

Ex-no: 8

Implement the K-Nearest Neighbor Learning Algorithm

Aim:

Write a program to Implement K-Nearest Algorithm to classify the iris data set.

Algorithm:

Input: Let m be the number of training data samples. Let P be an unknown point.

Method:

1. Store the training samples in an array of data points $arr[]$. This means each element of this array represents a tuple (x, y) .

2. for $i = 0$ to m

Calculate Euclidean distance $d(arr[i], P)$

3. Make set S of K smallest distances obtained. Each of these distances correspond to an already classified data point.

4. Return the Majority label among S .

Code:

```
from sklearn.model_selection import train_test_split
```

```
from sklearn import datasets
```

```
iris = datasets.load_iris()
```

```
X = iris.data
```

```
Y = iris.target
```

```
print("Iris Data set loaded...")
```

```

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.1)
print("Dataset is split into training and testing...")
print("Size of training Data and its label", x_train.shape, y_train.shape)
print("Size of testing data and its label", x_test.shape, y_test.shape)
for i in range(len(iris.target_names)):
    print("Label", i, "-", str(iris.target_names[i]))

from sklearn.neighbors import KNeighborsClassifier as Knn
classifier = Knn(n_neighbors=1)
classifier.fit(x_train, y_train)
y_pred = classifier.predict(x_test)
print("Results of classification using K-nn with k=1")
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]))
print("Classification Accuracy:", classifier.score(x_test, y_test))
from sklearn.metrics import classification_report, Confusion_Matrix
print("Confusion Matrix")
print(Confusion_Matrix(y_test, y_pred))
print("Accuracy Metrics")
print(classification_report(y_test, y_pred))

```

Result:

Thus the K-Nearest Neighbor Learning Algorithm has been implemented Successfully.

Output:

Iris Data are loaded...

Dataset is Split into training and testing...

Size of training Data and its label (135,4) (135,)

Size of testing Data and its label (15,4) (15,)

Label 0 - setosa

Label 1 - versicolor

Label 2 - virginica

Results of classification using K-nn with K=1

Sample: [6.9 3.1 5.4 2.1] Actual-label: 2 Predicted-label: 2

Sample: [5. 2.3 3.3 1.] Actual-label: 1 Predicted-label: 1

Sample: [6.4 3.2 5.3 2.3] Actual-label: 2 Predicted-label: 2

Sample: [5.5 2.4 3.8 1.8] Actual-label: 2 Predicted-label: 1

Sample: [4.9 2.4 3.8 1.1] Actual-label: 1 Predicted-label: 2

Sample: [6.7 3. 5.2 2.3] Actual-label: 1 Predicted-label: 0

Sample: [6. 2.7 5.1 1.6] Actual-label: 0 Predicted-label: 1

Sample: [4.9 3.1 1.5 0.1] Actual-label: 1 Predicted-label: 1

Sample: [4.4 3.1 1.5 0.2] Actual-label: 0 Predicted-label: 2

Sample: [4.9 3. 1.3 0.2] Actual-label: 1 Predicted-label: 1

Sample: [6. 3. 4.8 1.8] Actual-label: 2 Predicted-label: 0

Sample: [5.1 3.8 1.9 0.4] Actual-label: 2 Predicted-label: 1

Sample: [5.1 3.5 1.4 0.2] Actual-label: 0 Predicted-label: 0

Classification Accuracy = 0.933333333333

$\begin{bmatrix} 5 & 0 & 0 \end{bmatrix}$

$\begin{bmatrix} 0 & 3 & 1 \end{bmatrix}$

$\begin{bmatrix} 0 & 0 & 6 \end{bmatrix}$

Accuracy Matrices

	Precision	recall	F ₁ -Score	Support
0	1.00	1.00	1.00	5
1	1.00	0.75	0.86	4
2	0.86	1.00	0.92	6

Accuracy

(1.00 + 0.86 + 0.92) / 3 = 0.93

Macro avg

0.95

0.92

0.93

15

weighted avg

0.94

0.93

0.93

15

exp8 - Jupyter Notebo... ML/

localhost:8888/notebooks/ML/exp8.ipynb

jupyter exp8 Last Checkpoint: 9 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

