In [1]:
```
#Aim: KNN k nearest Nabour
#Exp no:11
#Name:Shrutika Vijay Ambekar
#Sec:B
#Roll no:01
#Sub:ET-1
#Date:10/10/2024
```

Importing Libraries

In [4]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from sklearn.model_selection import train_test_split
import warnings
warnings.filterwarnings('ignore')
```

In [6]:
```
import os
```

In [8]:
```
os.getcwd()
```

Out[8]:
```
'C:\\Users\\asus'
```

In [10]:
```
os.chdir("C:\\Users\\asus\\Desktop")
```

In [12]:
```
df=pd.read_csv("framingham.csv")
```

In [ ]:
```
#The "Framingham" heart disease dataset includes over 4,240 records, 15 attributes.
#The goal of the dataset is to predict whether the patient has 10-year risk of future (C
```

In [14]:
```
df.head()
```

Out[14]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	
0	1	39	4.0	0	0.0	0.0	0	0	0	195.0	106.0	70.0	26
1	0	46	2.0	0	0.0	0.0	0	0	0	250.0	121.0	81.0	28
2	1	48	1.0	1	20.0	0.0	0	0	0	245.0	127.5	80.0	25
3	0	61	3.0	1	30.0	0.0	0	1	0	225.0	150.0	95.0	28
4	0	46	3.0	1	23.0	0.0	0	0	0	285.0	130.0	84.0	23

In [16]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4238 entries, 0 to 4237
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   male             4238 non-null   int64  
 1   age              4238 non-null   int64  
 2   education        4133 non-null   float64
 3   currentSmoker    4238 non-null   int64  
 4   cigsPerDay       4209 non-null   float64
 5   BPMeds           4185 non-null   float64
 6   prevalentStroke  4238 non-null   int64  
 7   prevalentHyp     4238 non-null   int64  
 8   diabetes         4238 non-null   int64  
 9   totChol          4188 non-null   float64
 10  sysBP            4238 non-null   float64
 11  diaBP            4238 non-null   float64
 12  BMI              4219 non-null   float64
 13  heartRate        4237 non-null   float64
 14  glucose          3850 non-null   float64
 15  TenYearCHD       4238 non-null   int64  
dtypes: float64(9), int64(7)
memory usage: 529.9 KB
```

In [18]:
```
df.isna().sum()
```

Out[18]:
```
male                0
age                 0
education         105
currentSmoker       0
cigsPerDay         29
BPMeds             53
prevalentStroke     0
prevalentHyp        0
diabetes            0
totChol            50
sysBP               0
diaBP               0
BMI                19
heartRate           1
glucose           388
TenYearCHD          0
dtype: int64
```

In [20]:
```
#Since, only a few rows have null values in them, we are only removing those rows from t
#df = df.dropna(subset=['heartRate','BMI','cigsPerDay','totChol','BPMeds'])
```

In [22]:
```
df
```

Out[22]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP
0	1	39	4.0	0	0.0	0.0	0	0	0	195.0	106.0	70.0
1	0	46	2.0	0	0.0	0.0	0	0	0	250.0	121.0	81.0
2	1	48	1.0	1	20.0	0.0	0	0	0	245.0	127.5	80.0
3	0	61	3.0	1	30.0	0.0	0	1	0	225.0	150.0	95.0
4	0	46	3.0	1	23.0	0.0	0	0	0	285.0	130.0	84.0
...	...	...	...	...	...	...	...	...	...	...	...	...
4233	1	50	1.0	1	1.0	0.0	0	1	0	313.0	179.0	92.0
4234	1	51	3.0	1	43.0	0.0	0	0	0	207.0	126.5	80.0
4235	0	48	2.0	1	20.0	NaN	0	0	0	248.0	131.0	72.0
4236	0	44	1.0	1	15.0	0.0	0	0	0	210.0	126.5	87.0
4237	0	52	2.0	0	0.0	0.0	0	0	0	269.0	133.5	83.0

4238 rows × 16 columns

Missing Value Treatment

Since, 'glucose' and 'education' columns had a significant amount of null values, so we replaced them with the mean of values for their respective columns

In [26]:
```
df['glucose'].fillna(value = df['glucose'].mean(),inplace=True)
```

In [28]:
```
df['education'].fillna(value = df['education'].mean(),inplace=True)
```

In [30]:
```
df['heartRate'].fillna(value = df['heartRate'].mean(),inplace=True)
```

In [32]:
```
df['BMI'].fillna(value = df['BMI'].mean(),inplace=True)
```

In [34]:
```
df['cigsPerDay'].fillna(value = df['cigsPerDay'].mean(),inplace=True)
```

In [36]:
```
df['totChol'].fillna(value = df['totChol'].mean(),inplace=True)
```

In [38]:
```
df['BPMeds'].fillna(value = df['BPMeds'].mean(),inplace=True)
```

In [40]:
```
df.isna().sum()
```

Out[40]:
```
male               0
age                0
education          0
currentSmoker      0
cigsPerDay         0
BPMeds             0
prevalentStroke    0
prevalentHyp       0
diabetes           0
totChol            0
sysBP              0
diaBP              0
BMI                0
heartRate          0
glucose            0
TenYearCHD         0
dtype: int64
```

In [42]:
```
#Splitting the dependent and independent variables.
x = df.drop("TenYearCHD",axis=1)
y = df['TenYearCHD']
```

In [44]:
```
x #checking the features
```

Out[44]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP
0	1	39	4.0	0	0.0	0.00000	0	0	0	195.0	106.0	70.0
1	0	46	2.0	0	0.0	0.00000	0	0	0	250.0	121.0	81.0
2	1	48	1.0	1	20.0	0.00000	0	0	0	245.0	127.5	80.0
3	0	61	3.0	1	30.0	0.00000	0	1	0	225.0	150.0	95.0
4	0	46	3.0	1	23.0	0.00000	0	0	0	285.0	130.0	84.0
...	...	...	...	...	...	...	...	...	...	...	...	...
4233	1	50	1.0	1	1.0	0.00000	0	1	0	313.0	179.0	92.0
4234	1	51	3.0	1	43.0	0.00000	0	0	0	207.0	126.5	80.0
4235	0	48	2.0	1	20.0	0.02963	0	0	0	248.0	131.0	72.0
4236	0	44	1.0	1	15.0	0.00000	0	0	0	210.0	126.5	87.0
4237	0	52	2.0	0	0.0	0.00000	0	0	0	269.0	133.5	83.0

4238 rows × 15 columns

In [ ]: