874940     0.000000  5.545975e+00
50%        0.052046     0.104760      0.952930     0.003126  9.958200e+00
75%        0.131285     0.270845      0.996080     0.248650  2.093375e+01
max        2.763300   552.640000  18118.000000  7617.300000  3.660200e+06

             Attr61         Attr62       Attr63        Attr64        class
count   8388.000000    8367.000000  8389.000000   8225.000000  8402.000000
mean      14.266036     277.043245     9.061467     38.013129     0.047132
std       89.235008    7255.146439    29.760936    460.467365     0.211933
min        0.000000       0.000000     0.000000      0.000000     0.000000
25%        4.492450      40.635000     3.057300      1.997600     0.000000
50%        6.690100      70.494000     5.151600      4.021600     0.000000
75%       10.636250     118.490000     8.921600      9.690000     0.000000
max     4470.400000  501840.000000  1974.500000  21499.000000     1.000000

[8 rows x 65 columns]
```

3 Data Standardization

[10]:
```
def stand(df):
 # define standard scaler
 scaler = StandardScaler()
 # transform data
 scaled = scaler.fit_transform(df)
 scaled=pd.DataFrame(scaled)
 labels=[]
 for i in range(1,66):
    if(i!=65):
     labels.append("Attr"+str(i))
    else:
     labels.append('class')
 response = df1_train['class'].astype(np.int64)
 scaled.columns=labels
 i=scaled.loc[scaled['class']>0]['class'].index
 scaled.loc[i,'class']=scaled.loc[i,'class']/4
 scaled.loc[:,'class']=scaled.loc[:,'class'].astype(np.int64)

 return scaled
```

[11]:
```
df1_train_scaled=stand(df1_train)
df1_test_scaled=stand(df1_test)
```

[12]:
```
sum(df1_test_scaled['class']==0) #To check
```

[12]: 2002

4 Data Imputation

Mean Imputation

[13]:
```
def mean_imputation(df1_train):
    imputer = SimpleImputer(missing_values=np.nan, strategy='mean')
    mean_imputed_df = pd.DataFrame(imputer.fit_transform(df1_train))
    mean_imputed_df.columns=df1_train.columns
    return mean_imputed_df

mean_imputed_df_train=mean_imputation(df1_train_scaled)
mean_imputed_df_test=mean_imputation(df1_test_scaled)
```

KNN Imputation

[14]:
```
def knn_imputation(df1_train):
    knn_imputed_dataset = fancyimpute.KNN(k=100,verbose=True).
 ↪fit_transform(df1_train)
    knn_imputed_df=pd.DataFrame(data=knn_imputed_dataset)
    return knn_imputed_df

knn_imputed_df_train=knn_imputation(df1_train_scaled)
knn_imputed_df_test=knn_imputation(df1_test_scaled)
```

```
Imputing row 1/8402 with 1 missing, elapsed time: 39.236
Imputing row 101/8402 with 0 missing, elapsed time: 39.243
Imputing row 201/8402 with 1 missing, elapsed time: 39.249
Imputing row 301/8402 with 0 missing, elapsed time: 39.257
Imputing row 401/8402 with 0 missing, elapsed time: 39.267
Imputing row 501/8402 with 0 missing, elapsed time: 39.279
Imputing row 601/8402 with 0 missing, elapsed time: 39.286
Imputing row 701/8402 with 0 missing, elapsed time: 39.293
Imputing row 801/8402 with 0 missing, elapsed time: 39.306
Imputing row 901/8402 with 1 missing, elapsed time: 39.315
Imputing row 1001/8402 with 2 missing, elapsed time: 39.320
Imputing row 1101/8402 with 2 missing, elapsed time: 39.326
Imputing row 1201/8402 with 5 missing, elapsed time: 39.333
Imputing row 1301/8402 with 0 missing, elapsed time: 39.340
Imputing row 1401/8402 with 0 missing, elapsed time: 39.353
Imputing row 1501/8402 with 0 missing, elapsed time: 39.362
Imputing row 1601/8402 with 0 missing, elapsed time: 39.368
Imputing row 1701/8402 with 0 missing, elapsed time: 39.376
Imputing row 1801/8402 with 0 missing, elapsed time: 39.386
Imputing row 1901/8402 with 1 missing, elapsed time: 39.394
Imputing row 2001/8402 with 3 missing, elapsed time: 39.403
Imputing row 2101/8402 with 7 missing, elapsed time: 39.410
Imputing row 2201/8402 with 0 missing, elapsed time: 39.421
Imputing row 2301/8402 with 0 missing, elapsed time: 39.434
```

```
Imputing row 2401/8402 with 0 missing, elapsed time: 39.444
Imputing row 2501/8402 with 1 missing, elapsed time: 39.452
Imputing row 2601/8402 with 0 missing, elapsed time: 39.464
Imputing row 2701/8402 with 1 missing, elapsed time: 39.476
Imputing row 2801/8402 with 1 missing, elapsed time: 39.488
Imputing row 2901/8402 with 1 missing, elapsed time: 39.498
Imputing row 3001/8402 with 0 missing, elapsed time: 39.508
Imputing row 3101/8402 with 1 missing, elapsed time: 39.519
Imputing row 3201/8402 with 1 missing, elapsed time: 39.532
Imputing row 3301/8402 with 0 missing, elapsed time: 39.543
Imputing row 3401/8402 with 0 missing, elapsed time: 39.555
Imputing row 3501/8402 with 1 missing, elapsed time: 39.566
Imputing row 3601/8402 with 3 missing, elapsed time: 39.577
Imputing row 3701/8402 with 0 missing, elapsed time: 39.588
Imputing row 3801/8402 with 2 missing, elapsed time: 39.598
Imputing row 3901/8402 with 0 missing, elapsed time: 39.608
Imputing row 4001/8402 with 1 missing, elapsed time: 39.618
Imputing row 4101/8402 with 0 missing, elapsed time: 39.630
Imputing row 4201/8402 with 0 missing, elapsed time: 39.641
Imputing row 4301/8402 with 0 missing, elapsed time: 39.651
Imputing row 4401/8402 with 0 missing, elapsed time: 39.660
Imputing row 4501/8402 with 1 missing, elapsed time: 39.672
Imputing row 4601/8402 with 0 missing, elapsed time: 39.682
Imputing row 4701/8402 with 2 missing, elapsed time: 39.688
Imputing row 4801/8402 with 0 missing, elapsed time: 39.695
Imputing row 4901/8402 with 1 missing, elapsed time: 39.702
Imputing row 5001/8402 with 4 missing, elapsed time: 39.710
Imputing row 5101/8402 with 0 missing, elapsed time: 39.718
Imputing row 5201/8402 with 1 missing, elapsed time: 39.725
Imputing row 5301/8402 with 0 missing, elapsed time: 39.731
Imputing row 5401/8402 with 0 missing, elapsed time: 39.738
Imputing row 5501/8402 with 1 missing, elapsed time: 39.746
Imputing row 5601/8402 with 1 missing, elapsed time: 39.755
Imputing row 5701/8402 with 1 missing, elapsed time: 39.762
Imputing row 5801/8402 with 6 missing, elapsed time: 39.768
Imputing row 5901/8402 with 1 missing, elapsed time: 39.776
Imputing row 6001/8402 with 1 missing, elapsed time: 39.783
Imputing row 6101/8402 with 0 missing, elapsed time: 39.790
Imputing row 6201/8402 with 0 missing, elapsed time: 39.796
Imputing row 6301/8402 with 0 missing, elapsed time: 39.802
Imputing row 6401/8402 with 1 missing, elapsed time: 39.809
Imputing row 6501/8402 with 3 missing, elapsed time: 39.816
Imputing row 6601/8402 with 2 missing, elapsed time: 39.822
Imputing row 6701/8402 with 2 missing, elapsed time: 39.828
Imputing row 6801/8402 with 0 missing, elapsed time: 39.835
Imputing row 6901/8402 with 0 missing, elapsed time: 39.842
Imputing row 7001/8402 with 2 missing, elapsed time: 39.849
Imputing row 7101/8402 with 1 missing, elapsed time: 39.856
```

```
Imputing row 7201/8402 with 0 missing, elapsed time: 39.862
Imputing row 7301/8402 with 0 missing, elapsed time: 39.868
Imputing row 7401/8402 with 0 missing, elapsed time: 39.876
Imputing row 7501/8402 with 1 missing, elapsed time: 39.884
Imputing row 7601/8402 with 2 missing, elapsed time: 39.890
Imputing row 7701/8402 with 0 missing, elapsed time: 39.897
Imputing row 7801/8402 with 0 missing, elapsed time: 39.904
Imputing row 7901/8402 with 0 missing, elapsed time: 39.910
Imputing row 8001/8402 with 0 missing, elapsed time: 39.917
Imputing row 8101/8402 with 2 missing, elapsed time: 39.922
Imputing row 8201/8402 with 0 missing, elapsed time: 39.928
Imputing row 8301/8402 with 0 missing, elapsed time: 39.935
Imputing row 8401/8402 with 0 missing, elapsed time: 39.941
Imputing row 1/2101 with 0 missing, elapsed time: 1.924
Imputing row 101/2101 with 0 missing, elapsed time: 1.927
Imputing row 201/2101 with 1 missing, elapsed time: 1.930
Imputing row 301/2101 with 3 missing, elapsed time: 1.934
Imputing row 401/2101 with 3 missing, elapsed time: 1.939
Imputing row 501/2101 with 3 missing, elapsed time: 1.942
Imputing row 601/2101 with 1 missing, elapsed time: 1.945
Imputing row 701/2101 with 1 missing, elapsed time: 1.948
Imputing row 801/2101 with 1 missing, elapsed time: 1.952
Imputing row 901/2101 with 0 missing, elapsed time: 1.956
Imputing row 1001/2101 with 1 missing, elapsed time: 1.960
Imputing row 1101/2101 with 0 missing, elapsed time: 1.962
Imputing row 1201/2101 with 1 missing, elapsed time: 1.964
Imputing row 1301/2101 with 0 missing, elapsed time: 1.967
Imputing row 1401/2101 with 2 missing, elapsed time: 1.970
Imputing row 1501/2101 with 0 missing, elapsed time: 1.974
Imputing row 1601/2101 with 1 missing, elapsed time: 1.977
Imputing row 1701/2101 with 1 missing, elapsed time: 1.980
Imputing row 1801/2101 with 1 missing, elapsed time: 1.983
Imputing row 1901/2101 with 0 missing, elapsed time: 1.987
Imputing row 2001/2101 with 1 missing, elapsed time: 1.991
Imputing row 2101/2101 with 0 missing, elapsed time: 1.994
```

[15]:

```python
labels=[]
for i in range(1,66):
    if(i!=65):
     labels.append("Attr"+str(i))
    else:
     labels.append('class')

knn_imputed_df_train.columns=labels
knn_imputed_df_test.columns=labels
knn_imputed_df_train.head()
```

[15]:
```
      Attr1     Attr2     Attr3     Attr4     Attr5     Attr6     Attr7  \
0  0.009136  0.044060 -0.023776 -0.017587  0.012771  0.016023 -0.008635
1 -0.302844  0.158568 -0.101395 -0.018074  0.008621 -0.108234 -0.319247
2 -0.044806  0.069361 -0.009242 -0.017449  0.012753  0.008465 -0.053644
3 -0.028218 -0.039600 -0.007837 -0.017196  0.012430  0.016023 -0.045826
4 -0.104057  0.087673 -0.044439 -0.017763  0.012500  0.016023 -0.121332

      Attr8     Attr9    Attr10  …    Attr56    Attr57    Attr58    Attr59  \
0 -0.023416  0.162110 -0.041293  …  0.007313  0.013771 -0.012260 -0.019237
1 -0.024173 -0.008267 -0.155736  …  0.105909  0.019472 -0.015106 -0.019237
2 -0.023662 -0.037178 -0.066579  …  0.007620  0.014524 -0.012260 -0.010607
3 -0.021501 -0.130134  0.042322  …  0.010132  0.001210 -0.012396 -0.019237
4 -0.023803 -0.006280 -0.084883  …  0.003796  0.044461 -0.012113 -0.020600

     Attr60    Attr61    Attr62    Attr63    Attr64  class
0 -0.015328  0.320285 -0.025674 -0.169389 -0.046766    0.0
1 -0.015352 -0.106095  0.000941 -0.261286 -0.016468    0.0
2 -0.015355 -0.128932 -0.009393 -0.245782 -0.014508    0.0
3 -0.015389 -0.099410 -0.012731 -0.238080 -0.079589    0.0
4 -0.015350 -0.115056 -0.009412 -0.245742 -0.029761    0.0

[5 rows x 65 columns]
```

[18]:
```python
knn_imputed_df_train.isnull().sum().sum()
```

[18]: 0

MICE Imputation

[17]:
```python
def mice_imputation(df1_train):
    mice_imputed_dataset = fancyimpute.
 ↪IterativeImputer(verbose=True,max_iter=100,tol=0.01).fit_transform(df1_train)
    mice_imputed_df=pd.DataFrame(data=mice_imputed_dataset)
    return mice_imputed_df
```

[15]:
```python
mice_imputed_df_train=mice_imputation(df1_train_scaled.iloc[:,:-1])
```

```
[IterativeImputer] Completing matrix with shape (8402, 64)
[IterativeImputer] Change: 2354.0504907657614, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 1577.3659903441526, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 919.2456303804995, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 554.5785143855168, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 326.5443318038552, scaled tolerance:
```

```
0.9147375367136575
[IterativeImputer] Change: 192.2944348443667, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 113.34722115318401, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 66.70409624547095, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 39.49515187347934, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 23.61978040876983, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 14.294325542486137, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 11.901997692922613, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 10.645462846612682, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 9.673323711033255, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 8.906217410724356, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 8.292461428162222, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 7.79494137114533, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 7.385456614805407, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 7.044970503512117, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 6.75629496724924, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 6.508237763262328, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 6.2918555758663866, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 6.10102024709741, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 5.929481861954748, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 5.773535939879739, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 5.630148467814146, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 5.497140455282533, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 5.372742420243335, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 5.255544589830594, scaled tolerance:
```

```
0.9147375367136575
[IterativeImputer] Change: 5.1446557241291435, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 5.03918161108634, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.938512527497055, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.842193737586699, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.749701335615194, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.660727890535194, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.575056405723777, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.492423989709864, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.412581578629764, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.335407807911952, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.2607354472111725, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.188417191573588, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.118362105942558, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 4.050419825551883, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.984543459765497, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.920505483368502, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.8582997581785907, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.7978336721139976, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.7389204879657054, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.6817410905237233, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.6259698853009623, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.5716597019173455, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.518798442351475, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.467254683335239, scaled tolerance:
```

```
0.9147375367136575
[IterativeImputer] Change: 3.416964723488163, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.3682563724390673, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.3208223446451695, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.2744968775345513, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.229217737414808, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.1848891191589823, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.1415612947024703, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.0998962488096167, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.059415607812732, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 3.019851243020257, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.981032986377878, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.9430464245893386, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.9058361750043384, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.8692877790435265, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.833453368199243, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.7982846241506776, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.7637564468353433, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.7298709005098547, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.6966382535549, scaled tolerance: 0.9147375367136575
[IterativeImputer] Change: 2.664010640071939, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.631996583917062, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.6005351765455798, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5695904066953754, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5391792690430317, scaled tolerance:
0.9147375367136575
```

```
[IterativeImputer] Change: 2.5347238828438425, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5389470654474526, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.538905563820786, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.538754441108607, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5368915019393086, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.539163662674949, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5377064916937626, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5385904520514524, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.540012169738173, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.54024345675714, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5397894534968977, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5395294764987573, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5391719982451617, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.539860644597674, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.540913465321707, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5402653663147046, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5402643126720763, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5395905100334732, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5400451700735536, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5403772029253964, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5393765502380368, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.5400450853203917, scaled tolerance:
0.9147375367136575
[IterativeImputer] Change: 2.541515590848414, scaled tolerance:
0.9147375367136575
```

```
P:\Anaconda\lib\site-packages\sklearn\impute\_iterative.py:669:
```

```
ConvergenceWarning: [IterativeImputer] Early stopping criterion not reached.
  warnings.warn("[IterativeImputer] Early stopping criterion not"
```

[24]:
```
mice_imputed_df_test=mice_imputation(df1_test_scaled.iloc[:,:-1])
```

```
[IterativeImputer] Completing matrix with shape (2101, 64)
[IterativeImputer] Change: 438.09611079431085, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 287.28834954309445, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 122.85466675465615, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 83.9910599894613, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 37.07068035144622, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 12.05125410380262, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 2.901366186540069, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 2.4060641724286973, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 2.132884911983211, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.932808983834191, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.794765787026566, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.6932107889862344, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.6084519259481609, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.531095601355661, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.4573869141692803, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.38483959059294, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.3168805008123146, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.250427684540251, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.1865820341540232, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.124528638142088, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 1.0674914786841763, scaled tolerance:
0.4578464322505482
```

```
[IterativeImputer] Change: 1.0126492902679807, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.9606811227193351, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.9115441460402204, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.8650784996805242, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.8218272819205744, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.7807479873219076, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.7364961838685358, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.704514990471931, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.6703796761582337, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.6379690182797622, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.6131344885141659, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.611511951177164, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.6098596560905613, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.6081882857593011, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.606498863011441, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.6047947711721798, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.6030731760083042, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.6013477934388447, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5996165328709812, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5978831224815264, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5961505229867587, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5944213416933795, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5926966724137058, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5909804198221774, scaled tolerance:
0.4578464322505482
```

```
[IterativeImputer] Change: 0.589271366164832, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5875728758481346, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5863790547222569, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5853497682756671, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.584299284093963, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5832263025713037, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5821323329077052, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.581020304316936, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5798901908379159, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5787432945106638, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.577583399955966, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5764088238550512, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5752228939440548, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5740265959598665, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5728219544038677, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5716086804905581, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5703876220276154, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5691606366852036, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5679282634217213, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5666915153669103, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5654528793312104, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5642103529550985, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5629657836999694, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5617202105831893, scaled tolerance:
0.4578464322505482
```

```
[IterativeImputer] Change: 0.5604737314617609, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5592266369653676, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5579814838563206, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5567354331883285, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5554924781621556, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5542507159307937, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5530120972536723, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5517755809634464, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.550541328049493, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5493093868221186, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5480836356780731, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5468604338167455, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.545643615902602, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.54442747250144, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5432166992409914, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.542011314517572, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5408110078984865, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5396142231127703, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5384246070260408, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5372392352830259, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5360582374860975, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5348839459255553, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5337129950258495, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5325483901390844, scaled tolerance:
0.4578464322505482
```

```
[IterativeImputer] Change: 0.5313901155948946, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5302368192069452, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5290892352960138, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5279460969383099, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5268105535041484, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.525678550939009, scaled tolerance:
0.4578464322505482
[IterativeImputer] Change: 0.5245489229270716, scaled tolerance:
0.4578464322505482
```

```
P:\Anaconda\lib\site-packages\sklearn\impute\_iterative.py:669:
ConvergenceWarning: [IterativeImputer] Early stopping criterion not reached.
  warnings.warn("[IterativeImputer] Early stopping criterion not"
```

[25]:
```
mice_imputed_df_test['class']=df1_test_scaled['class']
mice_imputed_df_train['class']=df1_train_scaled['class']
```

[26]:
```
mice_imputed_df_train.columns=labels
mice_imputed_df_test.columns=labels
mice_imputed_df_test.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2101 entries, 0 to 2100
Data columns (total 65 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Attr1   2101 non-null   float64
 1   Attr2   2101 non-null   float64
 2   Attr3   2101 non-null   float64
 3   Attr4   2101 non-null   float64
 4   Attr5   2101 non-null   float64
 5   Attr6   2101 non-null   float64
 6   Attr7   2101 non-null   float64
 7   Attr8   2101 non-null   float64
 8   Attr9   2101 non-null   float64
 9   Attr10  2101 non-null   float64
 10  Attr11  2101 non-null   float64
 11  Attr12  2101 non-null   float64
 12  Attr13  2101 non-null   float64
 13  Attr14  2101 non-null   float64
 14  Attr15  2101 non-null   float64
 15  Attr16  2101 non-null   float64
 16  Attr17  2101 non-null   float64
```

```
17  Attr18  2101 non-null   float64
18  Attr19  2101 non-null   float64
19  Attr20  2101 non-null   float64
20  Attr21  2101 non-null   float64
21  Attr22  2101 non-null   float64
22  Attr23  2101 non-null   float64
23  Attr24  2101 non-null   float64
24  Attr25  2101 non-null   float64
25  Attr26  2101 non-null   float64
26  Attr27  2101 non-null   float64
27  Attr28  2101 non-null   float64
28  Attr29  2101 non-null   float64
29  Attr30  2101 non-null   float64
30  Attr31  2101 non-null   float64
31  Attr32  2101 non-null   float64
32  Attr33  2101 non-null   float64
33  Attr34  2101 non-null   float64
34  Attr35  2101 non-null   float64
35  Attr36  2101 non-null   float64
36  Attr37  2101 non-null   float64
37  Attr38  2101 non-null   float64
38  Attr39  2101 non-null   float64
39  Attr40  2101 non-null   float64
40  Attr41  2101 non-null   float64
41  Attr42  2101 non-null   float64
42  Attr43  2101 non-null   float64
43  Attr44  2101 non-null   float64
44  Attr45  2101 non-null   float64
45  Attr46  2101 non-null   float64
46  Attr47  2101 non-null   float64
47  Attr48  2101 non-null   float64
48  Attr49  2101 non-null   float64
49  Attr50  2101 non-null   float64
50  Attr51  2101 non-null   float64
51  Attr52  2101 non-null   float64
52  Attr53  2101 non-null   float64
53  Attr54  2101 non-null   float64
54  Attr55  2101 non-null   float64
55  Attr56  2101 non-null   float64
56  Attr57  2101 non-null   float64
57  Attr58  2101 non-null   float64
58  Attr59  2101 non-null   float64
59  Attr60  2101 non-null   float64
60  Attr61  2101 non-null   float64
61  Attr62  2101 non-null   float64
62  Attr63  2101 non-null   float64
63  Attr64  2101 non-null   float64
64  class   2101 non-null   int64
```

```
dtypes: float64(64), int64(1)
memory usage: 1.0 MB
```

5 Handling Data Imbalance using SMOTE

[16]:
```python
def balanced_data(df):
    y_train=df.iloc[:,-1]
    x_train=df.iloc[:,:-1]
    sm = SMOTE(random_state = 2)
    df1_train_res, y_train_res = sm.fit_sample(x_train, y_train)
    return df1_train_res,y_train_res
```

[28]:
```python
mice_imputed_df_balanced,y_balanced1=balanced_data(mice_imputed_df_train)
knn_imputed_df_balanced,y_balanced2=balanced_data(knn_imputed_df_train)
mean_imputed_df_balanced,y_balanced3=balanced_data(mean_imputed_df_train)
print('After OverSampling, the shape of mice_imputed_df: {}'.
 ↪format(mice_imputed_df_train.shape))
print('After OverSampling, the shape of knn_imputed_df: {}'.
 ↪format(knn_imputed_df_train.shape))
print('After OverSampling, the shape of mean_imputed_df: {}'.
 ↪format(mean_imputed_df_train.shape))
print('After OverSampling, the shape of train_y: {} \n'.format(y_balanced1.
 ↪shape))

print("After OverSampling, counts of label '1' in Mice Imputed df: {}".
 ↪format(sum(y_balanced1 == 1)))
print("After OverSampling, counts of label '0': {}".format(sum(y_balanced1 ==␣
 ↪0)))

print("After OverSampling, counts of label '1' in Knn Imputed df: {}".
 ↪format(sum(y_balanced2 == 1)))
print("After OverSampling, counts of label '0': {}".format(sum(y_balanced2 ==␣
 ↪0)))

print("After OverSampling, counts of label '1' in Mean Imputed df: {}".
 ↪format(sum(y_balanced3 == 1)))
print("After OverSampling, counts of label '0': {}".format(sum(y_balanced3 ==␣
 ↪0)))
```

```
After OverSampling, the shape of mice_imputed_df: (8402, 65)
After OverSampling, the shape of knn_imputed_df: (8402, 65)
After OverSampling, the shape of mean_imputed_df: (8402, 65)
After OverSampling, the shape of train_y: (16012,)

After OverSampling, counts of label '1' in Mice Imputed df: 8006
After OverSampling, counts of label '0': 8006
```

```
After OverSampling, counts of label '1' in Knn Imputed df: 8006
After OverSampling, counts of label '0': 8006
After OverSampling, counts of label '1' in Mean Imputed df: 8006
After OverSampling, counts of label '0': 8006
```

[31]:
```
mice_imputed_df_balanced_test,y_balanced1_test=balanced_data(mice_imputed_df_test)
knn_imputed_df_balanced_test,y_balanced2_test=balanced_data(knn_imputed_df_test)
mean_imputed_df_balanced_test,y_balanced3_test=balanced_data(mean_imputed_df_test)
```

[56]:
```
mice_imputed_df_balanced_test.head()
```

[56]:
```
      Attr1     Attr2     Attr3     Attr4     Attr5     Attr6     Attr7  \
0  0.021195 -0.018291  0.007223 -0.124372 -0.023005  0.033637  0.024844
1 -0.144941 -0.033177  0.021129 -0.107694 -0.019239  0.021543 -0.174797
2 -0.138871 -0.013065  0.012773 -0.122244 -0.019775  0.021711 -0.172395
3 -0.899060  0.000328 -0.003833 -0.132159 -0.024495  0.022988 -0.922784
4 -0.129732 -0.051662  0.020413 -0.097925 -0.019446  0.022988 -0.153188

      Attr8     Attr9    Attr10  …   Attr55    Attr56    Attr57    Attr58  \
0 -0.035387  0.113277  0.019821  … -0.113887  0.020382  0.039483 -0.030256
1 -0.034724 -0.353552  0.029419  … -0.015549  0.020363  0.024812 -0.030090
2 -0.035576  0.933716  0.014594  … -0.115339  0.020232  0.025238 -0.030060
3 -0.035934 -0.212108  0.001186  … -0.119264  0.018242 -0.094681 -0.030433
4 -0.031765 -0.156885  0.053193  … -0.102218  0.020427  0.025721 -0.030093

     Attr59    Attr60    Attr61    Attr62    Attr63    Attr64
0 -0.004817 -0.038437  0.171303  0.008735 -0.087601 -0.089674
1 -0.049566 -0.038424 -0.132889  0.009573 -0.157094 -0.094210
2 -0.056708 -0.031601 -0.108218  0.008604 -0.065792 -0.058998
3 -0.071110 -0.038856 -0.126532  0.010994 -0.195367 -0.089579
4 -0.071110 -0.038093 -0.100824  0.008557 -0.056557 -0.093325

[5 rows x 64 columns]
```

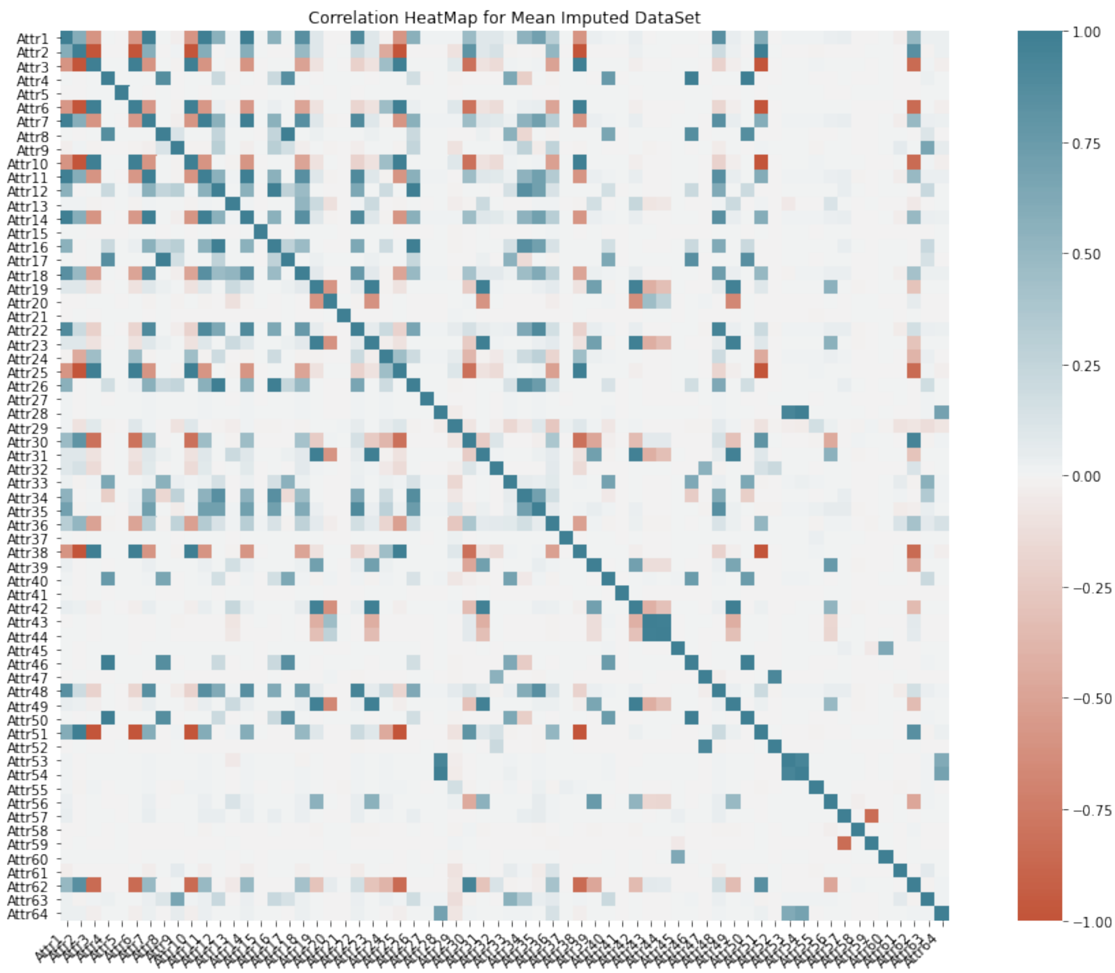
6 Checking Multicollinearity

[57]:
```
corr1=mean_imputed_df_balanced.corr()
corr2=knn_imputed_df_balanced.corr()
corr3=mice_imputed_df_balanced.corr()
```

[58]:
```
f, ax = plt.subplots(figsize=(20, 16))

# Generate a custom diverging colormap
ax = sns.heatmap(
    corr1,
    vmin=-1, vmax=1, center=0,
```

```
    cmap=sns.diverging_palette(20, 220, n=200),
    square=True
)
ax.set_xticklabels(
    ax.get_xticklabels(),
    rotation=45,
    horizontalalignment='right'
);
ax.set_title('Correlation HeatMap for Mean Imputed DataSet')
```

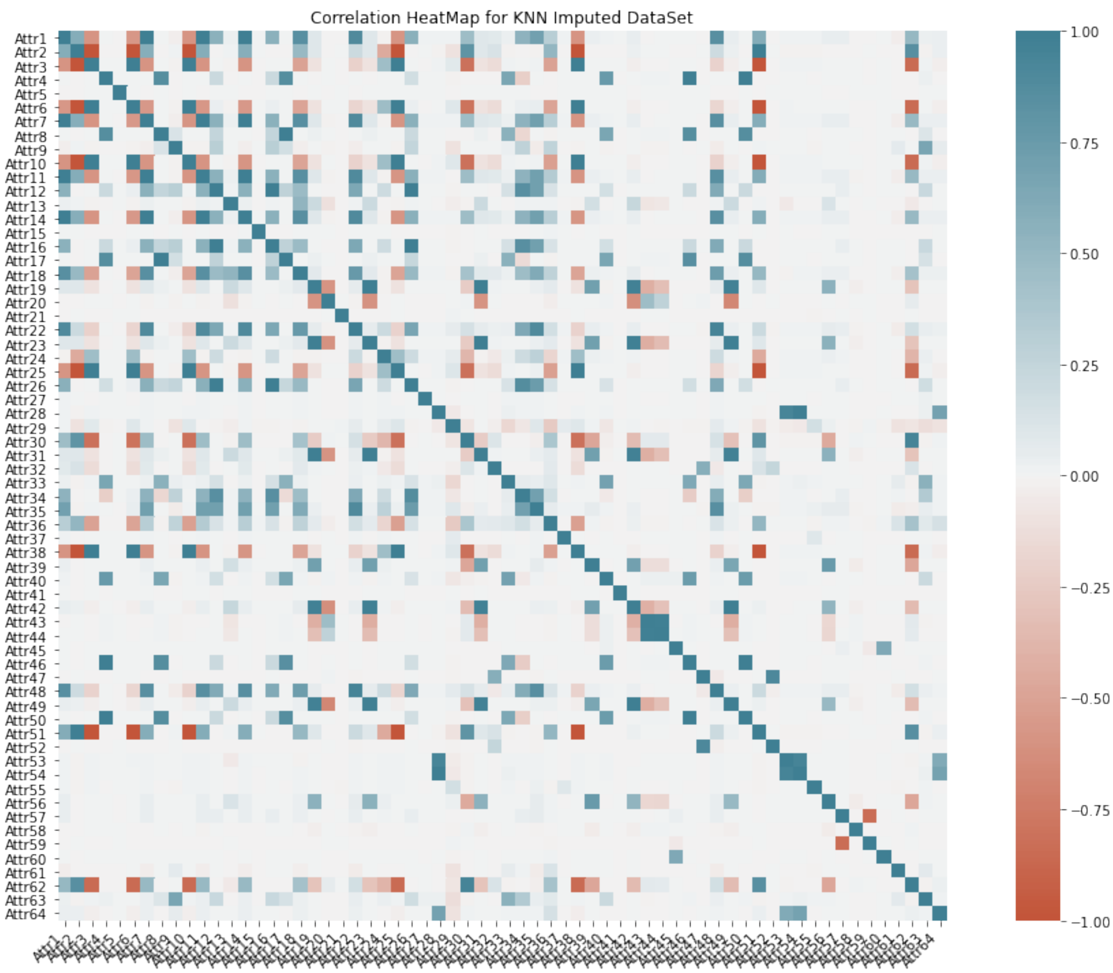
[58]: Text(0.5, 1.0, 'Correlation HeatMap for Mean Imputed DataSet')



[59]:
```
f, ax = plt.subplots(figsize=(18, 12))

# Generate a custom diverging colormap
ax = sns.heatmap(
    corr2,
```

```
    vmin=-1, vmax=1, center=0,
    cmap=sns.diverging_palette(20, 220, n=200),
    square=True
)
ax.set_xticklabels(
    ax.get_xticklabels(),
    rotation=45,
    horizontalalignment='right'
);
ax.set_title('Correlation HeatMap for KNN Imputed DataSet')
```

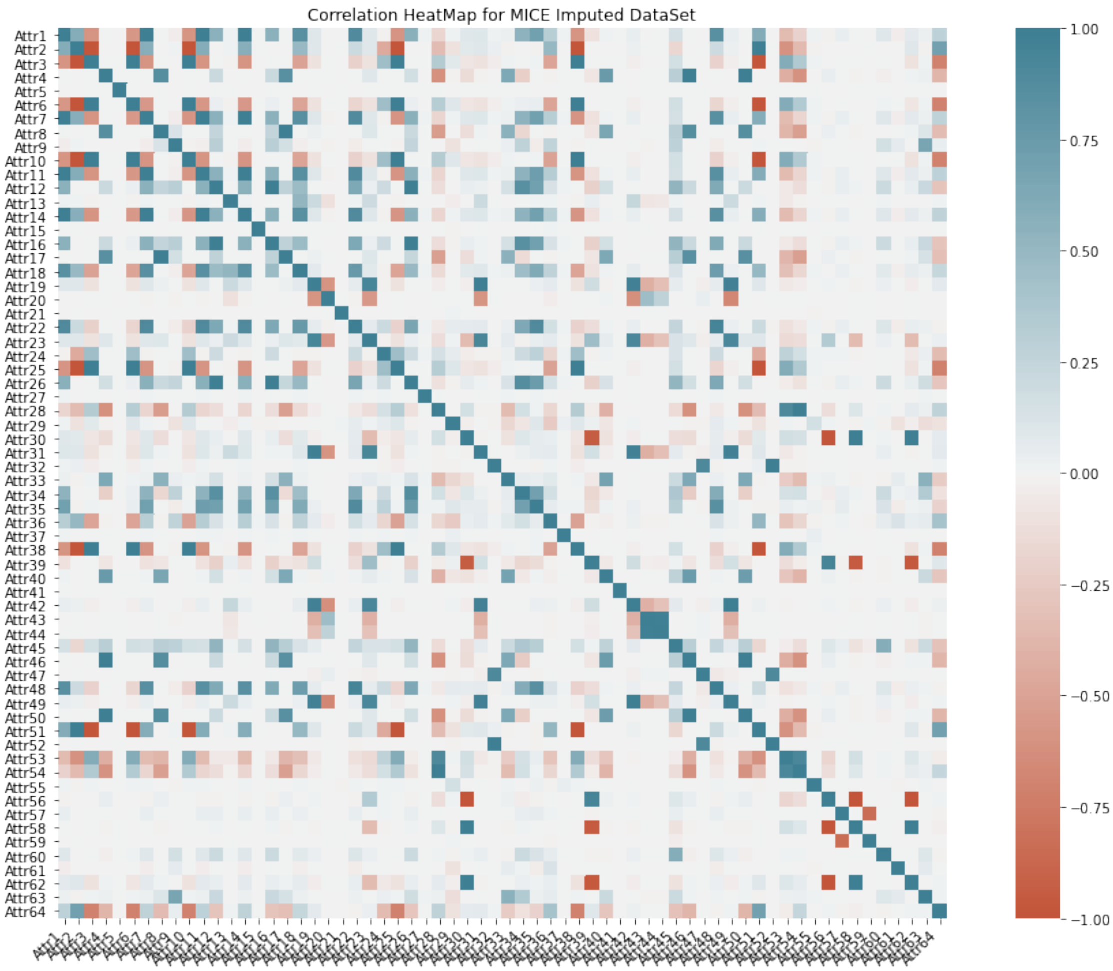
[59]: Text(0.5, 1.0, 'Correlation HeatMap for KNN Imputed DataSet')



[60]:
```
f, ax = plt.subplots(figsize=(18, 12))

# Generate a custom diverging colormap
ax = sns.heatmap(
```

```
    corr3,
    vmin=-1, vmax=1, center=0,
    cmap=sns.diverging_palette(20, 220, n=200),
    square=True
)
ax.set_xticklabels(
    ax.get_xticklabels(),
    rotation=45,
    horizontalalignment='right'
);
ax.set_title('Correlation HeatMap for MICE Imputed DataSet')
```

[60]: Text(0.5, 1.0, 'Correlation HeatMap for MICE Imputed DataSet')



7 Computing VIF

[61]:
```
Z1=pd.DataFrame(add_constant(mean_imputed_df_balanced))
Z2=add_constant(knn_imputed_df_balanced)
Z3=add_constant(mice_imputed_df_balanced)
```

[62]:
```
Z1
```

[62]:
```
       const     Attr1     Attr2     Attr3     Attr4     Attr5     Attr6  \
0        1.0  0.009136  0.044060 -0.023776 -0.017587  0.012771  0.016023
1        1.0 -0.302844  0.158568 -0.101395 -0.018074  0.008621 -0.108234
2        1.0 -0.044806  0.069361 -0.009242 -0.017449  0.012753  0.008465
3        1.0 -0.028218 -0.039600 -0.007837 -0.017196  0.012430  0.016023
4        1.0 -0.104057  0.087673 -0.044439 -0.017763  0.012500  0.016023
…        …         …         …         …         …         …         …
16007    1.0  0.102186 -0.012043  0.031581 -0.016449  0.013551  0.039229
16008    1.0 -0.065525  0.028729  0.026344 -0.016960  0.013306 -0.022324
16009    1.0  0.061463 -0.102202  0.092507 -0.006070  0.014026  0.033976
16010    1.0 -0.060358  0.026477 -0.011589 -0.017177  0.013537  0.016023
16011    1.0 -0.152979  0.018119 -0.057296 -0.018009  0.013032  0.016023

          Attr7     Attr8     Attr9  …    Attr55    Attr56    Attr57  \
0     -0.008635 -0.023416  0.162110  … -0.107653  0.007313  0.013771
1     -0.319247 -0.024173 -0.008267  … -0.110570  0.105909  0.019472
2     -0.053644 -0.023662 -0.037178  … -0.080205  0.007620  0.014524
3     -0.045826 -0.021501 -0.130134  … -0.095742  0.010132  0.001210
4     -0.121332 -0.023803 -0.006280  … -0.119587  0.003796  0.044461
…           …         …         …  …        …         …         …
16007  0.125378 -0.022939 -0.089953  …  0.340355  0.016248  0.020623
16008 -0.076606 -0.023190  0.212112  … -0.063341  0.002533  0.001033
16009  0.051243 -0.008703 -0.089869  …  0.259217  0.016030  0.003740
16010 -0.073053 -0.023189 -0.156423  … -0.060733  0.004209  0.000597
16011 -0.173052 -0.023068  0.346577  … -0.122320  0.013841 -0.009528

         Attr58    Attr59    Attr60    Attr61    Attr62    Attr63  \
0     -0.012260 -0.019237 -0.015328  0.320285 -0.025674 -0.169389
1     -0.015106 -0.019237 -0.015352 -0.106095  0.000941 -0.261286
2     -0.012260 -0.010607 -0.015355 -0.128932 -0.009393 -0.245782
3     -0.012396 -0.019237 -0.015389 -0.099410 -0.012731 -0.238080
4     -0.012113 -0.020600 -0.015350 -0.115056 -0.009412 -0.245742
…           …         …         …         …         …         …
16007 -0.012503 -0.016947 -0.014969 -0.111412 -0.024147 -0.146515
16008 -0.012209 -0.019237 -0.014216 -0.099774 -0.027435 -0.129295
16009 -0.012496 -0.019236 -0.015300 -0.102502 -0.035169  0.255547
16010 -0.012659 -0.009203 -0.014875 -0.106162 -0.009372 -0.243126
16011 -0.012490 -0.018557 -0.015113  0.358438 -0.031250 -0.059803
```

```
             Attr64
0     -4.676648e-02
1      3.887468e-18
2     -1.450816e-02
3     -7.958948e-02
4     -2.976098e-02
…               …
16007 -5.895125e-02
16008  4.009669e-01
16009 -7.266913e-02
16010 -8.076855e-02
16011 -6.190899e-02

[16012 rows x 65 columns]
```

[63]:
```python
def vif(Z3):
    vif_data = pd.DataFrame()
    vif_data["feature"] = Z3.columns

        # calculating VIF for each feature
    vif_data["VIF"] = [variance_inflation_factor(np.array(Z3), i)
                              for i in range(len(Z3.columns))]

    return vif_data
```

[64]:
```python
vif_mean=vif(mean_imputed_df_balanced)
vif_knn=vif(knn_imputed_df_balanced)
vif_mice=vif(mice_imputed_df_balanced)
```

```
P:\Anaconda\lib\site-packages\statsmodels\stats\outliers_influence.py:193:
RuntimeWarning: divide by zero encountered in double_scalars
  vif = 1. / (1. - r_squared_i)
```

[65]:
```python
vif_mean.sort_values(by=['VIF'],ascending=False)
```

[65]:
```
   feature           VIF
17  Attr18           inf
6    Attr7           inf
13  Attr14           inf
42  Attr43  9.918403e+10
43  Attr44  8.425739e+10
..     …             …
26  Attr27  1.002918e+00
14  Attr15  1.002749e+00
20  Attr21  1.001218e+00
40  Attr41  1.000434e+00
```

```
4     Attr5  1.000178e+00

[64 rows x 2 columns]
```

[58]:
```
vif_knn.sort_values(by=['VIF'],ascending=False)
```

[58]:
```
   feature           VIF
17  Attr18           inf
6    Attr7           inf
13  Attr14           inf
42  Attr43  1.108824e+11
43  Attr44  9.404443e+10
..     …           …
26  Attr27  1.003578e+00
20  Attr21  1.001957e+00
14  Attr15  1.001851e+00
40  Attr41  1.000534e+00
4    Attr5  1.000249e+00

[65 rows x 2 columns]
```

[66]:
```
vif_mice.sort_values(by=['VIF'],ascending=False)
```

[66]:
```
   feature           VIF
6    Attr7           inf
13  Attr14           inf
17  Attr18           inf
42  Attr43  1.031374e+11
43  Attr44  8.761611e+10
..     …           …
26  Attr27  1.002974e+00
14  Attr15  1.002687e+00
20  Attr21  1.001308e+00
40  Attr41  1.000492e+00
4    Attr5  1.000255e+00

[64 rows x 2 columns]
```

8 Iterative VIF Elimination Procedure

[67]:
```
def vif_func(X,thresh=10):
    var_names=X.columns
    vif_data = pd.DataFrame()
    vif_data["feature"] = X.columns
    Y=X
# calculating VIF for each feature
```

```
    vif_data["VIF"] = [variance_inflation_factor(X.values, i)
                          for i in range(len(X.columns))]
    vif_max=vif_data['VIF'].max()
    vif_features_max=vif_data[vif_data['VIF']==vif_max]['feature']

    if(vif_max<=thresh):
        print("All Variables have VIF <",thresh," Max VIF is ",vif_max)
        return var_names
    else:
       while(vif_max>thresh):
           X=X.drop(vif_features_max, inplace=False, axis=1)
           print("\n Dropped Features",list(vif_features_max),"\n Max VIF␣
↪was",vif_max)
           var_names=X.columns
           vif_data = pd.DataFrame()
           vif_data["feature"] = X.columns
           vif_data["VIF"] = [variance_inflation_factor(X.values, i)
                              for i in range(len(X.columns))]
           vif_max=vif_data['VIF'].max()
           vif_features_max=vif_data[vif_data['VIF']==vif_max]['feature']

           if(vif_max<=thresh):
            print("All Variables have VIF <",thresh," Max VIF is ",vif_max)
            break

    return X.columns
```

[68]:
```
#Don't run, takes time unless you need to!
col_mean=vif_func(Z1,5)[1:]
col_knn=vif_func(Z2,5)[1:]
col_mice=vif_func(Z3,5)[1:]
```

```
P:\Anaconda\lib\site-packages\statsmodels\stats\outliers_influence.py:193:
RuntimeWarning: divide by zero encountered in double_scalars
  vif = 1. / (1. - r_squared_i)


 Dropped Features ['Attr7', 'Attr14', 'Attr18']
 Max VIF was inf

 Dropped Features ['Attr43']
 Max VIF was 99198229677.76424

 Dropped Features ['Attr17']
 Max VIF was 90862.6491291998
```

```
Dropped Features ['Attr19']
Max VIF was 11758.582858795868

Dropped Features ['Attr16']
Max VIF was 11345.790558023062

Dropped Features ['Attr10']
Max VIF was 10458.37854718111

Dropped Features ['Attr4']
Max VIF was 7540.988670626193

Dropped Features ['Attr51']
Max VIF was 3335.1325747266674

Dropped Features ['Attr42']
Max VIF was 2184.9853287943715

Dropped Features ['Attr38']
Max VIF was 1627.133410692013

Dropped Features ['Attr23']
Max VIF was 1390.6620575515021

Dropped Features ['Attr2']
Max VIF was 1045.1536755524517

Dropped Features ['Attr46']
Max VIF was 638.9726420873458

Dropped Features ['Attr25']
Max VIF was 429.300323001772

Dropped Features ['Attr54']
Max VIF was 229.8054851871444

Dropped Features ['Attr3']
Max VIF was 195.896085515913

Dropped Features ['Attr26']
Max VIF was 167.94823085797367

Dropped Features ['Attr1']
Max VIF was 100.78083189196968

Dropped Features ['Attr62']
Max VIF was 89.18008033205179
```

```
 Dropped Features ['Attr31']
 Max VIF was 60.10726841557395

 Dropped Features ['Attr33']
 Max VIF was 39.80619949279961

 Dropped Features ['Attr22']
 Max VIF was 36.50138200829714

 Dropped Features ['Attr47']
 Max VIF was 35.00687937532918

 Dropped Features ['Attr12']
 Max VIF was 25.500457163234344

 Dropped Features ['Attr11']
 Max VIF was 15.85460147219793

 Dropped Features ['Attr28']
 Max VIF was 10.919535468989078

 Dropped Features ['Attr30']
 Max VIF was 9.374296653773948

 Dropped Features ['Attr49']
 Max VIF was 8.354846089937253

 Dropped Features ['Attr50']
 Max VIF was 5.782588326437771

 Dropped Features ['Attr35']
 Max VIF was 5.588356465638225
All Variables have VIF < 5  Max VIF is  3.4014603727935615

P:\Anaconda\lib\site-packages\statsmodels\stats\outliers_influence.py:193:
RuntimeWarning: divide by zero encountered in double_scalars
  vif = 1. / (1. - r_squared_i)


 Dropped Features ['Attr7', 'Attr14', 'Attr18']
 Max VIF was inf

 Dropped Features ['Attr43']
 Max VIF was 99313074091.63672

 Dropped Features ['Attr17']
 Max VIF was 90824.45844481561

 Dropped Features ['Attr19']
```

```
Max VIF was 11752.339662929407

Dropped Features ['Attr16']
Max VIF was 11346.173259295429

Dropped Features ['Attr10']
Max VIF was 10457.557041443071

Dropped Features ['Attr4']
Max VIF was 7564.2156916977

Dropped Features ['Attr51']
Max VIF was 3336.2272222156807

Dropped Features ['Attr42']
Max VIF was 2186.2803716236344

Dropped Features ['Attr38']
Max VIF was 1628.0126307731523

Dropped Features ['Attr23']
Max VIF was 1391.8632281057116

Dropped Features ['Attr2']
Max VIF was 1046.1801602979524

Dropped Features ['Attr46']
Max VIF was 641.1737689398424

Dropped Features ['Attr25']
Max VIF was 429.74687542544916

Dropped Features ['Attr54']
Max VIF was 230.26522212393039

Dropped Features ['Attr3']
Max VIF was 196.40722231703907

Dropped Features ['Attr26']
Max VIF was 167.98514031384246

Dropped Features ['Attr1']
Max VIF was 100.71437073617939

Dropped Features ['Attr62']
Max VIF was 89.0931334617361

Dropped Features ['Attr31']
```

```
 Max VIF was 60.14260109411895

 Dropped Features ['Attr33']
 Max VIF was 39.79808183883795

 Dropped Features ['Attr22']
 Max VIF was 36.980235433297096

 Dropped Features ['Attr47']
 Max VIF was 35.48805352177357

 Dropped Features ['Attr12']
 Max VIF was 25.53262333146463

 Dropped Features ['Attr11']
 Max VIF was 16.07674307316

 Dropped Features ['Attr28']
 Max VIF was 10.946497323728085

 Dropped Features ['Attr30']
 Max VIF was 9.386336274979882

 Dropped Features ['Attr49']
 Max VIF was 8.359862909907719

 Dropped Features ['Attr50']
 Max VIF was 5.800315247122059

 Dropped Features ['Attr35']
 Max VIF was 5.651511062304876
All Variables have VIF < 5  Max VIF is  3.401544217511379

P:\Anaconda\lib\site-packages\statsmodels\stats\outliers_influence.py:193:
RuntimeWarning: divide by zero encountered in double_scalars
  vif = 1. / (1. - r_squared_i)


 Dropped Features ['Attr7', 'Attr14', 'Attr18']
 Max VIF was inf

 Dropped Features ['Attr43']
 Max VIF was 103223727692.7422

 Dropped Features ['Attr32']
 Max VIF was 143456.16386818

 Dropped Features ['Attr17']
 Max VIF was 97573.68960836386
```

```
Dropped Features ['Attr56']
Max VIF was 27653.75968004387

Dropped Features ['Attr16']
Max VIF was 13066.998567704035

Dropped Features ['Attr19']
Max VIF was 11275.839991180235

Dropped Features ['Attr10']
Max VIF was 10504.889540562235

Dropped Features ['Attr62']
Max VIF was 10356.85087159006

Dropped Features ['Attr4']
Max VIF was 7620.953039930257

Dropped Features ['Attr51']
Max VIF was 3211.2494496756376

Dropped Features ['Attr42']
Max VIF was 2292.469038146789

Dropped Features ['Attr30']
Max VIF was 1953.3760377147444

Dropped Features ['Attr38']
Max VIF was 1621.6190855191605

Dropped Features ['Attr23']
Max VIF was 1567.2037884267374

Dropped Features ['Attr2']
Max VIF was 1031.397123764891

Dropped Features ['Attr46']
Max VIF was 638.2907277827107

Dropped Features ['Attr54']
Max VIF was 468.0610557067258

Dropped Features ['Attr25']
Max VIF was 428.9284996384918

Dropped Features ['Attr26']
Max VIF was 170.46937092725597
```

```
Dropped Features ['Attr3']
Max VIF was 140.3438179363889

Dropped Features ['Attr1']
Max VIF was 99.19181673292631

Dropped Features ['Attr39']
Max VIF was 92.17629775716549

Dropped Features ['Attr33']
Max VIF was 42.12793041996143

Dropped Features ['Attr22']
Max VIF was 37.76341676349815

Dropped Features ['Attr49']
Max VIF was 26.431450766459214

Dropped Features ['Attr12']
Max VIF was 26.135930576555214

Dropped Features ['Attr53']
Max VIF was 22.10995639882181

Dropped Features ['Attr11']
Max VIF was 16.13554858701738

Dropped Features ['Attr50']
Max VIF was 7.043986645131798

Dropped Features ['Attr6']
Max VIF was 6.544551116332987

Dropped Features ['Attr47']
Max VIF was 6.126569808799159

Dropped Features ['Attr35']
Max VIF was 5.52631088261081
All Variables have VIF < 5  Max VIF is  3.4009577965595836
```

[70]:
```
rem_var=pd.DataFrame()
rem_var['Mean Imputation']=col_mean
rem_var['KNN Imputation']=col_knn
rem_var1=pd.DataFrame(col_mice)

remaining_variables = pd.concat([rem_var, rem_var1], axis=1)
```

```
remaining_variables.columns=['Mean Imputation','KNN Imputation','MICE␣
↪Imputation']
remaining_variables            #Variables Remaining in Each  DataFrame after␣
↪Iterative VIF Elimination Method
```

[70]:

```
   Mean Imputation KNN Imputation MICE Imputation
0            Attr5          Attr5           Attr5
1            Attr6          Attr6           Attr8
2            Attr8          Attr8           Attr9
3            Attr9          Attr9          Attr13
4           Attr13         Attr13          Attr15
5           Attr15         Attr15          Attr20
6           Attr20         Attr20          Attr21
7           Attr21         Attr21          Attr24
8           Attr24         Attr24          Attr27
9           Attr27         Attr27          Attr28
10          Attr29         Attr29          Attr29
11          Attr32         Attr32          Attr31
12          Attr34         Attr34          Attr34
13          Attr36         Attr36          Attr36
14          Attr37         Attr37          Attr37
15          Attr39         Attr39          Attr40
16          Attr40         Attr40          Attr41
17          Attr41         Attr41          Attr44
18          Attr44         Attr44          Attr45
19          Attr45         Attr45          Attr48
20          Attr48         Attr48          Attr52
21          Attr52         Attr52          Attr55
22          Attr53         Attr53          Attr57
23          Attr55         Attr55          Attr58
24          Attr56         Attr56          Attr59
25          Attr57         Attr57          Attr60
26          Attr58         Attr58          Attr61
27          Attr59         Attr59          Attr63
28          Attr60         Attr60          Attr64
29          Attr61         Attr61             NaN
30          Attr63         Attr63             NaN
31          Attr64         Attr64             NaN
```

9 Exporting to R For Variance Decomposition Analysis and Variable Selection

[85]:

