GRIP – The Sparks Foundation

Data Science and Business Analytics Internship

Prediction using Decision Tree Algorithm

Name: Tejdeep Sai Iddum

Contact: linkedin.com/in/tejdeep-sai-iddum-61025a1a6

Aim:

- Create the Decision Tree classifier and visualize it graphically.
- The purpose is if we feed any new data to this classifier, it would be able to predict the right class accordingly.

Procedure:

In this task we shall

- Import the necessary libraries
- Load the dataset
- Perform basic analysis on the given dataset to observe and gain insight
- Define a Decision Tree algorithm
- Visualize the Decision Tree

In [46]:

```
# Importing Libraries in Python
import pandas as pd
import matplotlib.pyplot as plt
```

In [47]:

```
# Loading the iris dataset
from sklearn.datasets import load_iris
iris = load_iris()
data = pd.read_csv("C:/Users/User/Downloads/iris.csv")
```

In [48]:

```
# Obtaining the first 5 values from the dataset
data.head()
```

Out[48]:

	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 [49]:

```
# Obtaining the data type for each column
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Id             150 non-null    int64  
 1   SepalLengthCm  150 non-null    float64
 2   SepalWidthCm   150 non-null    float64
 3   PetalLengthCm  150 non-null    float64
 4   PetalWidthCm   150 non-null    float64
 5   Species        150 non-null    object 
dtypes: float64(4), int64(1), object(1)
memory usage: 6.5+ KB
```

In [50]:

```
# Checking if null values are present in the dataset
data.isnull().sum()
```

Out[50]:

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

In [51]:

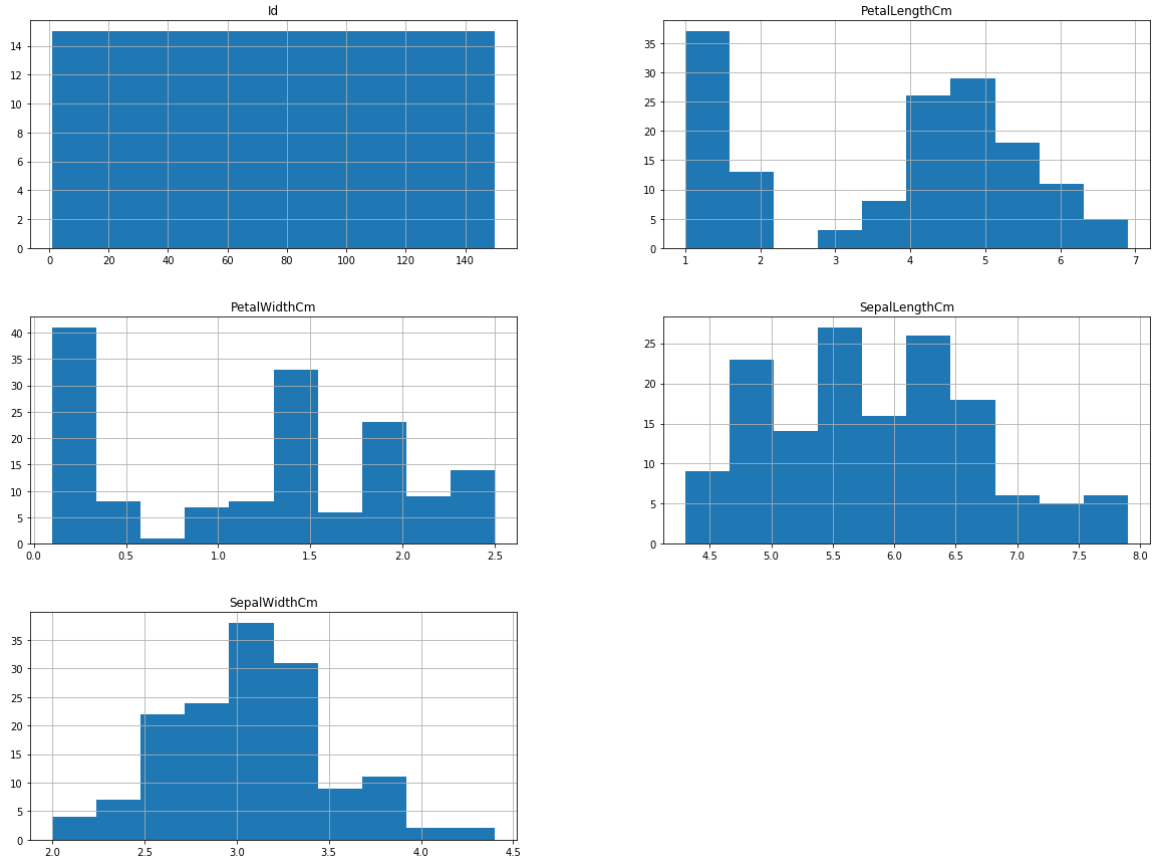
```
# Obtaining basic statistical details of the dataset
data.describe()
```

Out[51]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

In [52]:

```
# Plotting histogram for each column to obtain the distribution of data
num_bins = 10
data.hist(bins=num_bins, figsize=(20,15))
plt.show()
```



In [53]:

```
# Defining the decision tree algorithm
from sklearn.tree import DecisionTreeClassifier
dtree=DecisionTreeClassifier(criterion = 'entropy', random_state = 0)
dtree = dtree.fit(iris.data, iris.target)
```

In [54]:

```
# Visualization of Decision Tree
import graphviz
dot_data = tree.export_graphviz(dtree, out_file=None,feature_names=iris.feature_names,c
lass_names=iris.target_names,filled=True, rounded=True,special_characters=True)
graph = graphviz.Source(dot_data)
```

In [55]:

```
graph
```

Out[55]: