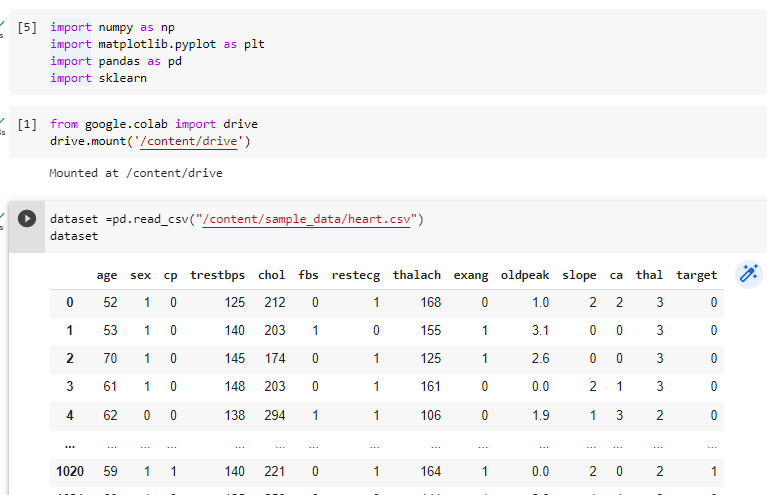
**LAB # 08**

# **Task # 1: Using python implement KNN with two different splitting ratios and two different k values (i.e k=3 and k=7) on Heart Attack Analysis & prediction dataset to predict the chances of heart failure in a person and performed the following steps:**

* **Data Pre-processing step**
* **Fitting KNN to the Training set**
* **Predicting the test result**
* **Test accuracy of the result(Creation of Confusion matrix)**
* **Visualizing the test set result.**
* **Compare the performance**

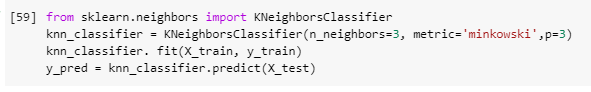
**Solution:**

1. **Data Pre-processing step:**

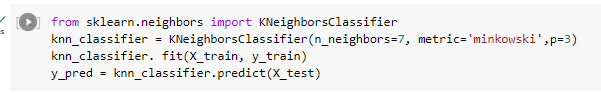


1. **Fitting KNN to the Training set:** 

**K=3:**

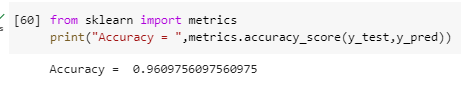


**K=7:**

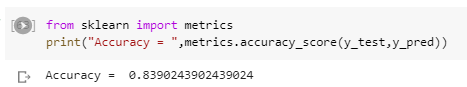


1. **Predicting the test result:**

**k=3:**



**K=7:**



1. **Test accuracy of the result(Creation of Confusion matrix):**

from sklearn.metrics import confusion\_matrix

conf\_matrix = confusion\_matrix(y\_true=y\_test, y\_pred=y\_pred)

fig, ax = plt.subplots(figsize=(4, 4))

ax.matshow(conf\_matrix, cmap=plt.cm.Blues, alpha=0.3)

for i in range(conf\_matrix.shape[0]):

    for j in range(conf\_matrix.shape[1]):

        ax.text(x=j, y=i,s=conf\_matrix[i, j], va='center', ha='center', size='xx-large')

plt.xlabel('Predictions', fontsize=18)

plt.ylabel('Actuals', fontsize=18)

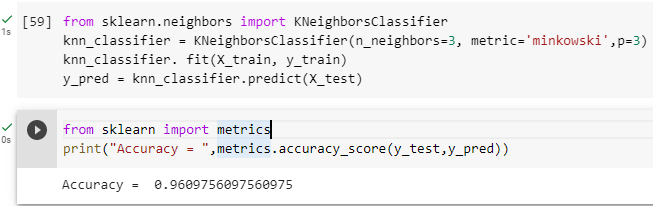
plt.title('Confusion Matrix', fontsize=18)

plt.show()

|  |  |
| --- | --- |
| **K=3** | **K=7** |

1. **Compare the performance**

**K=3:**



**K=7:**