```
def file_export(Z1,col,y_balanced,location):
    mean_df=Z1[col]
    mean_df['response']=y_balanced
    mean_df['response']=mean_df['response'].astype(int)
```

```
    mean_df.to_csv(location)

file_export(Z1,col_mean,y_balanced3,'C:/Users/rick7/Desktop/mean.csv')
file_export(Z2,col_knn,y_balanced2,'C:/Users/rick7/Desktop/knn.csv')
file_export(Z1,col_mice,y_balanced1,'C:/Users/rick7/Desktop/mice.csv')
```

```
<ipython-input-85-f416dd80e494>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  mean_df['response']=y_balanced
<ipython-input-85-f416dd80e494>:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  mean_df['response']=mean_df['response'].astype(int)
```

[140]:

```
file_export(mice_imputed_df_balanced,mice_imputed_df_balanced.
 ↪columns,y_balanced1,'C:/Users/rick7/Desktop/micefull.csv')

file_export(knn_imputed_df_balanced,knn_imputed_df_balanced.
 ↪columns,y_balanced2,'C:/Users/rick7/Desktop/knnfull.csv')

file_export(mean_imputed_df_balanced,mean_imputed_df_balanced.
 ↪columns,y_balanced3,'C:/Users/rick7/Desktop/meanfull.csv')
```

[250]:

```
file_export(mice_imputed_df_balanced_test,mice_imputed_df_balanced.
 ↪columns,y_balanced1_test,'C:/Users/rick7/Desktop/micefulltest.csv')

file_export(knn_imputed_df_balanced_test,knn_imputed_df_balanced.
 ↪columns,y_balanced2_test,'C:/Users/rick7/Desktop/knnfulltest.csv')

file_export(mean_imputed_df_balanced_test,mean_imputed_df_balanced.
 ↪columns,y_balanced3_test,'C:/Users/rick7/Desktop/meanfulltest.csv')
```

[88]:

```
mean_decomp=pd.read_csv('C:/Users/rick7/Desktop/Mean_Decomp.csv')
knn_decomp=pd.read_csv('C:/Users/rick7/Desktop/KNN_Decomp.csv')
mice_decomp=pd.read_csv('C:/Users/rick7/Desktop/MICE_Decomp.csv')
```

[88]:

```
   cond.index  Attr5   Attr6   Attr8   Attr9  Attr13  Attr15  Attr20  \
0      1.0000 0.0000  0.3759  0.0000  0.0000  0.0000  0.0000  0.0000
1      1.6884 0.0000  0.0302  0.0017  0.0295  0.0001  0.0000  0.0002
```

2	1.8638	0.0000	0.0082	0.0106	0.0009	0.0008	0.0000	0.0002
3	1.9174	0.0000	0.0004	0.0001	0.0008	0.0003	0.0000	0.0001
4	1.9658	0.0000	0.0169	0.0769	0.0017	0.0008	0.0002	0.0000
5	1.9888	0.0000	0.0031	0.0006	0.0003	0.0281	0.0001	0.0074
6	2.1079	0.0000	0.0003	0.0047	0.0043	0.0026	0.0000	0.0002
7	2.1887	0.0000	0.0000	0.0000	0.0029	0.0001	0.0000	0.0000
8	2.2093	0.0000	0.0001	0.0109	0.0620	0.0014	0.0006	0.0007
9	2.4183	0.0000	0.0012	0.0010	0.0004	0.0263	0.0007	0.3670
10	2.4657	0.0000	0.0002	0.0000	0.0005	0.0007	0.0544	0.0032
11	2.4785	0.0000	0.0008	0.0001	0.0000	0.0000	0.9347	0.0005
12	2.5291	0.0004	0.0481	0.0059	0.0044	0.0046	0.0034	0.0054
13	2.6627	0.0118	0.0303	0.0063	0.0027	0.0607	0.0016	0.0001
14	2.7122	0.9110	0.0005	0.0000	0.0000	0.0012	0.0002	0.0001
15	2.7258	0.0000	0.0017	0.0011	0.0005	0.0246	0.0000	0.0005
16	2.7340	0.0365	0.0040	0.0001	0.0001	0.0000	0.0013	0.0001
17	2.7506	0.0344	0.0225	0.0008	0.0010	0.0001	0.0021	0.0003
18	2.8049	0.0001	0.0036	0.0009	0.0014	0.5976	0.0000	0.0001
19	2.8144	0.0001	0.0100	0.0012	0.0015	0.0329	0.0002	0.0000
20	2.8320	0.0001	0.0000	0.0002	0.0001	0.1026	0.0000	0.0000
21	2.8879	0.0051	0.0051	0.0001	0.0027	0.0223	0.0001	0.0013
22	3.0732	0.0001	0.0163	0.0001	0.0003	0.0005	0.0000	0.0000
23	3.1597	0.0003	0.2675	0.0001	0.0064	0.0003	0.0000	0.0005
24	3.2589	0.0001	0.0000	0.0001	0.0001	0.0699	0.0000	0.6094
25	3.6320	0.0000	0.0000	0.0002	0.0002	0.0000	0.0000	0.0000
26	3.9790	0.0000	0.0001	0.3532	0.1749	0.0018	0.0002	0.0000
27	4.4011	0.0000	0.0010	0.0053	0.0011	0.0144	0.0000	0.0009
28	4.5887	0.0000	0.0001	0.0004	0.0013	0.0000	0.0000	0.0000
29	5.1038	0.0000	0.1510	0.0372	0.0161	0.0004	0.0001	0.0011
30	5.5096	0.0000	0.0000	0.0005	0.0013	0.0047	0.0000	0.0007
31	5.7922	0.0000	0.0008	0.4798	0.6803	0.0000	0.0001	0.0000
32	6.5246	0.0000	0.0000	0.0000	0.0002	0.0000	0.0000	0.0000

	Attr21	Attr24	…	Attr53	Attr55	Attr56	Attr57	Attr58	Attr59	\
0	0.0000	0.0051	…	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
1	0.0000	0.0109	…	0.0015	0.0012	0.0002	0.0009	0.0000	0.0004	
2	0.0000	0.0200	…	0.0008	0.0100	0.0062	0.0068	0.0002	0.0038	
3	0.0000	0.0019	…	0.0001	0.0008	0.0014	0.0823	0.0000	0.0633	
4	0.0000	0.0003	…	0.0050	0.0082	0.0041	0.0006	0.0001	0.0005	
5	0.0000	0.0010	…	0.0004	0.0026	0.0831	0.0000	0.0005	0.0001	
6	0.0000	0.0014	…	0.1559	0.0043	0.0000	0.0000	0.0000	0.0000	
7	0.0000	0.0012	…	0.0013	0.0012	0.0000	0.0009	0.0000	0.0006	
8	0.0001	0.0279	…	0.0024	0.0346	0.0000	0.0000	0.0002	0.0002	
9	0.0000	0.0005	…	0.0001	0.0020	0.0167	0.0000	0.0007	0.0000	
10	0.0000	0.0035	…	0.0000	0.0011	0.0001	0.0000	0.0003	0.0000	
11	0.0000	0.0002	…	0.0001	0.0024	0.0001	0.0000	0.0000	0.0000	
12	0.0003	0.0296	…	0.0102	0.1372	0.0000	0.0000	0.0020	0.0001	
13	0.0038	0.0011	…	0.0002	0.2874	0.0001	0.0000	0.1411	0.0000	

14	0.0030	0.0001	…	0.0000	0.0458	0.0000	0.0000	0.0065	0.0000
15	0.0070	0.0002	…	0.0000	0.0860	0.0008	0.0000	0.8092	0.0000
16	0.0077	0.0006	…	0.0002	0.0607	0.0000	0.0000	0.0075	0.0000
17	0.0327	0.0013	…	0.0017	0.1378	0.0000	0.0000	0.0089	0.0000
18	0.1280	0.0003	…	0.0005	0.0295	0.0051	0.0000	0.0000	0.0000
19	0.0016	0.0036	…	0.0023	0.0022	0.0002	0.0000	0.0149	0.0000
20	0.7721	0.0015	…	0.0003	0.0397	0.0009	0.0000	0.0022	0.0000
21	0.0427	0.1020	…	0.0032	0.0923	0.0003	0.0000	0.0047	0.0000
22	0.0003	0.1113	…	0.0005	0.0017	0.0000	0.0000	0.0000	0.0000
23	0.0004	0.6374	…	0.0018	0.0019	0.0001	0.0000	0.0000	0.0002
24	0.0001	0.0024	…	0.0007	0.0003	0.0065	0.0000	0.0002	0.0000
25	0.0000	0.0000	…	0.0000	0.0034	0.0000	0.0000	0.0000	0.0000
26	0.0000	0.0094	…	0.0005	0.0051	0.0000	0.0000	0.0001	0.0000
27	0.0000	0.0000	…	0.7846	0.0000	0.0003	0.0000	0.0000	0.0000
28	0.0000	0.0001	…	0.0000	0.0000	0.0000	0.0028	0.0000	0.0004
29	0.0001	0.0247	…	0.0247	0.0002	0.0001	0.0009	0.0002	0.0008
30	0.0000	0.0001	…	0.0001	0.0000	0.8714	0.0001	0.0002	0.0001
31	0.0000	0.0006	…	0.0007	0.0004	0.0019	0.0002	0.0000	0.0001
32	0.0000	0.0000	…	0.0000	0.0000	0.0003	0.9042	0.0000	0.9293

	Attr60	Attr61	Attr63	Attr64
0	0.0000	0.0000	0.0000	0.0000
1	0.0000	0.0047	0.0410	0.0044
2	0.0000	0.0031	0.0006	0.0016
3	0.0004	0.0001	0.0015	0.0001
4	0.0000	0.0016	0.0137	0.0093
5	0.0000	0.0010	0.0021	0.0013
6	0.0005	0.0023	0.0042	0.1716
7	0.1766	0.0001	0.0022	0.0014
8	0.0079	0.0031	0.0450	0.0034
9	0.0000	0.0006	0.0017	0.0002
10	0.0000	0.0001	0.0000	0.0000
11	0.0000	0.0001	0.0001	0.0000
12	0.0000	0.1801	0.0361	0.0010
13	0.0001	0.0013	0.0069	0.0002
14	0.0000	0.0106	0.0000	0.0000
15	0.0000	0.0099	0.0010	0.0000
16	0.0000	0.0545	0.0004	0.0000
17	0.0000	0.2828	0.0030	0.0001
18	0.0000	0.0009	0.0007	0.0048
19	0.0000	0.1472	0.0022	0.0004
20	0.0000	0.0498	0.0000	0.0012
21	0.0000	0.2222	0.0001	0.0007
22	0.0000	0.0057	0.0009	0.0000
23	0.0000	0.0080	0.0068	0.0001
24	0.0000	0.0005	0.0003	0.0015
25	0.0000	0.0000	0.0001	0.0000

```
26  0.0020  0.0044  0.1055  0.0003
27  0.0001  0.0017  0.0106  0.7596
28  0.8084  0.0000  0.0009  0.0000
29  0.0001  0.0008  0.2540  0.0353
30  0.0000  0.0000  0.0015  0.0001
31  0.0002  0.0030  0.4548  0.0015
32  0.0036  0.0000  0.0018  0.0000

[33 rows x 33 columns]
```

Condition Index for MCIE Imputed DataSet

[249]:
```
pd.DataFrame(mice_decomp['cond.index']) #Var Decomposition After Removal of␣
 ↪Variables for MICE Imputed DataSet
```

[249]:
```
    cond.index
0      1.00000
1      1.14160
2      1.18296
3      1.18935
4      1.22096
5      1.30265
6      1.35268
7      1.36501
8      1.38631
9      1.53160
10     1.60201
11     1.66901
12     1.67689
13     1.68037
14     1.68590
15     1.69159
16     1.71907
17     1.73288
18     1.74928
19     1.76960
20     1.79735
21     1.90007
22     2.24470
23     2.45304
24     2.67330
25     2.72809
26     2.83603
27     2.98147
28     3.58225
29     4.03229
```

10 Logisitc Regression with PCA

11 For Mean Model

[33]:
```
from sklearn.decomposition import PCA
# Make an instance of the Model
pca_mean = PCA(.95)
```

[230]:
```
pca_mean.fit(mean_imputed_df_balanced)
x_trainsmote = pca_mean.transform(mean_imputed_df_balanced)
x_test= pca_mean.transform(mean_imputed_df_balanced_test)
```

[231]:
```
lr_pca = sm.Logit(y_balanced3,x_trainsmote )
resultpca = lr_pca.fit_regularized()
resultpca.summary2()
```

```
P:\Anaconda\lib\site-packages\statsmodels\discrete\discrete_model.py:1799:
RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-X))
```

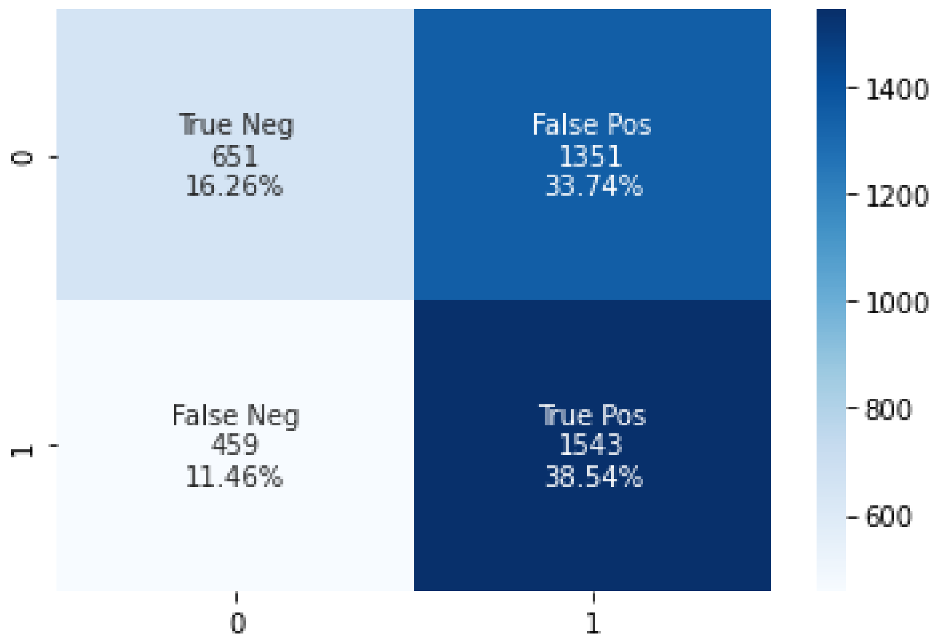
```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6471365348403904
            Iterations: 212
            Function evaluations: 213
            Gradient evaluations: 212
```

[231]:
```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                          Results: Logit
==================================================================
Model:              Logit            Pseudo R-squared: 0.066
Dependent Variable: class            AIC:              20771.9004
Date:               2021-04-17 00:35 BIC:              20956.2466
No. Observations:   16012            Log-Likelihood:   -10362.
Df Model:           23               LL-Null:          -11099.
Df Residuals:       15988            LLR p-value:      1.2103e-297
Converged:          1.0000           Scale:            1.0000
No. Iterations:     212.0000
--------------------------------------------------------------------
         Coef.     Std.Err.      z       P>|z|     [0.025     0.975]
--------------------------------------------------------------------
x1       1.0591     0.1020    10.3873   0.0000     0.8592     1.2589
x2      -0.8338     0.1165    -7.1563   0.0000    -1.0621    -0.6054
x3      -0.2565     0.0613    -4.1847   0.0000    -0.3766    -0.1363
x4      -0.3621     0.1036    -3.4935   0.0005    -0.5652    -0.1589
x5      -0.0723     0.0305    -2.3687   0.0179    -0.1321    -0.0125
x6      -0.5102     0.1351    -3.7759   0.0002    -0.7750    -0.2454
x7      -3.4491     1.4472    -2.3833   0.0172    -6.2856    -0.6127
```

```
x8          1.5500      0.6165      2.5143    0.0119      0.3417     2.7583
x9         -0.0992      0.0293     -3.3886    0.0007     -0.1566    -0.0418
x10        -0.5815      0.1573     -3.6973    0.0002     -0.8897    -0.2732
x11        -0.0533      0.0324     -1.6439    0.1002     -0.1168     0.0102
x12        -0.4534      0.2988     -1.5174    0.1292     -1.0390     0.1322
x13        -0.1963      0.1745     -1.1245    0.2608     -0.5383     0.1458
x14        -0.2539      0.3777     -0.6721    0.5015     -0.9942     0.4865
x15         0.0937      0.0823      1.1384    0.2550     -0.0676     0.2550
x16        -0.3700      0.1475     -2.5094    0.0121     -0.6591    -0.0810
x17         0.2312      0.8336      0.2774    0.7815     -1.4026     1.8651
x18        -0.1858      0.2065     -0.8996    0.3683     -0.5904     0.2189
x19       -15.8830      3.7944     -4.1859    0.0000    -23.3200    -8.4461
x20       -10.8572      3.0456     -3.5649    0.0004    -16.8264    -4.8880
x21         0.3467      0.1765      1.9643    0.0495      0.0008     0.6926
x22        -0.0861      0.0854     -1.0082    0.3133     -0.2533     0.0812
x23         0.0212      0.0986      0.2145    0.8301     -0.1721     0.2145
x24        -2.2338      0.1730    -12.9141    0.0000     -2.5728    -1.8948
==================================================================

"""
```

[266]:
```
y_pred_pca = resultpca.predict(x_test)
k1=[]
for val in y_pred_pca:
    if(val>=0.5):
        k1.append(1)
    else :
        k1.append(0)

recall1_pca=recall_score(y_balanced3_test,k1)
f1_pca=f1_score(y_balanced3_test,k1)
print(classification_report(y_balanced3_test,k1))
print("F1 Score is",f1_score(y_balanced3_test,k1))
cf_mean_pca=confusion_matrix(y_balanced3_test,k1)
conf_plot(cf_mean_pca)
```

```
              precision    recall  f1-score   support

         0.0       0.59      0.33      0.42      2002
         1.0       0.53      0.77      0.63      2002

    accuracy                           0.55      4004
   macro avg       0.56      0.55      0.52      4004
weighted avg       0.56      0.55      0.52      4004

F1 Score is 0.6303104575163399
```



12 For KNN Model

[239]:
```
pca_knn = PCA(.95)
pca_knn.fit(knn_imputed_df_balanced)
x_trainsmote3 = pca_knn.transform(knn_imputed_df_balanced)
x_test3= pca_knn.transform(knn_imputed_df_balanced_test)
```

[240]:
```
lr_pca1 = sm.Logit(y_balanced2,x_trainsmote3 )
resultpca1 = lr_pca1.fit_regularized()
resultpca1.summary2()
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6452490765128714
            Iterations: 203
            Function evaluations: 209
            Gradient evaluations: 203
```

[240]:
```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                          Results: Logit
==================================================================
Model:              Logit            Pseudo R-squared: 0.069
Dependent Variable: class            AIC:              20711.4564
Date:               2021-04-17 00:43 BIC:              20895.8027
No. Observations:   16012            Log-Likelihood:   -10332.
```

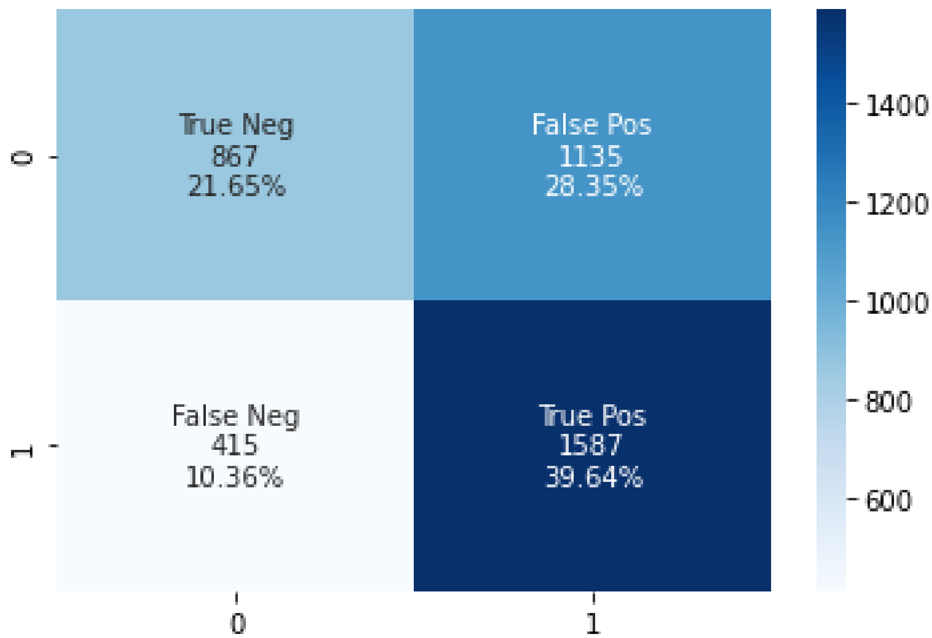
```
Df Model:              23               LL-Null:             -11099.
Df Residuals:          15988            LLR p-value:         1.3827e-310
Converged:             1.0000           Scale:               1.0000
No. Iterations:        203.0000
--------------------------------------------------------------------
           Coef.    Std.Err.      z       P>|z|     [0.025    0.975]
--------------------------------------------------------------------
x1         1.0019     0.0950    10.5461   0.0000    0.8157    1.1881
x2        -0.7332     0.1145    -6.4025   0.0000   -0.9576   -0.5087
x3        -0.2679     0.0394    -6.8023   0.0000   -0.3450   -0.1907
x4        -0.3323     0.1028    -3.2337   0.0012   -0.5337   -0.1309
x5        -0.1081     0.0285    -3.7897   0.0002   -0.1639   -0.0522
x6        -0.7966     0.1261    -6.3191   0.0000   -1.0437   -0.5495
x7        -5.7153     1.0861    -5.2620   0.0000   -7.8441   -3.5865
x8         0.9747     0.1600     6.0911   0.0000    0.6611    1.2883
x9        -0.0853     0.0238    -3.5862   0.0003   -0.1319   -0.0387
x10       -0.4330     0.1486    -2.9143   0.0036   -0.7241   -0.1418
x11       -0.0671     0.0406    -1.6530   0.0983   -0.1467    0.0125
x12       -0.4921     0.3092    -1.5915   0.1115   -1.0981    0.1139
x13       -0.3038     0.2019    -1.5049   0.1324   -0.6994    0.0919
x14        0.2173     0.3607     0.6025   0.5468   -0.4897    0.9244
x15       -0.1666     0.1343    -1.2403   0.2149   -0.4298    0.0967
x16        0.1084     0.0882     1.2289   0.2191   -0.0645    0.2814
x17       -3.0771     0.8790    -3.5005   0.0005   -4.8000   -1.3542
x18       -0.7635     0.2160    -3.5341   0.0004   -1.1869   -0.3401
x19       -5.0418     1.2950    -3.8933   0.0001   -7.5799   -2.5037
x20        0.6120     1.7531     0.3491   0.7270   -2.8241    4.0481
x21        1.5259     0.2579     5.9167   0.0000    1.0204    2.0314
x22       -0.5299     0.2758    -1.9213   0.0547   -1.0705    0.0107
x23        0.2498     0.2199     1.1358   0.2560   -0.1812    0.6808
x24       -2.1298     0.1684   -12.6448   0.0000   -2.4600   -1.7997
==================================================================

"""
```

[265]:

```
y_pred_pca1 = resultpca1.predict(x_test3)
k2=[]
for val in y_pred_pca1:
    if(val>=0.5):
        k2.append(1)
    else :
        k2.append(0)

recall2_pca=recall_score(y_balanced2_test,k2)
f2_pca=f1_score(y_balanced2_test,k2)
print(classification_report(y_balanced2_test,k2))
print("F1 Score is",f1_score(y_balanced2_test,k2))
```

```
cf_knn_pca=confusion_matrix(y_balanced2_test,k2)
conf_plot(cf_knn_pca)
```

```
              precision    recall  f1-score   support

         0.0       0.68      0.43      0.53      2002
         1.0       0.58      0.79      0.67      2002

    accuracy                           0.61      4004
   macro avg       0.63      0.61      0.60      4004
weighted avg       0.63      0.61      0.60      4004

F1 Score is 0.6718882303132937
```



13 For MICE Model

[39]:
```
import statsmodels.api as sm
pca_mice = PCA(.99)
pca_mice.fit(mice_imputed_df_balanced)
x_trainsmote1 = pca_mice.transform(mice_imputed_df_balanced)
x_test1= pca_mice.transform(mice_imputed_df_balanced_test)
```

[40]:
```
lr_pca2 = sm.Logit(y_balanced1,x_trainsmote1)
resultpca2 = lr_pca2.fit_regularized()
resultpca2.summary2()
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6611207495474264
            Iterations: 76
            Function evaluations: 80
            Gradient evaluations: 76

P:\Anaconda\lib\site-packages\statsmodels\discrete\discrete_model.py:1799:
RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-X))
```

[40]:

```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                          Results: Logit
==================================================================
Model:              Logit            Pseudo R-squared: 0.046
Dependent Variable: class            AIC:              21197.7309
Date:               2021-04-17 15:35 BIC:              21297.5851
No. Observations:   16012            Log-Likelihood:   -10586.
Df Model:           12               LL-Null:          -11099.
Df Residuals:       15999            LLR p-value:      5.8276e-212
Converged:          1.0000           Scale:            1.0000
No. Iterations:     76.0000
--------------------------------------------------------------------
           Coef.    Std.Err.      z       P>|z|     [0.025    0.975]
--------------------------------------------------------------------
x1        -0.0198    0.0214    -0.9248   0.3551   -0.0618    0.0222
x2         0.6914    0.0822     8.4162   0.0000    0.5304    0.8525
x3        -3.5283    0.4830    -7.3044   0.0000   -4.4751   -2.5816
x4        -0.4083    0.0255   -15.9934   0.0000   -0.4583   -0.3582
x5        -0.6881    0.0432   -15.9472   0.0000   -0.7727   -0.6036
x6        -0.0027    0.0390    -0.0690   0.9450   -0.0791    0.0737
x7        -0.0060    0.0322    -0.1859   0.8525   -0.0690    0.0571
x8        -0.1369    0.0381    -3.5926   0.0003   -0.2115   -0.0622
x9        -0.0088    0.0171    -0.5146   0.6068   -0.0422    0.0247
x10        0.0150    0.0292     0.5141   0.6072   -0.0422    0.0722
x11       -0.0621    0.0700    -0.8872   0.3750   -0.1994    0.0751
x12        0.1449    0.0357     4.0638   0.0000    0.0750    0.2148
x13        0.0169    0.0694     0.2434   0.8077   -0.1191    0.1529
==================================================================

"""
```

[41]:

```
y_pred_pca2 = resultpca2.predict(x_test1)
k3=[]
for val in y_pred_pca2:
    if(val>=0.5):
        k3.append(1)
    else :
```

