

In []:

In [1]:

```
import numpy as np
```

In [2]:

```
import pandas as pd
```

In [3]:

```
import seaborn as sns
```

In [4]:

```
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
from sklearn.model_selection import train_test_split
from sklearn import preprocessing as per
```

In [5]:

```
dataset = pd.read_csv('Data1.csv')
```

In [6]:

```
dataset
```

Out[6]:

	Region	Age	Income	Online Shopper
0	India	49.0	86400.0	No
1	Brazil	32.0	57600.0	Yes
2	USA	35.0	64800.0	No
3	Brazil	43.0	73200.0	No
4	USA	45.0	NaN	Yes
5	India	40.0	69600.0	Yes
6	Brazil	NaN	62400.0	No
7	India	53.0	94800.0	Yes
8	USA	55.0	99600.0	No
9	India	42.0	80400.0	Yes

In [7]:

```
dataset.isnull()
```

Out[7]:

	Region	Age	Income	Online Shopper
0	False	False	False	False
1	False	False	False	False
2	False	False	False	False
3	False	False	False	False
4	False	False	True	False
5	False	False	False	False
6	False	True	False	False
7	False	False	False	False
8	False	False	False	False
9	False	False	False	False

In [8]:

```
dataset.isnull().sum()
```

Out[8]:

```
Region      0
Age         1
Income      1
Online Shopper  0
dtype: int64
```

In [9]:

```
X = dataset.iloc[:, :-1].values
X
```

Out[9]:

```
array([[ 'India', 49.0, 86400.0],
       [ 'Brazil', 32.0, 57600.0],
       [ 'USA', 35.0, 64800.0],
       [ 'Brazil', 43.0, 73200.0],
       [ 'USA', 45.0, nan],
       [ 'India', 40.0, 69600.0],
       [ 'Brazil', nan, 62400.0],
       [ 'India', 53.0, 94800.0],
       [ 'USA', 55.0, 99600.0],
       [ 'India', 42.0, 80400.0]], dtype=object)
```

In [10]:

```
Y = dataset.iloc[:, 3].values
Y
```

Out[10]:

```
array([ 'No', 'Yes', 'No', 'No', 'Yes', 'Yes', 'No', 'Yes', 'No', 'Yes'],
      dtype=object)
```

In [11]:

```
imputer = SimpleImputer(missing_values = np.nan, strategy = 'mean')
```

In [12]:

```
imputer = imputer.fit(X[:, 1:-1])
```

In [13]:

```
X[:, 1:] = imputer.transform(X[:, 1:])
```

In [14]:

```
X
```

Out[14]:

```
array([[ 'India', 86400.0, 86400.0],
       [ 'Brazil', 57600.0, 57600.0],
       [ 'USA', 64800.0, 64800.0],
       [ 'Brazil', 73200.0, 73200.0],
       [ 'USA', 43.77777777777778, 43.77777777777778],
       [ 'India', 69600.0, 69600.0],
       [ 'Brazil', 62400.0, 62400.0],
       [ 'India', 94800.0, 94800.0],
       [ 'USA', 99600.0, 99600.0],
       [ 'India', 80400.0, 80400.0]], dtype=object)
```

In [15]:

```
le = LabelEncoder()
```

In [16]:

```
income = le.fit_transform(dataset['Region'])
income
```

Out[16]:

```
array([1, 0, 2, 0, 2, 1, 0, 1, 2, 1])
```

In [17]:

dataset

Out[17]:

	Region	Age	Income	Online Shopper
0	India	49.0	86400.0	No
1	Brazil	32.0	57600.0	Yes
2	USA	35.0	64800.0	No
3	Brazil	43.0	73200.0	No
4	USA	45.0	NaN	Yes
5	India	40.0	69600.0	Yes
6	Brazil	NaN	62400.0	No
7	India	53.0	94800.0	Yes
8	USA	55.0	99600.0	No
9	India	42.0	80400.0	Yes

In [18]:

```
y = dataset['Online Shopper']
y
```

Out[18]:

```
0    No
1    Yes
2    No
3    No
4    Yes
5    Yes
6    No
7    Yes
8    No
9    Yes
Name: Online Shopper, dtype: object
```

In [19]:

```
x = dataset.drop('Online Shopper',axis = 1)
print(x)
```

	Region	Age	Income
0	India	49.0	86400.0
1	Brazil	32.0	57600.0
2	USA	35.0	64800.0
3	Brazil	43.0	73200.0
4	USA	45.0	NaN
5	India	40.0	69600.0
6	Brazil	NaN	62400.0
7	India	53.0	94800.0
8	USA	55.0	99600.0
9	India	42.0	80400.0

In [20]:

```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.3)
```

In [21]:

```
print(x_train)
```

	Region	Age	Income
3	Brazil	43.0	73200.0
5	India	40.0	69600.0
4	USA	45.0	NaN
8	USA	55.0	99600.0
9	India	42.0	80400.0
7	India	53.0	94800.0
0	India	49.0	86400.0

In [22]:

```
print(y_train)
```

```
3      No
5      Yes
4      Yes
8      No
9      Yes
7      Yes
0      No
Name: Online Shopper, dtype: object
```

In [23]:

```
print(x_test)
```

```
      Region  Age  Income
2      USA  35.0  64800.0
6  Brazil   NaN  62400.0
1  Brazil  32.0  57600.0
```

In [24]:

```
print(y_test)
```

```
2      No
6      No
1      Yes
Name: Online Shopper, dtype: object
```

In [25]:

```
import sklearn as sk
```

In [26]:

```
scalar = sk.preprocessing.MinMaxScaler(feature_range=(0, 1))
```

In [29]:

```
v = dataset[['Age', 'Income']]
```

In [30]:

```
rescaleData = scalar.fit_transform(v)
```

In [34]:

```
rescaleData
```

Out[34]:

```
array([[0.73913043, 0.68571429],
       [0.        , 0.        ],
       [0.13043478, 0.17142857],
       [0.47826087, 0.37142857],
       [0.56521739, 0.        ],
       [0.34782609, 0.28571429],
       [0.        , 0.11428571],
       [0.91304348, 0.88571429],
       [1.        , 1.        ],
       [0.43478261, 0.54285714]])
```

In []:

In []: