2024

RAYYAN MASOOD.
Enrollment no: 01-134212-151
Class: BSCS 6-C
Date: May 17th ,2024

["ARTIFICIAL INTELLIGENCE LAB

["ASSIGNMENT NO:9"]

Lab Journal 9:

1. Implement each step of the PCA algorithm from scratch and apply it on either an existing dataset like “IRIS” or “MNIST” etc., or generate a dummy data by using the given command (it generates 6 variables for 10 samples, you can change as per your requirement).

> X = np.random.randint(10,50,100).reshape(10,10)

***NOTE: Do not use inbuilt function PCA() for this task, however, you can use inbuiltfunctions for implementing the steps. Show the output of each step as well.**

Code:

```
import numpy as np
import pandas as pd


X = np.random.randint(10, 50, 100).reshape(10, 10)

mean = np.mean(X, axis=0)
print("Mean of each feature:")
print(mean)

X_centered = X - mean
print("Centered data:")
print(X_centered)

covariance_matrix = np.cov(X_centered.T)
print("Covariance matrix:")
print(covariance_matrix)

eigenvalues, eigenvectors = np.linalg.eig(covariance_matrix)
print("Eigenvalues:")
print(eigenvalues)
print("Eigenvectors:")
print(eigenvectors)


sorted_indices = np.argsort(eigenvalues)[::-1]
sorted_eigenvalues = eigenvalues[sorted_indices]
sorted_eigenvectors = eigenvectors[:, sorted_indices]
```

```
print("Sorted eigenvalues:")
print(sorted_eigenvalues)
print("Sorted eigenvectors:")
print(sorted_eigenvectors)


k = 2
selected_eigenvectors = sorted_eigenvectors[:, :k]
print("Selected eigenvectors:")
print(selected_eigenvectors)


transformed_data = np.dot(X_centered, selected_eigenvectors)
print("Transformed data:")
print(transformed_data)
```

