

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

```
In [2]: dataset = pd.read_csv('C:\\Users\\Lenovo\\Downloads\\Social_Network_Ads.csv')
dataset.head()
```

```
Out[2]:
```

	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0

```
In [3]: X = dataset.iloc[:, [2, 3]].values
        Y = dataset.iloc[:, 4].values
```

```
In [4]: X
```

```
Out[4]: array([[ 19, 19000],
 [ 35, 20000],
 [ 26, 43000],
 [ 27, 57000],
 [ 19, 76000],
 [ 27, 58000],
 [ 27, 84000],
 [ 32, 150000],
 [ 25, 33000],
 [ 35, 65000],
 [ 26, 80000],
 [ 26, 52000],
 [ 20, 86000],
 [ 32, 18000],
 [ 18, 82000],
 [ 29, 80000],
 [ 47, 25000],
 [ 45, 26000],
 [ 46, 28000],
```

In [5]:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.25, random_state =
```

In [6]:

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

In [7]:

```
from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression(random_state = 0)
classifier.fit(X_train, y_train)
```

Out[7]:

```
LogisticRegression(random_state=0)
```

In [8]:

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

In [10]:

```
from sklearn.metrics import confusion_matrix, classification_report
cm = confusion_matrix(y_test, y_pred)
```

In [11]:

```
cm
```

Out[11]:

```
array([[65,  3],
       [ 8, 24]], dtype=int64)
```

In [12]:

```
cl_report=classification_report(y_test,y_pred)
```

In [13]:

```
cl_report
```

Out[13]:

```
'          precision    recall  f1-score   support\n\n 0.89      0.96      0.92      0.89         68\n 0.81      0.81      0.81      0.81         32\nmacro avg   0.89      0.85      0.87        100\n0.89      0.89      0.89      0.89        100'
```

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

```
In [6]: dataset = pd.read_csv('https://raw.githubusercontent.com/mk-gurucharan/Classification/master/iris.csv')
```

```
In [7]: X = dataset.iloc[:, :4].values
y = dataset['species'].values
```

```
In [8]: dataset.head(5)
```

```
Out[8]:
```

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

```
In [9]: 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)
```

```
In [10]: from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

```
In [11]: from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train, y_train)
```

```
Out[11]: GaussianNB()
```

```
In [12]: y_pred = classifier.predict(X_test)
y_pred
```

```
Out[12]: array(['virginica', 'setosa', 'virginica', 'setosa', 'virginica',
        'versicolor', 'setosa', 'versicolor', 'virginica', 'virginica',
        'versicolor', 'virginica', 'virginica', 'setosa', 'virginica',
        'setosa', 'versicolor', 'virginica', 'versicolor', 'virginica',
        'virginica', 'versicolor', 'setosa', 'setosa', 'virginica',
        'virginica', 'setosa', 'versicolor', 'versicolor', 'virginica'],
        dtype='<U10')
```

```
In [13]: 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 : 1.0
Out[13]: array([[ 8,  0,  0],
        [ 0,  8,  0],
        [ 0,  0, 14]], dtype=int64)
```