Date: ___ ___ _____

Data Analytics Laboratory
Task 3
Apply Decision Tree Classification technique on the given dataset

Introduction

- Decision tree models are the simplest form of supervised multivariate classification models.
- A series of logical tests (generally in the form of Boolean comparisons) are applied to the sample entries and their resulting subsets in turn to arrive at a final decision.
- It is very easy to visualize the decision process in a simple flowchart to trace the rational of every assignment made by a decision tree model, making it among the most interpretable of models.
- Decision trees are flow-chart-like tree structure, Internal node denotes a test on an attribute, Branch represents an outcome of the test, Leaf nodes represent class labels or class distribution.
- Decision tree generation consists of two phases.
    - Tree construction. Partition examples recursively based on selected attributes.
    - Tree pruning. Identify and remove branches that reflect noise or outliers

Prerequisites

1. What is the difference between classification and clustering? Justify the decision tree algorithm is used for classification or clustering.

2. Define information gain with an example.

3. For the same example dataset considered in question number 2, calculate entropy value.

Implement a Simple Decision Tree Classifier using Scikit Learn

Importing required laibraries and datasets

In [16]:

```
import pandas as pd
import numpy as np
from sklearn. datasets import load_iris
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
data=load_iris()
print('Classes to predict:',data.target_names)
```

```
Classes to predict: ['setosa' 'versicolor' 'virginica']
```

Storing the dependant and independent attributes in seperate variables

In [31]:

```
X=data.data
y=data.target
print("Number of records in dataset:",X.shape[0])
print(X[:4])
```

```
Number of records in dataset: 150
[[5.1 3.5 1.4 0.2]
 [4.9 3.  1.4 0.2]
 [4.7 3.2 1.3 0.2]
 [4.6 3.1 1.5 0.2]]
```

Splitting training and test dataset seperately

In [32]:

```
X_train,X_test,y_train, y_test=train_test_split
(X,y,random_state=22, test_size=0.45)
```

Initializing the DecisionTreeClassifier with entropy as splitting metrics

In [33]:

```
clf=DecisionTreeClassifier(criterion='entropy')
```

In [34]:

```
clf.fit(X_train,y_train)
```

Out[34]:

```
DecisionTreeClassifier(criterion='entropy')
```

In [35]:

```
y_pred=clf.predict(X_test)
```

In [36]:

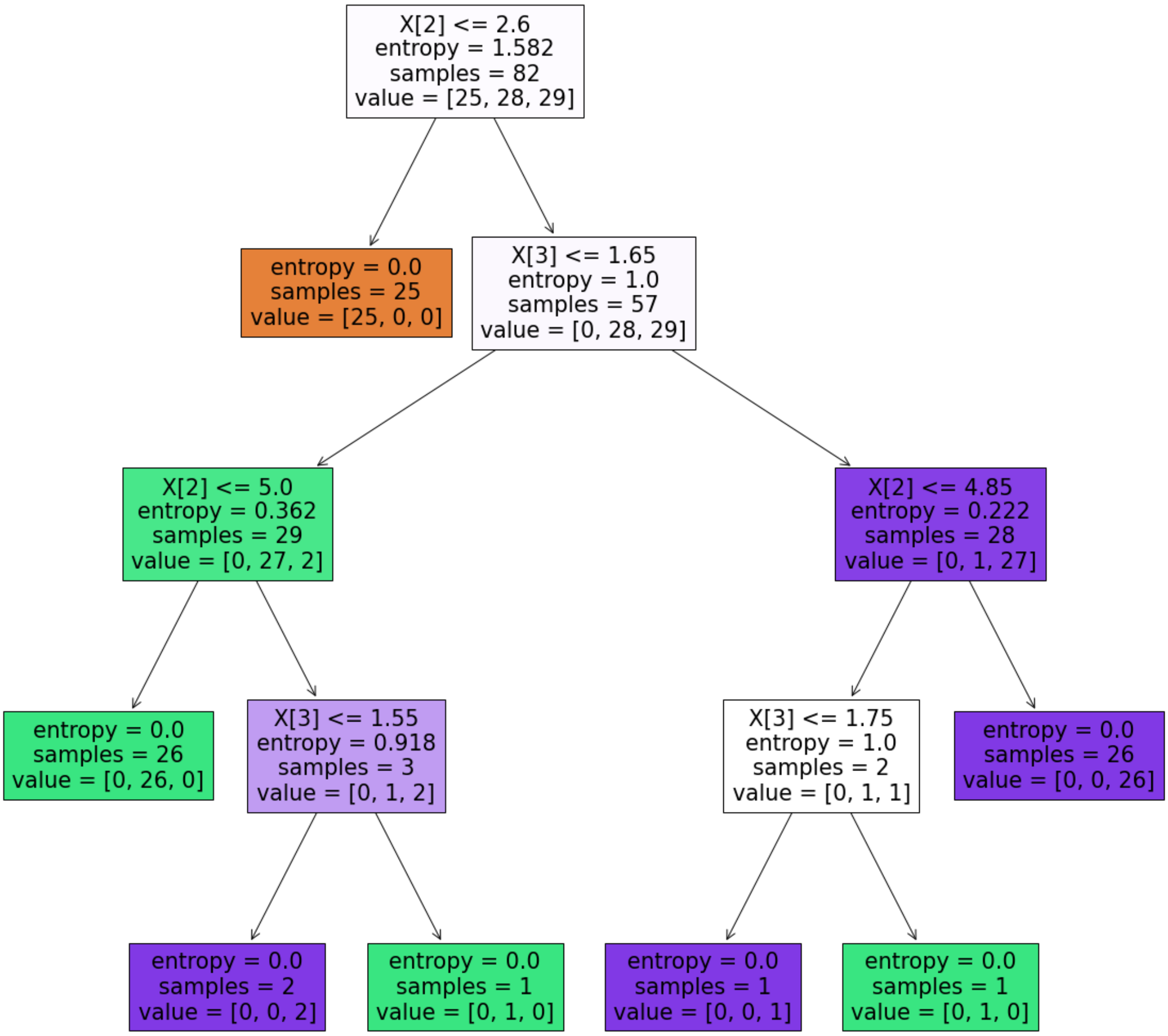
```
from sklearn.metrics import accuracy_score
print('Accuracy score on train data',accuracy_score
      (y_true=y_train,y_pred=clf.predict(X_train)))
print('Accuracy score on test data',accuracy_score
      (y_true=y_test,y_pred=y_pred))
```

```
Accuracy score on train data 1.0
Accuracy score on test data 0.9264705882352942
```

Printing the Decision tree

In [39]:

```
from sklearn.tree import DecisionTreeClassifier, plot_tree
plt.figure(figsize = (20,20))
plot_tree(clf,filled=True)
plt.show()
```



Apply Decision Tree Algorithm on the below given golf Dataset.

Weather	Temperature	Humidity	Wind	Golf Play
fine	hot	high	None	no
fine	hot	high	few	no
cloud	hot	high	none	yes
rain	warm	high	none	yes
rain	cold	medium	none	yes
rain	cold	medium	few	no
cloud	cold	medium	few	yes
fine	warm	high	none	no
fine	cold	medium	none	yes
rain	warm	medium	none	yes
fine	warm	medium	few	yes
cloud	warm	high	few	yes
cloud	hot	medium	none	yes
rain	warm	high	few	no

1. Use appropriate pre-processing techniques for encoding categorical data.
2. Draw the resultant decision tree.

Results

The program is implemented in python and the output is observed.

Faculty Signature