

```
In [3]: import pandas as pd
import numpy as np
```

```
In [4]: iris=pd.read_csv("Iris.csv")
iris
```

Out[4]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species	
	0	1	5.1	3.5	1.4	0.2	Iris-setosa
	1	2	4.9	3.0	1.4	0.2	Iris-setosa
	2	3	4.7	3.2	1.3	0.2	Iris-setosa
	3	4	4.6	3.1	1.5	0.2	Iris-setosa
	4	5	5.0	3.6	1.4	0.2	Iris-setosa
	...	...	...	...	...	...	...
	145	146	6.7	3.0	5.2	2.3	Iris-virginica
	146	147	6.3	2.5	5.0	1.9	Iris-virginica
	147	148	6.5	3.0	5.2	2.0	Iris-virginica
	148	149	6.2	3.4	5.4	2.3	Iris-virginica
	149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

```
In [7]: X=iris.drop(['Species'], axis=1)
```

```
In [8]: X
```

Out[8]:

	<b>Id</b>	<b>SepalLengthCm</b>	<b>SepalWidthCm</b>	<b>PetalLengthCm</b>	<b>PetalWidthCm</b>
<b>0</b>	1	5.1	3.5	1.4	0.2
<b>1</b>	2	4.9	3.0	1.4	0.2
<b>2</b>	3	4.7	3.2	1.3	0.2
<b>3</b>	4	4.6	3.1	1.5	0.2
<b>4</b>	5	5.0	3.6	1.4	0.2
...	...	...	...	...	...
<b>145</b>	146	6.7	3.0	5.2	2.3
<b>146</b>	147	6.3	2.5	5.0	1.9
<b>147</b>	148	6.5	3.0	5.2	2.0
<b>148</b>	149	6.2	3.4	5.4	2.3
<b>149</b>	150	5.9	3.0	5.1	1.8

150 rows × 5 columns

```
In [10]: Y=iris['Species']
```

```
In [11]: Y
```

```
Out[11]: 0      Iris-setosa
          1      Iris-setosa
          2      Iris-setosa
          3      Iris-setosa
          4      Iris-setosa
          ...
          145    Iris-virginica
          146    Iris-virginica
          147    Iris-virginica
          148    Iris-virginica
          149    Iris-virginica
          Name: Species, Length: 150, dtype: object
```

```
In [13]: from sklearn.preprocessing import MinMaxScaler #adjusting values measured on
          scaler=MinMaxScaler()
```

```
In [14]: X_scaled = scaler.fit_transform(X)
```

```
In [15]: X_scaled
```

```
Out[15]: array([[0.          , 0.22222222, 0.625        , 0.06779661, 0.04166667],
 [0.00671141, 0.16666667, 0.41666667, 0.06779661, 0.04166667],
 [0.01342282, 0.11111111, 0.5          , 0.05084746, 0.04166667],
 [0.02013423, 0.08333333, 0.45833333, 0.08474576, 0.04166667],
 [0.02684564, 0.19444444, 0.66666667, 0.06779661, 0.04166667],
 [0.03355705, 0.30555556, 0.79166667, 0.11864407, 0.125        ],
 [0.04026846, 0.08333333, 0.58333333, 0.06779661, 0.08333333],
 [0.04697987, 0.19444444, 0.58333333, 0.08474576, 0.04166667],
 [0.05369128, 0.02777778, 0.375        , 0.06779661, 0.04166667],
 [0.06040268, 0.16666667, 0.45833333, 0.08474576, 0.          ],
 [0.06711409, 0.30555556, 0.70833333, 0.08474576, 0.04166667],
 [0.0738255 , 0.13888889, 0.58333333, 0.10169492, 0.04166667],
 [0.08053691, 0.13888889, 0.41666667, 0.06779661, 0.          ],
 [0.08724832, 0.          , 0.41666667, 0.01694915, 0.          ],
 [0.09395973, 0.41666667, 0.83333333, 0.03389831, 0.04166667],
 [0.10067114, 0.38888889, 1.          , 0.08474576, 0.125        ],
 [0.10738255, 0.30555556, 0.79166667, 0.05084746, 0.125        ],
 [0.11409396, 0.22222222, 0.625        , 0.06779661, 0.08333333],
 [0.12080537, 0.38888889, 0.75          , 0.11864407, 0.08333333],
 [0.12751678, 0.22222222, 0.75          , 0.08474576, 0.08333333],
 [0.13422819, 0.30555556, 0.58333333, 0.11864407, 0.04166667],
 [0.1409396 , 0.22222222, 0.70833333, 0.08474576, 0.125        ],
 [0.14765101, 0.08333333, 0.66666667, 0.          , 0.04166667],
 [0.15436242, 0.22222222, 0.54166667, 0.11864407, 0.16666667],
 [0.16107383, 0.13888889, 0.58333333, 0.15254237, 0.04166667],
 [0.16778523, 0.19444444, 0.41666667, 0.10169492, 0.04166667],
 [0.17449664, 0.19444444, 0.58333333, 0.10169492, 0.125        ],
 [0.18120805, 0.25          , 0.625        , 0.08474576, 0.04166667],
 [0.18791946, 0.25          , 0.58333333, 0.06779661, 0.04166667],
 [0.19463087, 0.11111111, 0.5          , 0.10169492, 0.04166667],
 [0.20134228, 0.13888889, 0.45833333, 0.10169492, 0.04166667],
 [0.20805369, 0.30555556, 0.58333333, 0.08474576, 0.125        ],
 [0.2147651 , 0.25          , 0.875        , 0.08474576, 0.          ],
 [0.22147651, 0.33333333, 0.91666667, 0.06779661, 0.04166667],
 [0.22818792, 0.16666667, 0.45833333, 0.08474576, 0.          ],
 [0.23489933, 0.19444444, 0.5          , 0.03389831, 0.04166667],
 [0.24161074, 0.33333333, 0.625        , 0.05084746, 0.04166667],
 [0.24832215, 0.16666667, 0.45833333, 0.08474576, 0.          ],
 [0.25503356, 0.02777778, 0.41666667, 0.05084746, 0.04166667],
 [0.26174497, 0.22222222, 0.58333333, 0.08474576, 0.04166667],
 [0.26845638, 0.19444444, 0.625        , 0.05084746, 0.08333333],
 [0.27516779, 0.05555556, 0.125        , 0.05084746, 0.08333333],
 [0.28187919, 0.02777778, 0.5          , 0.05084746, 0.04166667],
 [0.2885906 , 0.19444444, 0.625        , 0.10169492, 0.20833333],
 [0.29530201, 0.22222222, 0.75          , 0.15254237, 0.125        ],
 [0.30201342, 0.13888889, 0.41666667, 0.06779661, 0.08333333],
 [0.30872483, 0.22222222, 0.75          , 0.10169492, 0.04166667],
 [0.31543624, 0.08333333, 0.5          , 0.06779661, 0.04166667],
 [0.32214765, 0.27777778, 0.70833333, 0.08474576, 0.04166667],
 [0.32885906, 0.19444444, 0.54166667, 0.06779661, 0.04166667],
 [0.33557047, 0.75          , 0.5          , 0.62711864, 0.54166667],
 [0.34228188, 0.58333333, 0.5          , 0.59322034, 0.58333333],
 [0.34899329, 0.72222222, 0.45833333, 0.66101695, 0.58333333],
 [0.3557047 , 0.33333333, 0.125        , 0.50847458, 0.5          ],
 [0.36241611, 0.61111111, 0.33333333, 0.61016949, 0.58333333],
 [0.36912752, 0.38888889, 0.33333333, 0.59322034, 0.5          ],
 [0.37583893, 0.55555556, 0.54166667, 0.62711864, 0.625        ],
 [0.38255034, 0.16666667, 0.16666667, 0.38983051, 0.375        ],
 [0.38926174, 0.63888889, 0.375        , 0.61016949, 0.5          ],
 [0.39597315, 0.25          , 0.29166667, 0.49152542, 0.54166667],
 [0.40268456, 0.19444444, 0.          , 0.42372881, 0.375        ],
 [0.40939597, 0.44444444, 0.41666667, 0.54237288, 0.58333333],
 [0.41610738, 0.47222222, 0.08333333, 0.50847458, 0.375        ],
 [0.42281879, 0.5          , 0.375        , 0.62711864, 0.54166667],
```

```
[0.4295302 , 0.36111111, 0.375 , 0.44067797, 0.5 ],
[0.43624161, 0.66666667, 0.45833333, 0.57627119, 0.54166667],
[0.44295302, 0.36111111, 0.41666667, 0.59322034, 0.58333333],
[0.44966443, 0.41666667, 0.29166667, 0.52542373, 0.375 ],
[0.45637584, 0.52777778, 0.08333333, 0.59322034, 0.58333333],
[0.46308725, 0.36111111, 0.20833333, 0.49152542, 0.41666667],
[0.46979866, 0.44444444, 0.5 , 0.6440678 , 0.70833333],
[0.47651007, 0.5 , 0.33333333, 0.50847458, 0.5 ],
[0.48322148, 0.55555556, 0.20833333, 0.66101695, 0.58333333],
[0.48993289, 0.5 , 0.33333333, 0.62711864, 0.45833333],
[0.4966443 , 0.58333333, 0.375 , 0.55932203, 0.5 ],
[0.5033557 , 0.63888889, 0.41666667, 0.57627119, 0.54166667],
[0.51006711, 0.69444444, 0.33333333, 0.6440678 , 0.54166667],
[0.51677852, 0.66666667, 0.41666667, 0.6779661 , 0.66666667],
[0.52348993, 0.47222222, 0.375 , 0.59322034, 0.58333333],
[0.53020134, 0.38888889, 0.25 , 0.42372881, 0.375 ],
[0.53691275, 0.33333333, 0.16666667, 0.47457627, 0.41666667],
[0.54362416, 0.33333333, 0.16666667, 0.45762712, 0.375 ],
[0.55033557, 0.41666667, 0.29166667, 0.49152542, 0.45833333],
[0.55704698, 0.47222222, 0.29166667, 0.69491525, 0.625 ],
[0.56375839, 0.30555556, 0.41666667, 0.59322034, 0.58333333],
[0.5704698 , 0.47222222, 0.58333333, 0.59322034, 0.625 ],
[0.57718121, 0.66666667, 0.45833333, 0.62711864, 0.58333333],
[0.58389262, 0.55555556, 0.125 , 0.57627119, 0.5 ],
[0.59060403, 0.36111111, 0.41666667, 0.52542373, 0.5 ],
[0.59731544, 0.33333333, 0.20833333, 0.50847458, 0.5 ],
[0.60402685, 0.33333333, 0.25 , 0.57627119, 0.45833333],
[0.61073826, 0.5 , 0.41666667, 0.61016949, 0.54166667],
[0.61744966, 0.41666667, 0.25 , 0.50847458, 0.45833333],
[0.62416107, 0.19444444, 0.125 , 0.38983051, 0.375 ],
[0.63087248, 0.36111111, 0.29166667, 0.54237288, 0.5 ],
[0.63758389, 0.38888889, 0.41666667, 0.54237288, 0.45833333],
[0.6442953 , 0.38888889, 0.375 , 0.54237288, 0.5 ],
[0.65100671, 0.52777778, 0.375 , 0.55932203, 0.5 ],
[0.65771812, 0.22222222, 0.20833333, 0.33898305, 0.41666667],
[0.66442953, 0.38888889, 0.33333333, 0.52542373, 0.5 ],
[0.67114094, 0.55555556, 0.54166667, 0.84745763, 1. ],
[0.67785235, 0.41666667, 0.29166667, 0.69491525, 0.75 ],
[0.68456376, 0.77777778, 0.41666667, 0.83050847, 0.83333333],
[0.69127517, 0.55555556, 0.375 , 0.77966102, 0.70833333],
[0.69798658, 0.61111111, 0.41666667, 0.81355932, 0.875 ],
[0.70469799, 0.91666667, 0.41666667, 0.94915254, 0.83333333],
[0.7114094 , 0.16666667, 0.20833333, 0.59322034, 0.66666667],
[0.71812081, 0.83333333, 0.375 , 0.89830508, 0.70833333],
[0.72483221, 0.66666667, 0.20833333, 0.81355932, 0.70833333],
[0.73154362, 0.80555556, 0.66666667, 0.86440678, 1. ],
[0.73825503, 0.61111111, 0.5 , 0.69491525, 0.79166667],
[0.74496644, 0.58333333, 0.29166667, 0.72881356, 0.75 ],
[0.75167785, 0.69444444, 0.41666667, 0.76271186, 0.83333333],
[0.75838926, 0.38888889, 0.20833333, 0.6779661 , 0.79166667],
[0.76510067, 0.41666667, 0.33333333, 0.69491525, 0.95833333],
[0.77181208, 0.58333333, 0.5 , 0.72881356, 0.91666667],
[0.77852349, 0.61111111, 0.41666667, 0.76271186, 0.70833333],
[0.7852349 , 0.94444444, 0.75 , 0.96610169, 0.875 ],
[0.79194631, 0.94444444, 0.25 , 1. , 0.91666667],
[0.79865772, 0.47222222, 0.08333333, 0.6779661 , 0.58333333],
[0.80536913, 0.72222222, 0.5 , 0.79661017, 0.91666667],
[0.81208054, 0.36111111, 0.33333333, 0.66101695, 0.79166667],
[0.81879195, 0.94444444, 0.33333333, 0.96610169, 0.79166667],
[0.82550336, 0.55555556, 0.29166667, 0.66101695, 0.70833333],
[0.83221477, 0.66666667, 0.54166667, 0.79661017, 0.83333333],
[0.83892617, 0.80555556, 0.5 , 0.84745763, 0.70833333],
[0.84563758, 0.52777778, 0.33333333, 0.6440678 , 0.70833333],
[0.85234899, 0.5 , 0.41666667, 0.66101695, 0.70833333],
```

```
[0.8590604 , 0.58333333, 0.33333333, 0.77966102, 0.83333333],
[0.86577181, 0.80555556, 0.41666667, 0.81355932, 0.625      ],
[0.87248322, 0.86111111, 0.33333333, 0.86440678, 0.75      ],
[0.87919463, 1.        , 0.75      , 0.91525424, 0.79166667],
[0.88590604, 0.58333333, 0.33333333, 0.77966102, 0.875      ],
[0.89261745, 0.55555556, 0.33333333, 0.69491525, 0.58333333],
[0.89932886, 0.5       , 0.25      , 0.77966102, 0.54166667],
[0.90604027, 0.94444444, 0.41666667, 0.86440678, 0.91666667],
[0.91275168, 0.55555556, 0.58333333, 0.77966102, 0.95833333],
[0.91946309, 0.58333333, 0.45833333, 0.76271186, 0.70833333],
[0.9261745  , 0.47222222, 0.41666667, 0.6440678 , 0.70833333],
[0.93288591, 0.72222222, 0.45833333, 0.74576271, 0.83333333],
[0.93959732, 0.66666667, 0.45833333, 0.77966102, 0.95833333],
[0.94630872, 0.72222222, 0.45833333, 0.69491525, 0.91666667],
[0.95302013, 0.41666667, 0.29166667, 0.69491525, 0.75      ],
[0.95973154, 0.69444444, 0.5       , 0.83050847, 0.91666667],
[0.96644295, 0.66666667, 0.54166667, 0.79661017, 1.        ],
[0.97315436, 0.66666667, 0.41666667, 0.71186441, 0.91666667],
[0.97986577, 0.55555556, 0.20833333, 0.6779661 , 0.75      ],
[0.98657718, 0.61111111, 0.41666667, 0.71186441, 0.79166667],
[0.99328859, 0.52777778, 0.58333333, 0.74576271, 0.91666667],
[1.        , 0.44444444, 0.41666667, 0.69491525, 0.70833333]])
```

```
In [29]: 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, random_state=42)
```

```
In [28]: Y_train
```

Out[28]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	
	73	74	6.1	2.8	4.7	1.2
	18	19	5.7	3.8	1.7	0.3
	118	119	7.7	2.6	6.9	2.3
	78	79	6.0	2.9	4.5	1.5
	76	77	6.8	2.8	4.8	1.4
	31	32	5.4	3.4	1.5	0.4
	64	65	5.6	2.9	3.6	1.3
	141	142	6.9	3.1	5.1	2.3
	68	69	6.2	2.2	4.5	1.5
	82	83	5.8	2.7	3.9	1.2
	110	111	6.5	3.2	5.1	2.0
	12	13	4.8	3.0	1.4	0.1
	36	37	5.5	3.5	1.3	0.2
	9	10	4.9	3.1	1.5	0.1
	19	20	5.1	3.8	1.5	0.3
	56	57	6.3	3.3	4.7	1.6
	104	105	6.5	3.0	5.8	2.2
	69	70	5.6	2.5	3.9	1.1
	55	56	5.7	2.8	4.5	1.3
	132	133	6.4	2.8	5.6	2.2
	29	30	4.7	3.2	1.6	0.2
	127	128	6.1	3.0	4.9	1.8
	26	27	5.0	3.4	1.6	0.4
	128	129	6.4	2.8	5.6	2.1
	131	132	7.9	3.8	6.4	2.0
	145	146	6.7	3.0	5.2	2.3
	108	109	6.7	2.5	5.8	1.8
	143	144	6.8	3.2	5.9	2.3
	45	46	4.8	3.0	1.4	0.3
	30	31	4.8	3.1	1.6	0.2

```
In [25]: from sklearn.naive_bayes import GaussianNB

In [30]: gnb = GaussianNB()
         gnb.fit(X_train, Y_train)

Out[30]: GaussianNB()

In [31]: Y_Pred = gnb.predict(X_test)

In [39]: from sklearn.metrics import accuracy_score ,precision_score,recall_score,f1
         accuracy_nb=round(accuracy_score(Y_test,Y_Pred)* 100, 2)
```

```
acc_gaussian = round(gnb.score(X_train, Y_train) * 100, 2)
cm = confusion_matrix(Y_test, Y_Pred)
accuracy = accuracy_score(Y_test, Y_Pred)
precision = precision_score(Y_test, Y_Pred, average='micro')
recall = recall_score(Y_test, Y_Pred, average='micro')
f1 = f1_score(Y_test, Y_Pred, average='micro')
```

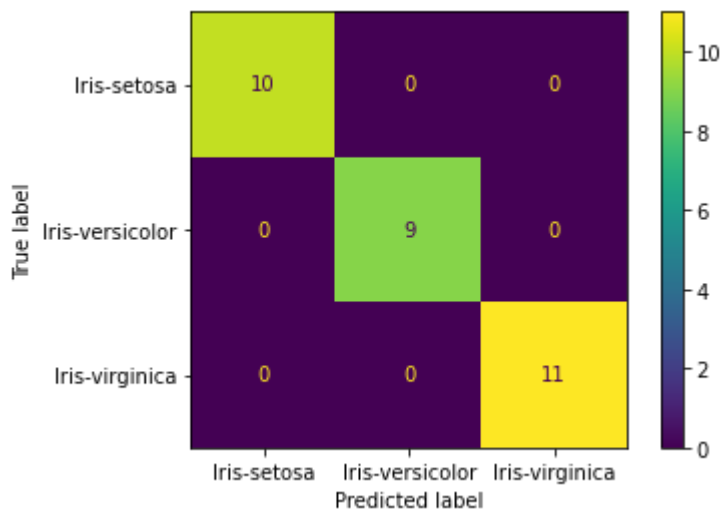
```
In [40]: print("Accuracy GNB:", accuracy)
print("Precision GNB:", precision)
print("Recall GNB:", recall)
print("F1-Score GNB:", f1)

print(cm)
```

```
Accuracy NB: 100.0
Accuracy GNB: 99.17
[[10  0  0]
 [ 0  9  0]
 [ 0  0 11]]
```

```
In [41]: from sklearn.metrics import ConfusionMatrixDisplay
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=gnb.classes)
disp.plot()
```

```
Out[41]: <sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7f7b4f5f1f10>
```



```
In [ ]:
```