```
        k3.append(0)
print(classification_report(y_balanced1_test,k3))
recall3_pca=recall_score(y_balanced1_test,k3)
f3_pca=f1_score(y_balanced1_test,k3)
print("F1 Score is",f1_score(y_balanced1_test,k3))

cf_knn_mice=confusion_matrix(y_balanced1_test,k3)
conf_plot(cf_knn_mice)
```

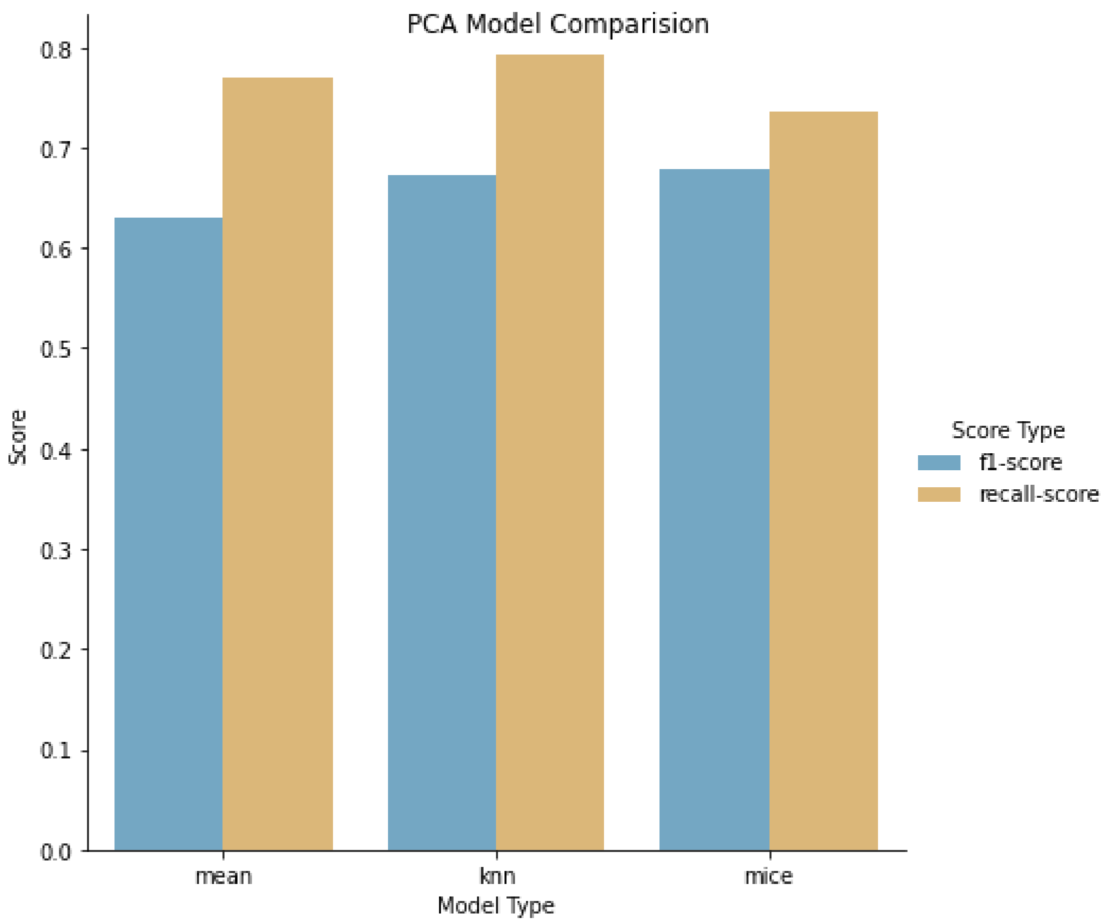
```
              precision    recall  f1-score   support

           0       0.68      0.56      0.62      2002
           1       0.63      0.74      0.68      2002

    accuracy                           0.65      4004
   macro avg       0.66      0.65      0.65      4004
weighted avg       0.66      0.65      0.65      4004
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-41-35fb338fd34e> in <module>
      8 print(classification_report(y_balanced1_test,k3))
      9 recall3_pca=recall_score(y_balanced1_test,k3)
---> 10 f3_pca=f1_score(y_balanced1_test,k3)
     11 print("F1 Score is",f1_score(y_balanced1_test,k3))
     12

NameError: name 'f1_score' is not defined
```

[285]:
```
def barchart(f1,f2,f3,recall1,recall2,recall3,title):
    data={'f1-score':[f1,f2,f3],'Model Type':
 ↪['mean','knn','mice'],'recall-score':[recall1,recall2,recall3]}
    s=pd.DataFrame(data)
    s = s.melt(id_vars=['Model Type'], var_name='recall-score',␣
 ↪value_name='f1-score')


    g = sns.catplot(
        data=s, kind="bar",
        x="Model Type", y="f1-score", hue="recall-score",
        ci="sd", palette='colorblind', alpha=.6, height=6,legend_out=True
    )
    g.set_axis_labels("Model Type", "Score")
    g._legend.set_title("Score Type")
    g.fig.suptitle(title)
```

```
barchart(f1_pca,f2_pca,f3_pca,recall1_pca,recall2_pca,recall3_pca,'PCA Model␣
↪Comparision')
```

```
<ipython-input-285-0f97beaf24d4>:4: FutureWarning: This dataframe has a column
name that matches the 'value_name' column name of the resultiing Dataframe. In
the future this will raise an error, please set the 'value_name' parameter of
DataFrame.melt to a unique name.
  s = s.melt(id_vars=['Model Type'], var_name='recall-score',
value_name='f1-score')
```



14 Variable Selection- Stepwise Regression

[188]:
```
m=[]
for val in y_pred:
    if(val>=0.5):
        m.append(1)
```

```
    else :
        m.append(0)
```

[71]:
```
#Variable Selection using Stepwise Regression on basis of AIC Criteria in R
variables_mean=['Attr24', 'Attr41', 'Attr63', 'Attr34', 'Attr48', 'Attr36',
'Attr58', 'Attr61', 'Attr9', 'Attr55', 'Attr52', 'Attr32', 'Attr60', 'Attr29',
'Attr44', 'Attr64', 'Attr40', 'Attr5', 'Attr37', 'Attr57', 'Attr59']

variables_knn=['Attr24', 'Attr63', 'Attr34', 'Attr52', 'Attr58', 'Attr55',
'Attr41', 'Attr48', 'Attr36', 'Attr9', 'Attr61', 'Attr60', 'Attr27', 'Attr37',
'Attr44','Attr40', 'Attr32', 'Attr64', 'Attr29', 'Attr5', 'Attr21', 'Attr8',
'Attr57', 'Attr59']

variables_mice=['Attr24' , 'Attr41' , 'Attr48' , 'Attr64' , 'Attr34' ,'Attr36' ,
    'Attr61' , 'Attr58' , 'Attr9' , 'Attr55' , 'Attr52' , 'Attr29' ,'Attr31' ,
    'Attr60' , 'Attr45' , 'Attr13' , 'Attr8' , 'Attr5']
```

[72]:
```
Z1 = add_constant(Z1[variables_mean])
y1 = np.array(y_balanced3)

Z2 = add_constant(Z2[variables_knn])
y2 = np.array(y_balanced2)

Z3 = add_constant(Z3[variables_mice])
y3 = np.array(y_balanced1)
```

[83]:
```
from scipy.stats import norm
def logit_pvalue(model, x):
    """ Calculate z-scores for scikit-learn LogisticRegression.
    parameters:
        model: fitted sklearn.linear_model.LogisticRegression with intercept␣
↪and large C
        x:     matrix on which the model was fit
    This function uses asymtptics for maximum likelihood estimates.
    """
    p = model.predict_proba(np.array(x))
    n = len(p)
    m = len(model.coef_[0]) + 1
    coefs = np.concatenate([model.intercept_, model.coef_[0]])
    x_full = np.matrix(np.insert(np.array(x), 0, 1, axis = 1))
    ans = np.zeros((m, m))
    for i in range(n):
        ans = ans + np.dot(np.transpose(x_full[i, :]), x_full[i, :]) * p[i,1] *␣
↪p[i, 0]
    vcov = np.linalg.inv(np.matrix(ans))
    se = np.sqrt(np.diag(vcov))
    t =  coefs/se
```

```
    p = (1 - norm.cdf(abs(t))) * 2

    return coefs,se,t,p
```

[109]:
```
def nice_output(model,X):
    coefs,se,t,p=logit_pvalue(model,X)
    output_df=pd.DataFrame()
    Y=add_constant(X)
    output_df['Variables']=list(Y.columns)
    output_df['Coefficients']=coefs
    output_df['Standard Error']=se
    output_df['z']=t
    output_df['P>|z|']=p

    return output_df
```

[162]:
```
import sklearn.metrics as metrics
def llr_full(X, y, model):
    llr_full= -1*metrics.log_loss(y, model.predict_proba(X), normalize=False)
    return llr_full

def llr_null(X, y,c):
    lr = LogisticRegression(C=c,max_iter=5000)
    model=lr.fit(np.array(X['const']).reshape(-1, 1),y)
    llr_null= -1*metrics.log_loss(y, model.predict_proba(np.array(X['const']).
 ↪reshape(-1, 1)), normalize=False)
    return llr_null
```

15 Logisitic Regression Mean Imputation Model after Stepwise Regression

[118]:
```
import statsmodels.api as sm
```

[204]:
```
Z1.iloc[:,1:]
```

[204]:
```
         Attr24    Attr41    Attr63    Attr34    Attr48    Attr36    Attr58  \
0     -0.019507 -0.011140 -0.169389 -0.020652  0.070976  0.440434 -0.012260
1     -0.474792 -0.011153 -0.261286 -0.065054 -0.279960 -0.082029 -0.015106
2     -0.077651 -0.010612 -0.245782 -0.050789 -0.011865 -0.170687 -0.012260
3     -0.053004 -0.011175 -0.238080 -0.048162  0.019158 -0.455736 -0.012396
4     -0.149437 -0.013764 -0.245742 -0.048843 -0.071690 -0.075936 -0.012113
…           …         …         …         …         …         …         …
16007 -0.028509 -0.011196 -0.146515 -0.068477  0.124022 -0.004984 -0.012503
16008 -0.052534 -0.012859 -0.129295 -0.003393 -0.002130  0.628315 -0.012209
16009 -0.052082 -0.011222  0.255547 -0.058746  0.072154 -0.191521 -0.012496
```

```
16010 -0.080037 -0.011146 -0.243126 -0.062467 -0.017016 -0.513252 -0.012659
16011  0.076830 -0.011961 -0.059803  0.022236 -0.023745  1.045285 -0.012490

         Attr61     Attr9    Attr55  …    Attr32    Attr60    Attr29  \
0      0.320285  0.162110 -0.107653  … -0.016962 -0.015328  0.098249
1     -0.106095 -0.008267 -0.110570  … -0.008899 -0.015352 -2.266791
2     -0.128932 -0.037178 -0.080205  … -0.015390 -0.015355  0.461760
3     -0.099410 -0.130134 -0.095742  … -0.015674 -0.015389  0.035529
4     -0.115056 -0.006280 -0.119587  … -0.015454 -0.015350 -0.147364
…           …         …         …  …         …         …         …
16007 -0.111412 -0.089953  0.340355  … -0.016722 -0.014969  1.242899
16008 -0.099774  0.212112 -0.063341  … -0.017169 -0.014216  0.030683
16009 -0.102502 -0.089869  0.259217  … -0.017859 -0.015300  0.752891
16010 -0.106162 -0.156423 -0.060733  … -0.015454 -0.014875  0.809920
16011  0.358438  0.346577 -0.122320  … -0.017472 -0.015113 -0.314912

         Attr44        Attr64    Attr40     Attr5        Attr37    Attr57  \
0     -0.041474 -4.676648e-02 -0.056661  0.012771  6.520852e-18  0.013771
1     -0.018853  3.887468e-18 -0.055793  0.008621  6.520852e-18  0.019472
2     -0.000058 -1.450816e-02 -0.058974  0.012753 -3.863726e-02  0.014524
3     -0.021669 -7.958948e-02 -0.057921  0.012430  6.520852e-18  0.001210
4     -0.013760 -2.976098e-02 -0.058110  0.012500  6.707617e-02  0.044461
…           …             …         …         …             …         …
16007 -0.001934 -5.895125e-02 -0.057383  0.013551 -3.239991e-02  0.020623
16008 -0.021073  4.009669e-01 -0.057381  0.013306  6.520852e-18  0.001033
16009 -0.019678 -7.266913e-02 -0.020540  0.014026  5.391318e-01  0.003740
16010 -0.018364 -8.076855e-02 -0.041306  0.013537 -4.345052e-02  0.000597
16011 -0.041555 -6.190899e-02 -0.058043  0.013032 -4.019021e-02 -0.009528

         Attr59
0     -0.019237
1     -0.019237
2     -0.010607
3     -0.019237
4     -0.020600
…           …
16007 -0.016947
16008 -0.019237
16009 -0.019236
16010 -0.009203
16011 -0.018557

[16012 rows x 21 columns]
```

[388]:
```
logistic_regression_mean = sm.Logit(y1, Z1)
result11_regularized = logistic_regression_mean.
  ↪fit_regularized(alpha=24,method='l1')
```

```python
result11 = logistic_regression_mean.fit_regularized()
result11.summary2()
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6593798151342252
            Iterations: 121
            Function evaluations: 121
            Gradient evaluations: 121
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6432456839343315
            Iterations: 206
            Function evaluations: 214
            Gradient evaluations: 206
```

```
P:\Anaconda\lib\site-packages\statsmodels\discrete\discrete_model.py:1799:
RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-X))
```

[388]:
```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                         Results: Logit
=================================================================
Model:              Logit            Pseudo R-squared: 0.072
Dependent Variable: y                AIC:              20643.2998
Date:               2021-04-17 03:04 BIC:              20812.2838
No. Observations:   16012            Log-Likelihood:   -10300.
Df Model:           21               LL-Null:          -11099.
Df Residuals:       15990            LLR p-value:      0.0000
Converged:          1.0000           Scale:            1.0000
No. Iterations:     206.0000
-----------------------------------------------------------------
             Coef.   Std.Err.    z     P>|z|    [0.025    0.975]
-----------------------------------------------------------------
const        -2.2973   0.2001 -11.4818 0.0000   -2.6894   -1.9051
Attr24       -3.4387   0.1468 -23.4254 0.0000   -3.7264   -3.1510
Attr41     -139.4685  15.2751  -9.1305 0.0000 -169.4072 -109.5299
Attr63       -0.5804   0.0544 -10.6679 0.0000   -0.6870   -0.4737
Attr34        0.6136   0.0509  12.0614 0.0000    0.5139    0.7134
Attr48        0.1731   0.0572   3.0279 0.0025    0.0610    0.2851
Attr36        0.5027   0.0633   7.9475 0.0000    0.3787    0.6267
Attr58       -3.6438   0.8236  -4.4244 0.0000   -5.2579   -2.0296
Attr61       -0.4093   0.0944  -4.3350 0.0000   -0.5944   -0.2243
Attr9        -1.1524   0.2095  -5.5015 0.0000   -1.5629   -0.7418
Attr55       -0.1543   0.0399  -3.8710 0.0001   -0.2324   -0.0762
Attr52      -37.6252   9.7760  -3.8487 0.0001  -56.7859  -18.4646
Attr32        7.0552   2.1217   3.3252 0.0009    2.8967   11.2137
Attr60       -2.8285   1.3831  -2.0450 0.0409   -5.5394   -0.1176
Attr29       -0.0567   0.0203  -2.7875 0.0053   -0.0965   -0.0168
```

```
Attr44        -0.2093   0.1032  -2.0285 0.0425   -0.4115   -0.0071
Attr64        -0.0615   0.0330  -1.8624 0.0625   -0.1262    0.0032
Attr40         0.0820   0.0390   2.0993 0.0358    0.0054    0.1585
Attr5          0.1686   0.1621   1.0397 0.2985   -0.1492    0.4863
Attr37        -0.1978   0.1364  -1.4498 0.1471   -0.4652    0.0696
Attr57        -0.5587   0.1371  -4.0736 0.0000   -0.8275   -0.2899
Attr59        -0.5544   0.1391  -3.9872 0.0001   -0.8270   -0.2819
=================================================================

"""
```

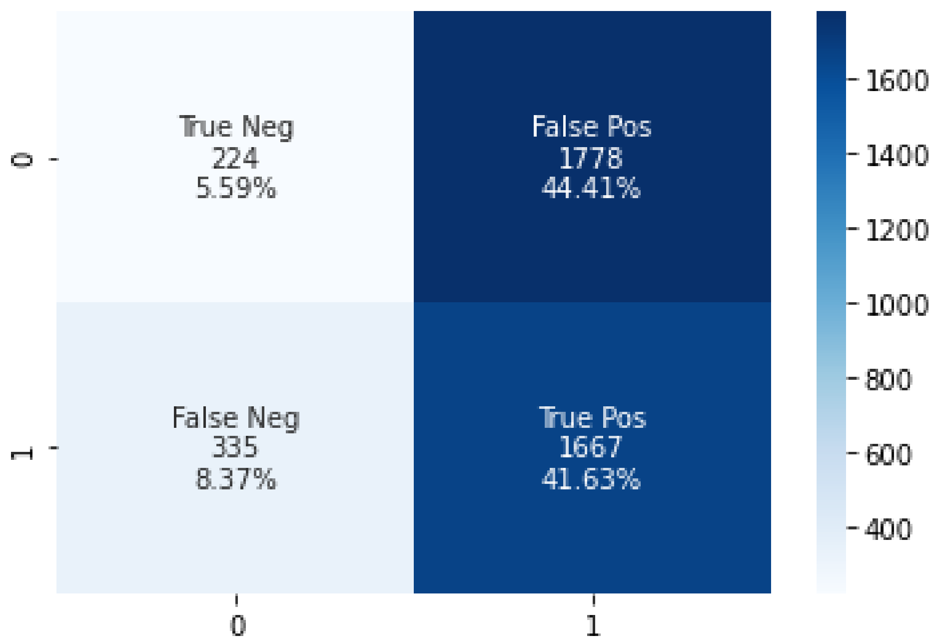
[216]:

```
#Using Sciket Learn
lr_mean = LogisticRegression(max_iter=5000)
result12 = lr_mean.fit(np.array(Z1.iloc[:,1:]),y1)
y_pred12=result12.predict(mean_imputed_df_balanced_test[variables_mean])
```

[390]:

```
#Using Statsmodels
#Without Regularization
y_pred11 = result11.
↪predict(add_constant(mean_imputed_df_balanced_test[variables_mean]))
m1=[]
for val in y_pred11:
    if(val>=0.5):
        m1.append(1)
    else :
        m1.append(0)
print(classification_report(y_balanced3_test,m1))
print("F1 Score is",f1_score(y_balanced3_test,m1))
cf_mean=confusion_matrix(y_balanced3_test,m1)

recall1_lr=recall_score(y_balanced3_test,m1)
f1_lr=f1_score(y_balanced3_test,m1)

def conf_plot(cf_matrix):
  group_names =['True Neg', 'False Pos' , 'False Neg', 'True Pos']

  group_counts = ["{0:0.0f}".format(value) for value in
                cf_matrix.flatten()]
  group_percentages = ["{0:.2%}".format(value) for value in
                     cf_matrix.flatten()/np.sum(cf_matrix)]
  labels = [f"{v1}\n{v2}\n{v3}" for v1, v2, v3 in
          zip(group_names,group_counts,group_percentages)]
  labels = np.asarray(labels).reshape(2,2)
  sns.heatmap(cf_matrix, annot=labels, fmt='', cmap='Blues')

conf_plot(cf_mean)
```

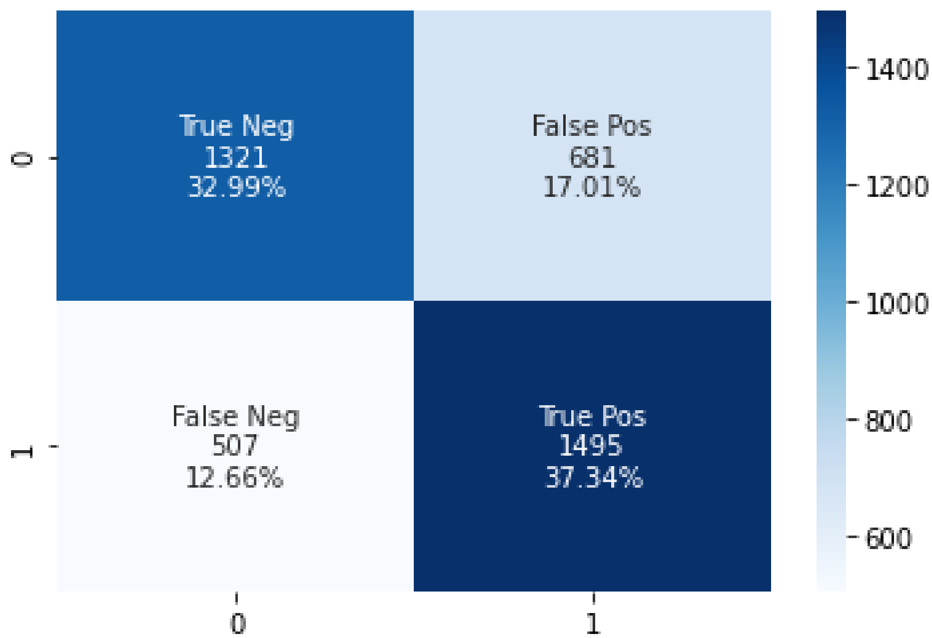
```
P:\Anaconda\lib\site-packages\statsmodels\discrete\discrete_model.py:1799:
RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-X))
              precision    recall  f1-score   support

         0.0       0.40      0.11      0.17      2002
         1.0       0.48      0.83      0.61      2002

    accuracy                           0.47      4004
   macro avg       0.44      0.47      0.39      4004
weighted avg       0.44      0.47      0.39      4004

F1 Score is 0.612080044060951
```



[389]:
```
#With L1 Regularization
y_pred11_reg = result11_regularized.
 ↪predict(add_constant(mean_imputed_df_balanced_test[variables_mean]))
m1=[]
for val in y_pred11_reg:
    if(val>=0.5):
        m1.append(1)
    else :
        m1.append(0)
```

```
print(classification_report(y_balanced3_test,m1))
print("F1 Score is",f1_score(y_balanced3_test,m1))
cf_mean1=confusion_matrix(y_balanced3_test,m1)

recall1_lr1=recall_score(y_balanced3_test,m1)
f1_lr1=f1_score(y_balanced3_test,m1)
conf_plot(cf_mean1)
```

```
              precision    recall  f1-score   support

         0.0       0.72      0.66      0.69      2002
         1.0       0.69      0.75      0.72      2002

    accuracy                           0.70      4004
   macro avg       0.70      0.70      0.70      4004
weighted avg       0.70      0.70      0.70      4004

F1 Score is 0.7156534226902824
```



16 Logisitic Regression KNN Imputation Model after Stepwise Regression

[392]:
```
logistic_regression_knn = sm.Logit(y2,Z2)
result2_regularized = logistic_regression_knn.
 ↪fit_regularized(alpha=24,method='l1')
result2=logistic_regression_knn.fit_regularized()
result2.summary2()
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6591577972529281
            Iterations: 140
            Function evaluations: 140
            Gradient evaluations: 140
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.644803019423566
            Iterations: 208
            Function evaluations: 210
            Gradient evaluations: 208
```

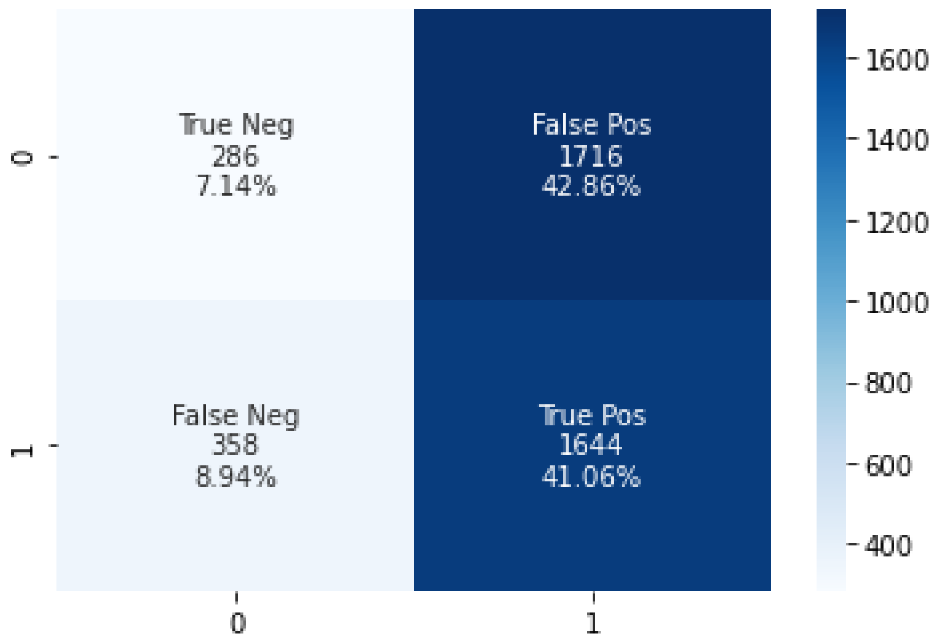
[392]:
```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                         Results: Logit
=================================================================
Model:              Logit            Pseudo R-squared: 0.070
Dependent Variable: y                AIC:              20699.1719
Date:               2021-04-17 03:11 BIC:              20891.1992
No. Observations:   16012            Log-Likelihood:   -10325.
Df Model:           24               LL-Null:          -11099.
Df Residuals:       15987            LLR p-value:      0.0000
Converged:          1.0000           Scale:            1.0000
No. Iterations:     208.0000
------------------------------------------------------------------
           Coef.    Std.Err.     z      P>|z|    [0.025    0.975]
------------------------------------------------------------------
const     -1.1706    0.2034   -5.7560  0.0000   -1.5692   -0.7720
Attr24    -3.3970    0.1525  -22.2707  0.0000   -3.6959   -3.0980
Attr63    -0.5975    0.0586  -10.1964  0.0000   -0.7123   -0.4826
Attr34     0.6364    0.0543   11.7125  0.0000    0.5299    0.7429
Attr52   -25.2945    8.8359   -2.8627  0.0042  -42.6126   -7.9763
Attr58    -7.0902    2.3810   -2.9779  0.0029  -11.7568   -2.4236
Attr55    -0.1206    0.0379   -3.1843  0.0015   -0.1949   -0.0464
Attr41   -43.0223   15.4662   -2.7817  0.0054  -73.3355  -12.7092
Attr48     0.1277    0.0667    1.9153  0.0555   -0.0030    0.2584
Attr36     0.5382    0.0635    8.4801  0.0000    0.4138    0.6626
Attr9     -1.2068    0.2103   -5.7396  0.0000   -1.6189   -0.7947
Attr61    -0.4200    0.0964   -4.3556  0.0000   -0.6089   -0.2310
```

```
Attr60     -1.5129    0.5201   -2.9092  0.0036   -2.5322   -0.4937
Attr27     -0.6900    0.3034   -2.2739  0.0230   -1.2847   -0.0953
Attr37     -0.2602    0.1337   -1.9454  0.0517   -0.5223    0.0019
Attr44     -0.1599    0.1075   -1.4871  0.1370   -0.3706    0.0508
Attr40      0.1592    0.0596    2.6726  0.0075    0.0425    0.2760
Attr32      3.2527    1.8644    1.7446  0.0811   -0.4015    6.9068
Attr64     -0.0623    0.0337   -1.8513  0.0641   -0.1283    0.0037
Attr29     -0.0462    0.0204   -2.2616  0.0237   -0.0863   -0.0062
Attr5       0.1911    0.1748    1.0937  0.2741   -0.1514    0.5337
Attr21     -0.1938    0.2247   -0.8627  0.3883   -0.6341    0.2465
Attr8      -0.5801    0.4739   -1.2240  0.2210   -1.5090    0.3488
Attr57     -0.5305    0.1427   -3.7186  0.0002   -0.8101   -0.2509
Attr59     -0.5292    0.1451   -3.6465  0.0003   -0.8137   -0.2448
=================================================================

"""
```

[402]:

```
y_pred2 = result2.
 ↪predict(add_constant(knn_imputed_df_balanced_test[variables_knn]))
m2=[]
for val in y_pred2:
    if(val>=0.5):
        m2.append(1)
    else :
        m2.append(0)

recall2_lr=recall_score(y_balanced2_test,m2)
f2_lr=f1_score(y_balanced2_test,m2)
print(classification_report(np.array(y_balanced2_test),m2))
print("F1 Score is",f1_score(y_balanced2_test,m2))
cf_knn=confusion_matrix(y_balanced2_test,m2)
conf_plot(cf_knn)
```

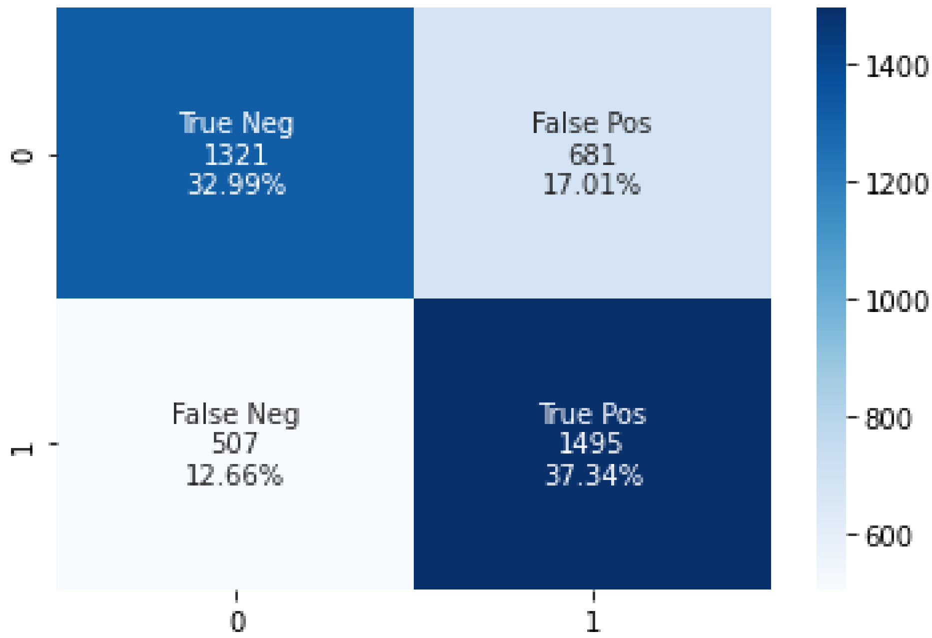
```
P:\Anaconda\lib\site-packages\statsmodels\discrete\discrete_model.py:1799:
RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-X))
              precision    recall  f1-score   support

         0.0       0.44      0.14      0.22      2002
         1.0       0.49      0.82      0.61      2002

    accuracy                           0.48      4004
   macro avg       0.47      0.48      0.41      4004
weighted avg       0.47      0.48      0.41      4004

F1 Score is 0.6132040283476314
```



[394]:
```
#With L1 Regularization
y_pred12_reg = result2_regularized.
 ↪predict(add_constant(knn_imputed_df_balanced_test[variables_knn]))
m2=[]
for val in y_pred11_reg:
    if(val>=0.5):
        m2.append(1)
    else :
        m2.append(0)

recall2_lr1=recall_score(y_balanced2_test,m2)
f2_lr1=f1_score(y_balanced2_test,m2)
print(classification_report(np.array(y_balanced2_test),m2))
print("F1 Score is",f1_score(y_balanced2_test,m2))
cf_knn1=confusion_matrix(y_balanced2_test,m2)
conf_plot(cf_knn1)
```

```
              precision    recall  f1-score   support

         0.0       0.72      0.66      0.69      2002
         1.0       0.69      0.75      0.72      2002

    accuracy                           0.70      4004
   macro avg       0.70      0.70      0.70      4004
weighted avg       0.70      0.70      0.70      4004
```

```
F1 Score is 0.7156534226902824
```



17 Logisitic Regression MICE Imputation Model after Stepwise Regression

[395]:
```
logistic_regression_mice = sm.
 ↪Logit(y3,add_constant(mice_imputed_df_balanced[variables_mice]))
result3 = logistic_regression_mice.fit_regularized()
result3_regularized=logistic_regression_mice.
 ↪fit_regularized(alpha=24,method='l1')
result3.summary2()
```

```
P:\Anaconda\lib\site-packages\statsmodels\discrete\discrete_model.py:1799:
RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-X))
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6477901228395612
            Iterations: 194
            Function evaluations: 197
            Gradient evaluations: 194
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6617449788705215
            Iterations: 105
            Function evaluations: 105
```

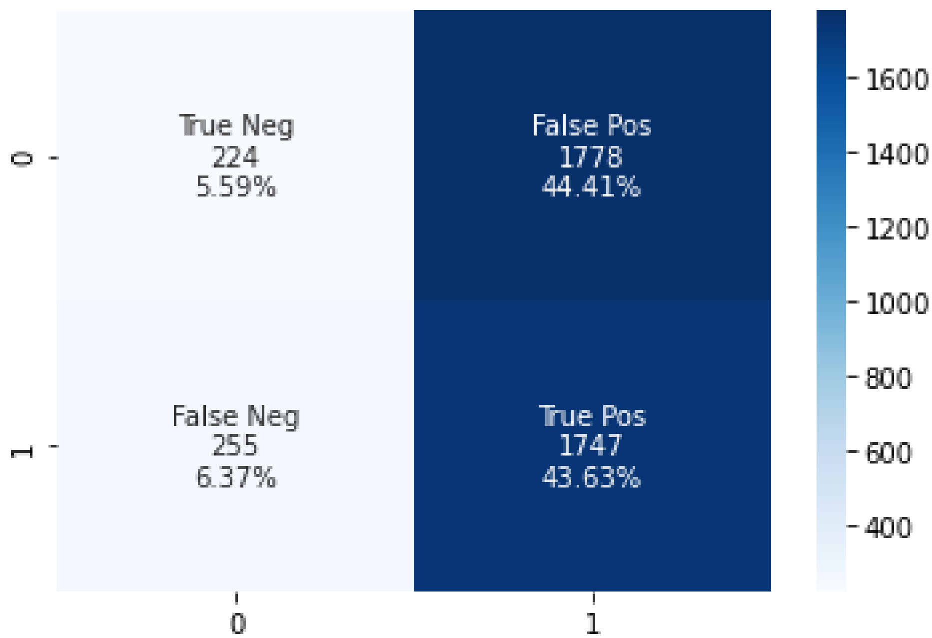
```
            Gradient evaluations: 105
```

[395]:
```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                          Results: Logit
==================================================================
Model:              Logit            Pseudo R-squared: 0.065
Dependent Variable: y                AIC:              20782.8309
Date:               2021-04-17 03:12 BIC:              20928.7717
No. Observations:   16012            Log-Likelihood:   -10372.
Df Model:           18               LL-Null:          -11099.
Df Residuals:       15993            LLR p-value:      7.5598e-298
Converged:          1.0000           Scale:            1.0000
No. Iterations:     194.0000
-------------------------------------------------------------------
          Coef.    Std.Err.    z       P>|z|     [0.025     0.975]
-------------------------------------------------------------------
const     -2.3372   0.2097  -11.1452  0.0000    -2.7482    -1.9262
Attr24    -3.6171   0.1453  -24.9002  0.0000    -3.9018    -3.3324
Attr41  -139.2764  15.1159   -9.2139  0.0000  -168.9031  -109.6497
Attr48     0.1874   0.0552    3.3922  0.0007     0.0791     0.2957
Attr64    -0.0668   0.0342   -1.9530  0.0508    -0.1339     0.0002
Attr34     0.1447   0.1214    1.1919  0.2333    -0.0933     0.3827
Attr36     0.4564   0.0644    7.0848  0.0000     0.3301     0.5826
Attr61    -0.4834   0.0919   -5.2575  0.0000    -0.6636    -0.3032
Attr58    -2.7945   1.3268   -2.1061  0.0352    -5.3951    -0.1939
Attr9     -1.0908   0.2087   -5.2253  0.0000    -1.4999    -0.6816
Attr55    -0.1879   0.0414   -4.5438  0.0000    -0.2690    -0.1069
Attr52    -5.7795   4.5493   -1.2704  0.2039   -14.6959     3.1368
Attr29    -0.0362   0.0197   -1.8367  0.0662    -0.0749     0.0024
Attr31     8.8137   2.1397    4.1192  0.0000     4.6201    13.0074
Attr60    -0.4055   0.3021   -1.3423  0.1795    -0.9975     0.1866
Attr45     0.0134   0.0488    0.2748  0.7834    -0.0823     0.1091
Attr13   -39.3023   9.6469   -4.0741  0.0000   -58.2098   -20.3947
Attr8     -0.7944   0.4045   -1.9639  0.0495    -1.5872    -0.0016
Attr5      0.1527   0.1553    0.9836  0.3253    -0.1516     0.4571
==================================================================

"""
```

[398]:
```
y_pred3 = result3.
 ↪predict(add_constant(mice_imputed_df_balanced_test[variables_mice]))
m3=[]
for val in y_pred3:
    if(val>=0.5):
        m3.append(1)
    else :
```

```
        m3.append(0)

recall3_lr=recall_score(y_balanced1_test,m3)
f3_lr=f1_score(y_balanced1_test,m3)
print(classification_report(np.array(y_balanced1_test),m3))
print("F1 Score is",f1_score(y_balanced1_test,m3))
cf_mice=confusion_matrix(y_balanced1_test,m3)
conf_plot(cf_mice)
```

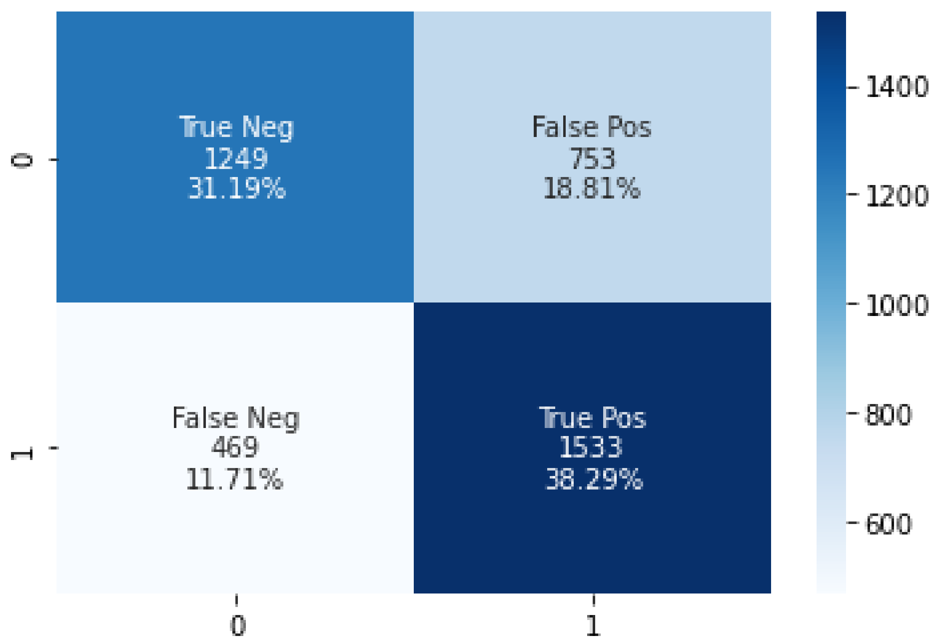
```
P:\Anaconda\lib\site-packages\statsmodels\discrete\discrete_model.py:1799:
RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-X))
              precision    recall  f1-score   support

           0       0.47      0.11      0.18      2002
           1       0.50      0.87      0.63      2002

    accuracy                           0.49      4004
   macro avg       0.48      0.49      0.41      4004
weighted avg       0.48      0.49      0.41      4004

F1 Score is 0.6321693504613715
```



[396]:
```
y_pred3_reg = result3_regularized.
 ↪predict(add_constant(mice_imputed_df_balanced_test[variables_mice]))
```

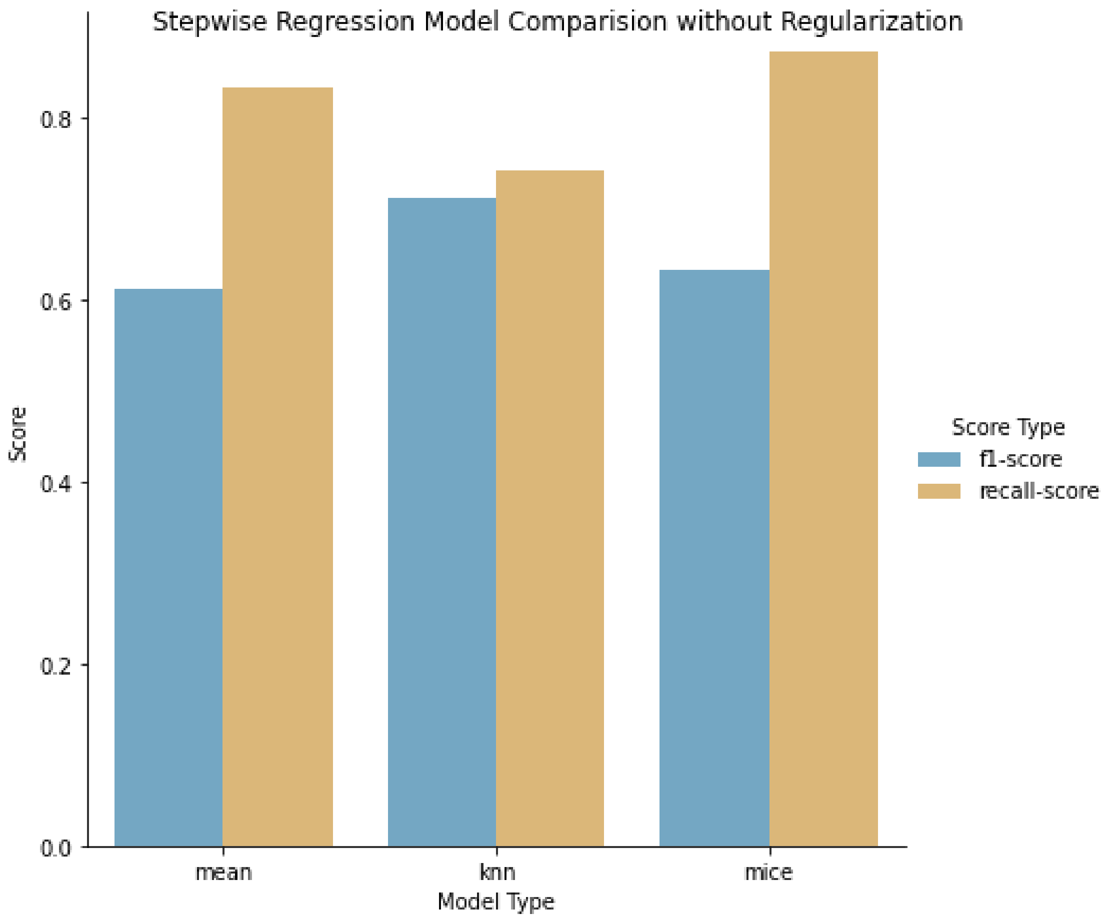
```
m3=[]
for val in y_pred3_reg:
    if(val>=0.5):
        m3.append(1)
    else :
        m3.append(0)

recall3_lr1=recall_score(y_balanced1_test,m3)
f3_lr1=f1_score(y_balanced1_test,m3)
print(classification_report(np.array(y_balanced1_test),m3))
print("F1 Score is",f1_score(y_balanced1_test,m3))
cf_mice1=confusion_matrix(y_balanced1_test,m3)
conf_plot(cf_mice1)
```

```
              precision    recall  f1-score   support

           0       0.73      0.62      0.67      2002
           1       0.67      0.77      0.72      2002

    accuracy                           0.69      4004
   macro avg       0.70      0.69      0.69      4004
weighted avg       0.70      0.69      0.69      4004

F1 Score is 0.715018656716418
```



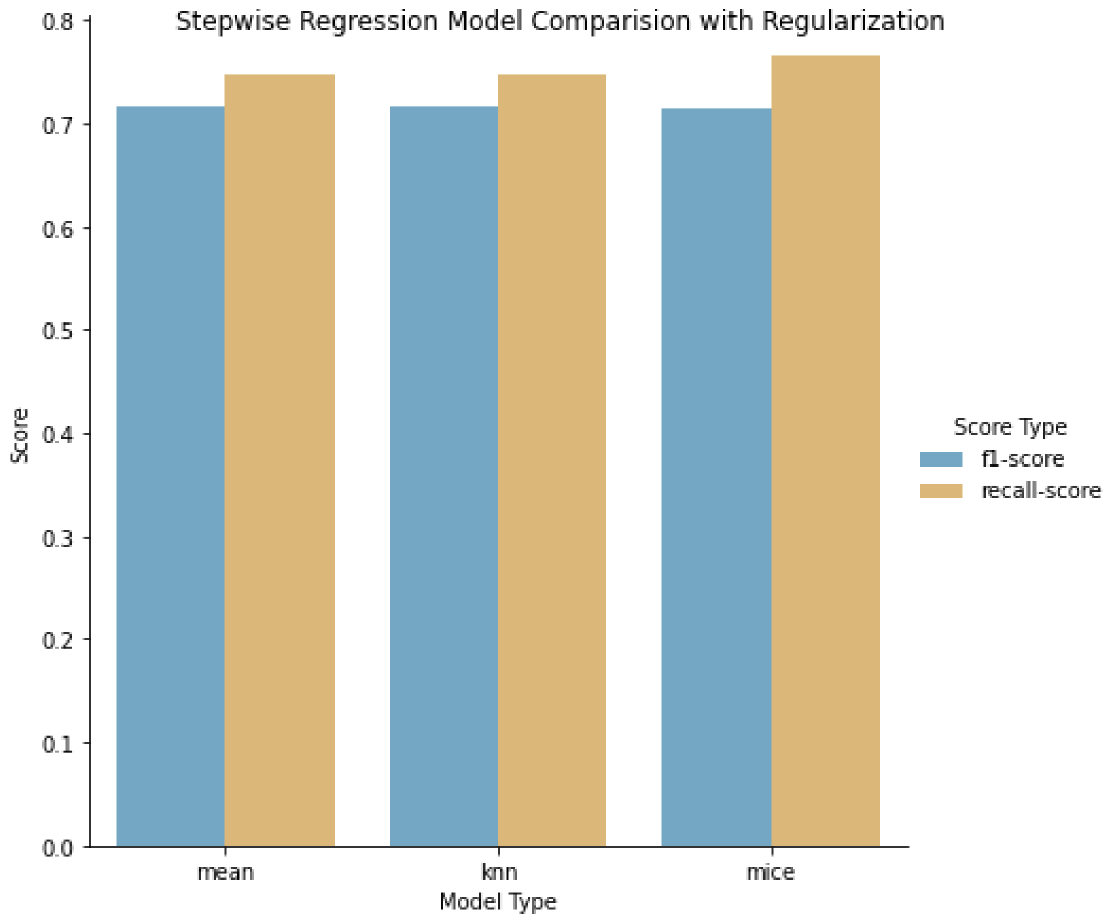
[399]:
```
barchart(f1_lr,f2_lr,f3_lr,recall1_lr,recall2_lr,recall3_lr,'Stepwise␣
↪Regression Model Comparision without Regularization')
```

```
<ipython-input-285-0f97beaf24d4>:4: FutureWarning: This dataframe has a column
name that matches the 'value_name' column name of the resultiing Dataframe. In
the future this will raise an error, please set the 'value_name' parameter of
DataFrame.melt to a unique name.
  s = s.melt(id_vars=['Model Type'], var_name='recall-score',
value_name='f1-score')
```



[400]:
```
barchart(f1_lr1,f2_lr1,f3_lr1,recall1_lr1,recall2_lr1,recall3_lr1,'Stepwise␣
↪Regression Model Comparision with Regularization')
```

```
<ipython-input-285-0f97beaf24d4>:4: FutureWarning: This dataframe has a column
name that matches the 'value_name' column name of the resultiing Dataframe. In
the future this will raise an error, please set the 'value_name' parameter of
DataFrame.melt to a unique name.
  s = s.melt(id_vars=['Model Type'], var_name='recall-score',
value_name='f1-score')
```



18 Variable Selection - Lasso Regression

19 For Mean Imputation Model

[372]:
```
lasso_regression_mean = sm.
 ↪Logit(y_balanced3,add_constant(mean_imputed_df_balanced))
lasso_results1 = lasso_regression_mean.
 ↪fit_regularized(alpha=24,method='l1',maxiter=5000,refit=True)
lasso_results1.summary2()
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6465892625917107
            Iterations: 282
            Function evaluations: 283
            Gradient evaluations: 282
```

```
[372]: <class 'statsmodels.iolib.summary2.Summary'>
"""
                         Results: Logit
=================================================================
Model:              Logit            Pseudo R-squared: 0.094
Dependent Variable: class            AIC:              20167.9412
Date:               2021-04-17 02:50 BIC:              20406.0551
No. Observations:   16012            Log-Likelihood:   -10053.
Df Model:           30               LL-Null:          -11099.
Df Residuals:       15981            LLR p-value:      0.0000
Converged:          1.0000           Scale:            1.0000
No. Iterations:     282.0000
------------------------------------------------------------------
             Coef.   Std.Err.     z      P>|z|    [0.025    0.975]
------------------------------------------------------------------
const       -0.2863   0.0241  -11.8587  0.0000   -0.3336   -0.2390
Attr1        0.0000      nan       nan     nan       nan       nan
Attr2        0.0000      nan       nan     nan       nan       nan
Attr3        0.0000      nan       nan     nan       nan       nan
Attr4        0.0000      nan       nan     nan       nan       nan
Attr5        0.0187   0.0280    0.6674  0.5045   -0.0362    0.0735
Attr6        0.0000      nan       nan     nan       nan       nan
Attr7        0.0000      nan       nan     nan       nan       nan
Attr8       -0.0221  19.9515   -0.0011  0.9991  -39.1262   39.0820
Attr9       -0.1333   0.1930   -0.6906  0.4898   -0.5114    0.2449
Attr10       0.0000      nan       nan     nan       nan       nan
Attr11      -0.1646   0.1580   -1.0417  0.2976   -0.4743    0.1451
Attr12       0.0000      nan       nan     nan       nan       nan
Attr13       0.0000      nan       nan     nan       nan       nan
Attr14       0.0000      nan       nan     nan       nan       nan
Attr15       0.0000      nan       nan     nan       nan       nan
Attr16       0.0000      nan       nan     nan       nan       nan
Attr17      -0.0012  20.1057   -0.0001  1.0000  -39.4076   39.4053
Attr18      -0.0438   0.0442   -0.9904  0.3220   -0.1303    0.0428
Attr19       0.0000      nan       nan     nan       nan       nan
Attr20       0.0000      nan       nan     nan       nan       nan
Attr21      -0.0163   0.0252   -0.6475  0.5173   -0.0657    0.0331
Attr22      -1.7670   0.2964   -5.9617  0.0000   -2.3480   -1.1861
Attr23       0.0000      nan       nan     nan       nan       nan
Attr24      -1.8356   0.1580  -11.6154  0.0000   -2.1453   -1.5258
Attr25       0.0000      nan       nan     nan       nan       nan
Attr26       0.0000      nan       nan     nan       nan       nan
Attr27      -0.0024   0.0286   -0.0841  0.9330   -0.0584    0.0536
Attr28       0.0000      nan       nan     nan       nan       nan
Attr29       0.0299   0.0205    1.4594  0.1445   -0.0103    0.0701
Attr30       0.0000      nan       nan     nan       nan       nan
Attr31       0.1052   0.1013    1.0394  0.2986   -0.0932    0.3037
```

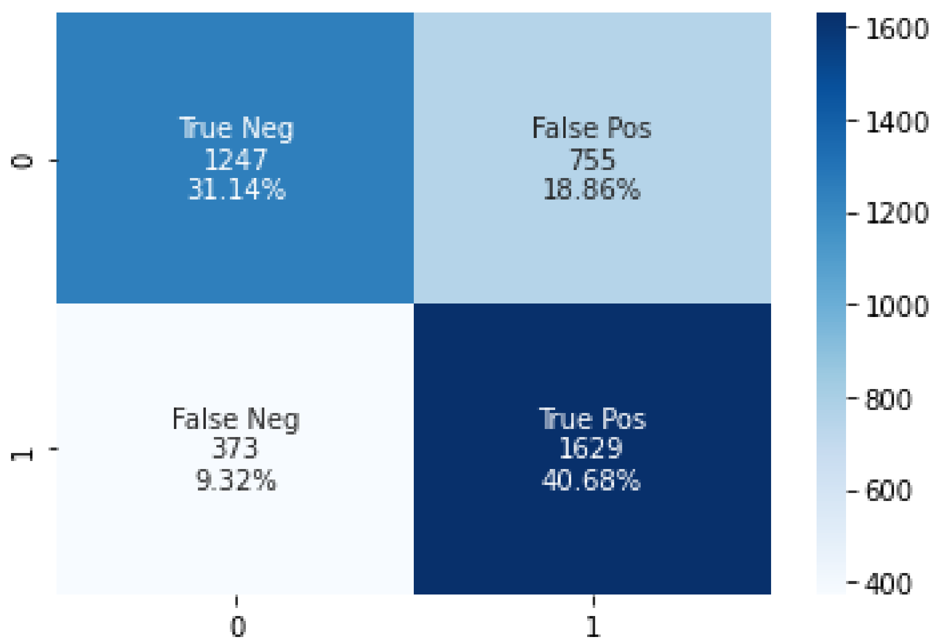
```
Attr32        0.0000       nan        nan      nan        nan        nan
Attr33        0.3428    0.0850     4.0350   0.0001     0.1763     0.5093
Attr34        1.0224    0.0719    14.2267   0.0000     0.8815     1.1633
Attr35       -1.1726    0.1278    -9.1722   0.0000    -1.4231    -0.9220
Attr36        0.2803    0.0542     5.1686   0.0000     0.1740     0.3866
Attr37       -0.0158    0.0433    -0.3647   0.7153    -0.1006     0.0691
Attr38       -1.0222    0.2715    -3.7648   0.0002    -1.5544    -0.4900
Attr39        0.0000       nan        nan      nan        nan        nan
Attr40        0.0000       nan        nan      nan        nan        nan
Attr41       -0.0131    0.0255    -0.5122   0.6085    -0.0630     0.0369
Attr42        0.0000       nan        nan      nan        nan        nan
Attr43        0.0000       nan        nan      nan        nan        nan
Attr44        0.0000       nan        nan      nan        nan        nan
Attr45        0.0000       nan        nan      nan        nan        nan
Attr46        0.0000       nan        nan      nan        nan        nan
Attr47        0.0000       nan        nan      nan        nan        nan
Attr48        2.5806    0.2508    10.2875   0.0000     2.0889     3.0722
Attr49        0.0000       nan        nan      nan        nan        nan
Attr50        0.0000       nan        nan      nan        nan        nan
Attr51        0.0000       nan        nan      nan        nan        nan
Attr52       -0.0461    0.0253    -1.8204   0.0687    -0.0958     0.0035
Attr53        0.0000       nan        nan      nan        nan        nan
Attr54        0.0000       nan        nan      nan        nan        nan
Attr55       -0.1322    0.0374    -3.5384   0.0004    -0.2055    -0.0590
Attr56        0.0447    0.0574     0.7786   0.4362    -0.0678     0.1571
Attr57       -0.0873    0.0476    -1.8334   0.0667    -0.1806     0.0060
Attr58       -0.0225    0.0275    -0.8195   0.4125    -0.0764     0.0314
Attr59       -0.0941    0.0498    -1.8879   0.0590    -0.1918     0.0036
Attr60       -0.0196    0.0303    -0.6456   0.5185    -0.0789     0.0398
Attr61       -0.1311    0.0644    -2.0363   0.0417    -0.2573    -0.0049
Attr62        0.0000       nan        nan      nan        nan        nan
Attr63       -0.9732    0.1116    -8.7195   0.0000    -1.1919    -0.7544
Attr64       -0.0822    0.0395    -2.0818   0.0374    -0.1597    -0.0048
=================================================================

