



**IARE**  
INSTITUTE OF  
AERONAUTICAL ENGINEERING  
(An Autonomous Institute affiliated to JNTUH, Hyderabad)  
Dundigal, Hyderabad - 500 043

## LABORATORY WORK BOOK

Name of the Student : HIMAKAR C

Class : CSE-B

Semester : VI

Course Code : ACIC08

Course Name : DMKD Laboratory

Roll Number

2 1 9 5 1 A 0 5 6 5

Name of the Course Faculty : DR. D. DURGA BHAVANI

Faculty ID : IARE 10921

Exercise Number : 9.1

Week Number : 09

Date :           

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	9.1	Build a decision tree classifier to determine kind of flower by given Dimensions	4	4		4	4	4	20
2									
3	9.2	Train with various split measures							
4									
5	9.3	Compare the accuracy							
6									
7									
8									
9									
0									
1									
2									

Himakar C  
Signature of the Student

DR. D. DURGA BHAVANI  
Signature of the Faculty

9.1 Build a decision tree classifier to determine the kind of flower by using given dimensions.

```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score,
ClassificationReport
```

```
i = load_iris()
```

```
x = i.data
```

```
y = i.target
```

```
x_t, x_te, y_t, y_te = train_test_split(x, y, test_size=0.2,
                                         random_state=42)
```

```
clf = DecisionTreeClassifier(random_state=42)
```

```
clf.fit(x_t, y_t)
```

```
y_p = clf.predict(x_te)
```

```
a = accuracy_score(y_te, y_p)
```

```
print('Accuracy:', a)
```

OUTPUT:

Accuracy: 1.0

q.2 Train with various split measures (Gini index, Entropy and Information Gain)

```

from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
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)

d = { 'Gini Index': DecisionTreeClassifier(
    criterion = 'gini', random_state=42),
      'Entropy': DecisionTreeClassifier(criterion = 'entropy',
    random_state = 42),
      'Information Gain': DecisionTreeClassifier(
    criterion = 'gini', splitter='best',
    random_state = 42) }

for j, k in d.items():
    clf = k.fit(xt, yt)
    yp = clf.predict(xte)
    a = accuracy_score(yte, yp)
    print(f'{j} - Accuracy: {a:.2f}')

```



OUTPUT:

Gini Index - Accuracy: 1.00

Entropy - Accuracy: 1.00

Information Gain - Accuracy: 1.00

## 9.3 Compare the accuracy

```

from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
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)

d = {
    'Decision Tree': DecisionTreeClassifier(random_state=42),
    'K-Nearest Neighbors': KNeighborsClassifier(),
    'Support Vector Machine': SVC(random_state=42)
}

```

```
for j, K in d.items():
```

```
    K.fit(Xt, yt)
```

```
    yp = K.predict(Xte)
```

```
    a = accuracy_score(yte, yp)
```

```
    print(f'{j} - Accuracy: {a:.2f}')
```

OUTPUT:

Decision Tree - Accuracy: 1.00

K-Nearest Neighbors - Accuracy: 1.00

Support Vector Machine - Accuracy: 1.00

BP