

```
In [4]: import pandas as pd
```

```
In [5]: import numpy as np
```

```
In [6]: df = pd.read_csv("Social_Network_Ads.csv")  
df
```

```
Out[6]:
```

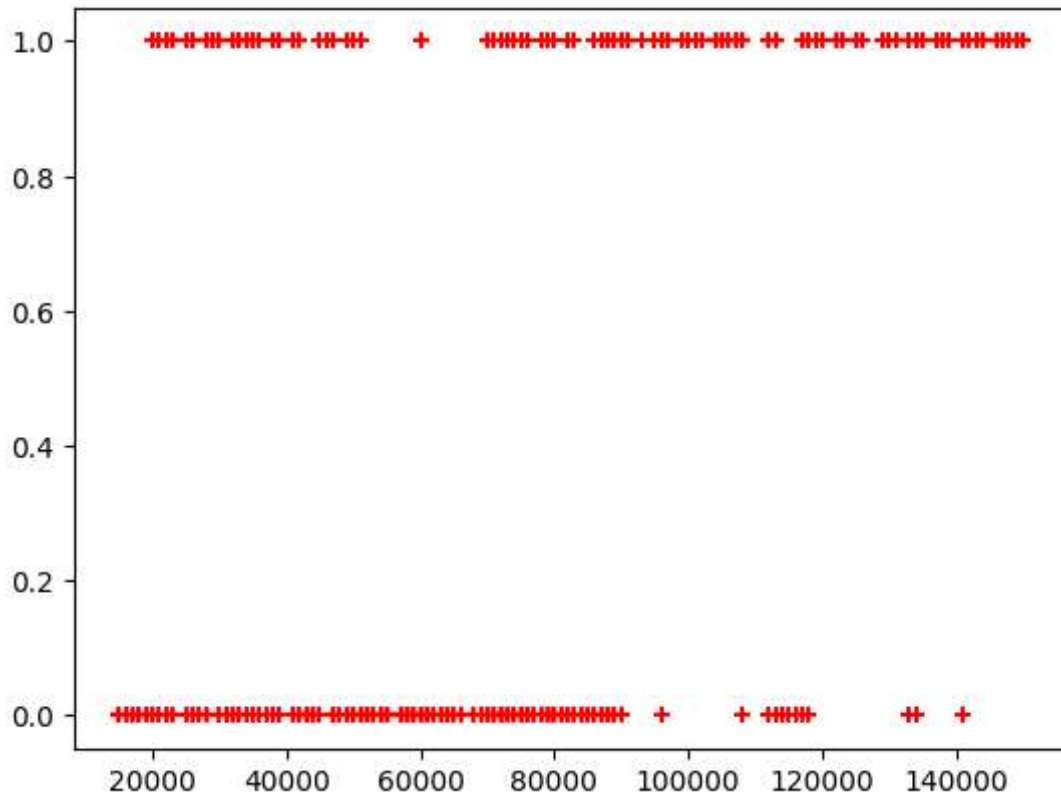
	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19.0	19000.0	0
1	15810944	Male	35.0	20000.0	0
2	15668575	Female	26.0	43000.0	0
3	15603246	Female	27.0	57000.0	0
4	15804002	Male	19.0	76000.0	0
...
395	15691863	Female	46.0	41000.0	1
396	15706071	Male	51.0	23000.0	1
397	15654296	Female	50.0	20000.0	1
398	15755018	Male	36.0	33000.0	0
399	15594041	Female	49.0	36000.0	1

400 rows × 5 columns

```
In [7]: import matplotlib.pyplot as plt
```

```
In [8]: plt.scatter(df.EstimatedSalary, df.Purchased, marker='+', color='red')
```

```
Out[8]: <matplotlib.collections.PathCollection at 0x267f768a410>
```



```
In [9]: x = df[['Age','EstimatedSalary']]
```

```
In [10]: y = df['Purchased']
```

```
In [11]: from sklearn.model_selection import train_test_split
```

```
In [12]: x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.1,random_state =
```

```
In [13]: from sklearn.linear_model import LogisticRegression
```

```
In [14]: regression = LogisticRegression()
```

```
In [15]: regression.fit(x_train,y_train)
```

```
Out[15]: ▼ LogisticRegression
          LogisticRegression()
```

```
In [16]: regression.predict(x_test)
```

```
Out[16]: array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], dtype=int64)
```

```
In [17]: regression.score(x_test,y_test)
```

```
Out[17]: 0.8
```

```
In [20]: from sklearn.preprocessing import StandardScaler
```

```
In [23]: scaler = StandardScaler()
```

```
In [28]: x_train = scaler.fit_transform(x_train)
```

```
In [29]: x_test = scaler.transform(x_test)
```

```
In [32]: x_train[0:10,:]
```

```
Out[32]: array([[ -1.05714987,   0.53420426],
 [   0.2798728 ,  -0.51764734],
 [ -1.05714987,   0.41733186],
 [ -0.29313691,  -1.45262654],
 [   0.47087604,   1.23543867],
 [ -1.05714987,  -0.34233874],
 [ -0.10213368,   0.30045946],
 [   1.33039061,   0.59264046],
 [ -1.15265148,  -1.16044554],
 [   1.04388575,   0.47576806]])
```

```
In [33]: regression.fit(x_train,y_train)
```

```
Out[33]: ▾ LogisticRegression
LogisticRegression()
```

```
In [34]: regression.predict(x_test)
```

```
Out[34]: array([0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1,
 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1], dtype=int64)
```

```
In [35]: regression.score(x_test,y_test)*100
```

```
Out[35]: 95.0
```

```
In [ ]:
```