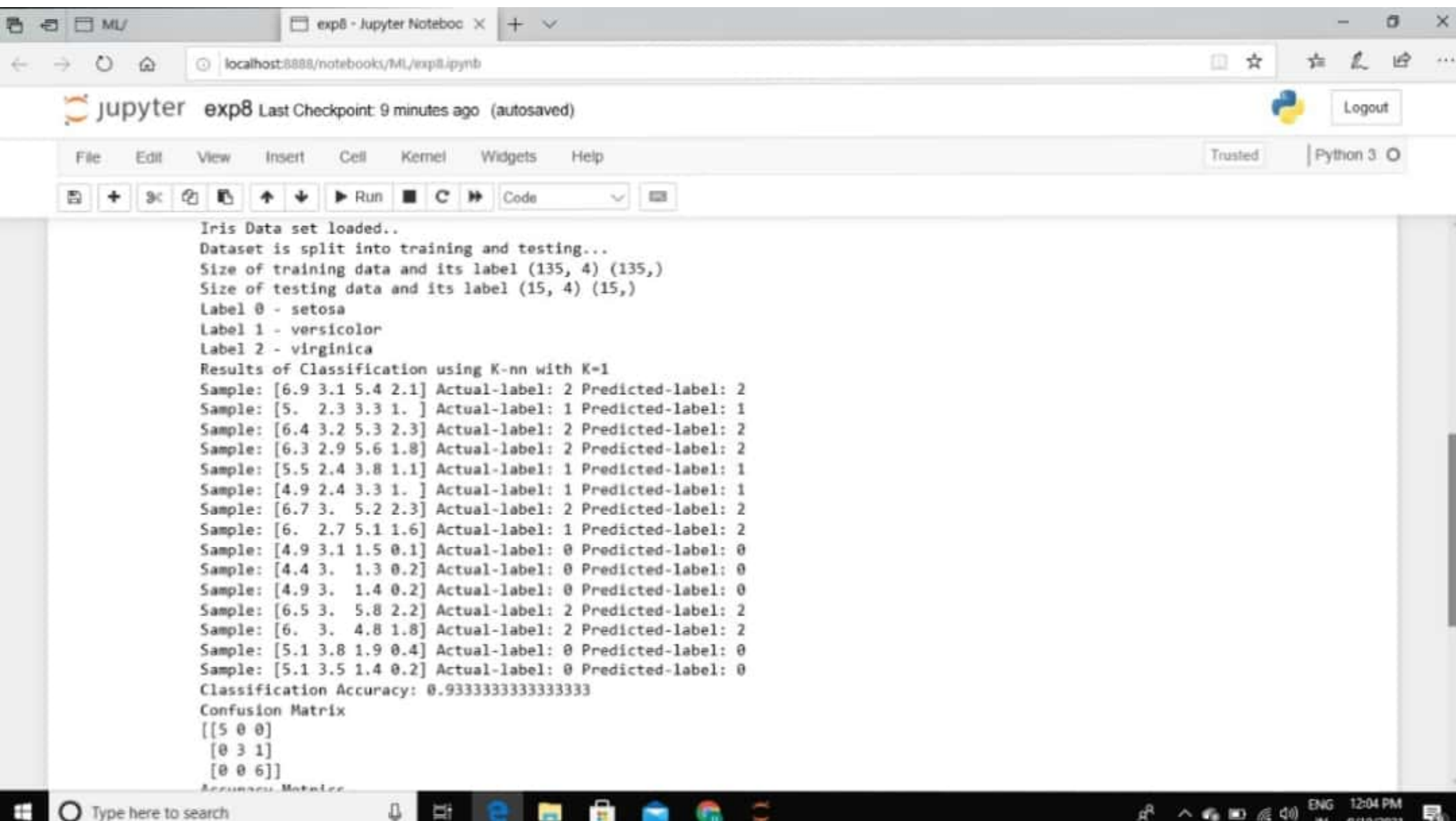
Code

```
In [4]: from sklearn.model_selection import train_test_split
from sklearn import datasets
iris = datasets.load_iris()
x = iris.data
y = iris.target
print("Iris Data set loaded..")
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.1)
print("Dataset is split into training and testing...")
print("Size of training data and its label", x_train.shape, y_train.shape)
print("Size of testing data and its label", x_test.shape, y_test.shape)
for i in range(len(iris.target_names)):
    print("Label", i, "-", str(iris.target_names[i]))
from sklearn.neighbors import KNeighborsClassifier as knn
classifier = knn(n_neighbors=1)
classifier.fit(x_train, y_train)
y_pred = classifier.predict(x_test)
print("Results of Classification using K-nn with K=1")
for r in range(0, len(x_test)):
    print("Sample:", str(x_test[r]), "Actual-label:", str(y_test[r]), "Predicted-label:",
    str(y_pred[r]))
print("Classification Accuracy:", classifier.score(x_test, y_test))
from sklearn.metrics import classification_report, confusion_matrix
print('Confusion Matrix')
print(confusion_matrix(y_test, y_pred))
print('Accuracy Metrics')
print(classification_report(y_test, y_pred))
```

Iris Data set loaded..

Type here to search

ENG 12:03 PM



The screenshot shows a Jupyter Notebook window titled 'exp8 - Jupyter Notebo...'. The browser address bar shows 'localhost:8888/notebooks/ML/exp8.ipynb'. The Jupyter logo and 'exp8' are visible, along with a 'Last Checkpoint: 9 minutes ago (autosaved)' message and a 'Logout' button. The menu bar includes 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. The toolbar shows icons for file operations, running, and code execution. The code cell contains the following text:

```
Iris Data set loaded..  
Dataset is split into training and testing...  
Size of training data and its label (135, 4) (135,)  
Size of testing data and its label (15, 4) (15,)  
Label 0 - setosa  
Label 1 - versicolor  
Label 2 - virginica  
Results of Classification using K-nn with K=1  
Sample: [6.9 3.1 5.4 2.1] Actual-label: 2 Predicted-label: 2  
Sample: [5. 2.3 3.3 1. ] Actual-label: 1 Predicted-label: 1  
Sample: [6.4 3.2 5.3 2.3] Actual-label: 2 Predicted-label: 2  
Sample: [6.3 2.9 5.6 1.8] Actual-label: 2 Predicted-label: 2  
Sample: [5.5 2.4 3.8 1.1] Actual-label: 1 Predicted-label: 1  
Sample: [4.9 2.4 3.3 1. ] Actual-label: 1 Predicted-label: 1  
Sample: [6.7 3. 5.2 2.3] Actual-label: 2 Predicted-label: 2  
Sample: [6. 2.7 5.1 1.6] Actual-label: 1 Predicted-label: 2  
Sample: [4.9 3.1 1.5 0.1] Actual-label: 0 Predicted-label: 0  
Sample: [4.4 3. 1.3 0.2] Actual-label: 0 Predicted-label: 0  
Sample: [4.9 3. 1.4 0.2] Actual-label: 0 Predicted-label: 0  
Sample: [6.5 3. 5.8 2.2] Actual-label: 2 Predicted-label: 2  
Sample: [6. 3. 4.8 1.8] Actual-label: 2 Predicted-label: 2  
Sample: [5.1 3.8 1.9 0.4] Actual-label: 0 Predicted-label: 0  
Sample: [5.1 3.5 1.4 0.2] Actual-label: 0 Predicted-label: 0  
Classification Accuracy: 0.9333333333333333  
Confusion Matrix  
[[5 0 0]  
 [0 3 1]  
 [0 0 6]]  
Accuracy: 0.9333333333333333
```

The Windows taskbar at the bottom shows the search bar and several application icons. The system tray on the right indicates 'ENG' and '12:04 PM'.

exp8 - Jupyter Notebook

localhost:8888/notebooks/ML/exp8/pynb

jupyter exp8 Last Checkpoint: 9 minutes ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Trusted Python 3

Run

```
Sample: [4.9 3.1 1.5 0.1] Actual-label: 0 Predicted-label: 0
Sample: [4.4 3. 1.3 0.2] Actual-label: 0 Predicted-label: 0
Sample: [4.9 3. 1.4 0.2] Actual-label: 0 Predicted-label: 0
Sample: [6.5 3. 5.8 2.2] Actual-label: 2 Predicted-label: 2
Sample: [6. 3. 4.8 1.8] Actual-label: 2 Predicted-label: 2
Sample: [5.1 3.8 1.9 0.4] Actual-label: 0 Predicted-label: 0
Sample: [5.1 3.5 1.4 0.2] Actual-label: 0 Predicted-label: 0
Classification Accuracy: 0.9333333333333333
Confusion Matrix
[[5 0 0]
 [0 3 1]
 [0 0 6]]
Accuracy Metrics
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	5
1	1.00	0.75	0.86	4
2	0.86	1.00	0.92	6
accuracy			0.93	15
macro avg	0.95	0.92	0.93	15
weighted avg	0.94	0.93	0.93	15

In []: