```python
#!/usr/bin/env python
# coding: utf-8

# In[2]:


from sklearn.datasets import load_iris
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split   #if it didn't
work, replace sklearn.model_selection with sklearn.cross_validation
import matplotlib.pyplot as plt
#%matplotlib inline                    #this is to plot the graphs in
this Jupyter notebook, but not outside of it

iris_dataset=load_iris()
X_train, X_test, y_train, y_test =
train_test_split(iris_dataset["data"], iris_dataset["target"],
random_state=0) #By default Train:Test ratio is 3:1

kn = KNeighborsClassifier()
kn.fit(X_train, y_train)
prediction = kn.predict(X_test)

import sklearn.metrics as sm
print('ACCURACY of KNN:',sm.accuracy_score(y_test,prediction))
print('Confusion Matrix for
KNN:\n',sm.confusion_matrix(y_test,prediction))    #Confusion matrix
for Data Samples in y_test

plt.plot(X_test,y_test,'ro')         #red colored circles represent
Actual Data
plt.plot(X_test,prediction,'b+')     #Blue colored plus represent
Predicted Data


# In[3]:


#To check detailed info about the predicted data and the actual data
of X_TEST
y_pred = kn.predict(X_test)
print("Classification Results are:\n")
for i in range(0,len(X_test)):
    print("Sample:", str(X_test[i]), " Actual label:",
str(y_test[i])," Predicted label:", str(y_pred[i]))


# In[ ]:
```