import pandas as pd

from sklearn.datasets import load\_iris

iris = load\_iris()

iris.feature\_names

iris.target\_names

Output

array(['setosa', 'versicolor', 'virginica'], dtype='<U10')

df = pd.DataFrame(iris.data,columns=iris.feature\_names)

df['target'] = iris.target

df['flower\_name'] =df.target.apply(lambda x: iris.target\_names[x])

df0 = df[:50]

df1 = df[50:100]

df2 = df[100:]

**Sepal length vs Sepal Width (Setosa vs Versicolor)**

import matplotlib.pyplot as plt

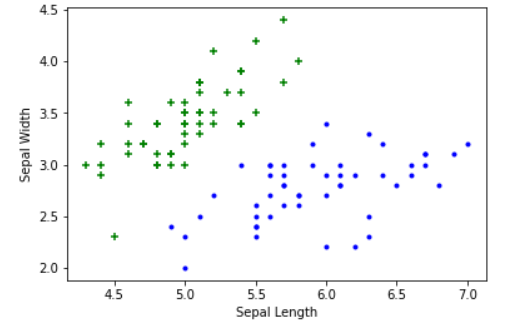
%matplotlib inline

plt.xlabel('Sepal Length')

plt.ylabel('Sepal Width')

plt.scatter(df0['sepal length (cm)'], df0['sepal width (cm)'],color="green",marker='+')

plt.scatter(df1['sepal length (cm)'], df1['sepal width (cm)'],color="blue",marker='.')



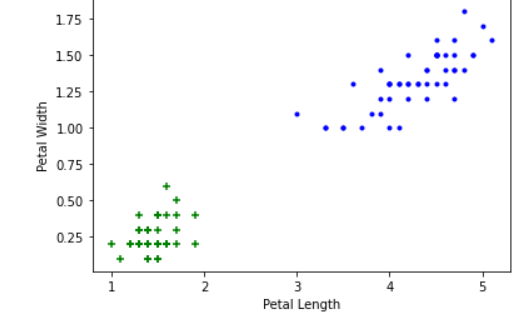
**Petal length vs Pepal Width (Setosa vs Versicolor)**

plt.xlabel('Petal Length')

plt.ylabel('Petal Width')

plt.scatter(df0['petal length (cm)'], df0['petal width (cm)'],color="green",marker='+')

plt.scatter(df1['petal length (cm)'], df1['petal width (cm)'],color="blue",marker='.')



from sklearn.model\_selection import train\_test\_split

X = df.drop(['target','flower\_name'], axis='columns')

y = df.target

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=1)

**Create KNN (K Neighrest Neighbour Classifier)**

from sklearn.neighbors import KNeighborsClassifier

knn = KNeighborsClassifier(n\_neighbors=10)

knn.fit(X\_train, y\_train)

knn.score(X\_test, y\_test)

Output

0.97

knn.predict([[4.8,3.0,1.5,0.3]])

output

array([0])

**Plot Confusion Matrix**

from sklearn.metrics import confusion\_matrix

y\_pred = knn.predict(X\_test)

cm = confusion\_matrix(y\_test, y\_pred)

cm

Output

array([[11, 0, 0],

[ 0, 12, 1],

[ 0, 0, 6]])

%matplotlib inline

import matplotlib.pyplot as plt

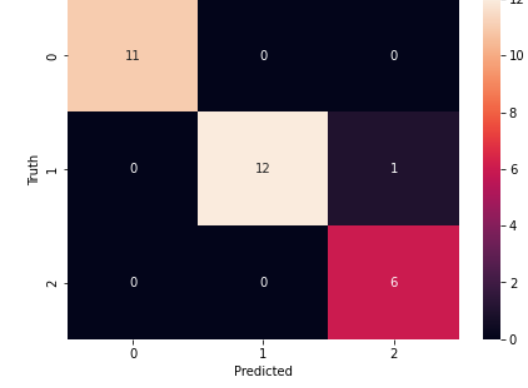
import seaborn as sn

plt.figure(figsize=(7,5))

sn.heatmap(cm, annot=True)

plt.xlabel('Predicted')

plt.ylabel('Truth')



**Print classification report for precesion, recall and f1-score for each classes**

:

from sklearn.metrics import classification\_report

print(classification\_report(y\_test, y\_pred))

precision recall f1-score support

0 1.00 1.00 1.00 11

1 1.00 0.92 0.96 13

2 0.86 1.00 0.92 6

accuracy 0.97 30

macro avg 0.95 0.97 0.96 30

weighted avg 0.97 0.97 0.97 30