CSE 310-L DATA WAREHOUSING AND MINING LAB PROJECT

HEART DISEASE PREDICTION USING DATA MINING ALGORITHMS

Submitted to

Dr. Saleti Sumalatha

Assistant Professor

Department of Computer Science Engineering

SRM University AP

Submitted by

Abhiram Thiriveedhi AP20110010457

Bhargav Kumbham AP20110010467

Mohan Krishna Komati AP20110010433

Vinay Kunisetty AP20110010464

Jasmitha Pusuluri AP20110010473

Introduction

Nowadays, heart diseases are a leading cause of death and disability in many parts of the world. Heart disease describes a range of conditions that affect your heart.

Prediction of cardiovascular disease is regarded as one of the most important subjects in the section of clinical data analysis.

One of the ways to predict heart diseases is to use data mining techniques. Data mining turns the large collection of raw healthcare data into information that can help to make informed decisions and predictions.

The amount of data in the healthcare industry is huge. Every year about 735,000 Americans have a heart attack. Of these, 525,000 are a first heart attack and 210,000 happen in people who have already had a heart attack. It is difficult to identify heart disease because of several contributory risk factors such as diabetes, high blood pressure, high cholesterol, abnormal pulse rate, and many other factors.

Due to such constraints modern approaches like Data Mining seems a viable technique for predicting the disease. It proves to be effective in assisting in making decisions and predictions from the large quantity of data produced by the healthcare industry.

Dataset information

The dataset we used is from the University of California Irvine's Machine Learning Repository at https://archive.ics.uci.edu/ml/datasets/Heart+Disease

The dataset contains 14 attributes which are used to predict whether the patient has a heart disease or not

1. age – age of patient

2. sex – gender of patient (1 for MALE / 0 for FEMALE)

3. cp – chest pain type (1: typical angina 2: atypical angina 3: non-anginal pain 4: asymptomatic)

4. bp – blood pressure

5. chol - cholestrol

6. fbs – fasting blood sugar over 120 (1 if YES / 0 if NO)

7. ekg result -

8. max hr – max heart rate

9. Exercise angina

10. ST depression

11. slope of st – the slope of the peak exercise ST segment

    Value 1: upsloping

    Value 2: flat

    Value 3: downsloping

12. number of vessels fluro

13. thallium – (3 = normal; 6 = fixed defect; 7 = reversable defect)

14. Heart disease (the predicted attribute)

SRM UNIVERSITY AP Andhra Pradesh

Algorithms used in the project

Naïve Bayes

Naive Bayes is a data mining algorithm that is used for classification tasks. It is based on the idea of applying Bayes' theorem, which is a statistical theorem that describes the probability of an event occurring based on certain conditions.

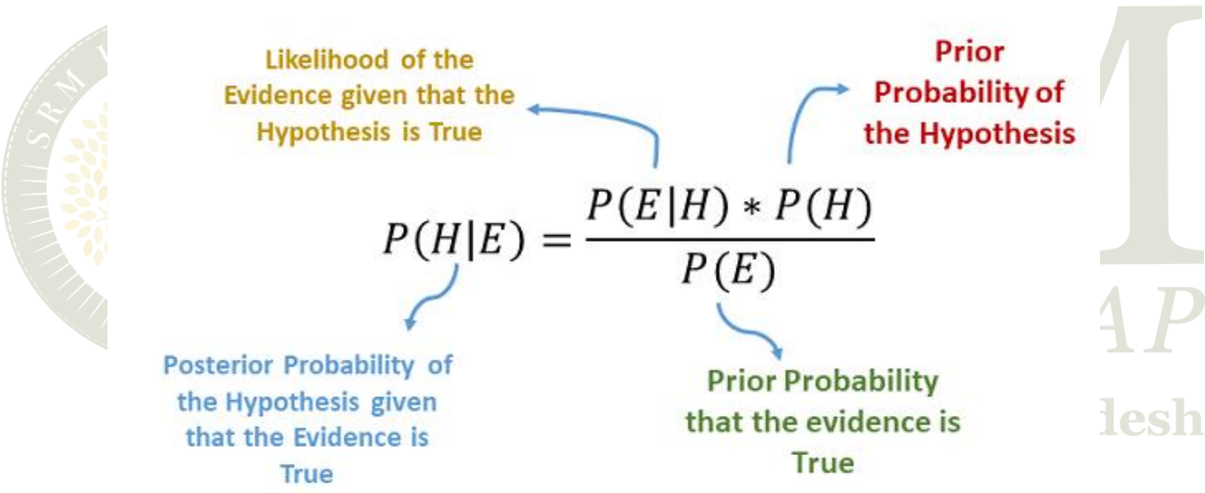
The algorithm works by using the training data to estimate the probabilities of different events occurring, and then using these probabilities to make predictions about new, unseen data.

