Program no:10

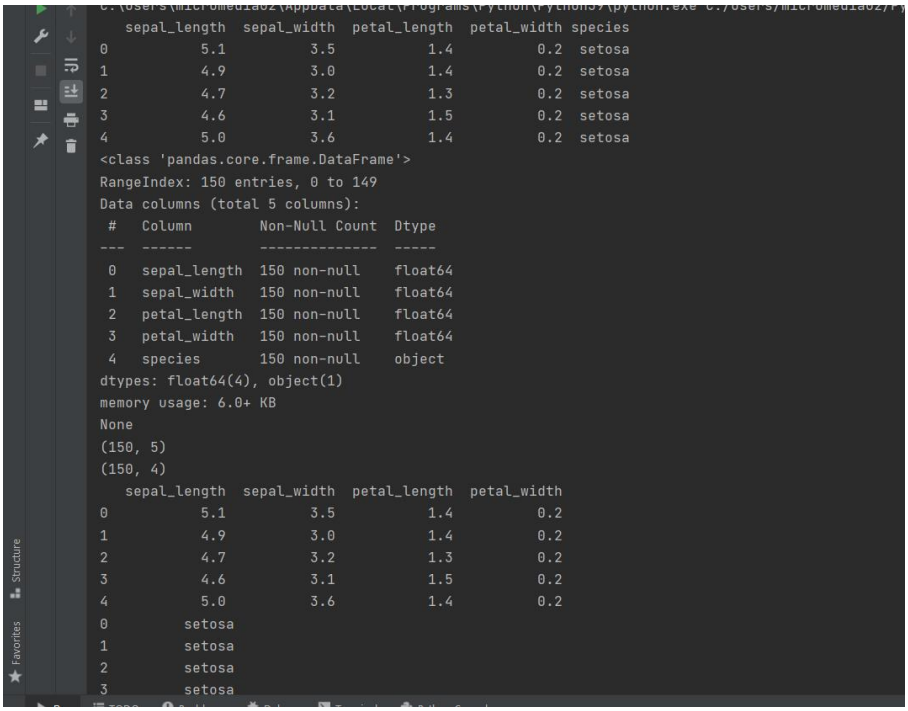
Aim :Program to implement decision tree using standard dataset available in the public domain and find the accuracy of the algorithm.

Program

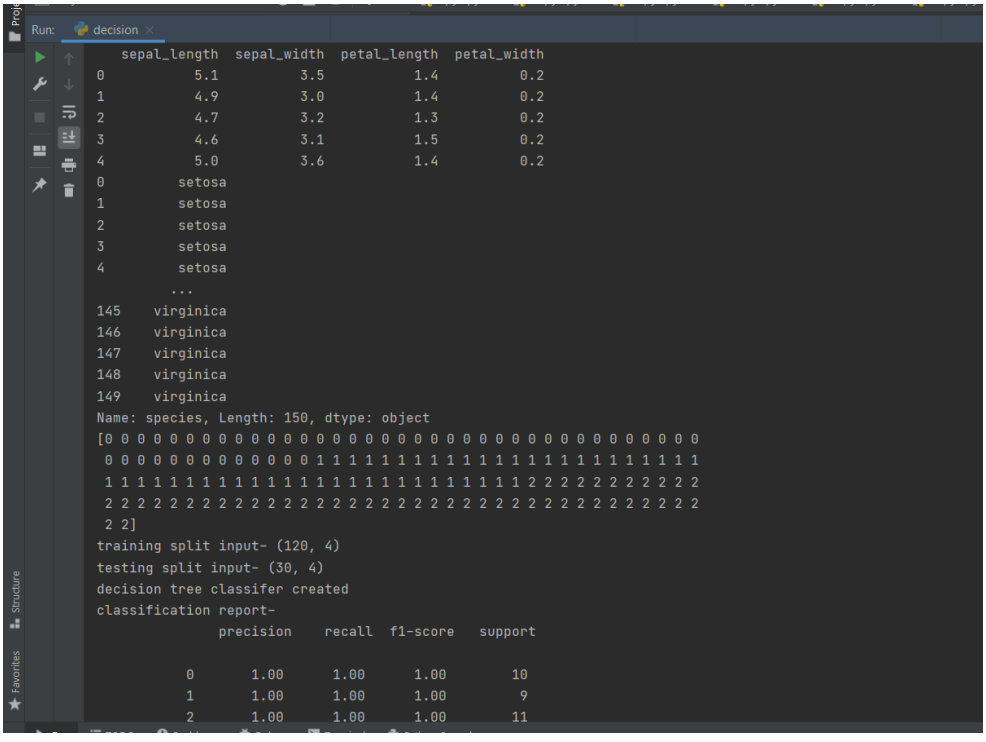
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import classification_report,confusion_matrix
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("pne.png")
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())
X = df1
print(target)
```

```
le = LabelEncoder()
target = le.fit_transform(target) # scaling parameter learn and printing
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)
dtree = DecisionTreeClassifier()
dtree.fit(X_train,y_train)
print("decision tree classifier created")
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,linewidths=.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("two.png")
plt.figure(figsize = (20,20))
dec_tree = plot_tree(decision_tree=dtree,feature_names= df1.columns,
                     class_names=["setosa","vercicolor","verginica"],filled = True, precision =4
,rounded =True)
plt.savefig("three.png")
```

OUTPUT



```
   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
1         setosa
2         setosa
3         setosa
```

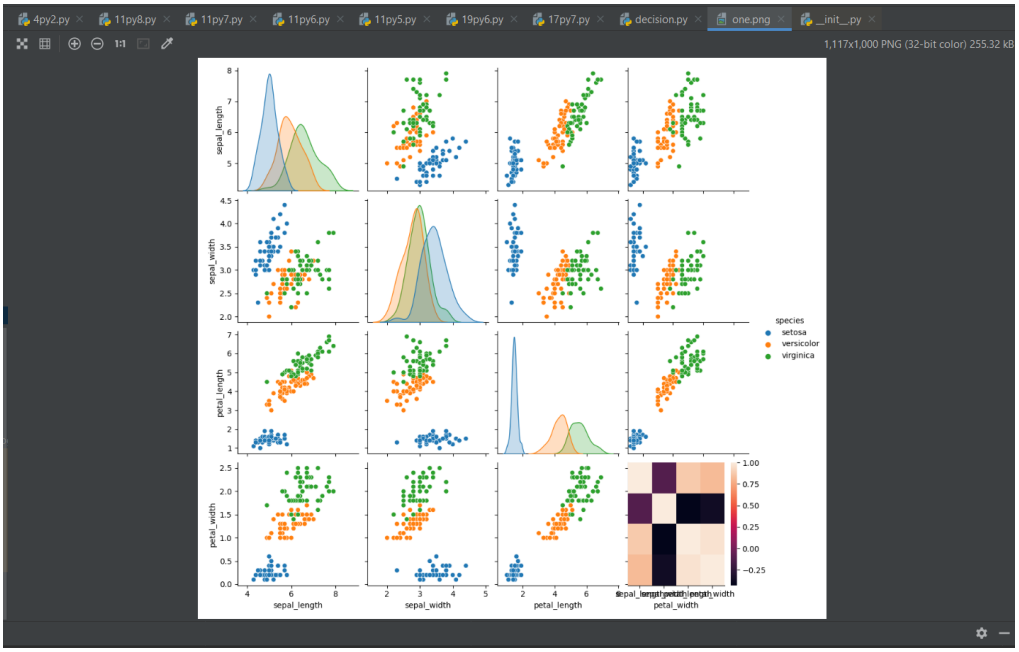


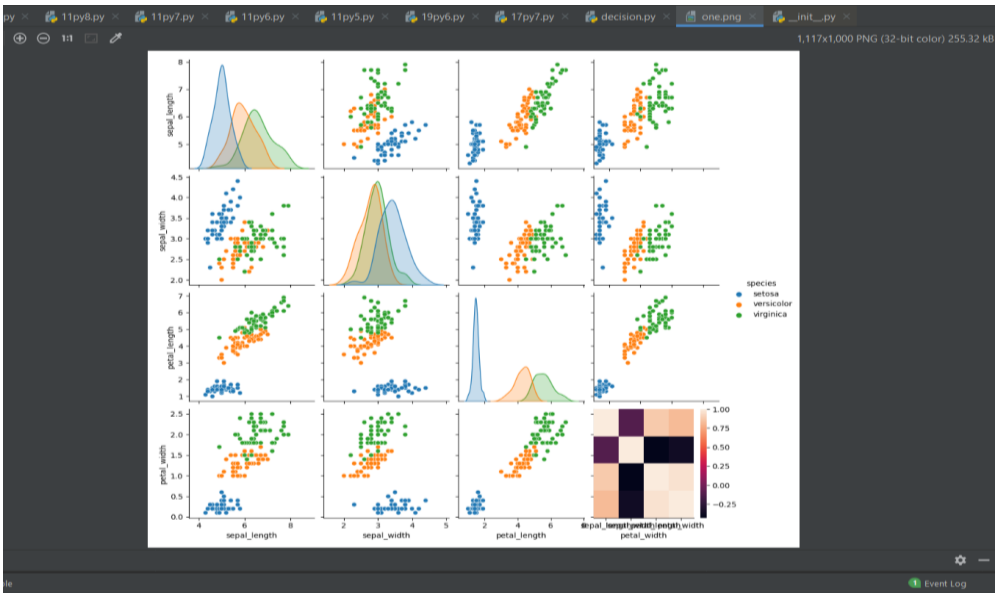
```
Run:  decision
   sepal_length  sepal_width  petal_length  petal_width
0           5.1          3.5           1.4          0.2
1           4.9          3.0           1.4          0.2
2           4.7          3.2           1.3          0.2
3           4.6          3.1           1.5          0.2
4           5.0          3.6           1.4          0.2
0         setosa
1         setosa
2         setosa
3         setosa
4         setosa
        ...
145    virginica
146    virginica
147    virginica
148    virginica
149    virginica
Name: species, Length: 150, dtype: object
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2]
training split input- (120, 4)
testing split input- (30, 4)
decision tree classifer created
classification report-
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        10
           1       1.00      1.00      1.00         9
           2       1.00      1.00      1.00        11
```

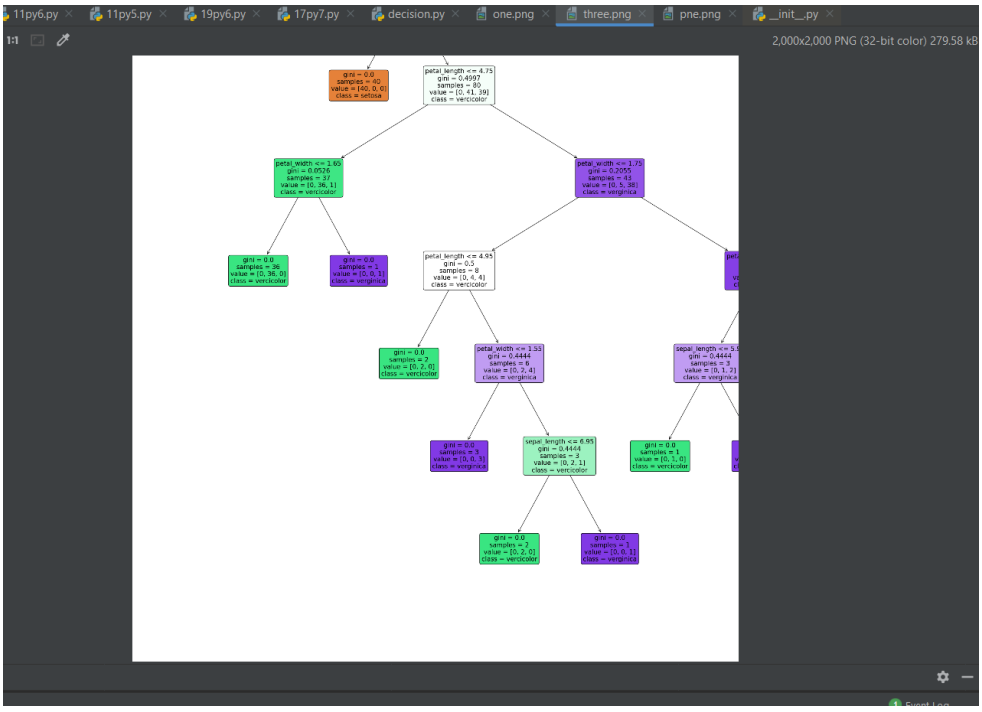
```
training split input- (120, 4)
testing split input- (30, 4)
decision tree classifer created
classification report-
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        10
           1       1.00      1.00      1.00         9
           2       1.00      1.00      1.00        11

    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
```

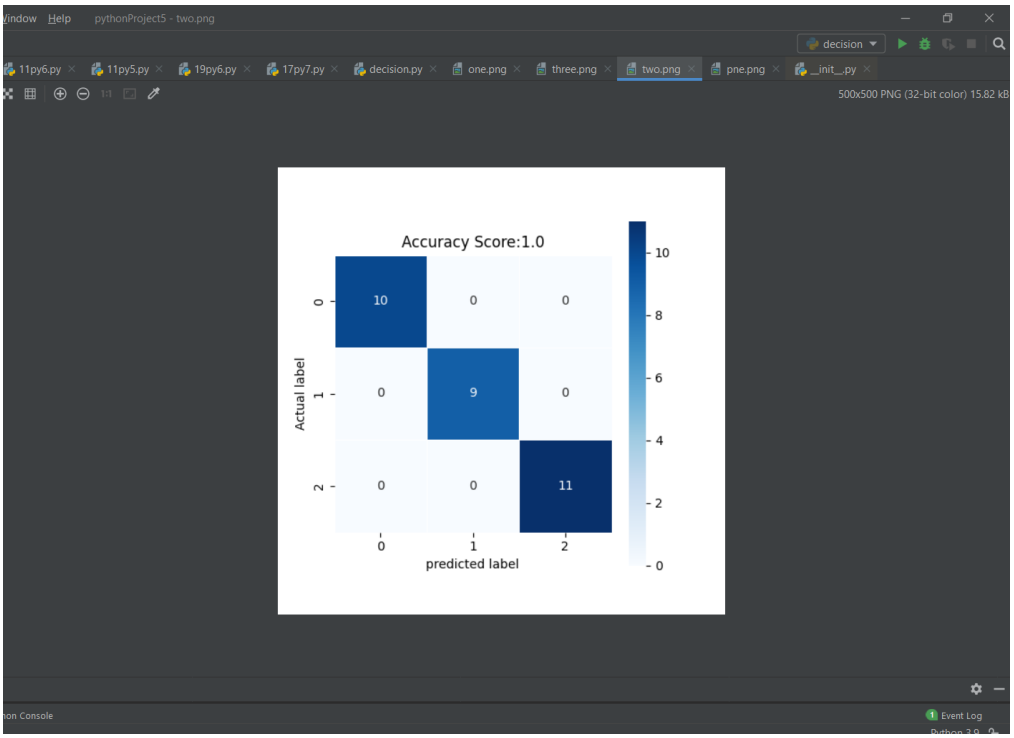
Structure Favorites

Run TODO Problems Debug Terminal Python Console




11py8.py
11py7.py
11py6.py
11py5.py
19py6.py
17py7.py
decision.py
one.png
__init__.py
1,117x1,000 PNG (32-bit color) 255.32 kB
sepal_length
sepal_width
petal_length
petal_width
species
setosa
versicolor
virginica
Event Log


11py6.py
11py5.py
19py6.py
17py7.py
decision.py
one.png
three.png
pne.png
__init__.py
2,000x2,000 PNG (32-bit color) 279.58 kB
petal_length <= 4.75
gini = 0.4997
samples = 80
value = [0, 41, 39]
class = versicolor
gini = 0.0
samples = 40
value = [40, 0, 0]
class = setosa
petal_width <= 1.65
gini = 0.0526
samples = 37
value = [0, 36, 1]
class = versicolor
petal_width <= 1.75
gini = 0.2055
samples = 43
value = [0, 5, 38]
class = virginica
gini = 0.0
samples = 36
value = [0, 36, 0]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica
petal_length <= 4.85
gini = 0.5
samples = 8
value = [0, 4, 4]
class = versicolor
gini = 0.0
samples = 2
value = [0, 2, 0]
class = versicolor
petal_width <= 1.53
gini = 0.4444
samples = 6
value = [0, 2, 4]
class = virginica
gini = 0.0
samples = 3
value = [0, 0, 3]
class = virginica
sepal_length <= 6.95
gini = 0.4444
samples = 3
value = [0, 2, 1]
class = versicolor
gini = 0.0
samples = 2
value = [0, 2, 0]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica


Window Help
pythonProject5 - two.png
decision
11py6.py
11py5.py
19py6.py
17py7.py
decision.py
one.png
three.png
two.png
pne.png
__init__.py
500x500 PNG (32-bit color) 15.82 kB
Accuracy Score:1.0
Actual label
predicted label
10
0
0
0
9
0
0
0
11
0
1
2
10
8
6
4
2
0
Event Log