SVM classifier

```python
In [ ]:   #Name : Vaibhav Laxman Karale
          #Roll no. 58
          #sub:E.T.1
          #Section :3A
          #Date:05/10/2024
```

```python
In [82]:   # Aim: to perform operation on svm classifier
```

```python
In [45]:  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')
```

```python
In [46]:  import os
```

```python
In [5]:   os.getcwd()
```

```
Out[5]:  'C:\\Users\\DELL'
```

```python
In [6]:  os.chdir("C:\\Users\\DELL\\OneDrive\\Desktop")
```

```python
In [48]:  df=pd.read_csv("C:\\Users\\DELL\\OneDrive\\Desktop\\framingham - Copy.csv")
```

```
In [49]: df
```

Out[49]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diab
0	1	39	4.0	0	0.0	0.0	0	0	
1	0	46	2.0	0	0.0	0.0	0	0	
2	1	48	1.0	1	20.0	0.0	0	0	
3	0	61	3.0	1	30.0	0.0	0	1	
4	0	46	3.0	1	23.0	0.0	0	0	
...	...	...	...	...	...	...	...	...	
4233	1	50	1.0	1	1.0	0.0	0	1	
4234	1	51	3.0	1	43.0	0.0	0	0	
4235	0	48	2.0	1	20.0	NaN	0	0	
4236	0	44	1.0	1	15.0	0.0	0	0	
4237	0	52	2.0	0	0.0	0.0	0	0	

4238 rows × 16 columns

```
In [50]: df.head()
```

Out[50]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabete
0	1	39	4.0	0	0.0	0.0	0	0	
1	0	46	2.0	0	0.0	0.0	0	0	
2	1	48	1.0	1	20.0	0.0	0	0	
3	0	61	3.0	1	30.0	0.0	0	1	
4	0	46	3.0	1	23.0	0.0	0	0	

```
In [51]: df.tail()
```

Out[51]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diab
4233	1	50	1.0	1	1.0	0.0	0	1	
4234	1	51	3.0	1	43.0	0.0	0	0	
4235	0	48	2.0	1	20.0	NaN	0	0	
4236	0	44	1.0	1	15.0	0.0	0	0	
4237	0	52	2.0	0	0.0	0.0	0	0	

```
In [52]: df.shape
```

Out[52]: (4238, 16)

```
In [53]: df.size
```

Out[53]: 67808

```
In [54]: df.ndim
```

Out[54]: 2

```
In [55]: 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 [56]: df.describe()
```

Out[56]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStro
count	4238.000000	4238.000000	4133.000000	4238.000000	4209.000000	4185.000000	4238.0000
mean	0.429212	49.584946	1.978950	0.494101	9.003089	0.029630	0.0058
std	0.495022	8.572160	1.019791	0.500024	11.920094	0.169584	0.0765
min	0.000000	32.000000	1.000000	0.000000	0.000000	0.000000	0.0000
25%	0.000000	42.000000	1.000000	0.000000	0.000000	0.000000	0.0000
50%	0.000000	49.000000	2.000000	0.000000	0.000000	0.000000	0.0000
75%	1.000000	56.000000	3.000000	1.000000	20.000000	0.000000	0.0000
max	1.000000	70.000000	4.000000	1.000000	70.000000	1.000000	1.0000

```
In [57]:  df.isnull().sum()
```

```
Out[57]:  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 [59]:  print(x)
```

```
[[  1.    39.     4.    ...  26.97  80.     77.  ]
 [  0.    46.     2.    ...  28.73  95.     76.  ]
 [  1.    48.     1.    ...  25.34  75.     70.  ]
 ...
 [  0.    48.     2.    ...  22.    84.     86.  ]
 [  0.    44.     1.    ...  19.16  86.     nan]
 [  0.    52.     2.    ...  21.47  80.    107.  ]]
```

```
In [60]:  print(y)
```

```
[0 0 0 ... 0 0 0]
```

```
In [61]:  print(y)
```

```
[0 0 0 ... 0 0 0]
```

```
In [62]:  import matplotlib.pyplot as plt
          import seaborn as sns
          import numpy as np
```

```
In [63]:  #Splitting testdata into X_train,X_test,y_train,y_test
          from sklearn.model_selection import train_test_split
          x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=.3,random_state=42)
```

```
In [64]:  print(x_train)
```

```
[[ 0.    42.    1.    ...  22.19 70.     nan]
 [ 0.    64.    1.    ...  29.97 65.    87.  ]
 [ 0.    39.    2.    ...  20.55 68.     nan]
 ...
 [ 0.    60.    2.    ...  25.09 83.    81.  ]
 [ 1.    39.    2.    ...  24.5  68.    62.  ]
 [ 0.    35.    2.    ...  20.64 90.    80.  ]]
```

```
In [65]:  print(y_train)
```

[0 0 0 ... 0 0 0]

missing value treatment

```
In [66]:  df['glucose'].fillna(value = df['glucose'].mean(),inplace=True)
```

```
In [67]:  df['education'].fillna(value = df['education'].mean(),inplace=True)
```

```
In [68]:  df['heartRate'].fillna(value = df['heartRate'].mean(),inplace=True)
```

```
In [69]:  df['BMI'].fillna(value = df['BMI'].mean(),inplace=True)
```

```
In [70]:  df['cigsPerDay'].fillna(value = df['cigsPerDay'].mean(),inplace=True)
```

```
In [71]:  df['totChol'].fillna(value = df['totChol'].mean(),inplace=True)
```

```
In [72]:  df['BPMeds'].fillna(value = df['BPMeds'].mean(),inplace=True)
```

```
In [73]:  df.isna().sum()
```

```
Out[73]:  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 [76]:  #Splitting the dependent and independent variables.
          x = df.drop("TenYearCHD",axis=1)
          y = df['TenYearCHD']
```

```
In [77]: x
```

Out[77]:

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diab
0	1	39	4.0	0	0.0	0.00000	0	0	
1	0	46	2.0	0	0.0	0.00000	0	0	
2	1	48	1.0	1	20.0	0.00000	0	0	
3	0	61	3.0	1	30.0	0.00000	0	1	
4	0	46	3.0	1	23.0	0.00000	0	0	
...	...	...	...	...	...	...	...	...	
4233	1	50	1.0	1	1.0	0.00000	0	1	
4234	1	51	3.0	1	43.0	0.00000	0	0	
4235	0	48	2.0	1	20.0	0.02963	0	0	
4236	0	44	1.0	1	15.0	0.00000	0	0	
4237	0	52	2.0	0	0.0	0.00000	0	0	

4238 rows × 15 columns

train test split

```
In [78]: x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=42
```

```
In [79]: y_train
```

Out[79]:
```
3252    0
3946    0
1261    0
2536    0
4089    0
       ..
3444    0
466     0
3092    0
3772    0
860     0
Name: TenYearCHD, Length: 3390, dtype: int64
```

SVM Classifier

```python
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score
svc=SVC()
svc.fit(x_test,y_test)
acc = svc.score(x_test,y_test)*100
print(acc)
```

85.37735849056604