One of the key assumptions of the Naive Bayes algorithm is that all of the features in the data are independent of one another, which is why it is called "naive." This assumption simplifies the calculations and makes it easier to apply the algorithm to large datasets.

Naive Bayes is a popular algorithm for text classification tasks, such as spam filtering and sentiment analysis, and it is also used in a variety of other applications, including medical diagnosis and credit risk assessment. It is relatively simple to implement and can perform well with large datasets, making it a useful tool in many different contexts.

The formula for Bayes' theorem is given as:

$$P(A|B)=\frac{P(B|A)P(A)}{P(B)}$$



KNN clustering technique

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.

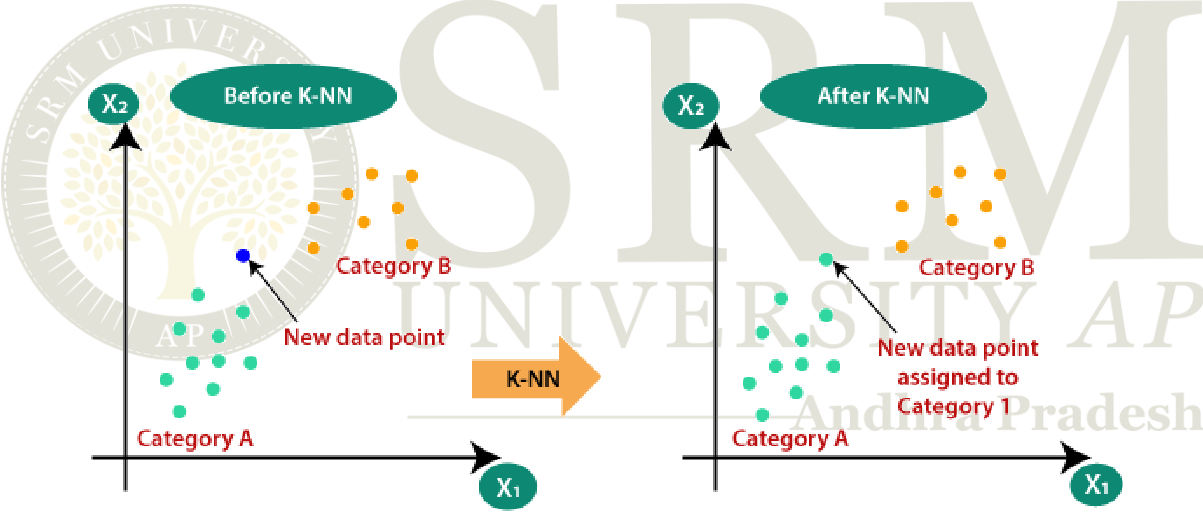
K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.



