

DATA SCIENCE LAB

Experiment No.: 8

Aim

Decision Tree

Procedure

```
import matplotlib.pyplot as plt
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import pandas as pd
import numpy as np
from sklearn import tree
from sklearn.datasets import load_iris

data = load_iris()
df = pd.DataFrame(data.data, columns=data.feature_names)
df['target'] = data.target

X_train, X_test, Y_train, Y_test = train_test_split(df[data.feature_name]
, df['target'], random_state=0)

# Step 1: Import the model you want to use
# This was already imported earlier in the notebook so commenting out
# from sklearn.tree import DecisionTreeClassifier
# Step 2: Make an instance of the Model
clf = DecisionTreeClassifier(max_depth=2, random_state=0)
# Step 3: Train the model on the data
clf.fit(X_train, Y_train)
# Step 4: Predict labels of unseen (test) data
# Not doing this step in the tutorial
# clf.predict(X_test)
# 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)
y_pred = clf.predict(X_test)
```

Name: Manya Madhu

Roll No:17

Batch: RMCA-B

Date: 13-10-2022

```
print("Train data accuracy:",accuracy_score(y_true = Y_train, y_pred=clf.  
predict(X_train)))  
print("Test data accuracy:",accuracy_score(y_true = Y_test, y_pred=y_pred  
) )  
plt.show()
```

Output

Train data accuracy: 0.9642857142857143
Test data accuracy: 0.8947368421052632

