



LABORATORY WORK BOOK

Name of the Student : HIMAKAR C

Class : CSE-B Semester : VI

Course Code : ACIC08 Course Name : DMKD Laboratory

Name of the Course Faculty : Dr. D. DURGA BHAVANI Faculty ID : IARE 10921

Exercise Number : _____ Week Number : 08 Date : 15/6/24

Roll Number									
2	1	9	5	1	A	0	5	6	5

S. No.	Exercise Number	EXERCISE NAME	MARKS AWARDED					
			Aim/ Preparation	Algorithm / Procedure	Source Code	Program Execution	Viva - Voce	Total
				Performance in the Lab	Calculations and Graphs	Results and Error Analysis		
			4	4	4	4	4	20
1	8-1	calculate Euclidean Distance	4	4	4	4	4	20
2								
3	8-2	Get Nearest Neighbors						
4	8-3	Make Predictions						
5								
6								
7								
8								
9								
10								
11								
12								


Signature of the Student


Signature of the Faculty

START WRITING FROM HERE

8.1 Calculate Euclidean Distance

```
from sklearn.datasets import load_iris
from scipy.spatial import distance
import numpy as np
```

```
i = load_iris()
```

```
d = i.data
```

```
def ed(p1, p2):
```

```
    return np.sqrt(np.sum((p1 - p2) ** 2))
```

```
n1 = int(input('Enter one point: '))
```

```
n2 = int(input('Enter two point: '))
```

```
de = ed(d[n1], d[n2])
```

```
print('Euclidean distance between point {} and  
point {}: {:.2f}'.format(n1, n2, de))
```

OUTPUT:

Enter one point: 0

Enter two point: 100

Euclidean distance between point 0 and point
100: 3.84

8.2 Get Nearest Neighbors

```

from sklearn.datasets import load_iris
from sklearn.neighbors import NearestNeighbors
i = load_iris()
d = i.data
n = int(input('Enter no. of neighbors to find:'))
kn = NearestNeighbors(n_neighbors = n)
kn.fit(d)
api = 0
di, in = kn.kneighbors([d[api]])
print('Nearest neighbors for point {} :='.format(
    api))

for i in range(n):
    print('Neighbor, {}, Index: {}, Distance: {:.2f}'.format(
        i + 1, in[i], di[i]))

```

OUTPUT:

```

Enter no. of neighbors to find: 5
Nearest neighbors for point 0:
Neighbor 1, Index: 0, Distance: 0.00
Neighbor 2, Index: 17, Distance: 0.14
Neighbor 3, Index: 4, Distance: 0.14
Neighbor 4, Index: 39, Distance: 0.14

```

8.3 Make Predictions

```

from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score

i = load_iris()
x = i.data
y = i.target

xt, xte, yt, yte = train_test_split(x, y, test_size=0.2,
                                     random_state=42)

kn = KNeighborsClassifier(int(input('Enter No. of
                                   Neighbors: ')))
kn.fit(xt, yt)

yp = kn.predict(xte)
a = accuracy_score(yte, yp)
print('Accuracy: ', a)

```

OUTPUT:

Enter No. of Neighbors: 3

Accuracy: 0.96666667