

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

```
In [2]: from sklearn.linear_model import LogisticRegression
```

```
In [3]: df=pd.read_csv("C4 framingham csv").dropna()

df
```

```
Out[3]:
```

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentH
0	1	39	4.0	0	0.0	0.0	0	
1	0	46	2.0	0	0.0	0.0	0	
2	1	48	1.0	1	20.0	0.0	0	
3	0	61	3.0	1	30.0	0.0	0	
4	0	46	3.0	1	23.0	0.0	0	
...	
4231	1	58	3.0	0	0.0	0.0	0	
4232	1	68	1.0	0	0.0	0.0	0	
4233	1	50	1.0	1	1.0	0.0	0	
4234	1	51	3.0	1	43.0	0.0	0	
4237	0	52	2.0	0	0.0	0.0	0	

```
In [8]: df.dropna(inplace=True)
```

In [5]: df.info()

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 3656 entries, 0 to 4237
Data columns (total 16 columns):
 #   Column                Non-Null Count  Dtype  
---  -
 0   male                  3656 non-null   int64  
 1   age                   3656 non-null   int64  
 2   education              3656 non-null   float64 
 3   currentSmoker          3656 non-null   int64  
 4   cigsPerDay             3656 non-null   float64 
 5   BPMeds                 3656 non-null   float64 
 6   prevalentStroke        3656 non-null   int64  
 7   prevalentHyp           3656 non-null   int64  
 8   diabetes               3656 non-null   int64  
 9   totChol               3656 non-null   float64 
10   sysBP                 3656 non-null   float64 
11   diaBP                 3656 non-null   float64 
12   BMI                   3656 non-null   float64 
13   heartRate             3656 non-null   float64 
14   TenYearCHD            3656 non-null   int64  
```

In [9]: feature_matrix = df[['male', 'age', 'education', 'currentSmoker', 'cigsPerDay', 'BPMeds', 'prevalentStroke', 'prevalentHyp', 'diabetes', 'totChol', 'sysBP', 'diaBP', 'BMI', 'heartRate', 'TenYearCHD']
target_vector = df['TenYearCHD']

In [10]: feature_matrix.shape

Out[10]: (3656, 15)

In [11]: target_vector.shape

Out[11]: (3656,)

In [12]: from sklearn.preprocessing import StandardScaler

In [13]: fs = StandardScaler().fit_transform(feature_matrix)

In [14]: logr = LogisticRegression()
logr.fit(fs, target_vector)

Out[14]: LogisticRegression()

In [15]: feature_matrix.shape

Out[15]: (3656, 15)

In [16]: target_vector.shape

Out[16]: (3656,)

```
In [17]: from sklearn.preprocessing import StandardScaler
```

```
In [18]: fs = StandardScaler().fit_transform(feature_matrix)
```

```
In [19]: logr = LogisticRegression()  
logr.fit(fs,target_vector)
```

```
Out[19]: LogisticRegression()
```

```
In [21]: observation=df[['male','age','education','currentSmoker','cigsPerDay','BPMeds',  
                        'diabetes','totChol','sysBP','diaBP','BMI','heartRate','g'
```

```
In [22]: prediction = logr.predict(observation)  
prediction
```

```
Out[22]: array([1, 1, 1, ..., 1, 1, 1], dtype=int64)
```

```
In [23]: logr.classes_
```

```
Out[23]: array([0, 1], dtype=int64)
```

```
In [24]: logr.predict_proba(observation)[0][1]
```

```
Out[24]: 1.0
```

```
In [ ]:
```