Data Science And Bussiness Analytics Intern At TheSparksFoundation

GRIPJAN21

Author- Rajat Kumar

Task 6:-Prediction using Decision Tree Algorithm

Problem Statement:- **Create the Decision Tree classifier and visualize it graphically**

Goal:- The goal of this project is if we feed any new data to this classifier, it would be able to predict the right class accordingly

Step:1 Importing Some Libraries

In [26]:

```
from sklearn import tree
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

import pandas as pd
import numpy as np
import seaborn as sns
from matplotlib import pyplot as plt
```

Step:2 Loading Dataset

In [28]:

```
df=pd.read_csv(r"E:\TSF\Task5\raj.csv")

data.head() # Finding first five Rows of dataset
```

Out[28]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

In [50]:

```
df.isnull().sum() # Checking for is there any null value or not
```

Out[50]:

```
Id               0
SepalLengthCm    0
SepalWidthCm     0
PetalLengthCm    0
PetalWidthCm     0
Species          0
dtype: int64
```

In [51]:

```
df.shape # This will give total number of rows and columns.
```

Out[51]:

```
(150, 6)
```

Step:3 Data Preprocessing

In [29]:

```
x=data.drop(['Id','Species'],axis=1)
y=data.Species
```

In [34]:

```
feature_names=data.columns[1:-1] # droping the id column and the target column
feature_names
```

Out[34]:

```
Index(['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm'], dtype='object')
```

Step:4 Total classes in our dataset

```
In [35]: target_names=y.unique()
         target_names
```

```
Out[35]: array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

Step:5 Mapping the classes into numerical

```
In [36]: y=y.map({'Iris-setosa':0, 'Iris-versicolor':1, 'Iris-virginica':2})
```

Step:6 Spliting the dataset into training and test dataset

```
In [37]: x_train,x_test,y_train,y_test,=train_test_split(x.values,y.values,test_size=0.2)
```

Step 7:Training and algorithm

```
In [41]: model=tree.DecisionTreeClassifier()
         model.fit(x_train,y_train)
```

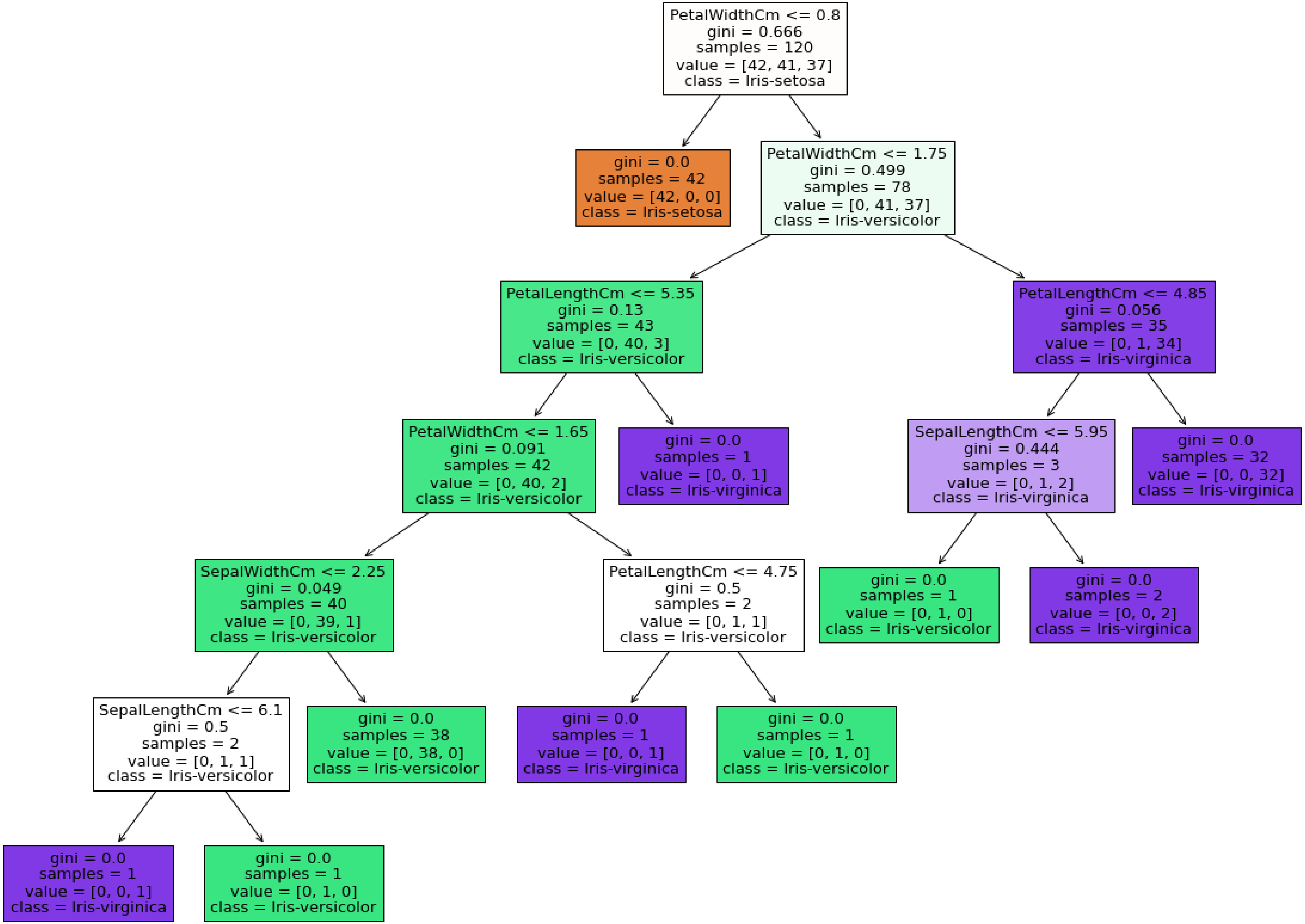
```
Out[41]: DecisionTreeClassifier()
```

```
In [42]: predictions=model.predict(x_test)
         acc=accuracy_score(predictions,y_test)
         print(f'Accuracy is:{acc*100} %')
```

```
Accuracy is:90.0 %
```

Step:8 Visualize Decision Tree

```
In [48]: fig=plt.figure(figsize=(20,15))
         _=tree.plot_tree(model,feature_names=feature_names,class_names=target_names,filled=True)
```


PetalWidthCm <= 0.8
gini = 0.666
samples = 120
value = [42, 41, 37]
class = Iris-setosa
gini = 0.0
samples = 42
value = [42, 0, 0]
class = Iris-setosa
PetalWidthCm <= 1.75
gini = 0.499
samples = 78
value = [0, 41, 37]
class = Iris-versicolor
PetalLengthCm <= 5.35
gini = 0.13
samples = 43
value = [0, 40, 3]
class = Iris-versicolor
PetalLengthCm <= 4.85
gini = 0.056
samples = 35
value = [0, 1, 34]
class = Iris-virginica
PetalWidthCm <= 1.65
gini = 0.091
samples = 42
value = [0, 40, 2]
class = Iris-versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = Iris-virginica
SepalLengthCm <= 5.95
gini = 0.444
samples = 3
value = [0, 1, 2]
class = Iris-virginica
gini = 0.0
samples = 32
value = [0, 0, 32]
class = Iris-virginica
SepalWidthCm <= 2.25
gini = 0.049
samples = 40
value = [0, 39, 1]
class = Iris-versicolor
PetalLengthCm <= 4.75
gini = 0.5
samples = 2
value = [0, 1, 1]
class = Iris-versicolor
gini = 0.0
samples = 1
value = [0, 1, 0]
class = Iris-versicolor
gini = 0.0
samples = 2
value = [0, 0, 2]
class = Iris-virginica
SepalLengthCm <= 6.1
gini = 0.5
samples = 2
value = [0, 1, 1]
class = Iris-versicolor
gini = 0.0
samples = 38
value = [0, 38, 0]
class = Iris-versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = Iris-virginica
gini = 0.0
samples = 1
value = [0, 1, 0]
class = Iris-versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = Iris-virginica
gini = 0.0
samples = 1
value = [0, 1, 0]
class = Iris-versicolor

Thank you

In [ ]: