

## Random Forest

In [ ]:

```
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report
from sklearn.datasets import load_iris
iris=load_iris()
```

In [ ]:

```
#dimensions of dataset
X,y=iris.data,iris.target
print(X.shape)
print(y.shape)
```

```
(150, 4)
(150,)
```

In [ ]:

```
#Splitting the dataset
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=100)
print("Dimensions for training data",X_train.shape)
print("Dimensions for testing data",y_train.shape)
```

```
Dimensions for training data (120, 4)
Dimensions for testing data (120,)
```

In [ ]:



```
#Random Forest classification
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators=30)
classifier.fit(X_train,y_train)
y_pred=classifier.predict(X_test)
accuracy=accuracy_score(y_test,y_pred)
print("Accuracy:",(accuracy)*100)
print("Confusion Matrix: ",confusion_matrix(y_test,y_pred))
print("Report :",classification_report(y_test,y_pred))
```

Accuracy: 96.66666666666667

Confusion Matrix: [[11 0 0]

[ 0 5 1]

[ 0 0 13]]

Report :	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	1.00	0.83	0.91	6
2	0.93	1.00	0.96	13
accuracy			0.97	30
macro avg	0.98	0.94	0.96	30
weighted avg	0.97	0.97	0.97	30