Ass_3_q5

April 5, 2021

[6]:
```
#import all the necessary libararies
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import math
from sympy import Matrix, solve, symbols
import sympy as sy
import seaborn as sns
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
```

[7]:
```
#read the db
df=pd.read_csv('face.csv')
df.shape
```

[7]:
```
(400, 4097)
```

[8]:
```
#all the classes(including redundant ones)
target=df["target"].values
class_arr=df["target"].unique()
target
```

[8]:
```
array([ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  2,  2,  2,  2,  2,  2,  2,  2,  2,  2,  3,  3,  3,  3,
        3,  3,  3,  3,  3,  3,  4,  4,  4,  4,  4,  4,  4,  4,  4,  4,  5,
        5,  5,  5,  5,  5,  5,  5,  5,  5,  6,  6,  6,  6,  6,  6,  6,  6,
        6,  6,  7,  7,  7,  7,  7,  7,  7,  7,  7,  7,  8,  8,  8,  8,  8,
        8,  8,  8,  8,  8,  9,  9,  9,  9,  9,  9,  9,  9,  9,  9, 10, 10,
       10, 10, 10, 10, 10, 10, 10, 10, 11, 11, 11, 11, 11, 11, 11, 11, 11,
       11, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 13, 13, 13, 13, 13, 13,
       13, 13, 13, 13, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 15, 15, 15,
       15, 15, 15, 15, 15, 15, 15, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16,
       17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 18, 18, 18, 18, 18, 18, 18,
       18, 18, 18, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 20, 20, 20, 20,
       20, 20, 20, 20, 20, 20, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 22,
       22, 22, 22, 22, 22, 22, 22, 22, 22, 23, 23, 23, 23, 23, 23, 23, 23,
       23, 23, 24, 24, 24, 24, 24, 24, 24, 24, 24, 24, 25, 25, 25, 25, 25,
       25, 25, 25, 25, 25, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 27, 27,
       27, 27, 27, 27, 27, 27, 27, 27, 28, 28, 28, 28, 28, 28, 28, 28, 28,
       28, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 30, 30, 30, 30, 30, 30,
       30, 30, 30, 30, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 32, 32, 32,
       32, 32, 32, 32, 32, 32, 32, 33, 33, 33, 33, 33, 33, 33, 33, 33, 33,
       34, 34, 34, 34, 34, 34, 34, 34, 34, 34, 35, 35, 35, 35, 35, 35, 35,
       35, 35, 35, 36, 36, 36, 36, 36, 36, 36, 36, 36, 36, 37, 37, 37, 37,
       37, 37, 37, 37, 37, 37, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 39,
       39, 39, 39, 39, 39, 39, 39, 39, 39])
```

[9]:

```
#all unique classes
class_arr
```

[9]:

```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
       17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
       34, 35, 36, 37, 38, 39])
```

[10]:

```
#number of feature vectors[dim]
num_features=df.shape[1]-1
num_features
```

[10]:

```
4096
```

[11]:

```
data=df.drop("target",axis=1).values
data.shape
```

[11]:

```
(400, 4096)
```

[12]:

```
#calculate mean
tot_mean=data.mean(axis=0)
tot_mean
```

[12]:

```
array([0.4001343 , 0.43423554, 0.47628099, …, 0.32141529, 0.31364669,
       0.31045455])
```

[13]:

```
#num of dims
n_dim=data.shape[1]
n_dim
```

[13]:

```
4096
```

1 Within class scatter and between class scatter

[14]:

```
sw=np.zeros((n_dim,n_dim))
sb=np.zeros((n_dim,n_dim))
for i in class_arr:
    x=df[df["target"] == i]
    x=x.drop("target",axis=1)
    n_dp=x.shape[0]
    mean=x.mean(axis=0)
    sw+=np.transpose(x-mean) @ (x-mean)
    sb+=n_dp*(np.transpose(tot_mean-mean) @ (tot_mean-mean))
```

[37]:

```
#calculate the eigen vectors and the values
eigen_vectors,eigen_values,_=np.linalg.svd(np.dot(np.linalg.inv(sw),sb))
```

[39]:

```
eigen_vectors.shape;eigen_values.shape
```

[39]:

```
(4096,)
```

[40]:

