20BCE529

Naman Thaker

PRACTICAL 6

Data Mining

PRINCIPAL COMPONENT ANALYSIS

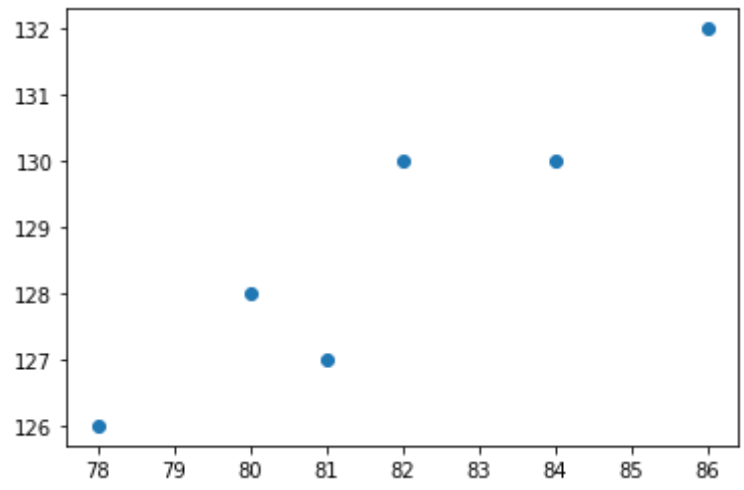
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
dataSet=pd.read_csv('BP.csv', delimiter=',')
dataSet
```

	Diastolic BP	Systolic BP
0	78	126
1	80	128
2	81	127
3	82	130
4	84	130
5	86	132

```
plt.scatter(dataSet['Diastolic BP'], dataSet['Systolic BP'])
plt.plot()
```

```
[]
```




```
DBP=np.array(dataSet['Diastolic BP'])
```

```
SBP=np.array(dataSet['Systolic BP'])
DBP, SBP
```

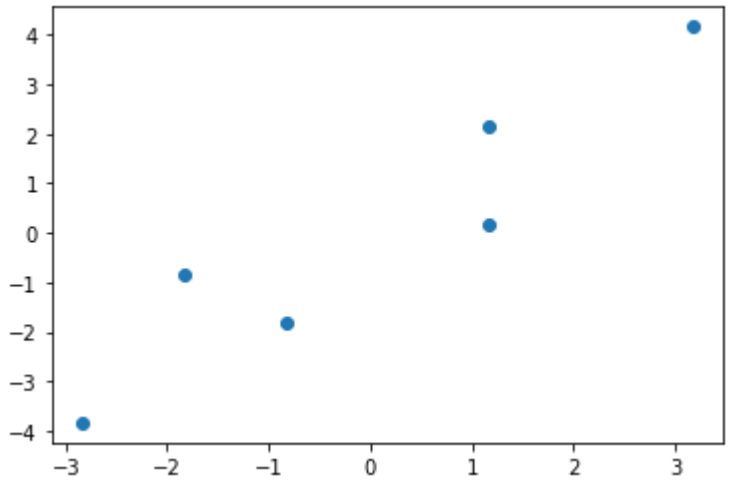
```
(array([78, 80, 81, 82, 84, 86]), array([126, 128, 127, 130, 130, 132]))
```

```
DBP=DBP-DBP.mean()
SBP=SBP-SBP.mean()
data=np.array([DBP, SBP]).T
DBP, SBP
```

```
(array([-3.83333333, -1.83333333, -0.83333333,  0.16666667,  2.16666667,
         4.16666667]),
 array([-2.83333333, -0.83333333, -1.83333333,  1.16666667,  1.16666667,
         3.16666667]))
```

```
plt.scatter(SBP, DBP)
plt.plot()
```

```
[]
```




```
covarianceMatrix=np.cov(dataSet, rowvar=False)
covarianceMatrix
```

```
array([[8.16666667, 5.96666667],
       [5.96666667, 4.96666667]])
```

```
eig_vals, eig_vecs = np.linalg.eig(covarianceMatrix)

print('Eigen - Vectors \n%s' %eig_vecs)
print('\nEigen - Values \n%s' %eig_vals)
```

```
Eigen - Vectors
[[ 0.79341219 -0.60868473]
 [ 0.60868473  0.79341219]]

Eigen - Values
[12.74413468  0.38919865]
```

```
eig_pairs = [(np.abs(eig_vals[i]), eig_vecs[:,i]) for i in range(len(eig_vals))]
```

```
eig_pairs.sort(key=lambda x: x[0], reverse=True)
print('Eigenvalues in descending order:')
for i in eig_pairs:
    print('{:.2f}'.format(i[0]))
```

```
 Eigenvalues in descending order:
 12.74
 0.39
```

```
tot = sum(eig_vals)
var_exp = [(i / tot)*100 for i in sorted(eig_vals, reverse=True)]
```

```
matrix_w = np.hstack((eig_pairs[0][1].reshape(2,1),
                      eig_pairs[1][1].reshape(2,1)
                    ))
print('Matrix W:\n', matrix_w)
```

```
 Matrix W:
  [[ 0.79341219 -0.60868473]
  [ 0.60868473  0.79341219]]
```

```
Y = np.dot(dataSet.values,matrix_w)
Y[:,0]
```

```
 array([138.58042679, 141.38462062, 141.56934808, 144.18881446,
        145.77563883, 148.57983267])
```

```
print(round(eig_vals[0]/(eig_vals[0]+eig_vals[1])*100, 2), '%')
```

```
97.04 %
```

+ Code + Text