Random Forest

Libraries

In [29]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix,accuracy_score
from sklearn.ensemble import RandomForestClassifier
```

Import dataset

In [30]:

```
dataset= pd.read_csv("F:/Data set/Iris_new.csv")
```

In [31]:

```
dataset.info()
```

```
<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   spectype      150 non-null    object 
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

Index location

In [32]:

```
x=dataset.iloc[:,0:4].values
y=dataset.iloc[:,-1].values
```

Split Train set and Test set

In [33]:

```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3,random_state=0)
```

Train the Random forest model in train set

In [34]:

```
classifier=RandomForestClassifier(n_estimators=160,criterion='entropy',random_state=0)
classifier.fit(x_train,y_train)
```

Out[34]:

```
RandomForestClassifier(criterion='entropy', n_estimators=160, random_state=0)
```

Predict the test set results

In [35]:

```
y_pred = classifier.predict(x_test)
```

Accuracy

In [37]:

```
cm=confusion_matrix(y_test,y_pred)
acc=sum(np.diag(cm)/len(y_test))
print(cm)
print(acc)
```

```
[[16  0  0]
 [ 0 17  1]
 [ 0  0 11]]
0.9777777777777779
```

In [38]:

```
#or else

accuracy=accuracy_score(y_test,y_pred)
print(accuracy)
```

```
0.9777777777777777
```