CS-23334 FUNDAMENTALS OF DATA SCIENCE

ABENANTHAN P 240701005

EXPERIMENT 9

9.Experiment to understand KNN algorithm for a given data set

Aim:

To conduct experiment to understand KNN Algorithm for a given data set

Description:

Understand the KNN algorithm for the dataset given.

Algorithm:

Step 1: Define the Classification or Regression Problem

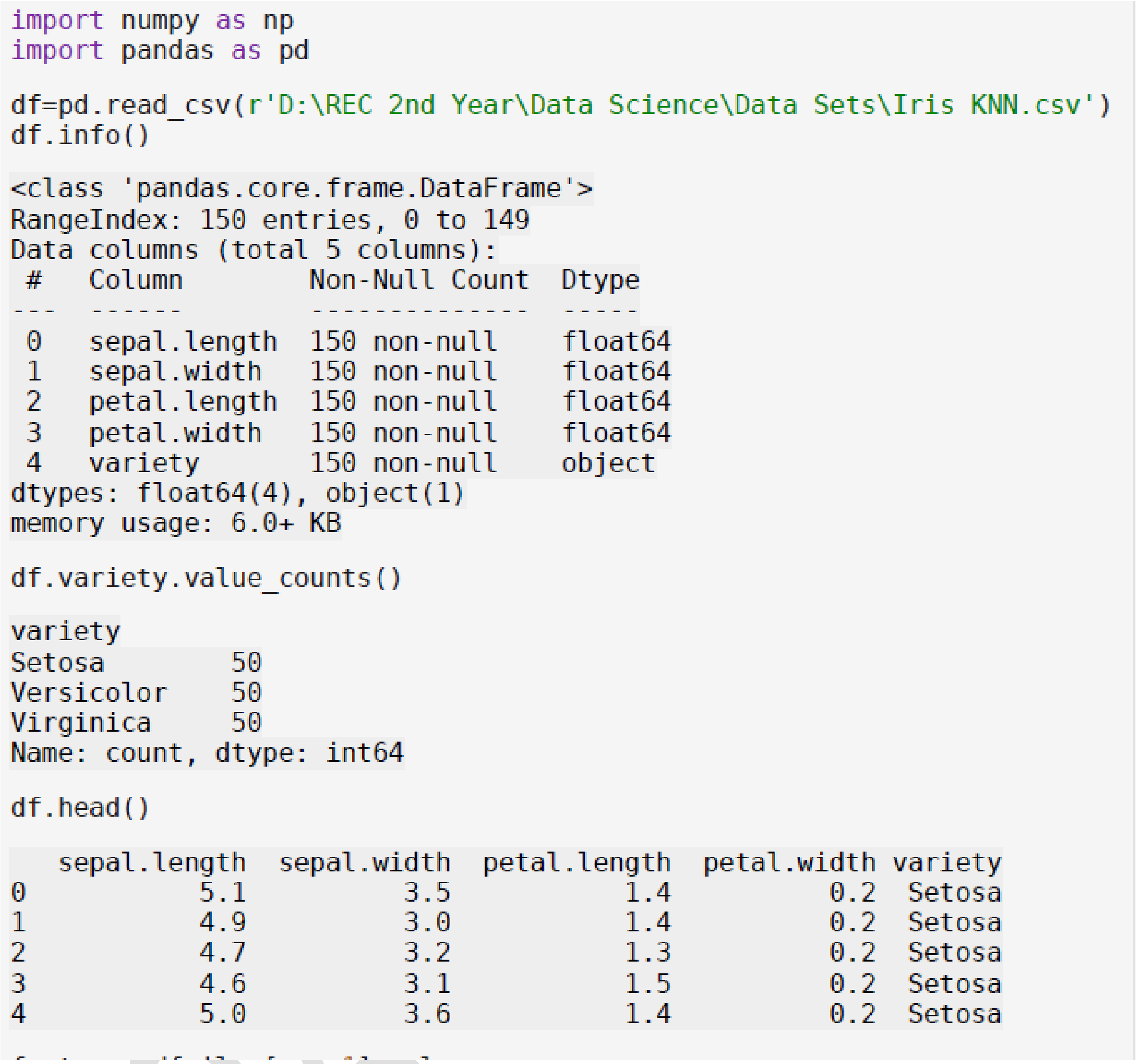
Step 2: Select Features and Normalize the Data

Step 3: Choose the Value of K and Train the Model

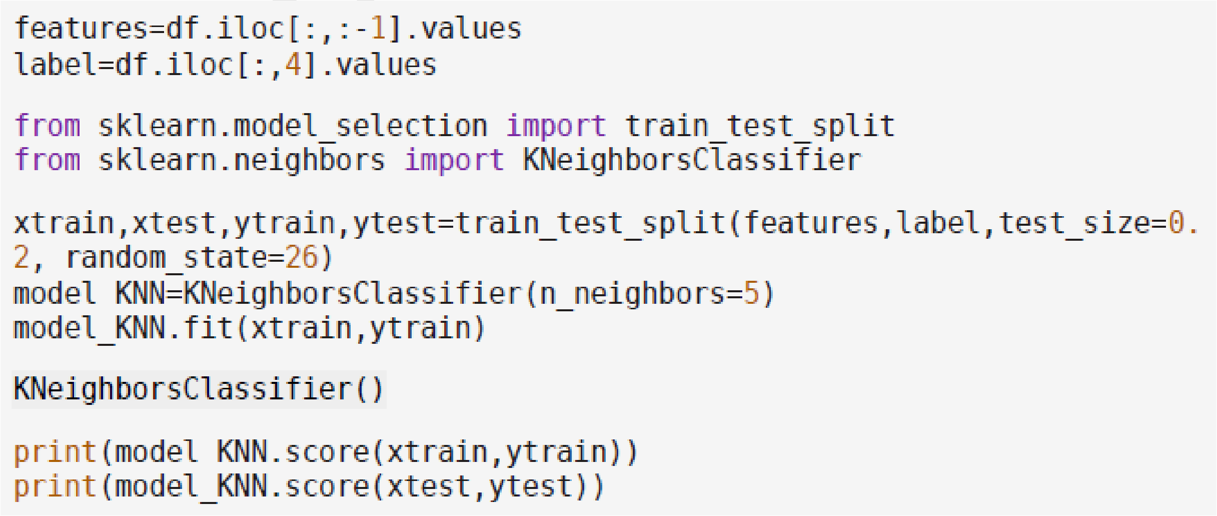
Step 4: Predict and Evaluate Model Performance

Step 5: Visualize Decision Boundaries or Accuracy Trends

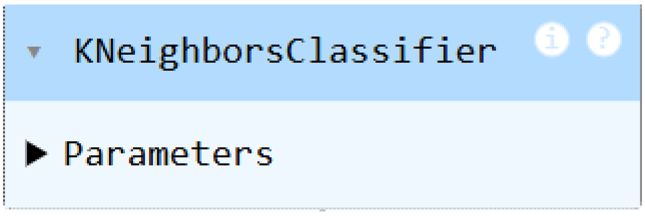
Code With Output:

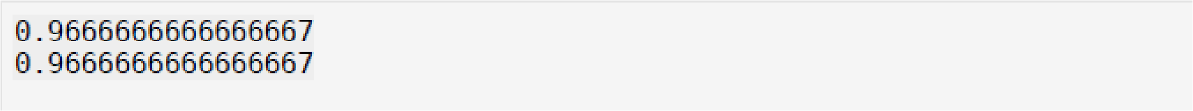


```
import numpy as np
import pandas as pd

df=pd.read_csv(r'D:\REC 2nd Year\Data Science\Data Sets\Iris KNN.csv')
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   sepal.length  150 non-null    float64
 1   sepal.width   150 non-null    float64
 2   petal.length  150 non-null    float64
 3   petal.width   150 non-null    float64
 4   variety       150 non-null    object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB

df.variety.value_counts()

variety
Setosa        50
Versicolor    50
Virginica     50
Name: count, dtype: int64

df.head()

   sepal.length  sepal.width  petal.length  petal.width variety
0           5.1          3.5           1.4          0.2  Setosa
1           4.9          3.0           1.4          0.2  Setosa
2           4.7          3.2           1.3          0.2  Setosa
3           4.6          3.1           1.5          0.2  Setosa
4           5.0          3.6           1.4          0.2  Setosa
```

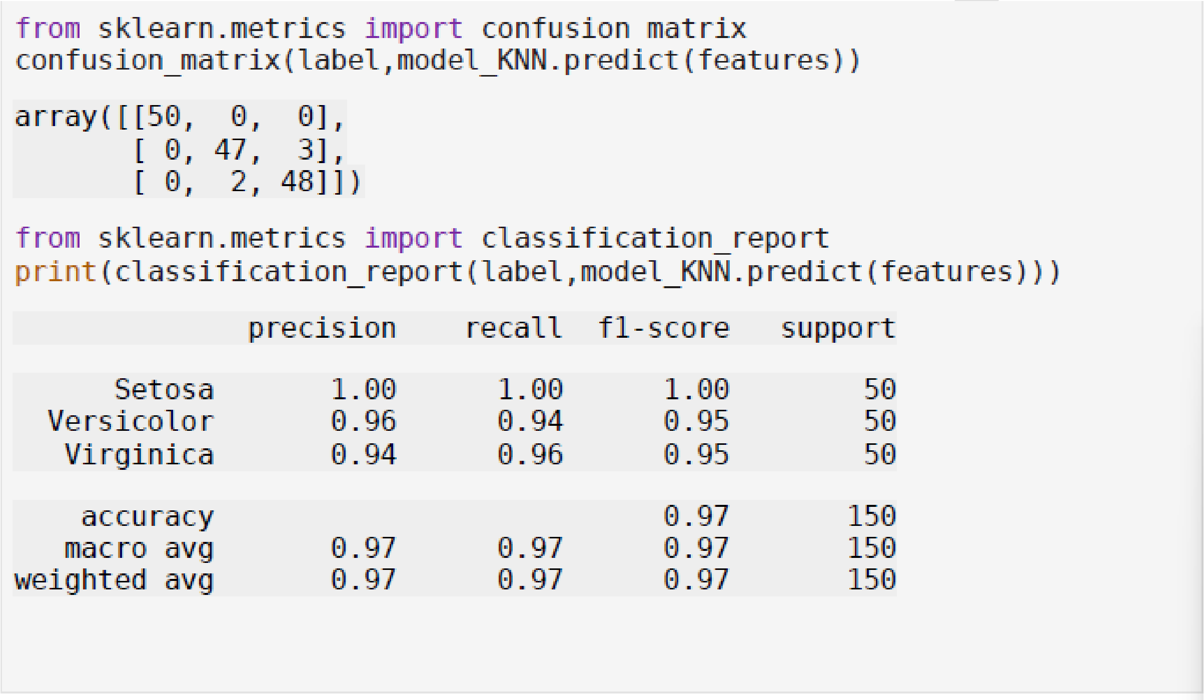


```
features=df.iloc[:,:-1].values
label=df.iloc[:,4].values

from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier

xtrain,xtest,ytrain,ytest=train_test_split(features,label,test_size=0.
2, random_state=26)
model KNN=KNeighborsClassifier(n_neighbors=5)
model_KNN.fit(xtrain,ytrain)

KNeighborsClassifier()

print(model KNN.score(xtrain,ytrain))
print(model_KNN.score(xtest,ytest))
```





```
0.9666666666666667
0.9666666666666667
```



```
from sklearn.metrics import confusion matrix
confusion_matrix(label,model_KNN.predict(features))
```

```
array([[50,  0,  0],
       [ 0, 47,  3],
       [ 0,  2, 48]])
```

```
from sklearn.metrics import classification_report
print(classification_report(label,model_KNN.predict(features)))
```

```
              precision    recall  f1-score   support

      Setosa       1.00      1.00      1.00        50
  Versicolor       0.96      0.94      0.95        50
   Virginica       0.94      0.96      0.95        50

    accuracy                           0.97       150
   macro avg       0.97      0.97      0.97       150
weighted avg       0.97      0.97      0.97       150
```

Result:

Thus python program to understand KNN algorithm for dataset is conducted successfully