"""
```

[373]:

```
lasso_pred1 = lasso_results1.
 ↪predict(add_constant(mean_imputed_df_balanced_test))
n1=[]
for val in lasso_pred1:
    if(val>=0.5):
        n1.append(1)
    else :
        n1.append(0)
recall1_lasso=recall_score(y_balanced3_test,n1)
f1_lasso=f1_score(y_balanced3_test,n1)
```

```
print(classification_report(y_balanced3_test,n1))
print("F1 Score is",f1_score(y_balanced3_test,n1))
cf_lasso1=confusion_matrix(y_balanced3_test,n1)
conf_plot(cf_lasso1)
```

```
              precision    recall  f1-score   support

         0.0       0.77      0.62      0.69      2002
         1.0       0.68      0.81      0.74      2002

    accuracy                           0.72      4004
   macro avg       0.73      0.72      0.72      4004
weighted avg       0.73      0.72      0.72      4004

F1 Score is 0.7428180574555403
```



20 For KNN Imputation Model

[43]:
```
lasso_regression_knn = sm.
↪Logit(y_balanced2,add_constant(knn_imputed_df_balanced))
lasso_results2 = lasso_regression_knn.
↪fit_regularized(alpha=24,method='l1',maxiter=1000)
lasso_results2.summary2()
```

```
Optimization terminated successfully    (Exit mode 0)
```

```
         Current function value: 0.6466960674447424
         Iterations: 287
         Function evaluations: 289
         Gradient evaluations: 287
```

[43]:
```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                           Results: Logit
=================================================================
Model:              Logit            Pseudo R-squared: 0.094
Dependent Variable: class            AIC:              20169.8449
Date:               2021-04-17 15:44 BIC:              20423.3210
No. Observations:   16012            Log-Likelihood:   -10052.
Df Model:           32               LL-Null:          -11099.
Df Residuals:       15979            LLR p-value:      0.0000
Converged:          1.0000           Scale:            1.0000
No. Iterations:     287.0000
------------------------------------------------------------------
              Coef.   Std.Err.      z      P>|z|    [0.025   0.975]
------------------------------------------------------------------
const        -0.2873    0.0242  -11.8555  0.0000   -0.3348  -0.2398
Attr1         0.0000       nan       nan     nan       nan      nan
Attr2         0.0000       nan       nan     nan       nan      nan
Attr3        -0.0082    0.3872   -0.0212  0.9831   -0.7671   0.7507
Attr4         0.0000       nan       nan     nan       nan      nan
Attr5         0.0192    0.0283    0.6757  0.4992   -0.0364   0.0747
Attr6         0.0000       nan       nan     nan       nan      nan
Attr7         0.0000       nan       nan     nan       nan      nan
Attr8        -0.0207   19.9874   -0.0010  0.9992  -39.1954  39.1539
Attr9        -0.1811    0.1939   -0.9343  0.3501   -0.5611   0.1989
Attr10        0.0000       nan       nan     nan       nan      nan
Attr11       -0.1325    0.1565   -0.8468  0.3971   -0.4393   0.1742
Attr12        0.0000       nan       nan     nan       nan      nan
Attr13        0.0000       nan       nan     nan       nan      nan
Attr14        0.0000       nan       nan     nan       nan      nan
Attr15        0.0000       nan       nan     nan       nan      nan
Attr16        0.0000       nan       nan     nan       nan      nan
Attr17       -0.0042   20.1420   -0.0002  0.9998  -39.4818  39.4734
Attr18       -0.0448    0.0446   -1.0035  0.3156   -0.1322   0.0427
Attr19        0.0000       nan       nan     nan       nan      nan
Attr20        0.0000       nan       nan     nan       nan      nan
Attr21       -0.0200    0.0275   -0.7282  0.4665   -0.0739   0.0339
Attr22       -1.8256    0.3002   -6.0806  0.0000   -2.4140  -1.2371
Attr23        0.0000       nan       nan     nan       nan      nan
Attr24       -1.8313    0.1599  -11.4509  0.0000   -2.1448  -1.5179
Attr25        0.0000       nan       nan     nan       nan      nan
Attr26        0.0000       nan       nan     nan       nan      nan
```

```
Attr27        -0.0963    0.0802    -1.2011   0.2297    -0.2535    0.0609
Attr28         0.0000       nan        nan      nan        nan       nan
Attr29         0.0275    0.0216     1.2706   0.2039    -0.0149    0.0699
Attr30         0.0000       nan        nan      nan        nan       nan
Attr31         0.1156    0.1133     1.0203   0.3076    -0.1065    0.3377
Attr32         0.0000       nan        nan      nan        nan       nan
Attr33         0.3052    0.1431     2.1331   0.0329     0.0248    0.5856
Attr34         1.0076    0.0758    13.3014   0.0000     0.8591    1.1561
Attr35        -1.1598    0.1303    -8.9025   0.0000    -1.4152   -0.9045
Attr36         0.2985    0.0555     5.3767   0.0000     0.1897    0.4073
Attr37        -0.0399    0.0688    -0.5802   0.5618    -0.1748    0.0950
Attr38        -0.9895    0.4114    -2.4055   0.0162    -1.7957   -0.1833
Attr39         0.0000       nan        nan      nan        nan       nan
Attr40         0.0000       nan        nan      nan        nan       nan
Attr41        -0.0131    0.0255    -0.5155   0.6062    -0.0630    0.0368
Attr42         0.0000       nan        nan      nan        nan       nan
Attr43         0.0000       nan        nan      nan        nan       nan
Attr44        -0.0028    0.0339    -0.0826   0.9341    -0.0693    0.0637
Attr45         0.0000       nan        nan      nan        nan       nan
Attr46         0.0000       nan        nan      nan        nan       nan
Attr47         0.0000       nan        nan      nan        nan       nan
Attr48         2.6101    0.2563    10.1849   0.0000     2.1078    3.1123
Attr49         0.0000       nan        nan      nan        nan       nan
Attr50         0.0000       nan        nan      nan        nan       nan
Attr51         0.0000       nan        nan      nan        nan       nan
Attr52        -0.0459    0.0253    -1.8123   0.0699    -0.0955    0.0037
Attr53         0.0000       nan        nan      nan        nan       nan
Attr54         0.0000       nan        nan      nan        nan       nan
Attr55        -0.1090    0.0360    -3.0295   0.0024    -0.1796   -0.0385
Attr56         0.0389    0.0628     0.6185   0.5362    -0.0843    0.1620
Attr57        -0.0884    0.0479    -1.8446   0.0651    -0.1824    0.0055
Attr58        -0.0305    0.0302    -1.0110   0.3120    -0.0897    0.0286
Attr59        -0.0949    0.0501    -1.8934   0.0583    -0.1931    0.0033
Attr60        -0.0268    0.0380    -0.7047   0.4810    -0.1014    0.0477
Attr61        -0.1321    0.0657    -2.0119   0.0442    -0.2609   -0.0034
Attr62         0.0000       nan        nan      nan        nan       nan
Attr63        -0.9294    0.1687    -5.5074   0.0000    -1.2601   -0.5986
Attr64        -0.0872    0.0412    -2.1159   0.0344    -0.1679   -0.0064
=================================================================

"""
```

[405]:

```python
# Finding perfect alpha
alpha=range(1,50)
f2=[]
for val in alpha:
```

```
    lasso_regression_knn = sm.
↪Logit(y_balanced2,add_constant(knn_imputed_df_balanced))
    lasso_results2 = lasso_regression_knn.
↪fit_regularized(alpha=val,method='l1',maxiter=1000)
    lasso_pred2 = lasso_results2.
↪predict(add_constant(knn_imputed_df_balanced_test))
    n2=[]
    for val in lasso_pred2:
        if(val>=0.5):
            n2.append(1)
        else :
            n2.append(0)

    f2.append(f1_score(y_balanced2_test,n2))
```

```
P:\Anaconda\lib\site-packages\statsmodels\discrete\discrete_model.py:1799:
RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-X))

Iteration limit reached    (Exit mode 9)
            Current function value: 0.604591918369656
            Iterations: 1000
            Function evaluations: 1010
            Gradient evaluations: 1000

P:\Anaconda\lib\site-packages\statsmodels\base\l1_solvers_common.py:71:
ConvergenceWarning: QC check did not pass for 26 out of 65 parameters
Try increasing solver accuracy or number of iterations, decreasing alpha, or
switch solvers
  warnings.warn(message, ConvergenceWarning)
P:\Anaconda\lib\site-packages\statsmodels\base\l1_solvers_common.py:144:
ConvergenceWarning: Could not trim params automatically due to failed QC check.
Trimming using trim_mode == 'size' will still work.
  warnings.warn(msg, ConvergenceWarning)
P:\Anaconda\lib\site-packages\statsmodels\base\model.py:566: ConvergenceWarning:
Maximum Likelihood optimization failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "

Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.610972710223412
            Iterations: 792
            Function evaluations: 808
            Gradient evaluations: 792

P:\Anaconda\lib\site-packages\statsmodels\base\l1_solvers_common.py:71:
ConvergenceWarning: QC check did not pass for 2 out of 65 parameters
Try increasing solver accuracy or number of iterations, decreasing alpha, or
switch solvers
```

```
  warnings.warn(message, ConvergenceWarning)
P:\Anaconda\lib\site-packages\statsmodels\base\l1_solvers_common.py:144:
ConvergenceWarning: Could not trim params automatically due to failed QC check.
Trimming using trim_mode == 'size' will still work.
  warnings.warn(msg, ConvergenceWarning)

Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6146103702664365
            Iterations: 617
            Function evaluations: 619
            Gradient evaluations: 617
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6176523266636811
            Iterations: 539
            Function evaluations: 539
            Gradient evaluations: 539
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6203703840916689
            Iterations: 514
            Function evaluations: 515
            Gradient evaluations: 514
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.622807650690014
            Iterations: 478
            Function evaluations: 480
            Gradient evaluations: 478
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6250200779814516
            Iterations: 479
            Function evaluations: 479
            Gradient evaluations: 479
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6270513870653619
            Iterations: 444
            Function evaluations: 444
            Gradient evaluations: 444
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6289355297009511
            Iterations: 426
            Function evaluations: 428
            Gradient evaluations: 426
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6306899798089884
            Iterations: 409
            Function evaluations: 410
            Gradient evaluations: 409
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6323280362651426
```

```
            Iterations: 395
            Function evaluations: 395
            Gradient evaluations: 395
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6338608083798044
            Iterations: 416
            Function evaluations: 416
            Gradient evaluations: 416
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6352985565408156
            Iterations: 378
            Function evaluations: 379
            Gradient evaluations: 378
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6366478350070602
            Iterations: 374
            Function evaluations: 374
            Gradient evaluations: 374
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.637918179623868
            Iterations: 371
            Function evaluations: 371
            Gradient evaluations: 371
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6391148550615061
            Iterations: 379
            Function evaluations: 379
            Gradient evaluations: 379
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.640241093570457
            Iterations: 364
            Function evaluations: 365
            Gradient evaluations: 364
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.641306369229854
            Iterations: 339
            Function evaluations: 340
            Gradient evaluations: 339
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6423175551258381
            Iterations: 326
            Function evaluations: 326
            Gradient evaluations: 326
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6432773950642582
            Iterations: 313
            Function evaluations: 314
            Gradient evaluations: 313
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6441876940036927
            Iterations: 326
            Function evaluations: 327
            Gradient evaluations: 326
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6450563814492222
            Iterations: 305
            Function evaluations: 306
            Gradient evaluations: 305
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6458921026346628
            Iterations: 304
            Function evaluations: 305
            Gradient evaluations: 304
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6466960674447424
            Iterations: 287
            Function evaluations: 289
            Gradient evaluations: 287
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6474690168136655
            Iterations: 291
            Function evaluations: 291
            Gradient evaluations: 291
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6482113693197643
            Iterations: 278
            Function evaluations: 278
            Gradient evaluations: 278
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6489238289053423
            Iterations: 267
            Function evaluations: 267
            Gradient evaluations: 267
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.649607605729085
            Iterations: 254
            Function evaluations: 255
            Gradient evaluations: 254
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6502641213425433
            Iterations: 263
            Function evaluations: 263
            Gradient evaluations: 263
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6508964685275717
            Iterations: 247
```

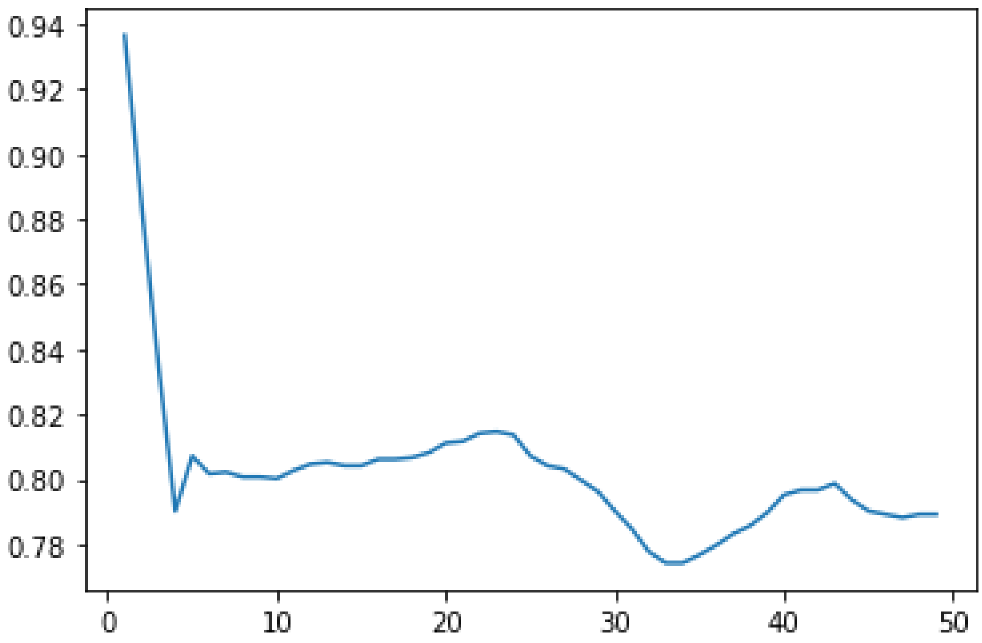
```
            Function evaluations: 248
            Gradient evaluations: 247
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6515083116112104
            Iterations: 245
            Function evaluations: 245
            Gradient evaluations: 245
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6521027486669113
            Iterations: 243
            Function evaluations: 243
            Gradient evaluations: 243
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6526803300922431
            Iterations: 218
            Function evaluations: 218
            Gradient evaluations: 218
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6532408850743363
            Iterations: 219
            Function evaluations: 219
            Gradient evaluations: 219
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6537826057082555
            Iterations: 215
            Function evaluations: 216
            Gradient evaluations: 215
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6543050703672674
            Iterations: 216
            Function evaluations: 216
            Gradient evaluations: 216
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6548089235532407
            Iterations: 217
            Function evaluations: 217
            Gradient evaluations: 217
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6552948349971003
            Iterations: 197
            Function evaluations: 198
            Gradient evaluations: 197
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6557632930417444
            Iterations: 215
            Function evaluations: 215
            Gradient evaluations: 215
Optimization terminated successfully    (Exit mode 0)
```

```
             Current function value: 0.6562149537731716
             Iterations: 204
             Function evaluations: 205
             Gradient evaluations: 204
Optimization terminated successfully    (Exit mode 0)
             Current function value: 0.656650311744796
             Iterations: 200
             Function evaluations: 200
             Gradient evaluations: 200
Optimization terminated successfully    (Exit mode 0)
             Current function value: 0.6570700237906317
             Iterations: 202
             Function evaluations: 203
             Gradient evaluations: 202
Optimization terminated successfully    (Exit mode 0)
             Current function value: 0.6574750950831346
             Iterations: 208
             Function evaluations: 208
             Gradient evaluations: 208
Optimization terminated successfully    (Exit mode 0)
             Current function value: 0.6578661244756996
             Iterations: 188
             Function evaluations: 189
             Gradient evaluations: 188
Optimization terminated successfully    (Exit mode 0)
             Current function value: 0.6582437395397427
             Iterations: 180
             Function evaluations: 181
             Gradient evaluations: 180
Optimization terminated successfully    (Exit mode 0)
             Current function value: 0.65860875378217
             Iterations: 191
             Function evaluations: 192
             Gradient evaluations: 191
Optimization terminated successfully    (Exit mode 0)
             Current function value: 0.6589620086531558
             Iterations: 206
             Function evaluations: 207
             Gradient evaluations: 206
Optimization terminated successfully    (Exit mode 0)
             Current function value: 0.6593045008712369
             Iterations: 190
             Function evaluations: 190
             Gradient evaluations: 190
Optimization terminated successfully    (Exit mode 0)
             Current function value: 0.6596369153451754
             Iterations: 188
             Function evaluations: 188
```

```
Gradient evaluations: 188
```

[407]:
```
g=sns.lineplot(x=alpha,y=f2)
ymax = max(f2)
xpos = f2.index(ymax)
xmax = alpha[xpos]
xmax
```

[407]: 1



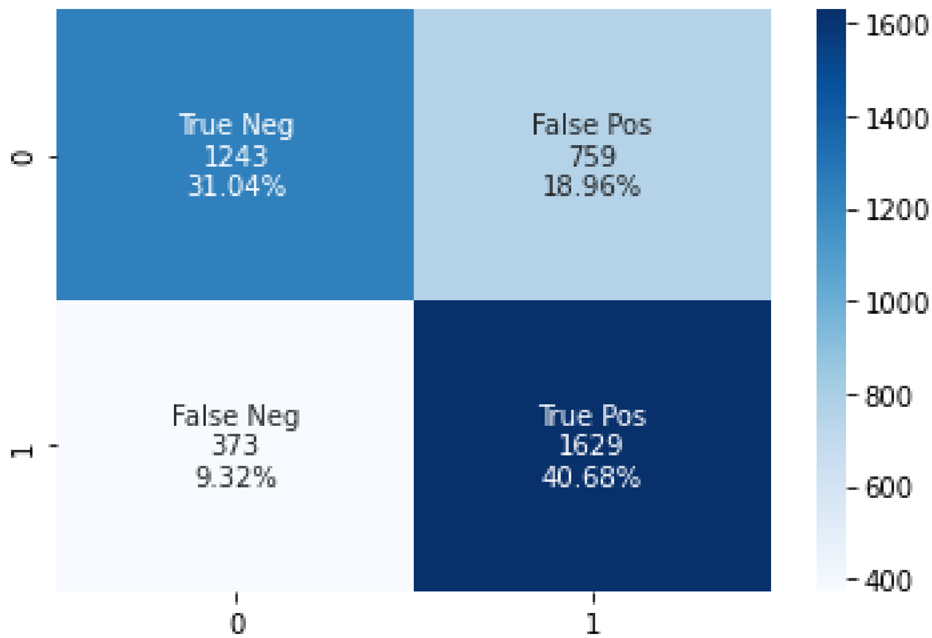
[358]:
```
xmax
```

[358]: 24

[360]:
```
lasso_pred2 = lasso_results2.predict(add_constant(knn_imputed_df_balanced_test))
n2=[]
for val in lasso_pred2:
    if(val>=0.5):
        n2.append(1)
    else :
        n2.append(0)

recall2_lasso=recall_score(y_balanced2_test,n2)
f2_lasso=f1_score(y_balanced2_test,n2)
print(classification_report(y_balanced2_test,n2))
print("F1 Score is",f1_score(y_balanced3_test,n2))
```

```
cf_lasso2=confusion_matrix(y_balanced2_test,n2)
conf_plot(cf_lasso2)
```

```
              precision    recall  f1-score   support

         0.0       0.77      0.62      0.69      2002
         1.0       0.68      0.81      0.74      2002

    accuracy                           0.72      4004
   macro avg       0.73      0.72      0.71      4004
weighted avg       0.73      0.72      0.71      4004

F1 Score is 0.7421412300683371
```



21 For MICE Impuation Model

[370]:
```
lasso_regression_mice = sm.
 ↪Logit(y_balanced1,add_constant(mice_imputed_df_balanced))
lasso_results3 = lasso_regression_mice.
 ↪fit_regularized(alpha=24,method='l1',maxiter=5000,refit=True)
lasso_results3.summary2()
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6463406191876027
            Iterations: 282
```

```
           Function evaluations: 282
           Gradient evaluations: 282
```

[370]:
```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                          Results: Logit
=================================================================
Model:              Logit            Pseudo R-squared: 0.094
Dependent Variable: class            AIC:              20181.3931
Date:               2021-04-17 02:49 BIC:              20442.5503
No. Observations:   16012            Log-Likelihood:   -10057.
Df Model:           33               LL-Null:          -11099.
Df Residuals:       15978            LLR p-value:      0.0000
Converged:          1.0000           Scale:            1.0000
No. Iterations:     282.0000
-------------------------------------------------------------------
          Coef.    Std.Err.      z      P>|z|     [0.025    0.975]
-------------------------------------------------------------------
const    -0.2778     0.0214  -12.9838   0.0000   -0.3198   -0.2359
Attr1     0.0000        nan       nan      nan       nan       nan
Attr2     0.0000        nan       nan      nan       nan       nan
Attr3    -0.2055     0.3899   -0.5270   0.5982   -0.9697    0.5587
Attr4     0.0000        nan       nan      nan       nan       nan
Attr5     0.0186     0.0280    0.6657   0.5056   -0.0362    0.0734
Attr6     0.0000        nan       nan      nan       nan       nan
Attr7     0.0000        nan       nan      nan       nan       nan
Attr8    -0.0104     0.0456   -0.2286   0.8192   -0.0999    0.0790
Attr9    -0.0608     0.1991   -0.3052   0.7602   -0.4510    0.3295
Attr10    0.0000        nan       nan      nan       nan       nan
Attr11   -0.0222     0.1432   -0.1549   0.8769   -0.3028    0.2585
Attr12    0.0000        nan       nan      nan       nan       nan
Attr13    0.0000        nan       nan      nan       nan       nan
Attr14    0.0000        nan       nan      nan       nan       nan
Attr15    0.0000        nan       nan      nan       nan       nan
Attr16    0.0000        nan       nan      nan       nan       nan
Attr17    0.0000        nan       nan      nan       nan       nan
Attr18   -0.0390     0.0415   -0.9392   0.3476   -0.1204    0.0424
Attr19    0.0000        nan       nan      nan       nan       nan
Attr20    0.0000        nan       nan      nan       nan       nan
Attr21   -0.0163     0.0252   -0.6458   0.5184   -0.0657    0.0332
Attr22   -1.9259     0.2957   -6.5129   0.0000   -2.5055   -1.3463
Attr23    0.0000        nan       nan      nan       nan       nan
Attr24   -1.8665     0.1594  -11.7089   0.0000   -2.1790   -1.5541
Attr25    0.0000        nan       nan      nan       nan       nan
Attr26    0.0000        nan       nan      nan       nan       nan
Attr27   -0.0031     0.0286   -0.1070   0.9148   -0.0591    0.0530
Attr28    0.0000        nan       nan      nan       nan       nan
```

```
Attr29     0.0235     0.0215     1.0936   0.2741   -0.0186    0.0656
Attr30     0.0000        nan        nan      nan       nan       nan
Attr31     0.0480     0.0642     0.7478   0.4546   -0.0778    0.1739
Attr32    -0.0147     0.0103    -1.4259   0.1539   -0.0348    0.0055
Attr33     0.1132     0.1453     0.7791   0.4359   -0.1715    0.3979
Attr34     1.0550     0.0883    11.9476   0.0000    0.8819    1.2280
Attr35    -1.1799     0.1305    -9.0444   0.0000   -1.4356   -0.9242
Attr36     0.2458     0.0571     4.3009   0.0000    0.1338    0.3577
Attr37    -0.0162     0.0441    -0.3680   0.7128   -0.1027    0.0703
Attr38    -1.0235     0.4133    -2.4761   0.0133   -1.8336   -0.2134
Attr39     0.0000        nan        nan      nan       nan       nan
Attr40     0.0000        nan        nan      nan       nan       nan
Attr41    -0.0136     0.0255    -0.5318   0.5949   -0.0636    0.0364
Attr42     0.0000        nan        nan      nan       nan       nan
Attr43     0.0000        nan        nan      nan       nan       nan
Attr44    -0.0035     0.0328    -0.1076   0.9143   -0.0679    0.0608
Attr45    -0.0000     0.0298    -0.0000   1.0000   -0.0584    0.0584
Attr46     0.0000        nan        nan      nan       nan       nan
Attr47     0.0000        nan        nan      nan       nan       nan
Attr48     2.6277     0.2556    10.2800   0.0000    2.1267    3.1286
Attr49     0.0000        nan        nan      nan       nan       nan
Attr50     0.0000        nan        nan      nan       nan       nan
Attr51     0.0000        nan        nan      nan       nan       nan
Attr52     0.0000        nan        nan      nan       nan       nan
Attr53     0.0188     0.0366     0.5136   0.6075   -0.0530    0.0906
Attr54     0.0000        nan        nan      nan       nan       nan
Attr55    -0.1285     0.0379    -3.3888   0.0007   -0.2028   -0.0542
Attr56     0.0597     0.0619     0.9652   0.3344   -0.0615    0.1810
Attr57    -0.0858     0.0472    -1.8173   0.0692   -0.1784    0.0067
Attr58     0.0000        nan        nan      nan       nan       nan
Attr59    -0.0925     0.0495    -1.8666   0.0620   -0.1896    0.0046
Attr60    -0.0180     0.0349    -0.5165   0.6055   -0.0865    0.0504
Attr61    -0.1353     0.0665    -2.0333   0.0420   -0.2657   -0.0049
Attr62    -0.0344     0.1017    -0.3380   0.7354   -0.2336    0.1649
Attr63    -0.6987     0.1685    -4.1453   0.0000   -1.0290   -0.3683
Attr64    -0.1057     0.0533    -1.9823   0.0474   -0.2102   -0.0012
=================================================================

"""
```

[20]:

```
np.random.seed(123)
N = 100
x1 = np.random.normal(size=N)
x2 = x1 + np.random.normal(size=N, scale=1)
y = x1 + np.random.normal(size=N)
x12 = np.vstack((x1, x2))
```

[26]:
```
x2
```

[26]:
```
array([-0.44357591, -0.98054248,  0.99524313,  1.09200921, -0.60322623,
        1.68557867, -2.24712976, -2.29088834,  1.6920829 , -2.47215015,
       -1.10656575,  1.14816058,  0.75617267, -0.13765301,  0.56875709,
       -0.15561042,  0.83498161,  1.85431081,  2.96346524, -1.63885936,
        0.46158256,  0.93862396, -0.81508651,  1.92404466,  0.3548103 ,
       -0.90798389,  1.71944653, -0.92894056,  0.33427858, -1.42567883,
       -1.25294084, -3.89863222, -2.52797031, -0.37819066,  1.68841182,
        0.14983317, -0.54610918,  2.49419282,  0.63932928, -0.07037279,
       -1.62879792, -1.59745454,  0.87639885,  0.90657084,  0.89513776,
       -0.22391062,  2.84863616,  1.95745661,  0.73906722,  2.38145107,
       -1.04026885, -0.75506285,  0.33182335, -2.67493139, -0.98997184,
        1.23725826,  1.44456256,  1.22421162,  2.87290162,  0.9262167 ,
       -0.75239272,  0.6008988 ,  0.44829879, -0.62179139,  2.08295248,
       -0.09118641,  1.56915386, -1.32811852, -1.11907413, -0.0748725 ,
       -0.58055256, -0.90627557,  0.42139028, -1.0242958 , -1.72001439,
        2.51165646, -0.09598182, -0.73725493, -1.22913653, -1.46598465,
        1.3486982 ,  0.77072028,  3.05630542,  0.44837226, -0.86340027,
       -3.64295701, -1.2813824 , -2.19058084,  1.7322889 ,  0.55602547,
        1.32739787, -1.29732006,  0.38061724,  1.05174417, -0.13799183,
       -2.19994055, -1.3538614 , -1.47432227,  0.0381389 , -0.5971227 ])
```

[363]:
```
alpha=range(1,50)
f3=[]
for val in alpha:
    lasso_regression_mice = sm.
 ↪Logit(y_balanced1,add_constant(mice_imputed_df_balanced))
    lasso_results3 = lasso_regression_mice.
 ↪fit_regularized(alpha=val,method='l1',maxiter=5000,refit=True)
    lasso_pred3 = lasso_results3.
 ↪predict(add_constant(mice_imputed_df_balanced_test))
    n3=[]
    for val in lasso_pred3:
        if(val>=0.5):
            n3.append(1)
        else :
            n3.append(0)

    f3.append(f1_score(y_balanced1_test,n3))
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.606043624177638
            Iterations: 1451
            Function evaluations: 1543
            Gradient evaluations: 1451
Optimization terminated successfully    (Exit mode 0)
```

```
            Current function value: 0.6139530438229962
            Iterations: 790
            Function evaluations: 799
            Gradient evaluations: 790
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6174715115094261
            Iterations: 599
            Function evaluations: 608
            Gradient evaluations: 599
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.62010855014221
            Iterations: 509
            Function evaluations: 510
            Gradient evaluations: 509
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6223706902219037
            Iterations: 481
            Function evaluations: 481
            Gradient evaluations: 481
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.624457896121029
            Iterations: 435
            Function evaluations: 436
            Gradient evaluations: 435
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6263824423793312
            Iterations: 424
            Function evaluations: 424
            Gradient evaluations: 424
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6281651153227342
            Iterations: 404
            Function evaluations: 407
            Gradient evaluations: 404
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6298388217042933
            Iterations: 389
            Function evaluations: 389
            Gradient evaluations: 389
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6314127961112469
            Iterations: 365
            Function evaluations: 365
            Gradient evaluations: 365
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6328936826893353
            Iterations: 357
            Function evaluations: 357
```

```
            Gradient evaluations: 357
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6342881362187002
            Iterations: 354
            Function evaluations: 355
            Gradient evaluations: 354
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6356033277916016
            Iterations: 356
            Function evaluations: 356
            Gradient evaluations: 356
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6368453873520399
            Iterations: 348
            Function evaluations: 350
            Gradient evaluations: 348
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6380189577899928
            Iterations: 348
            Function evaluations: 348
            Gradient evaluations: 348
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6391260809720553
            Iterations: 356
            Function evaluations: 357
            Gradient evaluations: 356
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6401748159263442
            Iterations: 346
            Function evaluations: 346
            Gradient evaluations: 346
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6411704733541381
            Iterations: 334
            Function evaluations: 334
            Gradient evaluations: 334
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6421218481765439
            Iterations: 306
            Function evaluations: 308
            Gradient evaluations: 306
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6430353052527171
            Iterations: 263
            Function evaluations: 265
            Gradient evaluations: 263
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6439124166045688
```

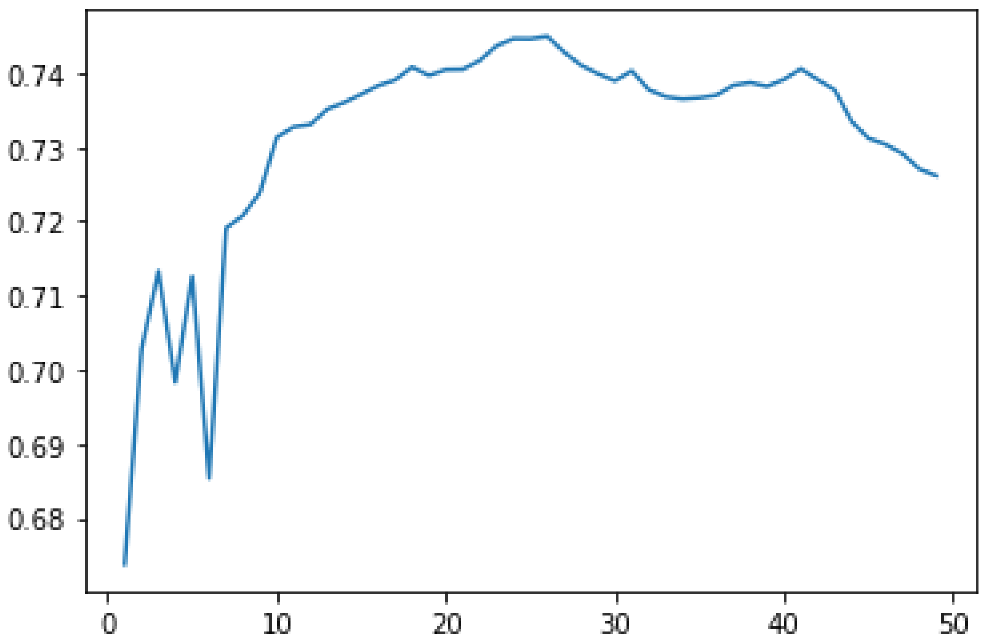
```
            Iterations: 269
            Function evaluations: 270
            Gradient evaluations: 269
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6447546849645833
            Iterations: 289
            Function evaluations: 290
            Gradient evaluations: 289
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6455637556879975
            Iterations: 287
            Function evaluations: 287
            Gradient evaluations: 287
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6463406191876027
            Iterations: 282
            Function evaluations: 282
            Gradient evaluations: 282
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6470862140994494
            Iterations: 282
            Function evaluations: 282
            Gradient evaluations: 282
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6478014683251971
            Iterations: 280
            Function evaluations: 280
            Gradient evaluations: 280
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6484892882819754
            Iterations: 292
            Function evaluations: 292
            Gradient evaluations: 292
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6491548193469728
            Iterations: 270
            Function evaluations: 271
            Gradient evaluations: 270
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6498005083438926
            Iterations: 264
            Function evaluations: 264
            Gradient evaluations: 264
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.650426896475759
            Iterations: 254
            Function evaluations: 254
            Gradient evaluations: 254
```

```
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6510330181275862
            Iterations: 252
            Function evaluations: 254
            Gradient evaluations: 252
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6516180914805076
            Iterations: 249
            Function evaluations: 250
            Gradient evaluations: 249
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6521828665969664
            Iterations: 241
            Function evaluations: 245
            Gradient evaluations: 241
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6527283842893542
            Iterations: 233
            Function evaluations: 233
            Gradient evaluations: 233
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6532555301529792
            Iterations: 223
            Function evaluations: 223
            Gradient evaluations: 223
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6537651694790638
            Iterations: 234
            Function evaluations: 234
            Gradient evaluations: 234
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6542581315154932
            Iterations: 217
            Function evaluations: 218
            Gradient evaluations: 217
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6547348066628141
            Iterations: 214
            Function evaluations: 215
            Gradient evaluations: 214
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6551957877583805
            Iterations: 204
            Function evaluations: 204
            Gradient evaluations: 204
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6556416903684865
            Iterations: 196
```

```
            Function evaluations: 197
            Gradient evaluations: 196
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6560733935660694
            Iterations: 192
            Function evaluations: 192
            Gradient evaluations: 192
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6564913627176624
            Iterations: 193
            Function evaluations: 193
            Gradient evaluations: 193
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6568960722711973
            Iterations: 182
            Function evaluations: 182
            Gradient evaluations: 182
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6572878956283246
            Iterations: 190
            Function evaluations: 191
            Gradient evaluations: 190
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.657667276296091
            Iterations: 172
            Function evaluations: 172
            Gradient evaluations: 172
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6580345309954598
            Iterations: 173
            Function evaluations: 174
            Gradient evaluations: 173
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6583900412855636
            Iterations: 168
            Function evaluations: 169
            Gradient evaluations: 168
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6587342320802563
            Iterations: 165
            Function evaluations: 165
            Gradient evaluations: 165
Optimization terminated successfully    (Exit mode 0)
            Current function value: 0.6590683403747615
            Iterations: 160
            Function evaluations: 160
            Gradient evaluations: 160
```

[369]:
```
g=sns.lineplot(x=alpha,y=f3)
ymax = max(f2)
xpos = f2.index(ymax)
xmax = alpha[xpos]
xmax
```

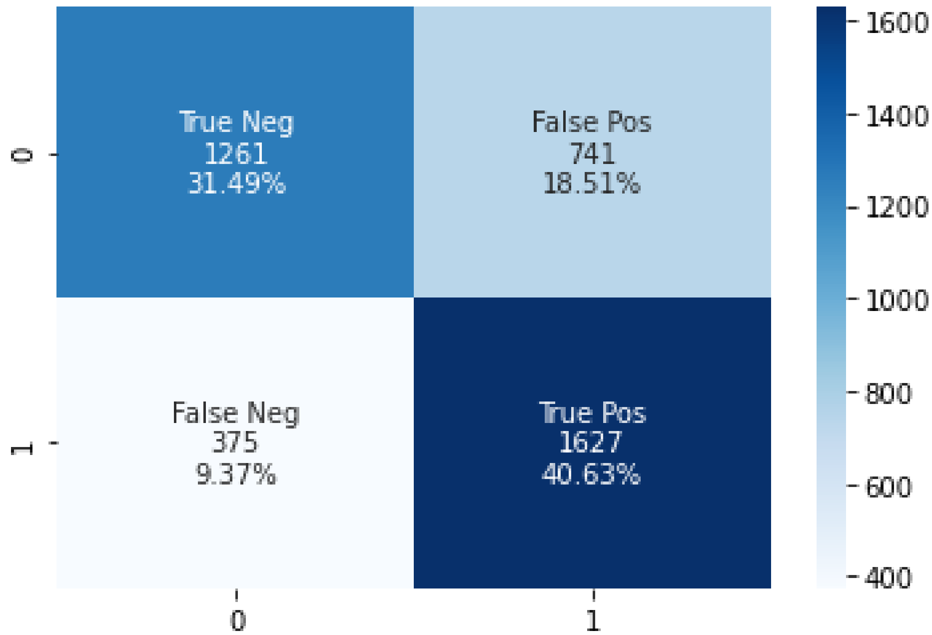
[369]: 24



[371]:
```
lasso_pred3 = lasso_results3.
 ↪predict(add_constant(mice_imputed_df_balanced_test))
n3=[]
for val in lasso_pred3:
    if(val>=0.5):
        n3.append(1)
    else :
        n3.append(0)

recall3_lasso=recall_score(y_balanced1_test,n3)
f3_lasso=f1_score(y_balanced1_test,n3)
print(classification_report(y_balanced1_test,n3))
print("F1 Score is",f1_score(y_balanced1_test,n3))
cf_lasso3=confusion_matrix(y_balanced1_test,n3)
conf_plot(cf_lasso3)
```

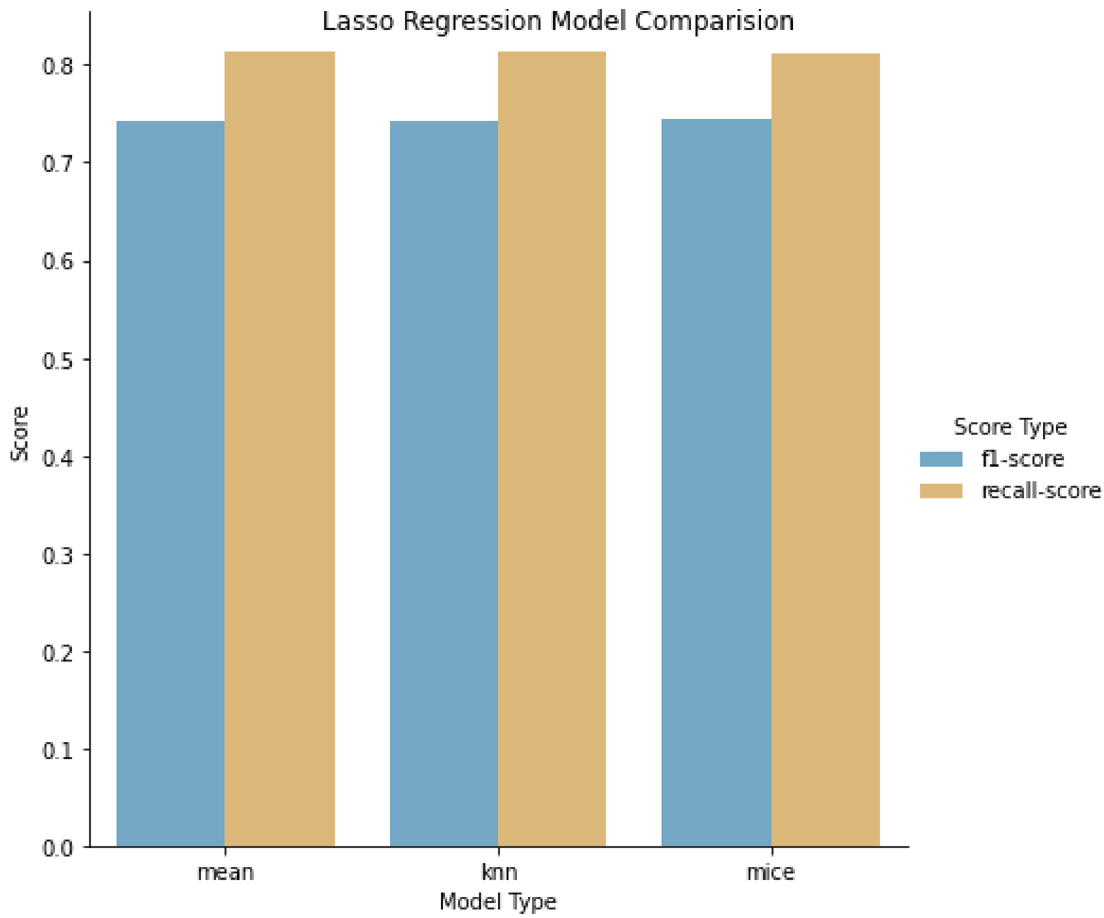
```
              precision    recall  f1-score   support
```

```
           0       0.77      0.63      0.69      2002
           1       0.69      0.81      0.74      2002

    accuracy                           0.72      4004
   macro avg       0.73      0.72      0.72      4004
weighted avg       0.73      0.72      0.72      4004

F1 Score is 0.7446224256292906
```



[374]:
```
barchart(f1_lasso,f2_lasso,f3_lasso,recall1_lasso,recall2_lasso,recall3_lasso,'Lasso␣
↪Regression Model Comparision')
```

```
<ipython-input-285-0f97beaf24d4>:4: FutureWarning: This dataframe has a column
name that matches the 'value_name' column name of the resultiing Dataframe. In
the future this will raise an error, please set the 'value_name' parameter of
DataFrame.melt to a unique name.
  s = s.melt(id_vars=['Model Type'], var_name='recall-score',
value_name='f1-score')
```

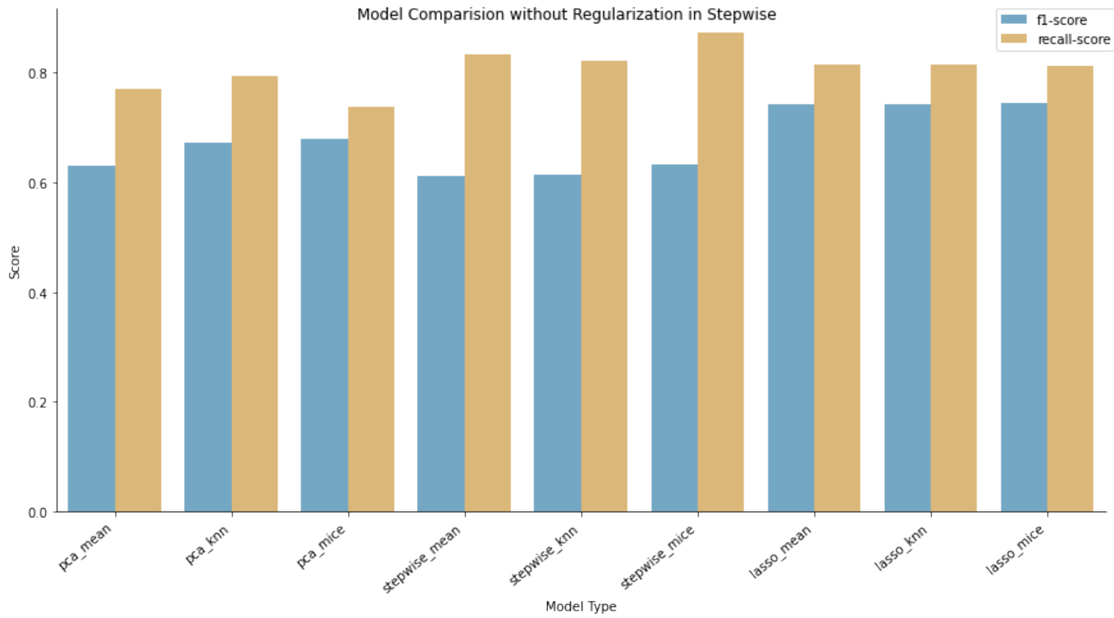


22 Overall Model Comparision

[403]:
```
def barchart1(f1,f2,f3,f4,f5,f6,f7,f8,f9,r1,r2,r3,r4,r5,r6,r7,r8,r9,title):
    data={'f1-score':[f1,f2,f3,f4,f5,f6,f7,f8,f9],'Model Type':
 ↪['pca_mean','pca_knn','pca_mice','stepwise_mean','stepwise_knn','stepwise_mice','lasso_mean
 ↪[r1,r2,r3,r4,r5,r6,r7,r8,r9]}
    s=pd.DataFrame(data)
    s = s.melt(id_vars=['Model Type'], var_name='recall-score',␣
 ↪value_name='f1-score')



    g = sns.catplot(
        data=s, kind="bar",
        x="Model Type", y="f1-score", hue="recall-score",
        ci="sd", palette='colorblind', alpha=.6, height=6,legend=False,aspect=2
    )
    plt.legend(bbox_to_anchor=(1.01, 1),
```

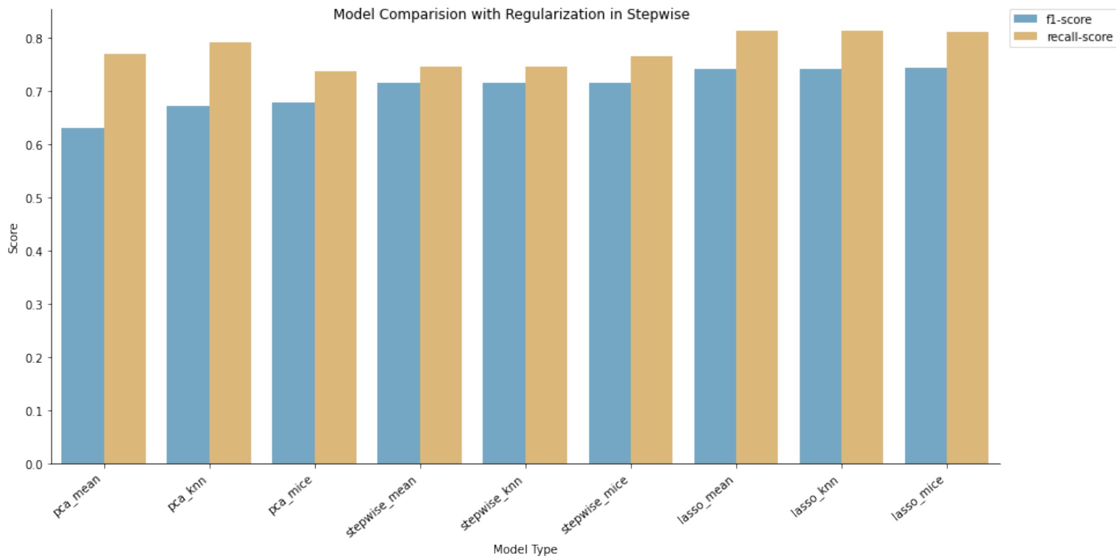
```
            borderaxespad=0)
    g.set_xticklabels(rotation=40, ha="right")

    g.set_axis_labels("Model Type", "Score")
    g.fig.suptitle(title)

barchart1(f1_pca,f2_pca,f3_pca,f1_lr,f2_lr,f3_lr,f1_lasso,f2_lasso,f3_lasso,recall1_pca,recall
 ↪Comparision without Regularization in Stepwise")
```

```
<ipython-input-403-47ea7e1c48d2>:4: FutureWarning: This dataframe has a column
name that matches the 'value_name' column name of the resultiing Dataframe. In
the future this will raise an error, please set the 'value_name' parameter of
DataFrame.melt to a unique name.
  s = s.melt(id_vars=['Model Type'], var_name='recall-score',
value_name='f1-score')
```



[404]:
```
barchart1(f1_pca,f2_pca,f3_pca,f1_lr1,f2_lr1,f3_lr1,f1_lasso,f2_lasso,f3_lasso,recall1_pca,rec
 ↪Comparision with Regularization in Stepwise")
```

```
<ipython-input-403-47ea7e1c48d2>:4: FutureWarning: This dataframe has a column
name that matches the 'value_name' column name of the resultiing Dataframe. In
the future this will raise an error, please set the 'value_name' parameter of
DataFrame.melt to a unique name.
  s = s.melt(id_vars=['Model Type'], var_name='recall-score',
value_name='f1-score')
```



23 Ignore after this. Solve before this

[ ]:

```
[168]: mice_imputed_df_train
```

[168]:

```
         Attr1     Attr2     Attr3     Attr4     Attr5     Attr6     Attr7  \
0     0.009136  0.044060 -0.023776 -0.017587  0.012771  0.016023 -0.008635
1    -0.302844  0.158568 -0.101395 -0.018074  0.008621 -0.108234 -0.319247
2    -0.044806  0.069361 -0.009242 -0.017449  0.012753  0.008465 -0.053644
3    -0.028218 -0.039600 -0.007837 -0.017196  0.012430  0.016023 -0.045826
4    -0.104057  0.087673 -0.044439 -0.017763  0.012500  0.016023 -0.121332
…          …         …         …         …         …         …         …
8397 -0.467961  0.113971 -0.212052 -0.019113  0.026127 -0.065392 -0.475304
8398 -0.160663 -0.100980 -0.008610 -0.016012  0.013771 -0.017757 -0.177689
8399 -0.044136  0.062241 -0.050132 -0.018120  0.012186  0.022015 -0.061674
8400  0.048549 -0.062627  0.053789 -0.014151  0.012897  0.104734  0.060301
8401 -0.055420 -0.080252 -0.023567 -0.017368  0.013364  0.005615 -0.068618

         Attr8     Attr9    Attr10  …    Attr56    Attr57    Attr58  \
0    -0.023416  0.162110 -0.041293  …  0.007313  0.013771 -0.012260
1    -0.024173 -0.008267 -0.155736  …  0.105909  0.019472 -0.015106
2    -0.023662 -0.037178 -0.066579  …  0.007620  0.014524 -0.012260
3    -0.021501 -0.130134  0.042322  …  0.010132  0.001210 -0.012396
4    -0.023803 -0.006280 -0.084883  …  0.003796  0.044461 -0.012113
…          …         …      …   …   …         …         …
8397 -0.023966 -0.153653 -0.111171  … -0.029915  0.088666 -0.009339
```

```
8398 -0.011694 -0.117003  0.073271  … -0.025945 -0.005377 -0.011241
8399 -0.023608 -0.086364 -0.060404  …  0.020624  0.009887 -0.012634
8400 -0.019991 -0.132290  0.065337  …  0.073087 -0.001605 -0.014197
8401 -0.017731 -0.098789  0.080778  …  0.004635 -0.000639 -0.012156

        Attr59    Attr60    Attr61    Attr62    Attr63    Attr64  class
0    -0.019237 -0.015328  0.320285 -0.025674 -0.169389 -0.046766      0
1    -0.019237 -0.015352 -0.106095  0.000941 -0.261286  0.111355      0
2    -0.010607 -0.015355 -0.128932 -0.009393 -0.245782 -0.014508      0
3    -0.019237 -0.015389 -0.099410 -0.012731 -0.238080 -0.079589      0
4    -0.020600 -0.015350 -0.115056 -0.009412 -0.245742 -0.029761      0
…          …         …         …         …         …         …    …
8397 -0.019237 -0.013054 -0.104351  0.059092 -0.287114 -0.081222      0
8398 -0.019237 -0.018137 -0.118041 -0.029267 -0.114989 -0.081171      0
8399  0.038413 -0.015290 -0.112388 -0.008668 -0.247224 -0.080540      0
8400 -0.017830 -0.015319 -0.098084 -0.024566 -0.180380 -0.078789      0
8401 -0.017789 -0.012840 -0.046151 -0.030049 -0.096769 -0.081121      0

[8402 rows x 65 columns]
```

[169]:
```
s=mice_imputed_df_test['class']
o=mice_imputed_df_train['class']
```

[191]:
```
from sklearn.metrics import f1_score
w = {0:1, 1:20}# define model
lg2 = LogisticRegression(random_state=13,␣
 ↪class_weight=w,max_iter=5000,penalty='l1',solver='liblinear')
# fit it
mod=lg2.fit(np.array(mice_imputed_df_train.iloc[:,:-1]),np.array(o))
# test
y_pred = lg2.predict(mice_imputed_df_test.iloc[:,:-1])# performance
print(f'Accuracy Score: {accuracy_score(np.array(s),y_pred)}')
print(f'Confusion Matrix: \n{confusion_matrix(np.array(s), y_pred)}')
print(f1_score(np.array(s),y_pred))
print(f'Recall score: {recall_score(np.array(s),y_pred)}')
print(classification_report(np.array(s),y_pred))
```

```
Accuracy Score: 0.2256068538791052
Confusion Matrix:
[[ 386 1616]
 [  11   88]]
0.0976150859678314
Recall score: 0.8888888888888888
              precision    recall  f1-score   support

           0       0.97      0.19      0.32      2002
           1       0.05      0.89      0.10        99
```

```
    accuracy                           0.23      2101
   macro avg       0.51      0.54      0.21      2101
weighted avg       0.93      0.23      0.31      2101
```

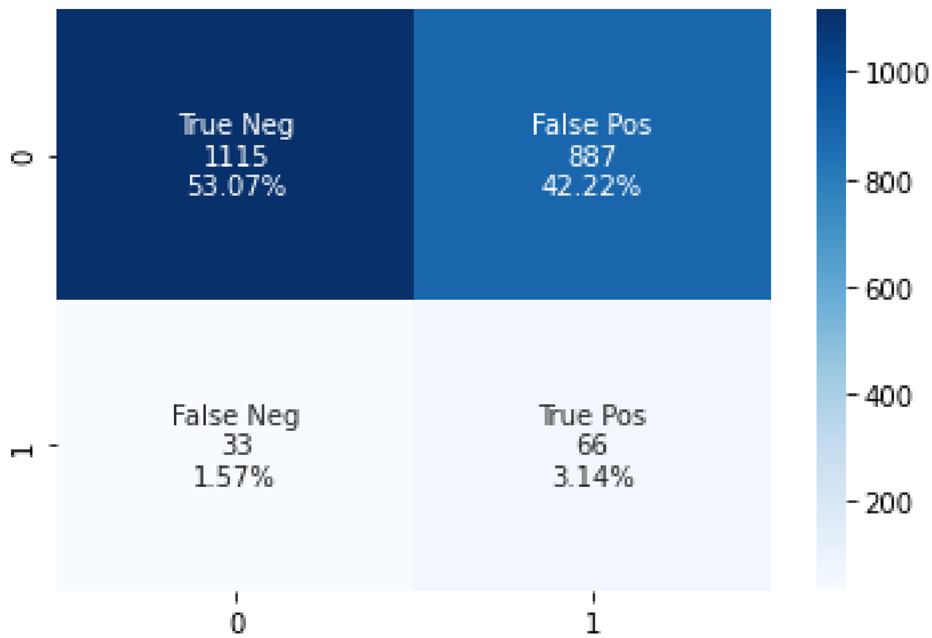
[188]:
```
nice_output(mod,mice_imputed_df_train)
```

[188]:
```
   Variables  Coefficients  Standard Error          z     P>|z|
0      const     -6.107917        2.966865  -2.058711  0.039522
1      Attr1     -0.010736        5.721712  -0.001876  0.998503
2      Attr2      0.082941       12.493448   0.006639  0.994703
3      Attr3     -0.076265        4.914893  -0.015517  0.987620
4      Attr4      0.019500       32.636825   0.000597  0.999523
..       …           …              …           …         …
61    Attr61     -0.047253        0.499604  -0.094581  0.924647
62    Attr62     -0.018386        2.293202  -0.008018  0.993603
63    Attr63     -0.164678        1.803306  -0.091320  0.927239
64    Attr64     -0.002795        0.384558  -0.007267  0.994202
65     class     11.839748        0.956619  12.376665  0.000000

