

Step 1: Importing the Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

Step 2: Importing the dataset

```
df = pd.read_csv("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	

Step 3: Checking for null values

```
df.isnull().sum()
```

```
sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

Saved successfully!

Step 4: Splitting the dataset into the Training set and Test set

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2)
```

Step 5: Feature Scaling

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.transform(x_test)
```

Step 6: Training the Naive Bayes Classification model on the Training Set

```
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(x_train, y_train)
```

```
▼ GaussianNB
GaussianNB()
```

Step 7: Predicting the Test set results

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

```
array(['Iris-virginica', 'Iris-virginica', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-versicolor', 'Iris-setosa', 'Iris-versicolor',
       'Iris-virginica', 'Iris-setosa', 'Iris-virginica', 'Iris-setosa',
       'Iris-setosa', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-virginica', 'Iris-versicolor', 'Iris-virginica',
       'Iris-virginica', 'Iris-setosa', 'Iris-virginica',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
```

```
'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa', 'Iris-setosa',  
'Iris-versicolor', 'Iris-versicolor'], dtype='<U15')
```

+ Code


+ Text

Step 8: Confusion Matrix and Accuracy

```
from sklearn.metrics import confusion_matrix  
cm = confusion_matrix(y_test, y_pred)  
from sklearn.metrics import accuracy_score  
print ("Accuracy : ", accuracy_score(y_test, y_pred))  
cm  
  
Accuracy : 0.9666666666666667  
array([[10,  0,  0],  
       [ 0, 12,  1],  
       [ 0,  0,  7]])
```

Step 9: Comparing the Real Values with Predicted Values

```
df = pd.DataFrame({'Real Values':y_test, 'Predicted Values':y_pred})  
df.head()
```

	Real Values	Predicted Values	
0	Iris-virginica	Iris-virginica	
1	Iris-virginica	Iris-virginica	
2	Iris-setosa	Iris-setosa	
3	Iris-setosa	Iris-setosa	
4	Iris-setosa	Iris-setosa	

Saved successfully!

×