Output:

```
Mean of each feature:
[30.4 37.  36.  28.3 29.  24.5 31.4 23.7 29.8 26.4]
Centered data:
[[  6.6 -11.    0.  -17.3  14.   -3.5 -17.4  15.3 -15.8   2.6]
 [  2.6  -4.   11.   -6.3  16.    5.5   8.6  -7.7  15.2  -5.4]
 [ -8.4   9.   -5.   20.7 -12.    5.5  11.6 -13.7   5.2  -7.4]
 [ 18.6   9.    8.   19.7  -6.   -7.5 -16.4  10.3   9.2 -16.4]
 [-11.4  -5.  -12.   -9.3  14.    1.5   8.6 -12.7  12.2 -12.4]
 [  4.6  -2.  -12.  -11.3  -3.   13.5  16.6   6.3 -12.8  14.6]
 [ -8.4   7.  -15.   12.7  -1.   -3.5   3.6  21.3   6.2  10.6]
 [ -2.4  -6.   13.   16.7  -4.   -6.5 -11.4  -8.7 -13.8 -11.4]
 [ 18.6  -9.    6.  -16.3  -6.    9.5 -21.4  -9.7 -10.8  22.6]
 [-20.4  12.    6.   -9.3 -12.  -14.5  17.6  -0.7   5.2   2.6]]
```

```
Covariance matrix:
[[ 161.82222222  -44.44444444   42.77777778  -16.02222222   11.22222222
    40.88888889 -141.84444444   26.35555556  -51.24444444   30.48888889]
 [ -44.44444444   70.88888889  -10.88888889   74.66666667  -57.44444444
   -31.22222222   59.44444444   18.11111111   54.77777778  -24.22222222]
 [  42.77777778  -10.88888889  107.11111111   11.66666667   -9.88888889
   -29.88888889  -69.11111111  -39.11111111   -9.33333333  -37.33333333]
 [ -16.02222222   74.66666667   11.66666667  240.23333333  -75.11111111
   -38.61111111   -7.02222222    4.43333333   52.62222222 -109.46666667]
 [  11.22222222  -57.44444444   -9.88888889  -75.11111111  114.88888889
    16.11111111  -13.            8.55555556   18.33333333  -23.55555556]
 [  40.88888889  -31.22222222  -29.88888889  -38.61111111   16.11111111
    74.27777778   15.          -31.38888889  -18.44444444   48.44444444]
 [-141.84444444   59.44444444  -69.11111111   -7.02222222  -13.
    15.          226.71111111  -32.64444444   78.86666667   -4.73333333]
 [  26.35555556   18.11111111  -39.11111111    4.43333333    8.55555556
   -31.38888889  -32.64444444  156.9         -24.17777778   40.8       ]
 [ -51.24444444   54.77777778   -9.33333333   52.62222222   18.33333333
   -18.44444444   78.86666667  -24.17777778  141.95555556  -73.13333333]
 [  30.48888889  -24.22222222  -37.33333333 -109.46666667  -23.55555556
    48.44444444   -4.73333333   40.8         -73.13333333  165.15555556]]
```

```
Sorted eigenvalues:
[ 4.69910538e+02  3.51104510e+02  2.19280876e+02  1.62134137e+02
  1.13241130e+02  8.17813611e+01  3.82371909e+01  2.32586811e+01
  9.96020386e-01 -8.21597740e-15]
Sorted eigenvectors:
[[ 0.42879567 -0.31108668  0.0056805   0.01159182 -0.47215838  0.34869523
   0.33977995  0.27167701 -0.30837057 -0.30053218]
 [-0.29857948 -0.0593927  -0.26514769  0.04012076  0.03861312  0.35050299
   0.01516125  0.24709679  0.66167233 -0.4604015 ]
 [ 0.08554097 -0.30900401  0.29706924  0.19795108  0.42254921  0.38742699
   0.42741099 -0.46086567  0.15416988  0.14923581]
 [-0.42060149 -0.57558778 -0.26565051  0.11010535 -0.32098534 -0.29780021
   0.00248626 -0.43545749 -0.10534623 -0.12905784]
 [ 0.13305238  0.1413233   0.40974272 -0.5754466  -0.14520888 -0.19528283
   0.05396996 -0.38204813  0.20341413 -0.46241222]
 [ 0.13946555  0.17879345  0.07227296  0.22253213 -0.61351286 -0.08018491
   0.16495362 -0.1095322   0.52117316  0.44741728]
 [-0.48092789  0.5370835  -0.03161525  0.09855992 -0.09011545  0.04311491
   0.59921791 -0.05441304 -0.28589708 -0.12568366]
 [ 0.11542876 -0.03632362 -0.6066407  -0.63212116  0.08342913  0.07928976
   0.29128926 -0.07842719  0.05148832  0.33296862]
 [-0.38599383  0.00634619  0.22715034 -0.28940834 -0.28220019  0.6283809
  -0.37700926 -0.10651446 -0.15949085  0.24939104]
 [ 0.33466618  0.36215339 -0.42085851  0.27062962 -0.04235867  0.26306146
  -0.29234127 -0.53357229 -0.0969889  -0.23646555]]
```

```
   -0.29234127 -0.53357229 -0.0969889  -0.23646555]]
Selected eigenvectors:
[[ 0.42879567 -0.31108668]
 [-0.29857948 -0.0593927 ]
 [ 0.08554097 -0.30900401]
 [-0.42060149 -0.57558778]
 [ 0.13305238  0.1413233 ]
 [ 0.13946555  0.17879345]
 [-0.48092789  0.5370835 ]
 [ 0.11542876 -0.03632362]
 [-0.38599383  0.00634619]
 [ 0.33466618  0.36215339]]
Transformed data:
[[ 31.8684753    0.85088987]
 [ -3.90325935   5.93988482]
 [-27.89665792  -4.92270131]
 [ -4.12097843 -37.38398193]
 [-12.89924959  15.81803214]
 [ 10.3501974   24.78267386]
 [-11.05628345   3.79335401]
 [ -0.5985125  -24.27671686]
 [ 39.46286016   0.10185679]
 [-21.20659163  15.29670861]]
```

2. Import onlineretail.csv dataset (https://www.kaggle.com/vijayuv/onlineretail). Detect outliers, visualize them through boxplot and remove them using **inter quantile range** equation.

Code:

```
import matplotlib.pyplot as plt
import pandas as pd

data = pd.read_csv("OnlineRetail.csv",encoding="ISO-8859-1")


Q1 = data['Quantity'].quantile(0.25)
Q3 = data['Quantity'].quantile(0.75)
IQR = Q3 - Q1
```

```
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

outliers = data[(data['Quantity'] < lower_bound) | (data['Quantity'] >
upper_bound)]


plt.boxplot(data['Quantity'])
plt.title('Boxplot of Quantity')
plt.show()


data = data[(data['Quantity'] >= lower_bound) & (data['Quantity'] <=
upper_bound)]


print("Updated dataset without outliers:")
print(data)
```

Output:

```
Updated dataset without outliers:
       InvoiceNo StockCode                          Description  ...  UnitPrice CustomerID         Country
0         536365    85123A   WHITE HANGING HEART T-LIGHT HOLDER  ...       2.55    17850.0  United Kingdom
1         536365     71053                  WHITE METAL LANTERN  ...       3.39    17850.0  United Kingdom
2         536365    84406B       CREAM CUPID HEARTS COAT HANGER  ...       2.75    17850.0  United Kingdom
3         536365    84029G  KNITTED UNION FLAG HOT WATER BOTTLE  ...       3.39    17850.0  United Kingdom
4         536365    84029E       RED WOOLLY HOTTIE WHITE HEART.  ...       3.39    17850.0  United Kingdom
...          ...       ...                                  ...  ...        ...        ...             ...
541904    581587     22613          PACK OF 20 SPACEBOY NAPKINS  ...       0.85    12680.0          France
541905    581587     22899         CHILDREN'S APRON DOLLY GIRL   ...       2.10    12680.0          France
541906    581587     23254        CHILDRENS CUTLERY DOLLY GIRL   ...       4.15    12680.0          France
541907    581587     23255      CHILDRENS CUTLERY CIRCUS PARADE  ...       4.15    12680.0          France
541908    581587     22138        BAKING SET 9 PIECE RETROSPOT   ...       4.95    12680.0          France

[483290 rows x 8 columns]
```

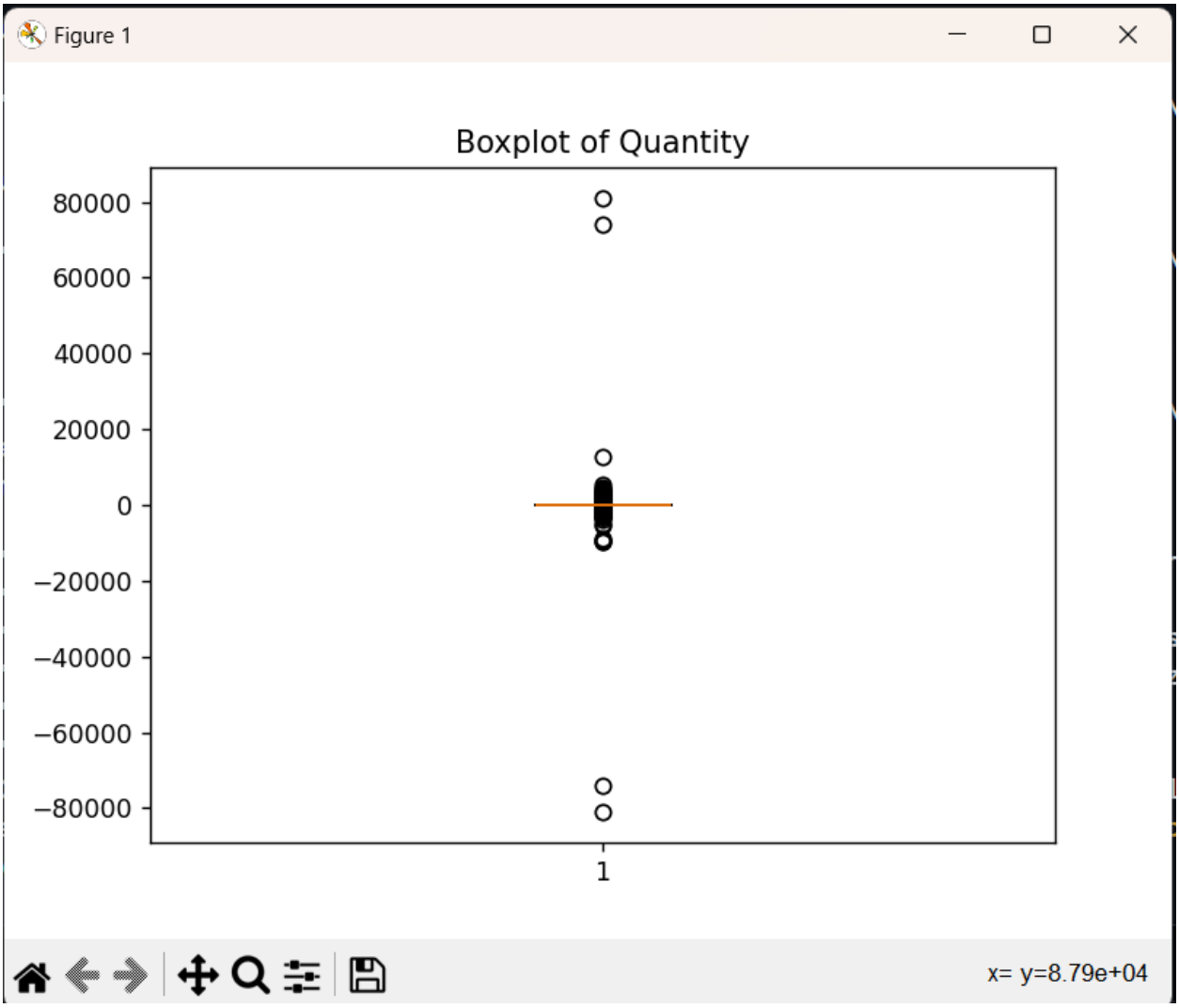
Figure 1
Boxplot of Quantity
80000
60000
40000
20000
0
−20000
−40000
−60000
−80000
1
x= y=8.79e+04