[66 rows x 5 columns]
```

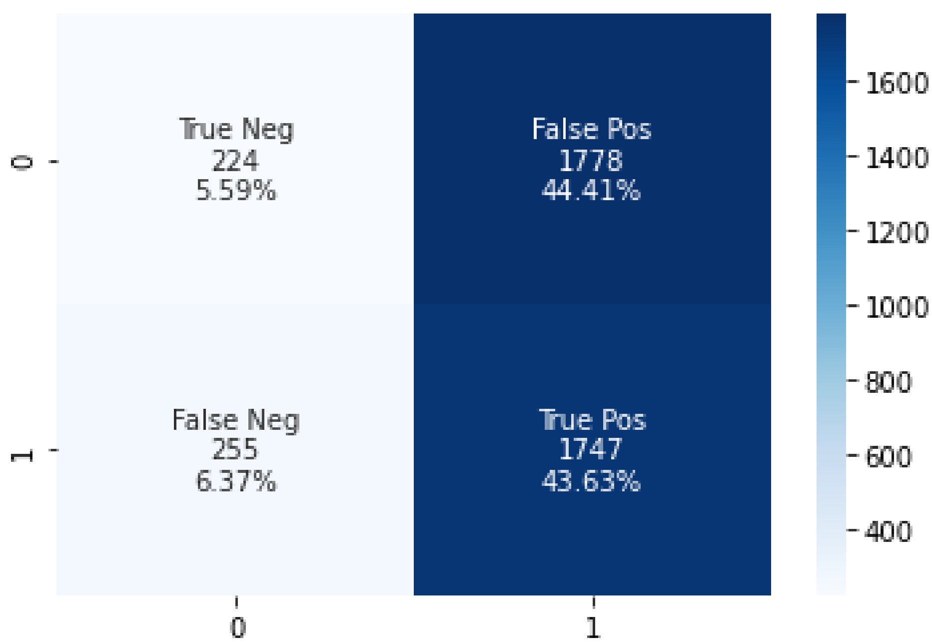
[181]:
```
print(classification_report(np.array(y_balanced1_test),m3))
```

```
              precision    recall  f1-score   support

           0       0.47      0.11      0.18      2002
           1       0.50      0.87      0.63      2002

    accuracy                           0.49      4004
   macro avg       0.48      0.49      0.41      4004
weighted avg       0.48      0.49      0.41      4004
```

[179]:
```
conf_plot(confusion_matrix(np.array(s), y_pred))
```



[180]:
```
conf_plot(cf_mice)
```



[194]:
```
from sklearn.linear_model import LogisticRegression
```

```
from sklearn.model_selection import train_test_split, GridSearchCV,␣
↪cross_val_score, RepeatedStratifiedKFold, StratifiedKFold
from sklearn.metrics import accuracy_score, confusion_matrix,roc_curve,␣
↪roc_auc_score, precision_score, recall_score, precision_recall_curve
from sklearn.metrics import f1_score
```

[192]:
```
w = [{0:1000,1:100},{0:1000,1:10}, {0:1000,1:1.0},
     {0:500,1:1.0}, {0:400,1:1.0}, {0:300,1:1.0}, {0:200,1:1.0},
     {0:150,1:1.0}, {0:100,1:1.0}, {0:99,1:1.0}, {0:10,1:1.0},
     {0:0.01,1:1.0}, {0:0.01,1:10}, {0:0.01,1:100},
     {0:0.001,1:1.0}, {0:0.005,1:1.0}, {0:1.0,1:1.0},
     {0:1.0,1:0.1}, {0:10,1:0.1}, {0:100,1:0.1},
     {0:10,1:0.01}, {0:1.0,1:0.01}, {0:1.0,1:0.001}, {0:1.0,1:0.005},
     {0:1.0,1:10}, {0:1.0,1:99}, {0:1.0,1:100}, {0:1.0,1:150},
     {0:1.0,1:200}, {0:1.0,1:300},{0:1.0,1:400},{0:1.0,1:500},
     {0:1.0,1:1000}, {0:10,1:1000},{0:100,1:1000} ]
crange = np.arange(0.5, 20.0, 0.5)
hyperparam_grid = {"class_weight": w
                   ,"penalty": ["l1", "l2"]}
```

[203]:
```
# logistic model classifier
lg4 = LogisticRegression(random_state=13,max_iter=5000)# define evaluation␣
↪procedure
grid = GridSearchCV(lg4,hyperparam_grid,scoring="f1", cv=10, n_jobs=-1,␣
↪refit=True)
grid.fit(np.array(mice_imputed_df_train[variables_mice]),np.array(o))
print(f'Best score: {grid.best_score_} with param: {grid.best_params_}')
```

```
Best score: 0.09075251120381389 with param: {'class_weight': {0: 1.0, 1: 400},
'penalty': 'l2'}
```

[200]:
```
col_mice
```

[200]:
```
Index(['Attr5', 'Attr8', 'Attr9', 'Attr13', 'Attr15', 'Attr20', 'Attr21',
       'Attr24', 'Attr27', 'Attr28', 'Attr29', 'Attr31', 'Attr34', 'Attr36',
       'Attr37', 'Attr40', 'Attr41', 'Attr44', 'Attr45', 'Attr48', 'Attr52',
       'Attr55', 'Attr57', 'Attr58', 'Attr59', 'Attr60', 'Attr61', 'Attr63',
       'Attr64'],
      dtype='object')
```

[197]:
```
from sklearn.metrics import SCORERS
sorted(SCORERS.keys())
```

[197]:
```
['accuracy',
 'adjusted_mutual_info_score',
 'adjusted_rand_score',
 'average_precision',
```

```
 'balanced_accuracy',
 'completeness_score',
 'explained_variance',
 'f1',
 'f1_macro',
 'f1_micro',
 'f1_samples',
 'f1_weighted',
 'fowlkes_mallows_score',
 'homogeneity_score',
 'jaccard',
 'jaccard_macro',
 'jaccard_micro',
 'jaccard_samples',
 'jaccard_weighted',
 'max_error',
 'mutual_info_score',
 'neg_brier_score',
 'neg_log_loss',
 'neg_mean_absolute_error',
 'neg_mean_gamma_deviance',
 'neg_mean_poisson_deviance',
 'neg_mean_squared_error',
 'neg_mean_squared_log_error',
 'neg_median_absolute_error',
 'neg_root_mean_squared_error',
 'normalized_mutual_info_score',
 'precision',
 'precision_macro',
 'precision_micro',
 'precision_samples',
 'precision_weighted',
 'r2',
 'recall',
 'recall_macro',
 'recall_micro',
 'recall_samples',
 'recall_weighted',
 'roc_auc',
 'roc_auc_ovo',
 'roc_auc_ovo_weighted',
 'roc_auc_ovr',
 'roc_auc_ovr_weighted',
 'v_measure_score']
```

[232]:
```
Z3
```

[232]:

```
        const    Attr24    Attr41    Attr48    Attr64    Attr34    Attr36  \
0         1.0 -0.019507 -0.011140  0.070976 -0.046766 -0.020652  0.440434
1         1.0 -0.474792 -0.011153 -0.279960  0.105123 -0.065054 -0.082029
2         1.0 -0.077651 -0.010612 -0.011865 -0.014508 -0.050789 -0.170687
3         1.0 -0.053004 -0.011175  0.019158 -0.079589 -0.048162 -0.455736
4         1.0 -0.149437 -0.013764 -0.071690 -0.029761 -0.048843 -0.075936
…       …       …       …       …       …       …       …
16007     1.0 -0.028509 -0.011196  0.124022 -0.058951 -0.068477 -0.004984
16008     1.0 -0.052534 -0.012859 -0.002130  0.400967 -0.003393  0.628315
16009     1.0 -0.052082 -0.011222  0.072154 -0.072669 -0.058746 -0.191521
16010     1.0 -0.080037 -0.011146 -0.017016 -0.080769 -0.062467 -0.513252
16011     1.0  0.076830 -0.011961 -0.023745 -0.061909  0.022236  1.045285

         Attr61    Attr58     Attr9    Attr55    Attr52    Attr29    Attr31  \
0      0.320285 -0.012260  0.162110 -0.107653 -0.014283  0.098249  0.021003
1     -0.106095 -0.015106 -0.008267 -0.110570 -0.012510 -2.266791  0.004498
2     -0.128932 -0.012260 -0.037178 -0.080205 -0.013937  0.461760  0.020842
3     -0.099410 -0.012396 -0.130134 -0.095742 -0.014000  0.035529  0.023965
4     -0.115056 -0.012113 -0.006280 -0.119587 -0.013951 -0.147364  0.018285
…         …       …       …       …       …       …       …
16007 -0.111412 -0.012503 -0.089953  0.340355 -0.014230  1.242899  0.025746
16008 -0.099774 -0.012209  0.212112 -0.063341 -0.014331  0.030683  0.019458
16009 -0.102502 -0.012496 -0.089869  0.259217 -0.014480  0.752891  0.024365
16010 -0.106162 -0.012659 -0.156423 -0.060733 -0.013959  0.809920  0.027438
16011  0.358438 -0.012490  0.346577 -0.122320 -0.014406 -0.314912  0.018897

         Attr60    Attr45    Attr13     Attr8     Attr5
0     -0.015328 -0.012383 -0.007094 -0.023416  0.012771
1     -0.015352 -0.012619 -0.007295 -0.024173  0.008621
2     -0.015355 -0.012402 -0.007098 -0.023662  0.012753
3     -0.015389 -0.012373 -0.006033 -0.021501  0.012430
4     -0.015350 -0.012455 -0.007545 -0.023803  0.012500
…         …       …       …       …       …
16007 -0.014969 -0.011717 -0.005726 -0.022939  0.013551
16008 -0.014216 -0.012407 -0.007441 -0.023190  0.013306
16009 -0.015300 -0.012251 -0.005854 -0.008703  0.014026
16010 -0.014875 -0.012101 -0.005166 -0.023189  0.013537
16011 -0.015113 -0.012507 -0.007753 -0.023068  0.013032

[16012 rows x 19 columns]
```

[239]:

```
# define model
lg4 = LogisticRegression(random_state=13, penalty='l1',class_weight={0:1, 1: 1}␣
 ↪)
# fit it
lg4.fit(X3,y)
# test
```

```
y_pred = lg4.predict(add_constant(mice_imputed_df_test[variables_mice]))#␣
↪performance
print(f'Accuracy Score: {accuracy_score(y1,y_pred)}')
print(f'Confusion Matrix: \n{confusion_matrix(y1, y_pred)}')
print(f'Area Under Curve: {roc_auc_score(y1, y_pred)}') # 0.5
print(f'Recall score: {recall_score(y1,y_pred)}')
print(f'f1 score: {f1_score(y1,y_pred)}')
print(f'Precision score: {precision_score(y1,y_pred)}')
```

```
Accuracy Score:               precision    recall  f1-score   support

         0.0       0.46      0.11      0.18      2002
         1.0       0.49      0.87      0.63      2002

    accuracy                           0.49      4004
   macro avg       0.48      0.49      0.41      4004
weighted avg       0.48      0.49      0.41      4004

Confusion Matrix:
[[ 223 1779]
 [ 259 1743]]
Recall score: 0.8706293706293706
f1 score: 0.6310644460535844
Precision score: 0.4948892674616695
```

[132]:
```
model4 = sm.Logit(y,X2)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-132-85a3a1e94ec7> in <module>
----> 1 model4 = sm.Logit(y,X2)

NameError: name 'sm' is not defined
```

[ ]:
```

```

[95]:
```
import statsmodels.api as sm
```

[414]:
```
result4 = model4.fit_regularized(method='l1',alpha=0)
```

```
P:\Anaconda\lib\site-packages\statsmodels\discrete\discrete_model.py:1799:
RuntimeWarning: overflow encountered in exp
  return 1/(1+np.exp(-X))
```

```
Iteration limit reached    (Exit mode 9)
            Current function value: 0.570337605516455
            Iterations: 1000
```

```
            Function evaluations: 1137
            Gradient evaluations: 1000

P:\Anaconda\lib\site-packages\statsmodels\base\model.py:566: ConvergenceWarning:
Maximum Likelihood optimization failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
```

[408]:
```python
y_pred1 = result4.predict(Z)
```

[409]:
```python
m1=[]
for val in y_pred1:
    if(val>=0.5):
        m1.append(1)
    else :
        m1.append(0)
```

[410]:
```python
print(classification_report(y,m1))
```

```
              precision    recall  f1-score   support

         0.0       0.79      0.66      0.72      8006
         1.0       0.71      0.82      0.76      8006

    accuracy                           0.74     16012
   macro avg       0.75      0.74      0.74     16012
weighted avg       0.75      0.74      0.74     16012
```

[415]:
```python
result4.summary2()
```

[415]:
```
<class 'statsmodels.iolib.summary2.Summary'>
"""
                             Results: Logit
=======================================================================
Model:                 Logit               Pseudo R-squared:   0.177
Dependent Variable:    y                   AIC:                18394.4915
Date:                  2021-04-13 02:27    BIC:                18893.7626
No. Observations:      16012               Log-Likelihood:     -9132.2
Df Model:              64                  LL-Null:            -11099.
Df Residuals:          15947               LLR p-value:        0.0000
Converged:             0.0000              Scale:              1.0000
No. Iterations:        1000.0000
-----------------------------------------------------------------------
         Coef.    Std.Err.      z     P>|z|      [0.025      0.975]
-----------------------------------------------------------------------
const    -6.2880       nan       nan    nan          nan         nan
Attr1    -2.0108    0.7858   -2.5590 0.0105      -3.5508     -0.4707
Attr2     8.7757    2.7776    3.1594 0.0016       3.3317     14.2197
```

Attr3	-1.3849	0.5744	-2.4109	0.0159	-2.5107	-0.2590
Attr4	49.7684	18.6072	2.6747	0.0075	13.2989	86.2379
Attr5	0.1509	0.1757	0.8586	0.3905	-0.1935	0.4953
Attr6	0.2143	0.2483	0.8631	0.3881	-0.2724	0.7010
Attr7	7.9120	1747049.0166	0.0000	1.0000	-3424145.2398	3424161.0638
Attr8	-13.1653	26.0517	-0.5054	0.6133	-64.2258	37.8951
Attr9	-2.9062	0.3106	-9.3571	0.0000	-3.5149	-2.2974
Attr10	7.1542	2.7999	2.5552	0.0106	1.6666	12.6418
Attr11	-1.0146	0.3844	-2.6397	0.0083	-1.7679	-0.2613
Attr12	25.4278	3.6664	6.9353	0.0000	18.2418	32.6139
Attr13	-7.6382	184.8223	-0.0413	0.9670	-369.8833	354.6069
Attr14	7.8531	1951053.8468	0.0000	1.0000	-3823987.4185	3824003.1247
Attr15	-0.0611	0.0399	-1.5291	0.1262	-0.1393	0.0172
Attr16	141.6692	16.1473	8.7736	0.0000	110.0211	173.3173
Attr17	-18.6304	29.0757	-0.6408	0.5217	-75.6178	38.3570
Attr18	-17.1633	nan	nan	nan	nan	nan
Attr19	-108.2497	42.7880	-2.5299	0.0114	-192.1127	-24.3868
Attr20	-0.0238	3006.4713	-0.0000	1.0000	-5892.5993	5892.5516
Attr21	-0.0539	0.1015	-0.5310	0.5954	-0.2529	0.1451
Attr22	-6.4461	0.4868	-13.2432	0.0000	-7.4001	-5.4921
Attr23	75.9780	10.4554	7.2669	0.0000	55.4858	96.4702
Attr24	-2.0018	0.1788	-11.1949	0.0000	-2.3523	-1.6513
Attr25	-0.3553	0.4415	-0.8048	0.4209	-1.2205	0.5100
Attr26	-170.8817	15.0803	-11.3314	0.0000	-200.4387	-141.3248
Attr27	0.0041	0.0453	0.0899	0.9284	-0.0848	0.0929
Attr28	-30.8321	4.4454	-6.9356	0.0000	-39.5450	-22.1191
Attr29	-0.0153	0.0255	-0.5983	0.5496	-0.0652	0.0347
Attr30	-1.5752	0.3573	-4.4081	0.0000	-2.2756	-0.8748
Attr31	46.5983	6.9515	6.7034	0.0000	32.9737	60.2229
Attr32	0.9052	2.9772	0.3040	0.7611	-4.9301	6.7405
Attr33	11.1885	0.7915	14.1363	0.0000	9.6372	12.7397
Attr34	5.9254	0.7071	8.3797	0.0000	4.5395	7.3113
Attr35	-0.2347	0.1676	-1.4000	0.1615	-0.5632	0.0939
Attr36	1.0832	0.1020	10.6232	0.0000	0.8834	1.2831
Attr37	-0.4836	0.2180	-2.2187	0.0265	-0.9109	-0.0564
Attr38	-17.8808	1.8369	-9.7344	0.0000	-21.4810	-14.2806
Attr39	0.2774	0.7427	0.3735	0.7088	-1.1783	1.7331
Attr40	3.4578	0.8758	3.9481	0.0001	1.7412	5.1743
Attr41	-191.1649	19.3115	-9.8990	0.0000	-229.0149	-153.3150
Attr42	-10.7759	39.6925	-0.2715	0.7860	-88.5718	67.0201
Attr43	2.0966	14283.0966	0.0001	0.9999	-27992.2584	27996.4516
Attr44	-2.7757	13153.5409	-0.0002	0.9998	-25783.2422	25777.6907
Attr45	0.2266	0.4172	0.5431	0.5871	-0.5912	1.0444
Attr46	-312.6484	18.6926	-16.7258	0.0000	-349.2852	-276.0116
Attr47	-10.2906	4.9588	-2.0752	0.0380	-20.0097	-0.5715
Attr48	7.6186	0.4212	18.0886	0.0000	6.7931	8.4441
Attr49	-2.1764	43.7659	-0.0497	0.9603	-87.9561	83.6032

```
Attr50   124.7501      15.7368   7.9273 0.0000      93.9066     155.5936
Attr51   -20.8773       1.7871 -11.6820 0.0000     -24.3800     -17.3746
Attr52    -4.4232      17.7036  -0.2498 0.8027     -39.1216      30.2751
Attr53     1.3578       0.8534   1.5911 0.1116      -0.3148       3.0304
Attr54    29.1595       4.5182   6.4538 0.0000      20.3040      38.0149
Attr55     0.0124       0.0268   0.4630 0.6434      -0.0401       0.0649
Attr56    -0.0840       1.1595  -0.0724 0.9423      -2.3566       2.1887
Attr57    -0.6349       0.1594  -3.9827 0.0001      -0.9473      -0.3224
Attr58    -9.8957      35.1337  -0.2817 0.7782     -78.7565      58.9651
Attr59    -0.9183       0.3409  -2.6938 0.0071      -1.5864      -0.2502
Attr60    -0.9737       0.7951  -1.2245 0.2208      -2.5321       0.5848
Attr61    -0.4455       0.1093  -4.0754 0.0000      -0.6598      -0.2313
Attr62     1.6637       0.6370   2.6116 0.0090       0.4151       2.9123
Attr63   -14.3319       0.8433 -16.9957 0.0000     -15.9846     -12.6791
Attr64    -0.1429       0.0782  -1.8277 0.0676      -0.2962       0.0103
=========================================================================

"""
```

[349]:
```python
model = LogisticRegression(solver='liblinear', random_state=0)
model.fit(X3, y)
```

[349]:
```
LogisticRegression(random_state=0, solver='liblinear')
```

[350]:
```python
p_pred = model.predict_proba(X3)
y_pred = model.predict(X3)
score_ = model.score(X3, y)
conf_m = confusion_matrix(y, y_pred)
```

[348]:
```python
print(model.param_)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-348-fd01b116a93c> in <module>
----> 1 print(model.param_)

AttributeError: 'LogisticRegression' object has no attribute 'param_'
```

[351]:
```python
print(classification_report(y, model.predict(X3)))
```

```
              precision    recall  f1-score   support

         0.0       0.75      0.62      0.68      8006
         1.0       0.68      0.79      0.73      8006

    accuracy                           0.71     16012
   macro avg       0.71      0.71      0.70     16012
```

```
weighted avg       0.71      0.71      0.70     16012
```

[57]:
```python
Z2=np.array(add_constant(df1_train_res))
```

[58]:
```python
Z2
```

[58]:
```
array([[ 1.00000000e+00,  9.13637728e-03,  4.40604351e-02, …,
        -2.56742407e-02, -1.69388740e-01, -4.67664800e-02],
       [ 1.00000000e+00, -3.02844201e-01,  1.58568438e-01, …,
         9.41009735e-04, -2.61286490e-01,  1.04450289e-01],
       [ 1.00000000e+00, -4.48059907e-02,  6.93614604e-02, …,
        -9.39296231e-03, -2.45782021e-01, -1.45081613e-02],
       …,
       [ 1.00000000e+00,  6.14630324e-02, -1.02202244e-01, …,
        -3.51686031e-02,  2.55546958e-01, -7.26691337e-02],
       [ 1.00000000e+00, -6.03577171e-02,  2.64767971e-02, …,
        -9.37237552e-03, -2.43126446e-01, -8.07685475e-02],
       [ 1.00000000e+00, -1.52978845e-01,  1.81188709e-02, …,
        -3.12504421e-02, -5.98029082e-02, -6.19089914e-02]])
```

[63]:
```python
model1 = LogisticRegression(solver='liblinear', penalty='l1', random_state=0)
model1.fit(Z2, y)
```

[63]:
```
LogisticRegression(penalty='l1', random_state=0, solver='liblinear')
```

[64]:
```python
print(classification_report(y, model1.predict(Z2)))
```

```
              precision    recall  f1-score   support

         0.0       0.76      0.65      0.70      8006
         1.0       0.69      0.79      0.74      8006

    accuracy                           0.72     16012
   macro avg       0.73      0.72      0.72     16012
weighted avg       0.73      0.72      0.72     16012
```

[30]:
```python
d=df1_train.dropna()
```

[53]:
```python
Z3=np.array(add_constant(knn_imputed_df.iloc[:,:-1]))
```

[35]:
```python

```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 3891 entries, 9126 to 5744
Data columns (total 65 columns):
 #   Column  Non-Null Count  Dtype
```

```
---  ------  --------------  -----
 0   Attr1   3891 non-null   float64
 1   Attr2   3891 non-null   float64
 2   Attr3   3891 non-null   float64
 3   Attr4   3891 non-null   float64
 4   Attr5   3891 non-null   float64
 5   Attr6   3891 non-null   float64
 6   Attr7   3891 non-null   float64
 7   Attr8   3891 non-null   float64
 8   Attr9   3891 non-null   float64
 9   Attr10  3891 non-null   float64
 10  Attr11  3891 non-null   float64
 11  Attr12  3891 non-null   float64
 12  Attr13  3891 non-null   float64
 13  Attr14  3891 non-null   float64
 14  Attr15  3891 non-null   float64
 15  Attr16  3891 non-null   float64
 16  Attr17  3891 non-null   float64
 17  Attr18  3891 non-null   float64
 18  Attr19  3891 non-null   float64
 19  Attr20  3891 non-null   float64
 20  Attr21  3891 non-null   float64
 21  Attr22  3891 non-null   float64
 22  Attr23  3891 non-null   float64
 23  Attr24  3891 non-null   float64
 24  Attr25  3891 non-null   float64
 25  Attr26  3891 non-null   float64
 26  Attr27  3891 non-null   float64
 27  Attr28  3891 non-null   float64
 28  Attr29  3891 non-null   float64
 29  Attr30  3891 non-null   float64
 30  Attr31  3891 non-null   float64
 31  Attr32  3891 non-null   float64
 32  Attr33  3891 non-null   float64
 33  Attr34  3891 non-null   float64
 34  Attr35  3891 non-null   float64
 35  Attr36  3891 non-null   float64
 36  Attr37  3891 non-null   float64
 37  Attr38  3891 non-null   float64
 38  Attr39  3891 non-null   float64
 39  Attr40  3891 non-null   float64
 40  Attr41  3891 non-null   float64
 41  Attr42  3891 non-null   float64
 42  Attr43  3891 non-null   float64
 43  Attr44  3891 non-null   float64
 44  Attr45  3891 non-null   float64
 45  Attr46  3891 non-null   float64
 46  Attr47  3891 non-null   float64
```

```
 47  Attr48  3891 non-null   float64
 48  Attr49  3891 non-null   float64
 49  Attr50  3891 non-null   float64
 50  Attr51  3891 non-null   float64
 51  Attr52  3891 non-null   float64
 52  Attr53  3891 non-null   float64
 53  Attr54  3891 non-null   float64
 54  Attr55  3891 non-null   float64
 55  Attr56  3891 non-null   float64
 56  Attr57  3891 non-null   float64
 57  Attr58  3891 non-null   float64
 58  Attr59  3891 non-null   float64
 59  Attr60  3891 non-null   float64
 60  Attr61  3891 non-null   float64
 61  Attr62  3891 non-null   float64
 62  Attr63  3891 non-null   float64
 63  Attr64  3891 non-null   float64
 64  class   3891 non-null   int64
dtypes: float64(64), int64(1)
memory usage: 2.0 MB
```

[56]:
```
model2 = LogisticRegression(solver='liblinear', random_state=0)
model2.fit(Z3, np.array(knn_imputed_df.iloc[:,-1]))
```

[56]:
```
LogisticRegression(random_state=0, solver='liblinear')
```

[57]:
```
print(classification_report(np.array(knn_imputed_df.iloc[:,-1]), model2.
 ↪predict(Z3)))
```

```
              precision    recall  f1-score   support

         0.0       0.95      1.00      0.98      8006
         1.0       0.33      0.01      0.01       396

    accuracy                           0.95      8402
   macro avg       0.64      0.50      0.50      8402
weighted avg       0.92      0.95      0.93      8402
```

[58]:
```
sm = SMOTE(random_state = 3)
e, f = sm.fit_sample(d.iloc[:,:-1], d.iloc[:,-1])
```

[38]:
```
model3 = LogisticRegression(solver='liblinear', penalty='l1', random_state=0)
model3.fit(np.array(add_constant(e)), np.array(f))
```

```
P:\Anaconda\lib\site-packages\sklearn\svm\_base.py:976: ConvergenceWarning:
Liblinear failed to converge, increase the number of iterations.
  warnings.warn("Liblinear failed to converge, increase "
```

[38]: LogisticRegression(penalty='l1', random_state=0, solver='liblinear')

[42]:
```
print(classification_report(np.array(f), model2.predict(np.
 ↪array(add_constant(e)))))
```

```
              precision    recall  f1-score   support

           0       0.51      1.00      0.68      3801
           1       0.98      0.05      0.10      3801

    accuracy                           0.52      7602
   macro avg       0.75      0.52      0.39      7602
weighted avg       0.75      0.52      0.39      7602
```

[ ]: