

## PROGRAM NO : 11

AIM : Program to implement Decision Tree using any standard dataset available in the public domain and find the accuracy of the algorithm

### PROGRAM

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.tree import plot_tree
df = sns.load_dataset('iris')
print(df.head())
print(df.info())
df.isnull().any()
print(df.shape)
sns.pairplot(data=df, hue='species')
plt.savefig("decision tree.png")
#correlation matrix
sns.heatmap(df.corr())
plt.savefig("one.png")
target=df['species']
df1=df.copy()
df1=df1.drop('species',axis=1)
print(df1.shape)
print(df1.head())
#defining the attribute
x=df1;
print(target)
#label encoding
le=LabelEncoder()
target=le.fit_transform(target)
print(target)
y=target
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=42)
print('Training split input- ',x_train.shape)
print('testing split input- ',x_test.shape)
#Defing the Decision tree algorithm
dtree=DecisionTreeClassifier()
dtree.fit(x_train,y_train)
y_pred=dtree.predict(x_test)
print('Classification Report - \n',classification_report(y_test,y_pred))
cm=confusion_matrix(y_test,y_pred)
plt.figure(figsize=(5,5))
```

```

sns.heatmap(data=cm,linewidth=5,annot=True,square=True,cmap="Blues")
plt.ylabel("Actual Label")
plt.xlabel("Predicted Label")
all_sample_title = 'Accuracy Score : {0}'.format(dtree.score(x_test,y_test))
plt.title(all_sample_title,size= 15)
plt.savefig("2.png")
#visualizong the graph without the use of graphics
plt.figure(figsize=(20,20))
dec_tre=plot_tree(decision_tree=dtree,feature_names=df1.columns,class_names=["s
etosa","vercolor","venginica"],filled=True,precision=4,rounded=True)
plt.savefig("3.png")

```

## OUTPUT

"C:\Users\ajcemca\PycharmProjects\decission tree\venv\Scripts\python.exe"

"C:/Users/ajcemca/PycharmProjects/decission tree/main.py"

```

sepal_length sepal_width petal_length petal_width species
0      5.1      3.5      1.4      0.2 setosa
1      4.9      3.0      1.4      0.2 setosa
2      4.7      3.2      1.3      0.2 setosa
3      4.6      3.1      1.5      0.2 setosa
4      5.0      3.6      1.4      0.2 setosa

```

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 150 entries, 0 to 149

Data columns (total 5 columns):

```
# Column      Non-Null Count  Dtype

```

```

---  ---
0  sepal_length  150 non-null  float64
1  sepal_width   150 non-null  float64
2  petal_length  150 non-null  float64
3  petal_width   150 non-null  float64
4  species       150 non-null  object

```

dtypes: float64(4), object(1)

memory usage: 6.0+ KB

None

(150, 5)

(150, 4)

```

sepal_length sepal_width petal_length petal_width
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
0      setosa

```

...

Name: species, Length: 150, dtype: object

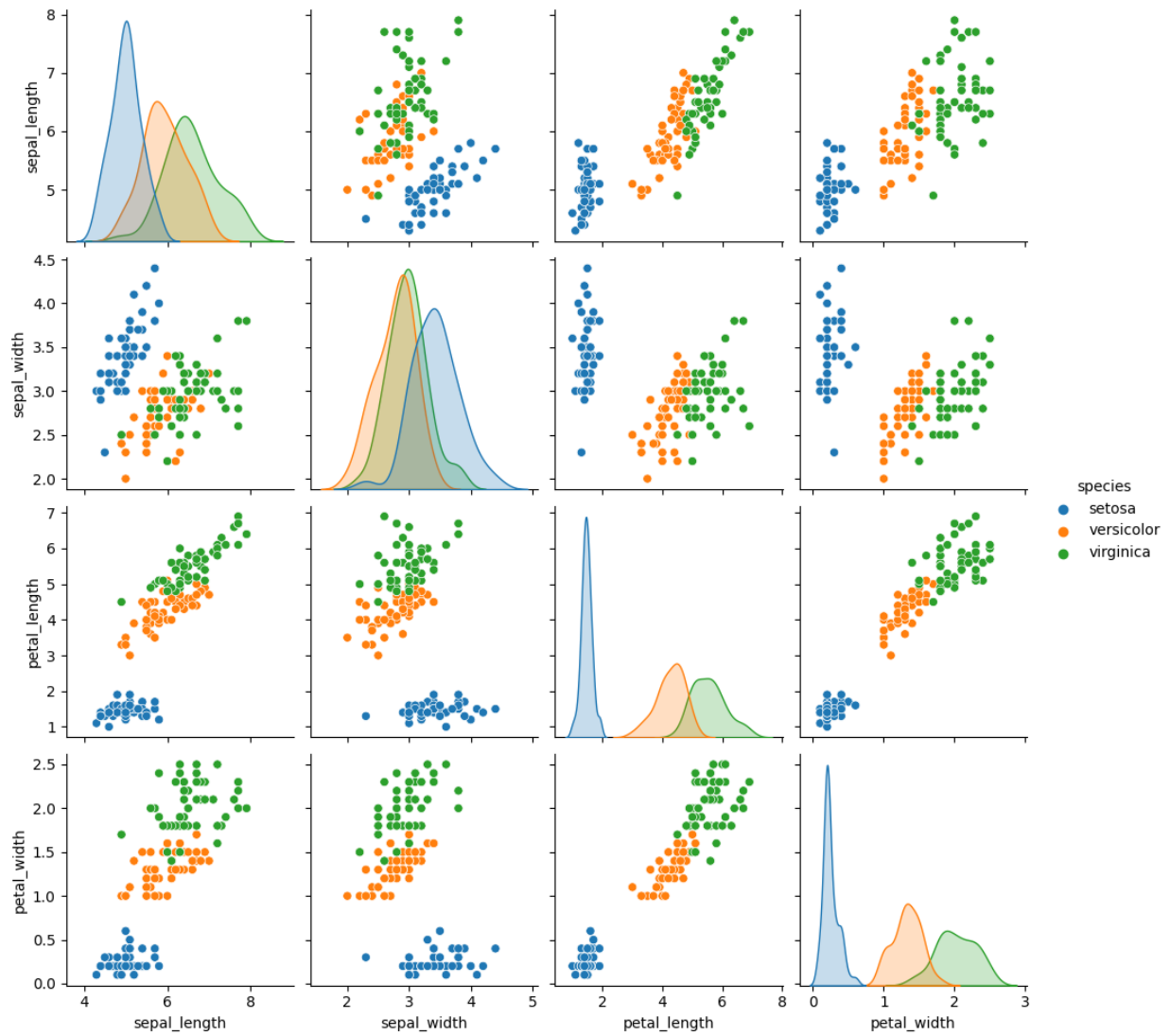
Training split input- (120, 4)

## Classification Report -

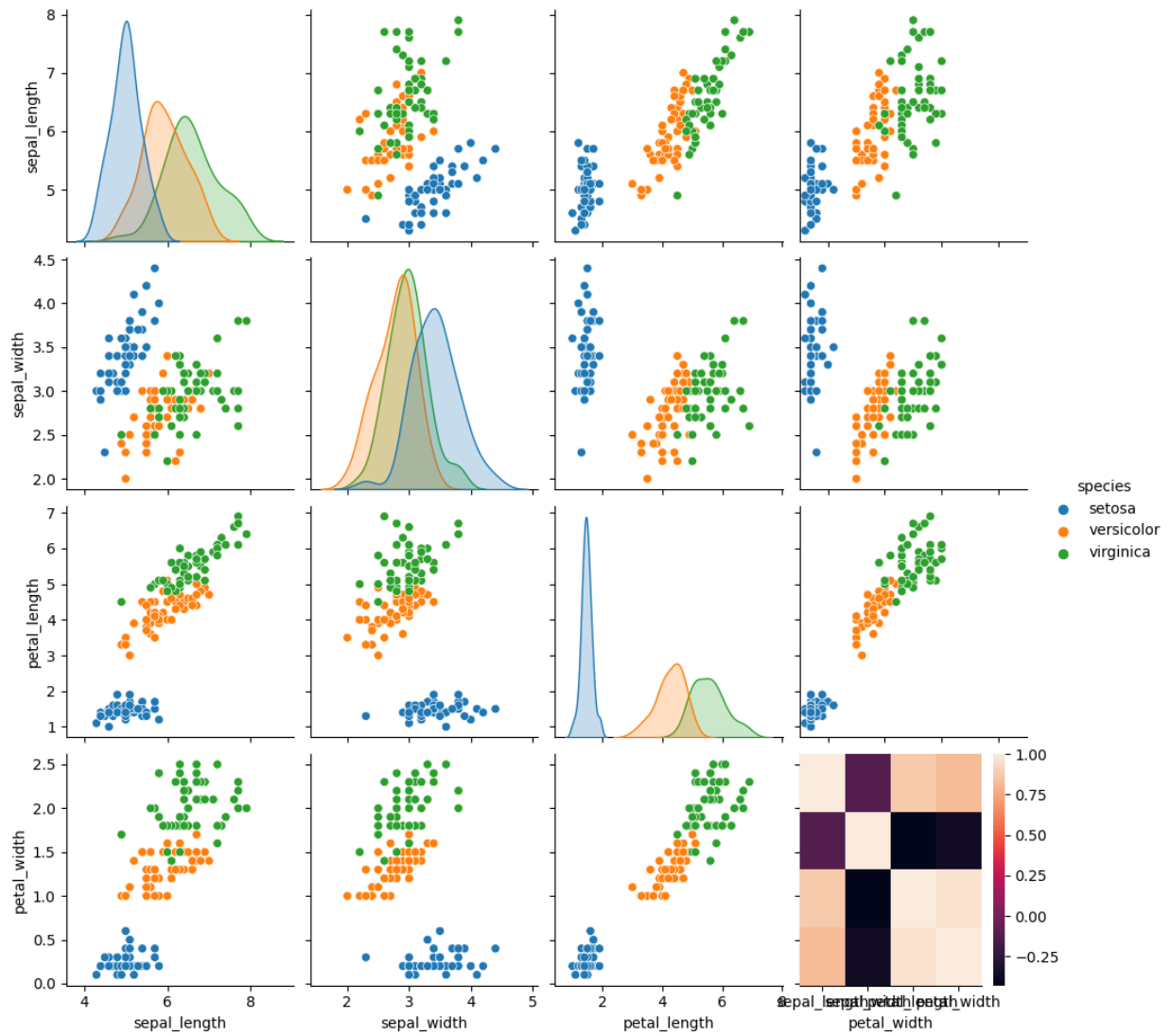
accuracy		1.00	30	
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

Process finished with exit code 0

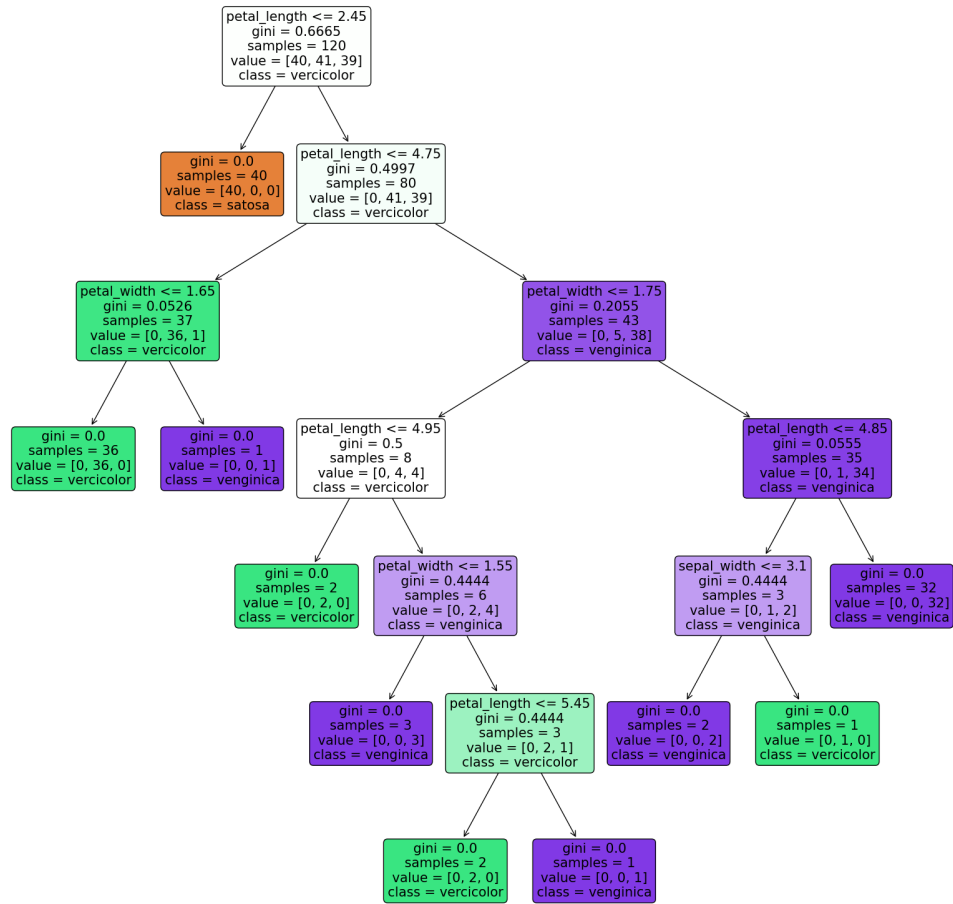
Decision Tree.png



One.png



3.png



2.png

