

```

# import necessary lib
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import confusion_matrix, accuracy_score,
precision_score, recall_score

iris = pd.read_csv('iris.csv')

print(iris.columns)

Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm',
       'PetalWidthCm',
       'Species'],
      dtype='object')

# split the features and target
X =
iris[['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm']]
y = iris['Species']

#Split into training and testing set

X_train,X_test,y_train,y_test = train_test_split(X,y,test_size =
.25,random_state = 0)

# initialise naive bayes classifier
nb_classifier = GaussianNB()

# train the model
nb_classifier.fit(X_train,y_train)

GaussianNB()

y_pred = nb_classifier.predict(X_test)

#confusion matrix

cm = confusion_matrix(y_test,y_pred)
print(f"Confusion matrix :\n {cm}")

Confusion matrix :
[[13  0  0]
 [ 0 16  0]
 [ 0  0  9]]

# metrics
accuracy = accuracy_score(y_test,y_pred)
error_rate = 1 - accuracy
precision = precision_score(y_test,y_pred,average = 'macro')
recall = recall_score(y_test,y_pred, average = 'macro')

```

```
# print matrix
print(f"Accuracy:{accuracy}")
print(f"Error Rate:{error_rate}")
print(f"Precision: {precision}")
print(f"Recall: {recall}")
```

```
Accuracy:1.0
Error Rate:0.0
Precision: 1.0
Recall: 1.0
```