Python Libraries Used

Pandas

Numpy

Matplotlib

Sklearn

Seaborn

Implementation

Step-1

Preprocess the dataset

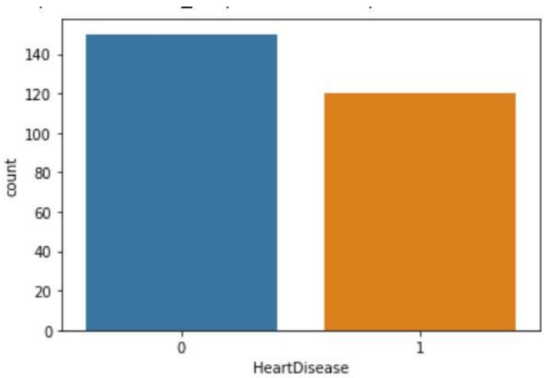
Since the BP column have some values missing,to fill them, we use the mean of the BP column

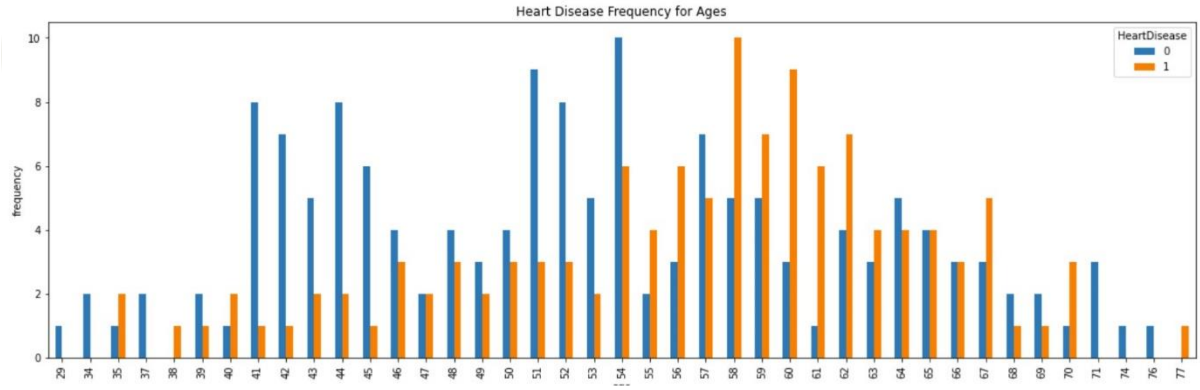
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 270 entries, 0 to 269
Data columns (total 14 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Age                      270 non-null    int64  
 1   Sex                      270 non-null    int64  
 2   Chest pain type          270 non-null    int64  
 3   BP                       253 non-null    float64
 4   Cholesterol              270 non-null    int64  
 5   FBS over 120             270 non-null    int64  
 6   EKG results              270 non-null    int64  
 7   Max HR                   270 non-null    int64  
 8   Exercise angina          270 non-null    int64  
 9   ST depression            270 non-null    float64
 10  Slope of ST              270 non-null    int64  
 11  Number of vessels fluro  270 non-null    int64  
 12  Thallium                 270 non-null    int64  
 13  HeartDisease             270 non-null    object 
dtypes: float64(2), int64(11), object(1)
memory usage: 29.7+ KB

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 270 entries, 0 to 269
Data columns (total 14 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Age                      270 non-null    int64  
 1   Sex                      270 non-null    int64  
 2   Chest pain type          270 non-null    int64  
 3   BP                       270 non-null    float64
 4   Cholesterol              270 non-null    int64  
 5   FBS over 120             270 non-null    int64  
 6   EKG results              270 non-null    int64  
 7   Max HR                   270 non-null    int64  
 8   Exercise angina          270 non-null    int64  
 9   ST depression            270 non-null    float64
 10  Slope of ST              270 non-null    int64  
 11  Number of vessels fluro  270 non-null    int64  
 12  Thallium                 270 non-null    int64  
 13  HeartDisease             270 non-null    object 
dtypes: float64(2), int64(11), object(1)
memory usage: 29.7+ KB
```

Since we are only concerned about the effect of age,bp and cholesterol on heart diseases we drop the remaining columns

Step-2

Plot the bar graphs depicting the genders which are most diagnosed with heart diseases and ages which are most effected by heart diseases





Step 3

For classification apply the Naïve-Bayes algorithm

a) Divide the data set into training and testing data
b) Using sklearn library,apply naïve-bayes theorem
c) Create the model for the training dataset and use the same model to predict for the testing dataset
d) Find the accuracy and precision
e) Construct the confusion matrix

Step 4

For clustering apply the KNN clustering algorithm

a) Divide the data set into training and testing data
b) Use the Kneighbourclassifier from sklearn library to form the clusters
c) Create the model for the training dataset and use the same model to predict the result for the testing dataset
d) Find the accuracy and precision
e) Construct the confusion matrix

Observations/Results

Accuracy and Precision for:

a)Naïve-Bayes's algorithm

```
score_knn = round(accuracy_score(Y_pred,Y_test)*100,2)

print("The accuracy score achieved using KNN is: "+str(score_knn)+" %")
```

```
The accuracy score achieved using KNN is: 49.47 %
```

```
[39] from sklearn.metrics import precision_score
precision = precision_score(Y_test, Y_pred)
print("Precision: ",precision)
```

```
Precision:  0.4090909090909091
```

b) KNN algorithm

```
score_nb = round(accuracy_score(y_pred,y_test)*100,2)

print("The accuracy score achieved using Naive Bayes is: "+str(score_nb)+" %")
```

```
The accuracy score achieved using Naive Bayes is: 66.67 %
```

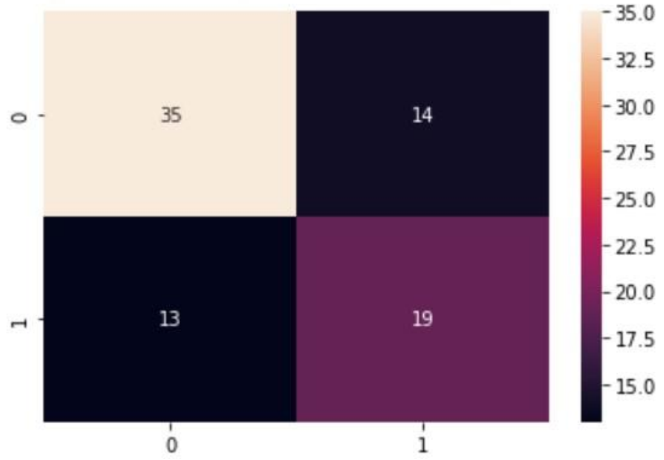
```
[37] from sklearn.metrics import precision_score
precision = precision_score(y_test, y_pred)
print("Precision: ",precision)
```

```
Precision:  0.5757575757575758
```

Confusion matrix for:

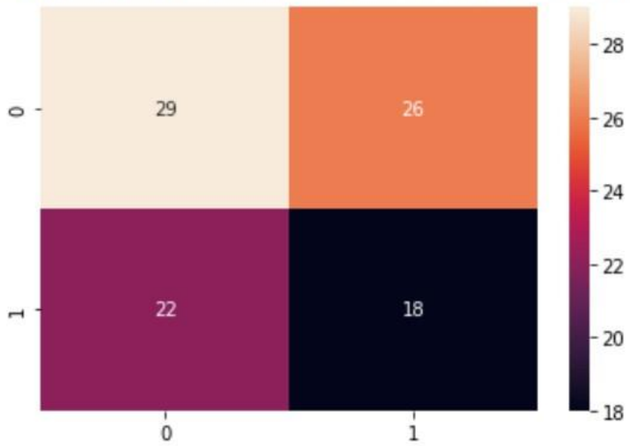
a)Naïve-Bayes theorem

<matplotlib.axes._subplots.AxesSubplot at 0x7ff015e890d0>



b)KNN algorithm

<matplotlib.axes._subplots.AxesSubplot at 0x7f36774d8b20>



References

1-https://www.kaggle.com/datasets/rishidamarla/heart-disease-prediction

2- https://archive.ics.uci.edu/ml/datasets/Heart+Disease

3-https://www.javatpoint.com/machine-learning-naive-bayes-classifier

4-https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

SRM UNIVERSITY AP
Andhra Pradesh