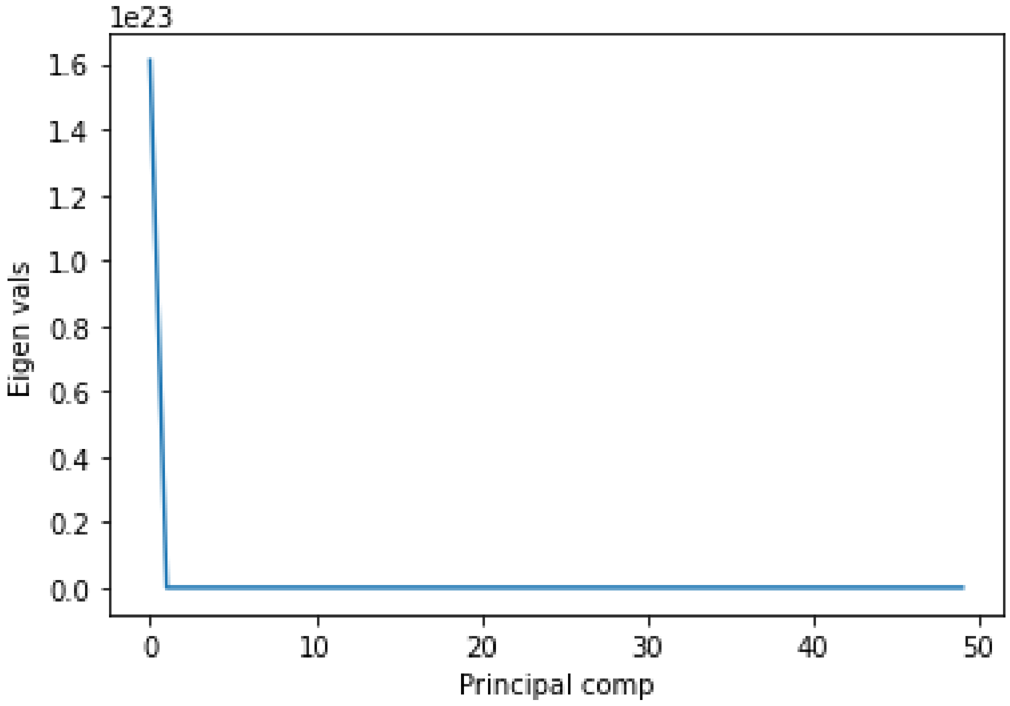
```
exp_var=eigen_values.cumsum()/np.sum(eigen_values)
```

[41]:

```
plt.plot(eigen_values[:50])
plt.xlabel("Principal comp")
plt.ylabel("Eigen vals")
```

[41]:

```
Text(0, 0.5, 'Eigen vals')
```



[42]:

```
reduced_data=np.dot(data,eigen_vectors[:,:])
```

2 LDA for dataset

```
[43]: lda=LinearDiscriminantAnalysis(n_components=39)
```

```
[22]: X_lda=lda.fit_transform(data,target)
```

```
[23]: X_lda.shape
```

```
[23]: (400, 39)
```

```
[26]: eigen_vectors[:,:39].shape
```

```
[26]: (39,)
```

```
[27]: X_lda
```

```
[27]: array([[-0.62777794,  3.82875022,  1.4675847 , …, -3.25503111,
         -0.27431701,  1.25042126],
        [-2.22147119,  3.29157602,  2.50522997, …, -2.38164033,
         -2.80681649,  1.72181352],
        [-1.64522877,  3.80214848,  2.65051006, …, -1.46811339,
          1.44905459,  1.41007756],
        …,
        [ 2.29939628,  8.02357838,  0.69362997, …, -3.91929104,
         -4.52063252, -2.15401061],
        [ 0.16842079,  7.38778541,  0.33385385, …, -2.02113982,
         -0.87457498, -1.16563861],
        [ 2.0934787 ,  8.33510906, -0.25304638, …, -3.19873545,
         -2.04304882, -0.29292432]])
```

```
[28]: reduced_data
```

```
[28]: array([-9.58888564e+18-9.60891400e+04j, -1.40636992e+19+1.60016331e+03j,
        -9.84458855e+18-4.74445278e+05j, -6.13688649e+18-1.61425802e+06j,
        -1.54700686e+19+3.00796918e+04j, -1.70042910e+19-5.72770190e+05j,
        -1.02281441e+19-4.72895238e+05j, -3.96340589e+18-9.61596564e+05j,
        -7.54325650e+18-7.76132632e+05j, -1.17623659e+19-6.96292100e+05j,
        -1.67485862e+19-9.15818799e+04j, -1.99448819e+19+4.38955251e+05j,
        -2.03284375e+19-1.04312973e+05j, -1.94334754e+19+7.04810804e+04j,
        -1.96891771e+19-1.73219947e+05j, -2.07119931e+19+5.49099223e+04j,
        -1.94334754e+19-2.59517814e+05j, -1.81549562e+19-3.21907056e+05j,
        -2.10955487e+19+7.85915004e+04j, -1.75156974e+19-8.60698254e+05j,
        -1.78992530e+19+8.78896496e+03j, -1.94334754e+19+1.38937112e+05j,
        -1.95613263e+19+1.56060523e+05j, -6.13688649e+18-1.60020515e+06j,
        -4.85836886e+18-1.89709092e+06j, -5.88118296e+18-7.25357088e+05j,
        -5.24192445e+18-4.29259860e+05j, -5.36977591e+18-1.05882041e+06j,
        -1.61093290e+19-2.13744612e+05j, -1.67485862e+19-2.32276185e+05j,
        -5.24192445e+18-5.87274000e+05j, -1.73878457e+19-2.22222800e+05j,
        -8.94962651e+18-2.08703009e+06j, -1.57257734e+19-2.16076572e+05j,
        -2.07119931e+19+5.07258093e+04j, -1.18902180e+19-1.25610619e+06j,
        -5.75333120e+18-4.76985580e+05j, -3.19629532e+18-7.78552497e+05j,
        -2.23740663e+19-2.22822430e+05j, -2.39082871e+19+1.24065117e+05j,
        -1.40636992e+19-1.82729982e+05j, -2.09676978e+19+1.32397903e+05j,
        -2.04562883e+19+1.47295037e+04j, -1.32965874e+19-2.30103887e+05j,
        -1.57257734e+19-5.13366672e+05j, -1.04838489e+19-4.19591446e+05j,
        -1.62371798e+19-4.51253243e+04j, -1.61093290e+19-4.74248936e+05j,
        -1.62371798e+19-7.76453980e+04j, -1.01002933e+19-8.29282689e+05j,
        -2.22462145e+19-1.30675878e+05j, -2.12233995e+19-3.12047850e+05j,
        -2.19905107e+19+7.85295122e+04j, -1.96891771e+19-6.80642350e+05j,
        -1.02281441e+19-8.61521254e+05j, -2.03284375e+19-5.28119366e+05j,
        -8.18251532e+18-8.63904973e+05j, -2.23740663e+19-5.44190900e+05j,
        -7.92681210e+18-1.50843547e+06j, -1.71321418e+19-6.24274870e+05j,
        -6.39259032e+18-1.88005986e+05j, -9.46103357e+18+2.59445598e+04j,
        -6.77614592e+18-5.85964640e+05j, -5.49762767e+18-9.32817856e+05j,
        -7.92681210e+18-1.30237985e+05j, -6.52044209e+18-1.63847072e+06j,
        -7.54325650e+18-2.97983410e+05j, -6.77614592e+18-8.51624195e+05j,
        -7.15970121e+18-1.15338000e+06j, -9.33318149e+18-7.84140845e+05j,
        -5.75333120e+18-3.90131216e+03j, -6.00903503e+18+7.35458096e+04j,
        -1.21459212e+19+4.18979088e+04j, -2.49311015e+19+7.23468196e+05j,
        -8.05466448e+18-1.25393513e+05j, -1.08674039e+19-1.43028401e+04j,
        -2.32690296e+19+6.52904407e+05j, -6.26473825e+18-2.44063907e+05j,
        -6.90399768e+18-2.10240118e+06j, -2.30133251e+19+3.78276647e+05j,
        -1.54700686e+19-2.17765786e+05j, -1.63650306e+19-4.21235263e+05j,
        -1.61093290e+19-3.10183614e+05j, -1.15066627e+19-7.10700327e+05j,
        -1.61093290e+19-5.74580992e+05j, -4.09125766e+18-6.27705508e+05j,
        -1.07395518e+19-1.17768111e+06j, -6.64829385e+18-1.10000379e+06j,
        -1.32965874e+19-4.46205540e+05j, -1.68764389e+19-4.97611973e+05j,
        -1.01002933e+19-1.87095336e+05j, -9.07747828e+18-4.17688822e+04j,
        -1.76435498e+19+3.54792480e+05j, -2.02005866e+19+1.94904100e+05j,
        -7.67110826e+18-7.29150396e+05j, -1.54700686e+19-3.40698572e+05j,
        -1.58536242e+19-2.06422612e+05j, -1.20180701e+19-5.75332608e+05j,
        -3.06844324e+18-6.40032521e+05j, -1.24016254e+19-3.36447708e+05j,
        -8.94962651e+18+6.18889347e+04j, -5.75333120e+18+4.60020467e+05j,
        -1.01002933e+19-1.47176483e+03j, -5.24192445e+18+1.70766324e+05j,
        -1.06116997e+19+3.61534038e+05j, -5.75333120e+18+3.03963972e+05j,
        -8.05466448e+18+4.24281083e+05j, -1.63650306e+19+4.39000873e+05j,
        -2.41639913e+19+2.55918564e+05j, -7.92681210e+18+3.53375996e+05j,
        -1.49586612e+19-3.83021132e+05j, -1.59814750e+19-2.47523781e+05j,
        -1.09952553e+19-5.83407205e+05j, -1.89220671e+19-3.04394805e+05j,
        -1.47029574e+19-3.14020275e+05j, -1.27851810e+19-9.40929405e+05j,
        -1.66207354e+19-4.71914988e+05j, -1.72599942e+19-1.48013688e+05j,
        -1.84106604e+19-1.59855161e+05j, -1.58536242e+19-3.47134395e+05j,
        -1.15066627e+19-4.21521006e+05j, -1.71321418e+19-5.73608849e+05j,
        -8.69392268e+18-1.08927901e+06j, -8.31036739e+18-8.56469523e+05j,
        -1.54700686e+19-3.17210081e+05j, -2.03284375e+19+6.51568281e+04j,
        -8.82177475e+18-1.67312334e+06j, -5.75333120e+18-6.91429223e+05j,
        -1.68764389e+19-5.06711557e+05j, -1.40636992e+19-7.21442926e+05j,
        -1.95613263e+19-5.15093617e+05j, -1.90499183e+19-4.02823309e+04j,
        -2.07119931e+19+3.18446896e+05j, -1.18902180e+19-5.97808712e+05j,
        -1.36801436e+19-5.33829548e+05j, -2.00727327e+19+1.33739392e+05j,
        -1.17623659e+19-4.98503792e+05j, -1.24016254e+19-3.79793724e+05j,
        -1.94334754e+19+3.07856231e+05j, -1.26573286e+19-3.01844447e+05j,
        -1.86663630e+19+1.76677858e+05j, -1.71321418e+19-2.44017583e+05j,
        -1.63650306e+19-4.15278906e+05j, -1.55979194e+19-4.35619895e+05j,
        -1.63650306e+19-1.49938141e+05j, -1.53422165e+19-5.22716236e+05j,
        -1.68764389e+19-2.83194774e+04j, -1.66207354e+19-7.41883066e+04j,
        -1.61093290e+19-2.68545613e+05j, -1.75156974e+19-3.53488411e+05j,
        -1.95613263e+19+1.64027998e+05j, -1.50865130e+19-3.87296079e+05j,
        -1.68764389e+19-4.18086074e+05j, -1.82828074e+19-5.30603070e+05j,
        -1.81549562e+19-4.14479033e+05j, -1.34244398e+19-1.07763952e+05j,
        -1.93056224e+19+3.72007098e+05j, -8.05466448e+18-3.17084958e+05j,
        -7.41540505e+18-3.72676533e+05j, -1.68764389e+19-2.49460367e+05j,
        -1.34244398e+19-1.06287167e+06j, -1.29130321e+19-2.27236064e+05j,
        -1.09952553e+19-4.41594563e+05j, -1.72599942e+19-3.90544145e+05j,
        -1.47029574e+19-7.60116843e+05j, -6.13688649e+18-9.97148440e+05j,
        -4.73051678e+18-1.98780108e+06j, -7.28755297e+18-8.56679236e+05j,
        -5.49762767e+18-6.26047905e+05j, -7.54325650e+18-1.09157572e+06j,
        -1.62371798e+19-5.65720533e+05j, -1.70042910e+19-1.21533517e+05j,
        -1.22737730e+19-1.48203016e+06j, -1.62371798e+19-8.01889498e+05j,
        -1.45751056e+19-6.17997863e+05j, -1.43194024e+19-1.17207103e+06j,
        -1.32965874e+19-8.33210265e+05j, -1.36801436e+19-3.06164489e+05j,
        -1.30408842e+19-9.59941700e+05j, -1.12509592e+19-1.13182068e+06j,
        -7.79895972e+18-1.51039359e+06j, -7.41540505e+18-4.32450930e+05j,
        -7.79895972e+18-2.52040230e+05j, -1.22737730e+19+2.19120457e+05j,
        -1.04838489e+19+2.92007087e+05j, -7.79895972e+18-6.17734756e+05j,
        -1.39358471e+19+3.67521125e+05j, -1.20180701e+19+3.40564102e+05j,
        -1.12509592e+19+4.16908570e+05j, -7.67110826e+18-4.02106027e+04j,
        -1.48308098e+19+4.41331399e+04j, -1.30408842e+19+1.29850640e+05j,
        -1.32965874e+19+1.72060615e+05j, -4.85836886e+18+6.20050553e+05j,
        -9.84458855e+18-5.91077544e+05j, -6.26473825e+18+6.71314292e+05j,
        -8.69392268e+18-6.07043524e+05j, -7.79895972e+18-6.53130695e+05j,
        -7.41540505e+18-1.04428236e+06j, -2.07119931e+19+3.69656917e+05j,
        -1.59814750e+19+6.65849074e+04j, -7.28755297e+18-4.14467970e+05j,
        -9.97244093e+18-5.05427290e+05j, -1.44472545e+19-4.08558465e+05j,
        -1.54700686e+19-6.40067078e+05j, -1.57257734e+19-2.18157351e+05j,
        -7.03184945e+18-7.96157635e+05j, -8.82177475e+18-9.52581051e+05j,
        -1.47029574e+19-1.39384947e+05j, -6.13688649e+18-1.00238748e+06j,
        -4.21910973e+18-8.97373923e+04j, -4.60266502e+18-6.33712224e+05j,
        -6.39259032e+18-2.60760847e+05j, -4.09125766e+18-1.45464382e+05j,
        -4.73051678e+18-4.75571767e+04j, -6.39259032e+18-1.14331525e+06j,
        -5.62547974e+18-8.44078300e+04j, -7.28755297e+18-1.10750522e+05j,
        -4.73051678e+18-7.31183719e+05j, -4.98622062e+18-4.95564712e+05j,
        -1.26573286e+19-1.47942047e+05j, -2.28854740e+19+3.95182986e+04j,
        -2.02005866e+19+1.91611138e+05j, -3.96340589e+18-2.58837650e+05j,
        -2.19905107e+19+3.64713860e+05j, -4.09125766e+18-5.63724930e+05j,
        -1.68764389e+19-2.84166719e+05j, -2.30133251e+19-4.56956465e+04j,
        -1.40636992e+19-5.91248452e+05j, -5.36977591e+18-9.00194636e+05j,
        -1.34244398e+19+5.16614003e+03j, -1.38079954e+19+2.86296884e+05j,
        -3.45199884e+18-3.68974831e+05j, -4.21910973e+18-8.25953776e+05j,
        -1.13788109e+19+1.06905653e+05j, -6.64829385e+18-6.03632318e+04j,
        -6.90399768e+18-5.32841250e+05j, -1.73878457e+19-3.96696051e+05j,
        -4.47481326e+18-3.75901153e+05j, -4.98622062e+18-3.56100173e+05j,
        -1.72599942e+19-2.06787410e+05j, -2.45475459e+19+3.70616637e+05j,
        -1.95613263e+19-1.51252004e+05j, -5.24192445e+18-1.09693878e+05j,
        -7.03184945e+18-8.18202614e+05j, -2.08398439e+19+1.74476898e+05j,
        -1.16345148e+19-4.60046250e+05j, -5.88118296e+18-8.64033526e+05j,
        -1.09952553e+19-1.17678463e+05j, -1.98170310e+19+1.88486421e+04j,
        -1.90499183e+19-1.84919265e+05j, -8.82177475e+18-5.63497906e+05j,
        -1.87942151e+19-2.72649517e+04j, -2.41639913e+19+1.89934418e+05j,
        -2.35247318e+19+2.25422748e+04j, -1.41915503e+19-4.25024870e+05j,
        -1.94334754e+19-5.22187478e+05j, -2.35247318e+19-8.94439703e+04j,
        -1.91777707e+19+6.95681516e+04j, -2.32690296e+19-5.12424291e+04j,
        -1.29130321e+19-8.43946910e+04j, -1.20180701e+19+3.66500598e+05j,
        -1.32965874e+19-3.47130341e+04j, -1.36801436e+19-8.90215501e+04j,
        -2.04562883e+19+1.91575140e+05j, -1.34244398e+19+2.68801157e+05j,
        -1.03559965e+19-1.35775473e+04j, -1.40636992e+19+4.06731437e+04j,
        -2.05841422e+19+2.48766385e+05j, -1.18902180e+19+3.01599019e+05j,
        -1.89220671e+19-2.89091764e+05j, -1.84106604e+19-9.10760324e+05j,
        -1.21459212e+19-3.94898010e+05j, -7.54325650e+18+1.76935109e+05j,
        -2.02005866e+19+1.62875000e+05j, -1.86663630e+19-2.72285794e+05j,
        -2.18626598e+19+3.79229699e+05j, -2.22462145e+19-5.10440699e+04j,
        -2.37804360e+19-3.01321106e+05j, -1.12509592e+19-8.16241820e+05j,
        -5.75333120e+18-6.72786620e+05j, -6.77614592e+18-1.90798037e+05j,
        -9.71673771e+18-3.28724297e+05j, -1.02281441e+19-3.44238794e+05j,
        -1.09952553e+19-1.50634538e+06j, -1.07395518e+19-8.36210715e+05j,
        -8.69392268e+18-9.13615799e+05j, -1.03559965e+19-7.01560458e+05j,
        -7.79895972e+18+1.01584031e+05j, -5.49762767e+18-4.41328596e+05j,
        -1.35522915e+19-2.24112287e+05j, -1.34244398e+19+1.15243276e+05j,
        -1.49586612e+19+4.77761071e+04j, -1.18902180e+19-2.38431642e+05j,
        -1.40636992e+19-5.47209050e+05j, -9.97244093e+18-4.08277977e+05j,
        -9.84458855e+18+1.38967041e+04j, -7.28755297e+18+1.12213293e+05j,
        -1.32965874e+19-2.10294315e+05j, -1.25294768e+19+1.80524787e+04j,
        -1.45751056e+19-5.29555555e+05j, -2.16069551e+19+5.09487415e+05j,
        -2.19905107e+19+5.00625484e+05j, -1.89220671e+19+7.43437997e+05j,
        -2.17348078e+19+6.29286081e+05j, -1.30408842e+19-5.04998546e+05j,
        -1.48308098e+19-1.92265930e+05j, -1.50865130e+19-5.00683780e+05j,
        -1.49586612e+19-3.72112795e+05j, -2.37804360e+19+4.57626756e+03j,
        -1.18902180e+19+1.28685780e+06j, -1.53422165e+19-9.04792610e+05j,
        -1.57257734e+19+2.93031379e+05j, -1.66207354e+19-6.63212792e+04j,
        -1.21459212e+19-1.66089758e+04j, -1.82828074e+19+8.98614345e+05j,
        -2.39082871e+19+1.34761277e+05j, -1.59814750e+19-5.81943164e+04j,
        -1.94334754e+19+2.72149328e+05j, -1.45751056e+19-1.06527374e+05j,
        -3.83555413e+18-5.62854095e+05j, -4.21910973e+18-2.62805144e+05j,
        -4.21910973e+18-4.05351857e+05j, -4.47481326e+18-8.02415090e+05j,
        -3.83555413e+18-2.60347378e+05j, -4.21910973e+18-6.35352729e+05j,
        -4.47481326e+18-6.95762762e+05j, -3.83555413e+18-7.22771631e+05j,
        -3.70770237e+18-6.00292620e+05j, -4.21910973e+18-5.61512344e+05j,
        -1.38079954e+19+4.59122520e+05j, -1.34244398e+19-3.94919426e+05j,
        -1.31687362e+19-9.26560969e+05j, -6.39259032e+18-5.70137599e+05j,
        -1.49586612e+19-9.17062742e+05j, -1.52143654e+19+5.96402720e+05j,
        -1.44472545e+19-8.17504552e+05j, -1.27851810e+19-7.42790262e+05j,
        -1.31687362e+19-5.71617730e+04j, -1.30408842e+19-2.24896865e+05j,
        -9.33318149e+18-7.22373159e+05j, -1.15066627e+19-6.69511230e+05j,
        -1.16345148e+19-4.49816678e+05j, -1.16345148e+19-4.79961487e+05j,
        -1.01002933e+19-1.98458615e+06j, -1.16345148e+19-6.46702466e+05j,
        -1.29130321e+19-7.53363142e+05j, -1.02281441e+19-1.04593483e+06j,
        -8.82177475e+18-1.78979377e+06j, -9.07747828e+18-1.39162553e+06j,
        -7.28755297e+18-2.16455423e+04j, -3.06844324e+18-7.90316060e+05j,
        -6.00903503e+18-6.62904460e+05j, -8.56607091e+18-5.43918337e+05j,
        -7.41540505e+18-2.60965268e+05j, -4.21910973e+18-8.73436288e+05j,
        -2.81273972e+18-1.10318714e+06j, -1.40636992e+19-4.33660015e+05j,
        -1.58536242e+19+6.41767090e+04j, -6.64829385e+18-4.99285431e+05j,
        -8.69392268e+18-3.24508272e+05j, -1.06116997e+19-2.62861120e+05j,
        -6.77614592e+18-4.81337692e+05j, -1.02281441e+19+3.29163751e+05j,
        -6.77614592e+18-1.01965294e+05j, -1.17623659e+19+1.09896164e+05j,
        -9.71673771e+18-3.98950489e+05j, -1.50865130e+19+1.26531532e+06j,
        -6.13688649e+18-2.87241982e+05j, -8.94962651e+18-1.29781366e+06j,
        -3.83555413e+18-2.23803756e+06j, -4.47481326e+18-4.31900929e+05j,
        -3.83555413e+18-3.22815836e+05j, -4.21910973e+18-4.69009834e+05j,
        -3.57985061e+18-6.60701143e+05j, -2.68488795e+18-1.41474929e+06j,
        -5.11407207e+18-2.68923812e+05j, -8.82177475e+18-5.73673643e+04j,
        -4.21910973e+18-1.78376191e+06j, -3.19629532e+18-7.92550560e+05j,
        -7.79895972e+18-3.49864250e+05j, -1.09952553e+19+9.41807562e+03j,
        -9.33318149e+18-1.09841903e+06j, -4.60266502e+18-4.13169137e+05j,
        -1.01002933e+19-9.75480592e+05j, -5.11407207e+18-1.46130664e+05j,
        -8.69392268e+18-3.67441047e+05j, -1.13788109e+19-1.08438728e+05j,
        -9.46103357e+18-4.41375783e+05j, -4.60266502e+18-6.58900755e+05j,
        -1.68764389e+19-4.19349578e+05j, -1.03559965e+19-1.12521498e+06j,
        -1.40636992e+19-5.16694406e+05j, -1.59814750e+19-1.47284499e+06j,
        -8.94962651e+18-9.24273459e+05j, -1.24016254e+19-2.90263463e+05j,
        -1.13788109e+19-1.71421844e+06j, -1.54700686e+19-4.41397719e+05j,
        -6.64829385e+18-1.78924045e+06j, -1.59814750e+19-1.83132058e+06j])
```

3 Classifying and finding the error

[44]:

```
from sklearn.model_selection import  train_test_split
```

[45]:

```
X_tr , X_test , Y_tr ,Y_test = train_test_split(reduced_data,target,test_size=0.
↪1 , random_state = 0)
```

[46]:

```
from sklearn.naive_bayes import  GaussianNB
```

[47]:

```
gnb=GaussianNB()
```

[48]:

```
y_pred=gnb.fit(X_tr,Y_tr).predict(X_test)
y_pred
```

[48]:

```
array([13, 30, 34, 19, 24,  6, 15, 26, 14, 21,  3, 13, 11, 34,  1,  5, 29,
       14, 20, 19,  2, 26, 12, 34, 17, 31,  7,  1, 28, 10, 17, 30, 33, 22,
        6,  7,  9,  2, 17, 39])
```

[49]:

```
Y_test
```

[49]:

```
array([13, 30, 34, 19, 24,  6, 15, 26, 14, 21,  3, 13, 11, 34,  1,  5, 29,
       14, 20, 19, 17, 26, 12, 34, 17, 31,  7,  1, 28, 10, 17, 30, 33, 22,
        6,  7,  9, 17, 17, 39])
```

[50]:

```
#calculating the error
error=0
for i in range(0,len(y_pred)):
    if y_pred[i] != Y_test[i]:
        error+=1
error/len(y_pred)
```

[50]:

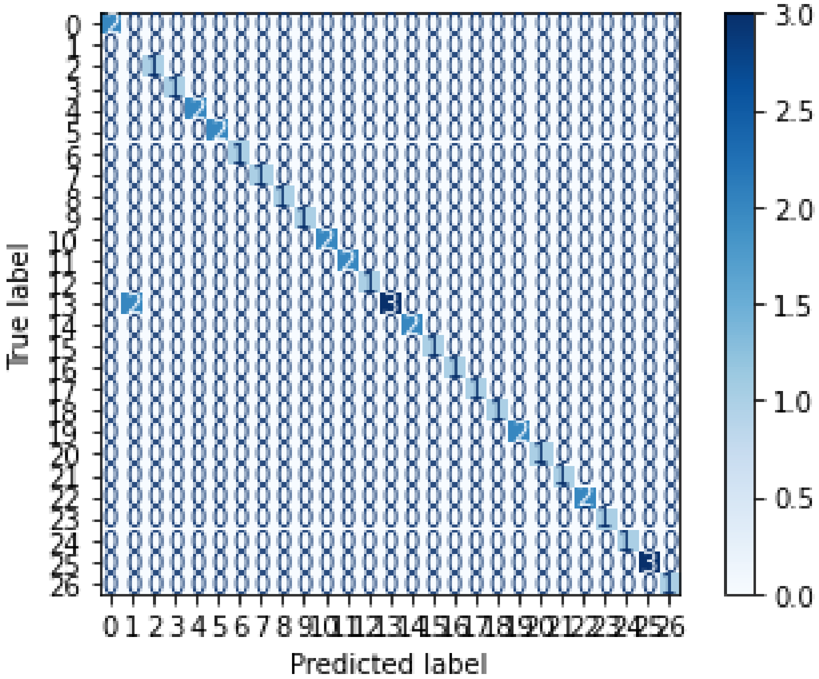
```
0.05
```

[51]:

```
#find the number of false postives using the confusion matrix
from sklearn.metrics import confusion_matrix, plot_confusion_matrix
plot_confusion_matrix(gnb, X_test, Y_test, cmap=plt.cm.Blues)
```

[51]:

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at
0x7f3211017190>
```



[54]:
```
# Verifying the results

def accuracy(pred,true_class):
    true=0
    total=len(pred)
    for i in range(total):
        if pred[i] == true_class[i]:
            true+=1

    return true/total

accuracy(y_pred,Y_test)
```

[54]:
```
0.95
```

[ ]: