

```
In [1]: 1 import numpy as np
        2 import matplotlib.pyplot as plt
        3 import pandas as pd
```

```
In [2]: 1 dataset=pd.read_csv("Iris.csv - Iris.csv.csv")
        2 dataset
```

```
Out[2]:
```

	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 [3]: 1 dataset.describe()
```

```
Out[3]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

In [4]: 1 dataset.head()

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

In [6]: 1 dataset.shape

Out[6]: (150, 6)

In [21]: 1 X=dataset.iloc[:,4].values
2 X

```
[ 42. ,  4.5,  2.3,  1.3],  
[ 43. ,  4.4,  3.2,  1.3],  
[ 44. ,  5. ,  3.5,  1.6],  
[ 45. ,  5.1,  3.8,  1.9],  
[ 46. ,  4.8,  3. ,  1.4],  
[ 47. ,  5.1,  3.8,  1.6],  
[ 48. ,  4.6,  3.2,  1.4],  
[ 49. ,  5.3,  3.7,  1.5],  
[ 50. ,  5. ,  3.3,  1.4],  
[ 51. ,  7. ,  3.2,  4.7],  
[ 52. ,  6.4,  3.2,  4.5],  
[ 53. ,  6.9,  3.1,  4.9],  
[ 54. ,  5.5,  2.3,  4. ],  
[ 55. ,  6.5,  2.8,  4.6],  
[ 56. ,  5.7,  2.8,  4.5],  
[ 57. ,  6.3,  3.3,  4.7],  
[ 58. ,  4.9,  2.4,  3.3],  
[ 59. ,  6.6,  2.9,  4.6],  
[ 60. ,  5.2,  2.7,  3.9],  
[ 61. ,  5. ,  2. ,  3.5],
```

```
In [11]: 1 y=dataset['Species'].values
          2 y
```

[illegible]

```
In [12]: 1 from sklearn.model_selection import train_test_split
```

```
In [22]: 1 X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2)
```

```
In [16]: 1 from sklearn.preprocessing import StandardScaler
```

In [23]:

```
1 from sklearn.preprocessing import StandardScaler
2 sc = StandardScaler()
3 X_train = sc.fit_transform(X_train)
4 X_test = sc.transform(X_test)
5 X_train
[ 7.70275188e-01,  5.58025000e-01,  1.01149868e+00,
 -1.36390459e+00],
[-8.61006990e-01, -3.66365318e-01,  1.01149868e+00,
 -1.36390459e+00],
[-1.08783649e+00, -9.98029660e-01,  7.73498988e-01,
 -1.19150092e+00],
[-3.39299132e-01, -7.45363923e-01, -8.92498832e-01,
  1.30260551e-01],
[-9.51738791e-01, -7.45363923e-01,  2.43949681e+00,
 -1.24896881e+00],
[-6.79543387e-01, -8.71696791e-01,  1.72549774e+00,
 -1.01909725e+00],
[ 1.36192214e+00,  3.91631892e-01, -1.13049852e+00,
  1.10721468e+00],
[ 9.53629037e-01,  8.96963365e-01, -1.78499766e-01,
  1.04974679e+00],
[ 1.04436084e+00,  1.40229484e+00,  2.97499611e-01,
  1.16468257e+00],
[-9.97104692e-01, -1.25069540e+00,  5.94999222e-02,
 -1.19150092e+00]])
```

In [24]:

```
1 X_test
```

```
Out[24]: array([[ 0.09167692,  0.26529902, -0.41649946,  0.47506789],
 [-0.06710373,  0.39163189, -0.65449914,  0.18772844],
 [-1.2693001 , -0.11369958,  1.72549774, -1.13403303],
 [ 1.31655624,  0.7706305 , -0.65449914,  1.10721468],
 [ 0.99899494,  2.41295778, -1.13049852,  1.85429725],
 [-1.11051944, -0.99802966, -0.17849977, -1.19150092],
 [-1.15588534, -0.87169679,  0.5354993 , -1.13403303],
 [ 0.63606773,  1.65496057, -0.17849977,  1.27961835],
 [-0.90637289, -1.12436253,  0.05949992, -1.24896881],
 [-1.3600319 ,  0.01263329,  2.20149712, -1.42137248],
 [ 1.06704379, -0.24003245, -0.65449914,  0.70493945],
 [ 0.13704282, -0.36636532, -1.6064979 ,  0.07279266],
 [-0.31661618, -0.99802966, -2.55849665, -0.09961101],
 [-1.20125124, -0.87169679,  1.48749805, -1.24896881],
 [ 0.29582348,  0.64429763, -1.84449759,  0.4176   ],
 [-0.54344569,  1.52862771,  0.29749961,  0.59000367],
 [-1.4053978 , -1.2506954 , -0.17849977, -1.3064367 ],
 [ 0.70411658,  2.28662492, -0.17849977,  1.68189358],
 [ 0.04631102,  1.27596197, -0.65449914,  0.64747156],
 [-1.47344665, -1.12436253,  0.05949992, -1.24896881],
 [ 0.97631199,  2.41295778,  1.72549774,  1.73936147],
 [ 1.11240969,  0.64429763, -0.89249883,  0.70493945],
 [-0.18051848, -0.24003245, -0.17849977,  0.47506789],
 [ 0.54533593, -0.87169679, -1.36849821, -0.38695046],
 [ 1.29387329,  2.66562352,  1.72549774,  1.5669578 ],
 [ 0.88558019, -0.11369958, -1.36849821,  0.76240734],
 [ 0.84021429,  0.7706305 , -0.89249883,  0.93481101],
 [ 0.86289724,  1.27596197, -0.17849977,  1.04974679],
 [ 0.65875068,  0.64429763, -0.41649946,  1.10721468],
 [ 0.36387233, -0.36636532, -1.13049852,  0.4176   ]])
```

```
In [27]: 1 from sklearn.naive_bayes import GaussianNB
```

```
In [30]: 1 classifier=GaussianNB()  
2 classifier.fit(X_train,y_train)
```

```
Out[30]: ▾ GaussianNB  
GaussianNB()
```

```
In [32]: 1 y_pred=classifier.predict(X_test)  
2 y_pred
```

```
Out[32]: array(['Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',  
                'Iris-virginica', 'Iris-virginica', 'Iris-setosa', 'Iris-setosa',  
                'Iris-virginica', 'Iris-setosa', 'Iris-setosa', 'Iris-virginica',  
                'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',  
                'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',  
                'Iris-virginica', 'Iris-versicolor', 'Iris-setosa',  
                'Iris-virginica', 'Iris-virginica', 'Iris-versicolor',  
                'Iris-versicolor', 'Iris-virginica', 'Iris-virginica',  
                'Iris-virginica', 'Iris-virginica', 'Iris-virginica',  
                'Iris-versicolor'], dtype='<U15')
```

```
In [34]: 1 from sklearn.metrics import confusion_matrix  
2 cm=confusion_matrix(y_test,y_pred)  
3  
4 from sklearn.metrics import accuracy_score  
5 print("Accuracy: ",accuracy_score(y_test,y_pred))  
6 cm
```

Accuracy: 1.0

```
Out[34]: array([[ 8,  0,  0],  
               [ 0, 10,  0],  
               [ 0,  0, 12]], dtype=int64)
```

```
In [35]: 1 df=pd.DataFrame({'Real Values':y_test,'Predicted_values':y_pred})
```

In [36]:

1 df

Out[36]:

	Real Values	Predicted_values
0	Iris-versicolor	Iris-versicolor
1	Iris-versicolor	Iris-versicolor
2	Iris-setosa	Iris-setosa
3	Iris-virginica	Iris-virginica
4	Iris-virginica	Iris-virginica
5	Iris-setosa	Iris-setosa
6	Iris-setosa	Iris-setosa
7	Iris-virginica	Iris-virginica
8	Iris-setosa	Iris-setosa
9	Iris-setosa	Iris-setosa
10	Iris-virginica	Iris-virginica
11	Iris-versicolor	Iris-versicolor
12	Iris-versicolor	Iris-versicolor
13	Iris-setosa	Iris-setosa
14	Iris-versicolor	Iris-versicolor
15	Iris-versicolor	Iris-versicolor
16	Iris-setosa	Iris-setosa
17	Iris-virginica	Iris-virginica
18	Iris-versicolor	Iris-versicolor
19	Iris-setosa	Iris-setosa
20	Iris-virginica	Iris-virginica
21	Iris-virginica	Iris-virginica
22	Iris-versicolor	Iris-versicolor
23	Iris-versicolor	Iris-versicolor
24	Iris-virginica	Iris-virginica
25	Iris-virginica	Iris-virginica
26	Iris-virginica	Iris-virginica
27	Iris-virginica	Iris-virginica
28	Iris-virginica	Iris-virginica
29	Iris-versicolor	Iris-versicolor

In []:

1

In []:

1

