

Pattern & Anomaly Detection Lab

Experiment 6

Principal Component Analysis

Submitted By:

Dhruv Singhal

500075346

R177219074

AIML B3

Submitted To:

Dr. Gopal Phartiyal

Asst. Professor

SOCS

UPES

CODE:

```
1 # -*- coding: utf-8 -*-
2 *****
3
4 @author: Dhruv Singhal
5 *****
6
7 #####
8 import numpy as np
9 import pandas as pd
10 import matplotlib.pyplot as plt
11 import seaborn as sns
12 from sklearn.datasets import make_regression
13 #####
14 X,y=make_regression(n_samples=100,n_features=10,n_informative=4,random_state=0)
15 sns.distplot(y)
16
17 #####
18 from sklearn.decomposition import PCA
19 pca=PCA(n_components=5)
20 principalComponents=pca.fit_transform(X)
21 X_new=pd.DataFrame(data=principalComponents,columns=('PC1','PC2','PC3','PC4','PC5'))
22 print(X_new.head())
23 #####
24 from sklearn.model_selection import train_test_split
25 from sklearn.linear_model import LinearRegression
26 X_train , X_test , Y_train , Y_test = train_test_split(X_new, y, train_size=0.20,random_state=0)
27 lin=LinearRegression()
28 model=lin.fit(X_train,Y_train)
29 y_pred=model.predict(X_test)
30 #####
31 print('Coefficients: ', model.coef_)
32 print('Variance score: {}'.format(model.score(X_test, Y_test)))
33 print("Test Score",model.score(X_train,Y_train))
34 print("Test Score",model.score(X_test,Y_test))
35 #####
36 #training
37 plt.scatter(model.predict(X_train),model.predict(X_train) - Y_train,color = "green", s = 10, label = 'Train data')
38 #testing
39 plt.scatter(model.predict(X_test),model.predict(X_test) - Y_test,color = "orange", s = 10, label = 'Testing data')
40
41 #####
42 import math
43 from sklearn.metrics import mean_squared_error
44 print("RMSE",math.sqrt(mean_squared_error(Y_test,y_pred)))
45 from sklearn.metrics import r2_score
46 print("R^2",r2_score(Y_test,y_pred))
47
48
49
50 #####
51 print(np.max(Y_train),np.min(Y_train))
52
53 #####
54 plt.figure(figsize=(20,10))
55 plt.plot(y_pred)
56 plt.plot(Y_test)
```

```

57
58
59
60
61 #0.0/
62 X_train2 , X_test2 , Y_train2 , Y_test2 = train_test_split(X, y, train_size=0.20,random_state=0)
63 lin=LinearRegression()
64 model2=lin.fit(X_train2,Y_train2)
65 y_pred2=model.predict(X_test)
66 #0.0/
67 print('Coefficients: ', model2.coef_)
68 print('Variance score: {}'.format(model2.score(X_test2, Y_test2)))
69 print("Test Score",model2.score(X_train2,Y_train2))
70 print("Test Score",model2.score(X_test2,Y_test2))
71
72 #0.0/
73 import math
74 from sklearn.metrics import mean_squared_error
75 print("RMSE",math.sqrt(mean_squared_error(Y_test2,y_pred2)))
76 from sklearn.metrics import r2_score
77 print("R^2",r2_score(Y_test2,y_pred2))
78
79
80 #0.0/
81 print(np.max(Y_train2),np.min(Y_train2))
82
83 #0.0/
84 plt.figure(figsize=(20,10))
85 plt.plot(y_pred2)
86 plt.plot(Y_test2)
87
88
89
90

```

OUTPUT:

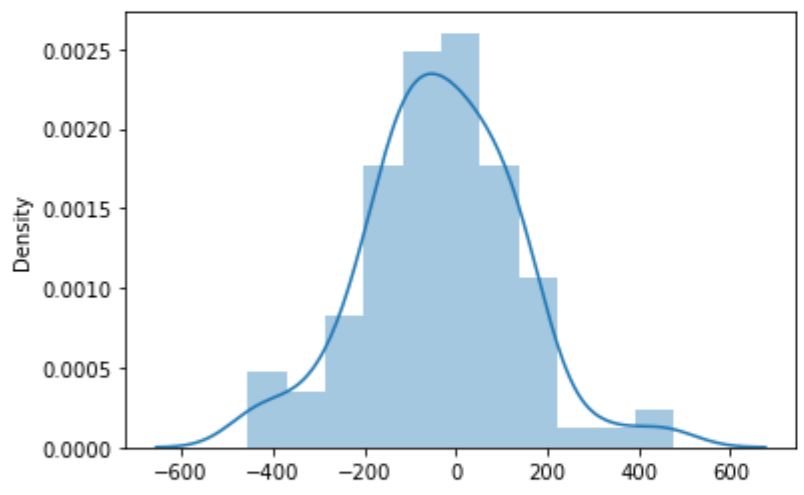
```
Console 1/A x
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.26.0 -- An enhanced Interactive Python.

In [1]: runcell(0, 'B:/3rd year/5th sem/P&AD/PCA.py')

In [2]: runcell(1, 'B:/3rd year/5th sem/P&AD/PCA.py')

In [3]: runcell(2, 'B:/3rd year/5th sem/P&AD/PCA.py')
C:\Users\Dhruv Singhal\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



```
In [4]: runcell(3, 'B:/3rd year/5th sem/P&AD/PCA.py')
      PC1      PC2      PC3      PC4      PC5
0  2.689187  0.188731  0.553922  0.677884  0.069860
1 -2.718170 -2.513221  0.564428  0.308534  0.148173
2 -0.150486  3.214459 -0.133615 -1.756744  0.593628
3 -1.597885 -2.291277 -0.211204 -0.593296 -0.201198
4 -0.621526 -0.492717  0.588593  0.407506 -0.956159

In [5]: runcell(4, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

```
In [6]: runcell(5, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

Output from spyder call 'get_namespace_view':

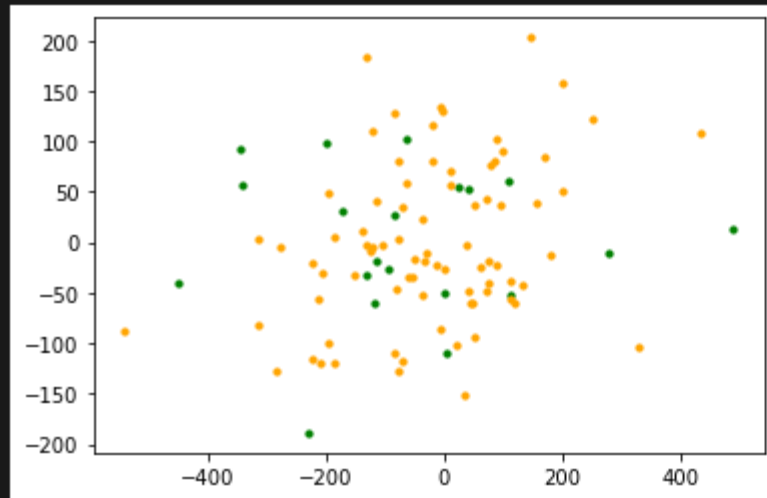
Coefficients: [-14.02798066 29.78262828 -44.53078748 120.20573166 -94.01743173]

Variance score: 0.715357937146422

Test Score 0.8948220992728023

Test Score 0.715357937146422

```
In [7]: runcell(6, 'B:/3rd year/5th sem/P&AD/PCA.py')
```



```
In [8]: runcell(7, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

RMSE 79.41012724129493

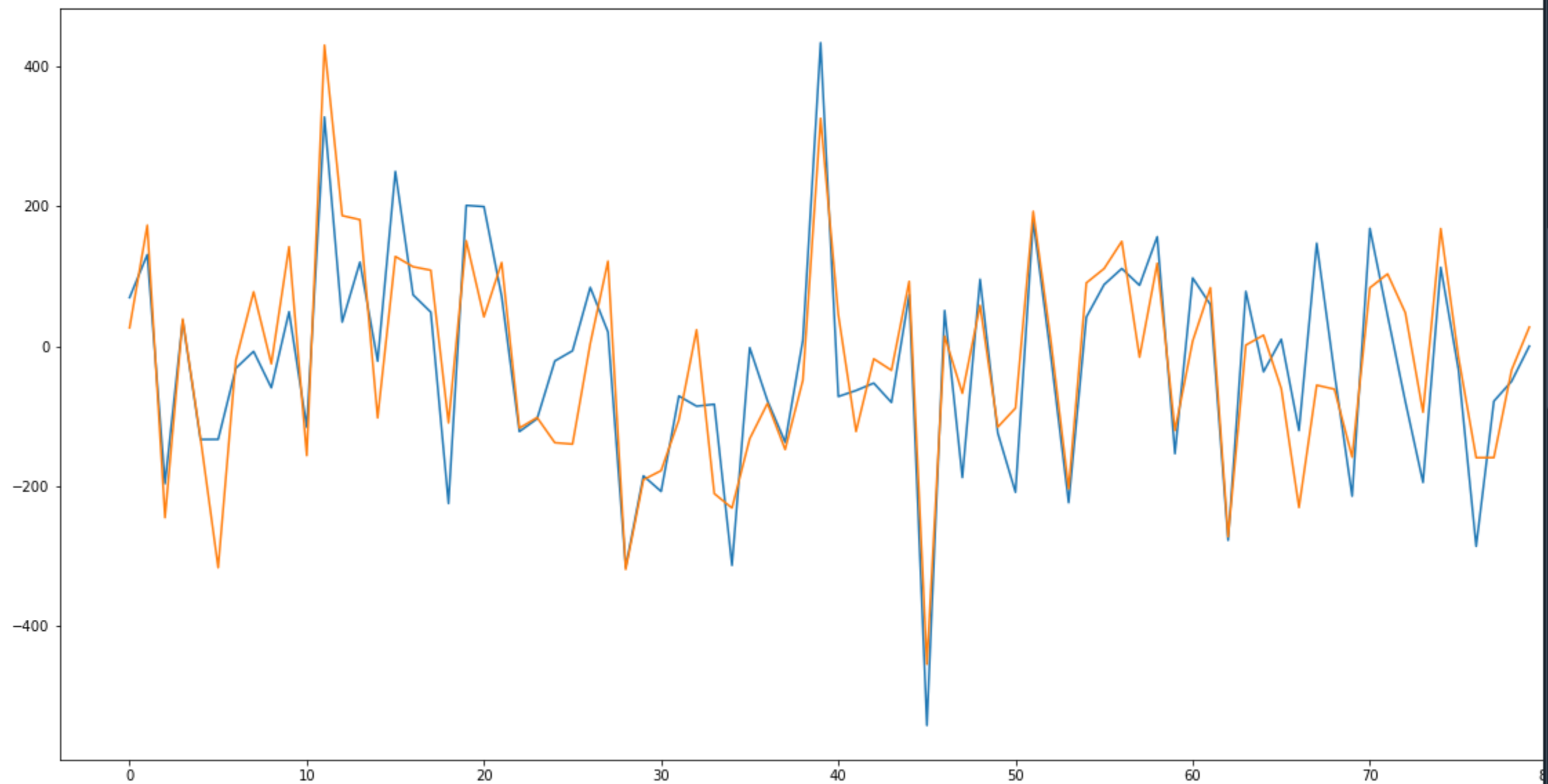
R² 0.715357937146422

```
In [9]: runcell(8, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

476.0496266907876 -436.9209738094973

```
In [10]: runcell(9, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

```
In [10]: runcell(9, 'B:/3rd year/5th sem/P&AD/PCA.py')
```



```
In [11]: runcell(10, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

```
In [12]: runcell(11, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

```
Coefficients: [-1.05459713e-14  9.83472758e-14 -7.33134857e-14  7.00528623e+01  
 8.83077597e+01  9.66575107e+01  8.21903908e+01  3.77754737e-14  
 4.10586745e-14 -4.61401623e-14]
```

```
Variance score: 1.0
```

```
Test Score 1.0
```

```
Test Score 1.0
```

```
In [13]: runcell(12, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

```
RMSE 79.41012724129493
```

```
R^2 0.715357937146422
```

```
In [14]: runcell(13, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

```
476.0496266907876 -436.9209738094973
```

```
In [15]: runcell(14, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

```
In [15]: runcell(14, 'B:/3rd year/5th sem/P&AD/PCA.py')
```

