#Name:-Divya Vijay Alahit.

Roll No:-03.

Practical No:-

Practical Name:-Write a program to implement decision tree using the Python/R/Programming language of your choice.

==================================================================================

import matplotlib.pyplot as plt

import pandas as pd

from sklearn.datasets import load\_iris # load\_iris

data\_b = load\_iris() # lo

df = pd.DataFrame(data\_b.data, columns=data\_b.feature\_names)

df['target'] = data\_b.target

# df['target']

print(df)

print("Dataset Labels=", data\_b.target\_names)

from sklearn.tree import DecisionTreeClassifier

from sklearn import metrics

from sklearn.model\_selection import train\_test\_split

# import numpy as np

from sklearn import tree

X\_train, X\_test, Y\_train, y\_test = train\_test\_split(df[data\_b.feature\_names], df['target'], random\_state=1)

print(X\_train)

print(X\_test)

print(Y\_train)

print(y\_test)

clf = DecisionTreeClassifier(max\_depth=5, random\_state=1, criterion='gini') # 'gini'/'entropy'

clf.fit(X\_train, Y\_train)

y\_pred = clf.predict(X\_test)

print(y\_test, y\_pred)

print("Accuracy: ", metrics.accuracy\_score(y\_test, y\_pred))

# tree.plot\_tree(clf)

fn = ['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width (cm)']

cn = ['setosa', 'versicolor', 'virginica']

fig, axes = plt.subplots(nrows=1, ncols=1, figsize=(4, 4), dpi=300)

tree.plot\_tree(clf, feature\_names=fn, class\_names=cn, filled=True); fig.savefig('Dicision\_tree.png')

Output:-

sepal length (cm) sepal width (cm) petal length (cm) petal width (cm) \

0 5.1 3.5 1.4 0.2

1 4.9 3.0 1.4 0.2

2 4.7 3.2 1.3 0.2

3 4.6 3.1 1.5 0.2

4 5.0 3.6 1.4 0.2

.. ... ... ... ...

145 6.7 3.0 5.2 2.3

146 6.3 2.5 5.0 1.9

147 6.5 3.0 5.2 2.0

148 6.2 3.4 5.4 2.3

149 5.9 3.0 5.1 1.8

target

0 0

1 0

2 0

3 0

4 0

.. ...

145 2

146 2

147 2

148 2

149 2

[150 rows x 5 columns]

Dataset Labels= ['setosa' 'versicolor' 'virginica']

sepal length (cm) sepal width (cm) petal length (cm) petal width (cm)

54 6.5 2.8 4.6 1.5

108 6.7 2.5 5.8 1.8

112 6.8 3.0 5.5 2.1

17 5.1 3.5 1.4 0.3

119 6.0 2.2 5.0 1.5

.. ... ... ... ...

133 6.3 2.8 5.1 1.5

137 6.4 3.1 5.5 1.8

72 6.3 2.5 4.9 1.5

140 6.7 3.1 5.6 2.4

37 4.9 3.6 1.4 0.1

[112 rows x 4 columns]

sepal length (cm) sepal width (cm) petal length (cm) petal width (cm)

14 5.8 4.0 1.2 0.2

98 5.1 2.5 3.0 1.1

75 6.6 3.0 4.4 1.4

16 5.4 3.9 1.3 0.4

131 7.9 3.8 6.4 2.0

56 6.3 3.3 4.7 1.6

141 6.9 3.1 5.1 2.3

44 5.1 3.8 1.9 0.4

29 4.7 3.2 1.6 0.2

120 6.9 3.2 5.7 2.3

94 5.6 2.7 4.2 1.3

5 5.4 3.9 1.7 0.4

102 7.1 3.0 5.9 2.1

51 6.4 3.2 4.5 1.5

78 6.0 2.9 4.5 1.5

42 4.4 3.2 1.3 0.2

92 5.8 2.6 4.0 1.2

66 5.6 3.0 4.5 1.5

31 5.4 3.4 1.5 0.4

35 5.0 3.2 1.2 0.2

90 5.5 2.6 4.4 1.2

84 5.4 3.0 4.5 1.5

77 6.7 3.0 5.0 1.7

40 5.0 3.5 1.3 0.3

125 7.2 3.2 6.0 1.8

99 5.7 2.8 4.1 1.3

33 5.5 4.2 1.4 0.2

19 5.1 3.8 1.5 0.3

73 6.1 2.8 4.7 1.2

146 6.3 2.5 5.0 1.9

91 6.1 3.0 4.6 1.4

135 7.7 3.0 6.1 2.3

69 5.6 2.5 3.9 1.1

128 6.4 2.8 5.6 2.1

114 5.8 2.8 5.1 2.4

48 5.3 3.7 1.5 0.2

53 5.5 2.3 4.0 1.3

28 5.2 3.4 1.4 0.2

54 1

108 2

112 2

17 0

119 2

..

133 2

137 2

72 1

140 2

37 0

Name: target, Length: 112, dtype: int32

14 0

98 1

75 1

16 0

131 2

56 1

141 2

44 0

29 0

120 2

94 1

5 0

102 2

51 1

78 1

42 0

92 1

66 1

31 0

35 0

90 1

84 1

77 1

40 0

125 2

99 1

33 0

19 0

73 1

146 2

91 1

135 2

69 1

128 2

114 2

48 0

53 1

28 0

Name: target, dtype: int32

14 0

98 1

75 1

16 0

131 2

56 1

141 2

44 0

29 0

120 2

94 1

5 0

102 2

51 1

78 1

42 0

92 1

66 1

31 0

35 0

90 1

84 1

77 1

40 0

125 2

99 1

33 0

19 0

73 1

146 2

91 1

135 2

69 1

128 2

114 2

48 0

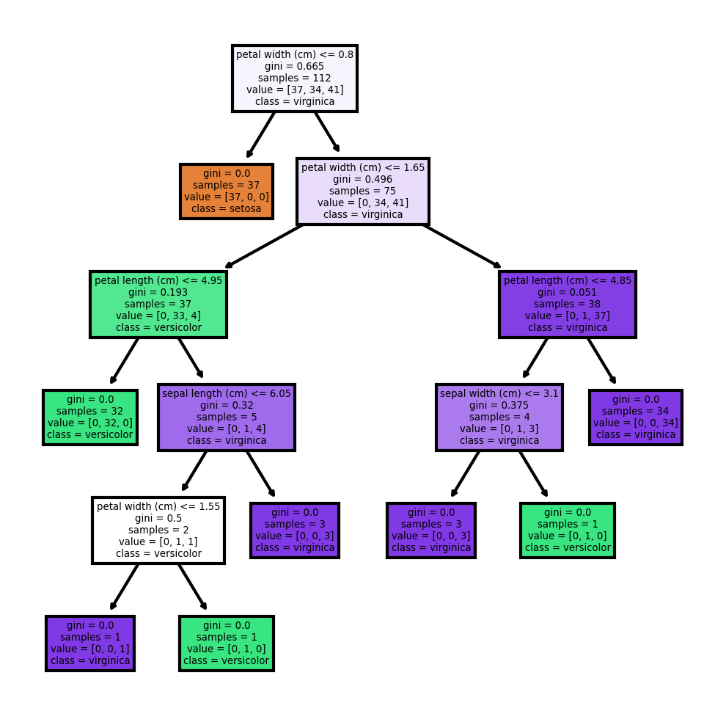
53 1

28 0

Name: target, dtype: int32 [0 1 1 0 2 1 2 0 0 2 1 0 2 1 1 0 1 1 0 0 1 1 2 0 2 1 0 0 1 2 1 2 1 2 2 0 1

0]

Accuracy: 0.9736842105263158



[ ]: