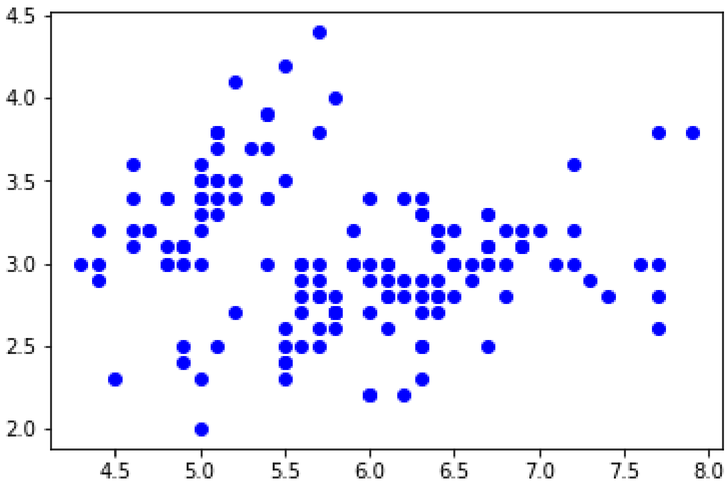
Excercise 1

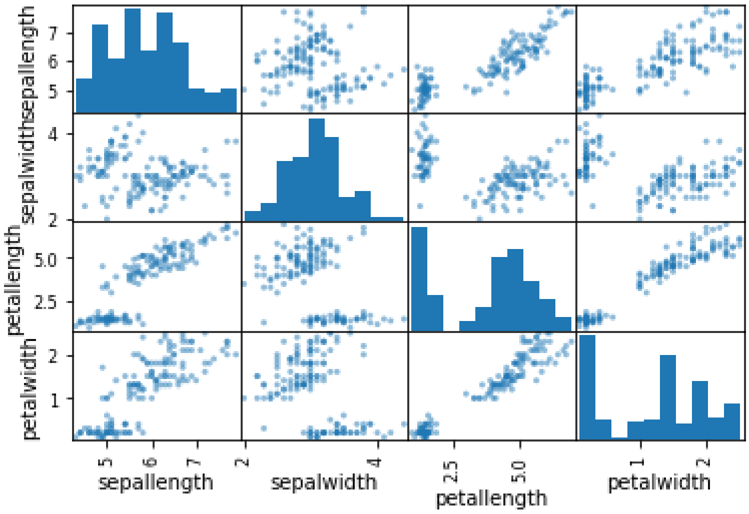
In [171…

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as stats
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans


#Excercise 1

iris = pd.read_csv('iris_csv.csv')
length = iris['sepallength']
width = iris['sepalwidth']

plt.plot(length, width, 'bo')
plt.show()

pd.plotting.scatter_matrix(iris)
```



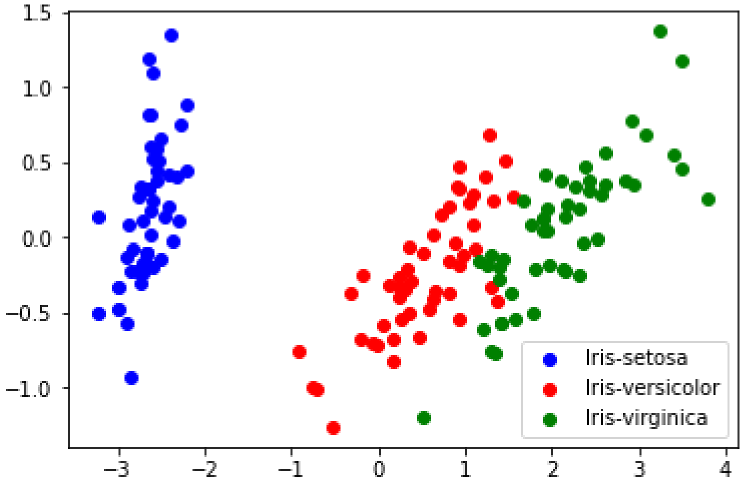
Out[171…

```
array([[<AxesSubplot:xlabel='sepallength', ylabel='sepallength'>,
        <AxesSubplot:xlabel='sepalwidth', ylabel='sepallength'>,
        <AxesSubplot:xlabel='petallength', ylabel='sepallength'>,
        <AxesSubplot:xlabel='petalwidth', ylabel='sepallength'>],
       [<AxesSubplot:xlabel='sepallength', ylabel='sepalwidth'>,
        <AxesSubplot:xlabel='sepalwidth', ylabel='sepalwidth'>,
        <AxesSubplot:xlabel='petallength', ylabel='sepalwidth'>,
        <AxesSubplot:xlabel='petalwidth', ylabel='sepalwidth'>],
       [<AxesSubplot:xlabel='sepallength', ylabel='petallength'>,
        <AxesSubplot:xlabel='sepalwidth', ylabel='petallength'>,
        <AxesSubplot:xlabel='petallength', ylabel='petallength'>,
        <AxesSubplot:xlabel='petalwidth', ylabel='petallength'>],
       [<AxesSubplot:xlabel='sepallength', ylabel='petalwidth'>,
        <AxesSubplot:xlabel='sepalwidth', ylabel='petalwidth'>,
        <AxesSubplot:xlabel='petallength', ylabel='petalwidth'>,
        <AxesSubplot:xlabel='petalwidth', ylabel='petalwidth'>]],
      dtype=object)
```



In [123…

```
#Excercise 2

irisclass = iris['class']
irisclass.to_numpy()
irisnumbers = iris.drop(['class'], axis=1)
pca = PCA(n_components = 2)
irispca = pca.fit_transform(irisnumbers)


label_color = {'Iris-setosa' : 'blue', 'Iris-versicolor' : 'yellow', 'Iris-virginica

setosa = irispca[irisclass == 'Iris-setosa']
versicolor = irispca[irisclass == 'Iris-versicolor']
virginica = irispca[irisclass == 'Iris-virginica']


plt.scatter(setosa[:,0], setosa[:,1], color='blue', label="Iris-setosa")
plt.scatter(versicolor[:,0], versicolor[:,1], color='red', label="Iris-versicolor")
plt.scatter(virginica[:,0], virginica[:,1], color='green', label="Iris-virginica")
plt.legend()
plt.show()
```



Iris setosa is clearly distinct. There is some overlap between iris versicolor and iris virginica.

In [137…

```
#Excercise 3

loadings = pd.DataFrame(pca.components_.T, index=irisnumbers.columns, columns=['1',
print(loadings)
```

```
                    1         2
sepallength  0.361590  0.656540
sepalwidth  -0.082269  0.729712
petallength  0.856572 -0.175767
petalwidth   0.358844 -0.074706
```

For component 1 the most important feature was petal length, while for component 2 it was sepal width followed by sepal ength.

In [170…

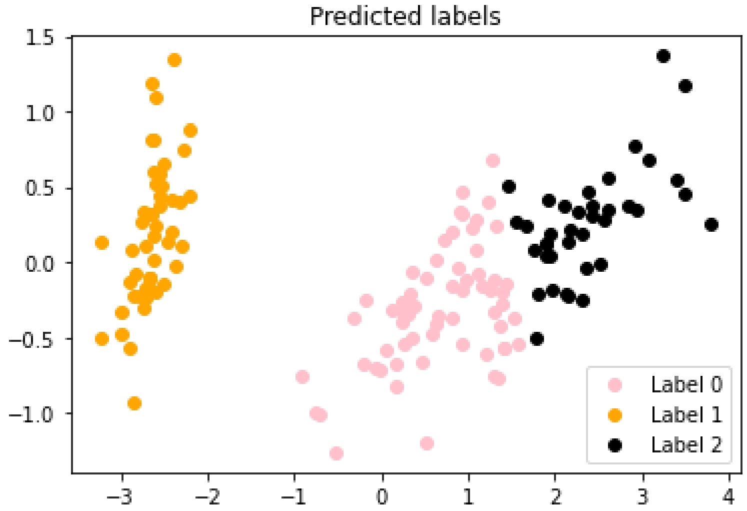
```
#Excercise 4

iriscorrD = iris.corr(method = 'spearman')
pcadf = pd.DataFrame(irispca)
iriscorrP = pcadf.corr(method='spearman')

print(iriscorrD)
print(iriscorrP)
```

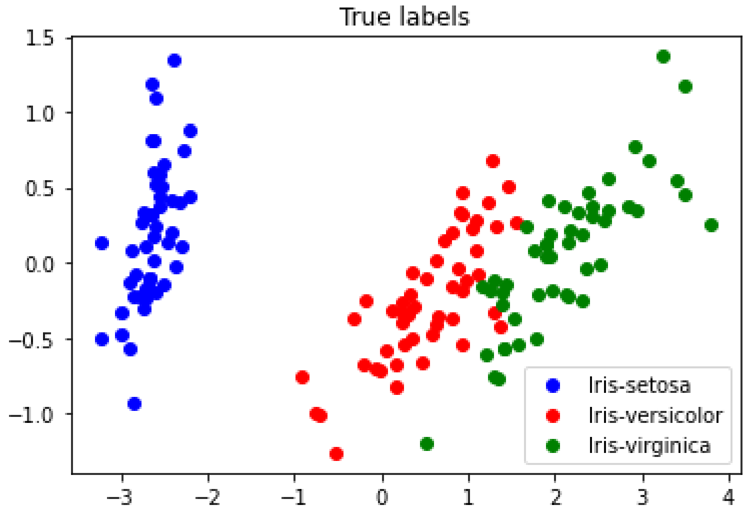
```
             sepallength  sepalwidth  petallength  petalwidth
sepallength     1.000000   -0.159457     0.881386    0.834421
sepalwidth     -0.159457    1.000000    -0.303421   -0.277511
petallength     0.881386   -0.303421     1.000000    0.936003
petalwidth      0.834421   -0.277511     0.936003    1.000000
          0         1
0  1.000000  0.141512
1  0.141512  1.000000
```

Petal length + sepal length, petal width + sepal length, petal length + petal width

In [187…

```
#Excercise 5

kmeans = KMeans(n_clusters=3)

clusters = kmeans.fit_predict(irisnumbers)

label0 = irispca[clusters == 0]
label1 = irispca[clusters == 1]
label2 = irispca[clusters == 2]

plt.scatter(label0[:,0], label0[:,1], color='pink', label="Label 0")
plt.scatter(label1[:,0], label1[:,1], color='orange', label="Label 1")
plt.scatter(label2[:,0], label2[:,1], color='black', label="Label 2")
plt.title("Predicted labels")
plt.legend()
plt.show()

plt.scatter(setosa[:,0], setosa[:,1], color='blue', label="Iris-setosa")
plt.scatter(versicolor[:,0], versicolor[:,1], color='red', label="Iris-versicolor")
plt.scatter(virginica[:,0], virginica[:,1], color='green', label="Iris-virginica")
plt.title('True labels')
plt.legend()
plt.show()
```





The label 1 does correspond with Iris-setosa almost perfectly. The problem is with clusters of label 0 (mostly iris versicolor) and label 2 (iris virginica), which do differ a little bit.