Supervised Machine Learning Classification

Data Set :-

The **Heart disease data** is chosen by me for this analysis as I have interest in the Medical related field.

The dataset contains – (4238,16) observations.

Rows – 4238, columns – 16

- The data types used in this specific problem is numeric and no Object datatype is used.
- The target column or feature is TenYearCHD(chronic Heart disease)

```
In [5]: data.dtypes

Out[5]: male               int64
        age                int64
        education        float64
        currentSmoker      int64
        cigsPerDay       float64
        BPMeds           float64
        prevalentStroke    int64
        prevalentHyp       int64
        diabetes           int64
        totChol          float64
        sysBP            float64
        diaBP            float64
        BMI              float64
        heartRate        float64
        glucose          float64
        TenYearCHD         int64
        dtype: object
```

- The complete description of the features are listed below for this dataset.
  Feature 1)Male – 1 and Female – 0 which is Nominal data.
  Feature 2) Age - int data type which is continous.
  Feature 3) education - is a Nominal data which is not going to provide any medical evidence to the heart disease prediction which will be dropped in further analysis.
  Feature 4) currentSmoker – 0 or 1 whether the person smokes or not.
  Feature 5) cigsPerDay – The total number of cigrattees smoked by the patient and the count value is provided in the column.
  Featuer 6) BPMeds - whether or not the patient was on blood pressure medication (Nominal)
  Feature 7) prevalentStroke: whether or not the patient had previously had a stroke (Nominal)
  Feature 8) prevalentHyp: whether or not the patient was hypertensive (Nominal)
  Feature 9) diabetes: whether or not the patient had diabetes (Nominal)
  Feature 10) totChol: total cholesterol level (Continuous)
  Feature 11) sysBP: systolic blood pressure (Continuous)
  Feature 12) diaBP: diastolic blood pressure (Continuous)
  Feature 13) BMI: Body Mass Index (Continuous)

Feature 14) heartRate: heart rate (Continuous - In medical research, variables such as heart rate though in fact discrete, yet are considered continuous because of large number of possible values.)
Feature 15) glucose: glucose level (Continuous)
Feature16) Target Feature - 10 year risk of coronary heart disease CHD (binary: “1”, means “Yes”, “0” means “No”)

- Checking if there are any NAN values present in the dataset by below method.

```
In [17]: data.isnull().sum()

Out[17]: male               0
         age                0
         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
```

- From the observation we can determine that the features like glucose, BPMeds, cigsPerDay BMI, totChol has around total of 500 missing value which is just 10% of the total observation o I will be dropping the NAN values from this dataset using the dropna command.

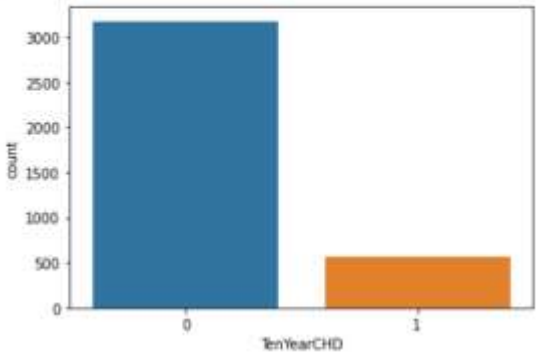
```
In [18]: data.dropna(axis=0,inplace=True)

In [19]: data.isnull().sum()

Out[19]: male             0
         age              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
```

- We may need to analyze the dataset is balanced for predicting the target column. Which can be performed by the valuecounts() method.

```
In [32]: sn.countplot(x='TenYearCHD',data=data)
Out[32]: <AxesSubplot:xlabel='TenYearCHD', ylabel='count'>
```



From the observation its clear that the target value which is binary in nature has the decent evidance for the positive and negative class. If the negative or the positive class has a very less observations then we can do a up or down sampling for the class and make it appear equal for the machine learning model.

- The sns.pairplot will be plotted in order to understand the correlation between the features and the dependent variable.
- The features like age, totChol, sysBP, BMI, heartrate, Glucose are continuous in nature and they are measured in different scale. So I will be using the min_max scaler to make sure all the features are measures in the same scale and range.
- Before the features are used for training a model, they should be scaled using the MinMax scaler or the standard Scaler.
- The sklearn has the Logistic Regression I will be importing that specific algorithm for fitting the training data of the dataset.

```
[ ]: new_features=data[['age','Sex_male','cigsPerDay','totChol','sysBP','glucose','TenYearCHD']]
     x=new_features.iloc[:,:-1]
     y=new_features.iloc[:,-1]
     from sklearn.cross_validation import train_test_split
     x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=.20,random_state=5)
```

The xtrain values then be fitted to the Logistic Regression algorithm.

```
In [37]: from sklearn.linear_model import LogisticRegression
         log=LogisticRegression()
         log.fit(x_train,y_train)
         y_pred=log.predict(x_test)
```

- The model accuracy is found by the sklearn library. sklearn.metrics.accuracy_score(y_test,y_pred)

- The sklearn confusion matrix is used to find the FP, TP, FN, TN values, with which the furthe r hyperpara meter tuning can be done for the model.Further improvement in the model we ca n use the SVM in order to do the same activity for predicting the heart diesease.