# Import Library

```
import pandas as pd
```

```
import numpy as np
```

# Import Dataset

```
df = pd.read_csv(r'https://github.com/YBI-Foundation/Dataset/raw/main/IRIS.csv')
```

```
df.head()
```

| | sepal_length | sepal_width | petal_length | petal_width | species |
|---|---|---|---|---|---|
| **0** | 5.1 | 3.5 | 1.4 | 0.2 | Iris-setosa |
| **1** | 4.9 | 3.0 | 1.4 | 0.2 | Iris-setosa |
| **2** | 4.7 | 3.2 | 1.3 | 0.2 | Iris-setosa |
| **3** | 4.6 | 3.1 | 1.5 | 0.2 | Iris-setosa |
| **4** | 5.0 | 3.6 | 1.4 | 0.2 | Iris-setosa |

```
df.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   species       150 non-null    object 
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

# Define independent and dependent variable

```
X = df.drop('species',axis = 1)
```

```
y = df['species']
```

```
X.shape,y.shape
```

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

## Train_test_split

```
from sklearn.model_selection import train_test_split
```

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size = 0.3,random_state =
```

```
X_train.shape,X_test.shape,y_train.shape,y_test.shape
```

```
((105, 4), (45, 4), (105,), (45,))
```

```
from sklearn.ensemble import BaggingClassifier
```

```
from sklearn.svm import SVC
```

```
bagging = BaggingClassifier(base_estimator=SVC(),n_estimators= 20,max_samples = 0.7,
```

```
bagging.fit(X_train,y_train)
```

```
BaggingClassifier(base_estimator=SVC(), max_samples=0.7, n_estimators=20,
                  random_state=2529)
```

```
y_pred = bagging.predict(X_test)
```

## Model Evaluation

```
from sklearn.metrics import confusion_matrix,classification_report
```

```
confusion_matrix(y_pred,y_test)
```

```
array([[15,  0,  0],
       [ 0, 15,  4],
       [ 0,  0, 11]])
```

```
print(classification_report(y_pred,y_test))
```

```
              precision    recall  f1-score   support
```

```
     Iris-setosa       1.00      1.00      1.00        15
 Iris-versicolor       1.00      0.79      0.88        19
  Iris-virginica       0.73      1.00      0.85        11

        accuracy                           0.91        45
       macro avg       0.91      0.93      0.91        45
    weighted avg       0.93      0.91      0.91        45
```

## Ensemble Model

```
from sklearn.ensemble import AdaBoostClassifier
```

```
ada = AdaBoostClassifier()
```

```
ada.fit(X_train,y_train)

     AdaBoostClassifier()
```

```
y_pred_ada = ada.predict(X_test)
```

## Model Evaluation

```
confusion_matrix(y_pred_ada,y_test)

     array([[15,  0,  0],
            [ 0, 13,  3],
            [ 0,  2, 12]])
```

```
print(classification_report(y_pred_ada,y_test))

                      precision    recall  f1-score   support

         Iris-setosa       1.00      1.00      1.00        15
     Iris-versicolor       0.87      0.81      0.84        16
      Iris-virginica       0.80      0.86      0.83        14

            accuracy                           0.89        45
           macro avg       0.89      0.89      0.89        45
        weighted avg       0.89      0.89      0.89        45
```

## Hyper Parameter Tunning

● ✕