In [56]:
```
import pandas as pd
import numpy as np
# import matplot.lib.pyplot as plt
```

In [57]:
```
df = pd.read_csv("Social_Network_Ads.csv")
df
```

Out[57]:

	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 [58]:
```
x = df.drop(['User ID','Gender'], axis=1)
# x
# x.dtypes

y = df['Purchased']    # Labeled data to be separated
x = x.drop('Purchased', axis=1)    # Input data
```

In [71]:
```
# Data is divided into 2 parts for : Training(75%) and testing(25%)
from sklearn.model_selection import train_test_split    # Importing the splitting m
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.25, random_st
# print(x_train)
# print(x_test)
# y_train
# y_test
```

In [87]:
```
from sklearn.preprocessing import StandardScaler, MinMaxScaler
std = StandardScaler()
# std = MinMaxScaler()
x_train = std.fit_transform(x_train)
# x_train
```

```
x_test = std.fit_transform(x_test)
# print(x_train)
```

In [88]:
```
from sklearn.linear_model import LogisticRegression   # Importing the training mode
model = LogisticRegression()
model.fit(x_train, y_train)    # Training
```

Out[88]:
```
LogisticRegression()
```

In [90]:
```
y_pred = model.predict(x_test)   # Predicting
# print(y_pred)
# print(y_test)

from sklearn.metrics import accuracy_score, confusion_matrix, precision_score, reca

print(accuracy_score(y_test, y_pred))
print(confusion_matrix(y_pred, y_test))
print(precision_score(y_pred, y_test))
print(recall_score(y_pred, y_test))
print()
print(classification_report(y_pred, y_test))
```

```
0.88
[[64  7]
 [ 5 24]]
0.7741935483870968
0.8275862068965517

              precision    recall  f1-score   support

           0       0.93      0.90      0.91        71
           1       0.77      0.83      0.80        29

    accuracy                           0.88       100
   macro avg       0.85      0.86      0.86       100
weighted avg       0.88      0.88      0.88       100
```