STAT 8670 Project : Predict Heart Disease Using Maximum Likelihood Classifier

Anusha Palisetty

April 2020

1 Introduction

One of the main reasons for death in the world is heart disease. Several techniques are being used to help the professionals in the diagnosis of heart disease. Here we use the Maximum Likelihood statistical method to predict if a patient has a Heart Disease or not. Given a set of parameters and i.i.d samples we calculate the maximum likelihood of a function for new data point and predict if he has a heart disease or not. We estimate the parameters of assumed distribution for the data and evaluate the PDF for each class label. The label is classified for each data point which has the maximum PDF value.

The main goal of Maximum Likelihood Classification is to predict the class label y, that maximizes the likelihood of the observed data $X = (X_1, X_2 \ldots . X_n)$. We assume x to be a random vector and y is a prediction column which depends on the distribution of x. Initially we assume that the distribution is a Multivariate Gaussian distribution. We split the dataset corresponding to each label of y. We estimate the parameters of distribution of X for each split of the dataset and provided to the PDF function.
For example consider a 1-dimensional input x and two classes y=0 and y=1

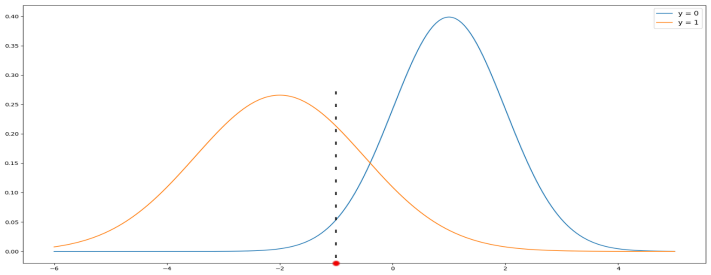


Figure 1: Example

Assume after the parameters are estimated under y=0 and y=1, we get the above two PDF plots. For y=0(blue) the mean μ=1 and standard deviation σ=1; for y=1(orange) the mean μ= -2 and σ=1.5. Now for a new data point x=-1, we want to predict the label y by evaluating both the PDF's: $f_{y=0}(-1) \approx 0.05$; $f_{y=1}(-1) \approx 0.21$. Now comparing both the values 0.21 is biggest value when y=1, so we predict the label as y=1.

2 Implementation

The above example is simple one, but in real-world situations we will have more input input variables to be used in order to make prediction.We use Multivariate Guassian distribution here

2.1 Notation

The multivariate normal distribution of a k-dimensional random vector $X = (X_1, X_2...X_n)^T$ can be written as

$$X \sim N(\mu, \Sigma) \tag{1}$$

with k dimensional mean vector

$$\mu = E[X] = (E[X_1], E[X_2], ...E[X_n]) \tag{2}$$

and kxk covariance martix

$$\Sigma_{i,j} = E[(X_i - \mu_i)(X_j - \mu_j)] = Cov[X_i, X_j] \tag{3}$$

2.2 Density Function

The multivariate normal distribution is said to be non-generate when the symmetric covariance matrix Σ is positive definite. The density function is given as

$$f(x) = \frac{1}{\sqrt{(2\pi)^d det\Sigma}}.e^{\frac{-1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \tag{4}$$

where x= a column vector with data from one observation
d=dimension of x
μ= mean vector
Σ= Covariance matrix of x

The covariance matrix should be a positive definite, i.e it should be symmetric and all the eigen values should be positive. The covariance matrix contains the covariances between all pairs of components of x.

$$\Sigma_{ij} = cov(x_i, x_j) \tag{5}$$

So, $cov(x_j, x_i) = cov(x_i, x_j)$ and we check if all the eigen values are positive. If ther are more observations than variables and variables don't have high correlation between them this condition will be met and Σ should be definite positive.

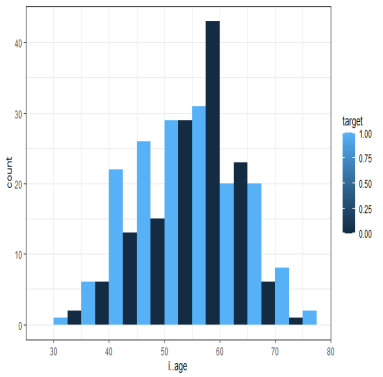
3 Data Analysis

We consider data from https://www.kaggle.com/ronitf/heart-disease-uci for 303 patients. This csv file has 303 rows each one has 13 columns that we can use for prediction and 1 label column. A brief description of each field is given the table.

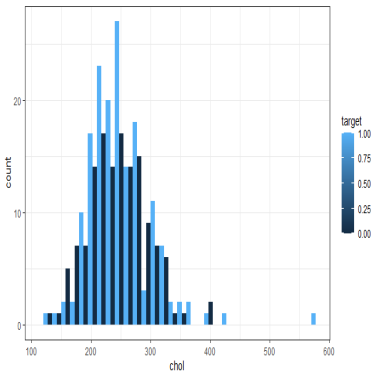
Column Name	Description
age	age in years
sex	1 = male; 0 = female
cp	chest pain type
trestbps	resting blood pressure (in mm Hg on admission to the hospital)
chol	serum cholestoral in mg/dl
fbs	fasting blood sugar 120 mg/dl (1 = true; 0 = false)
restecg	resting electrocardiographic results
thalach	maximum heart rate achieved
exang	exercise induced angina (1 = yes; 0 = no)
oldpeak	ST depression induced by exercise relative to rest
slope	the slope of the peak exercise ST segment
ca	number of major vessels (0-3) colored by flourosopy
thal	3 = normal; 6 = fixed defect; 7 = reversable defect
target	1 = heart disease; 0 = no heart disease

3.1 Data Visualizations

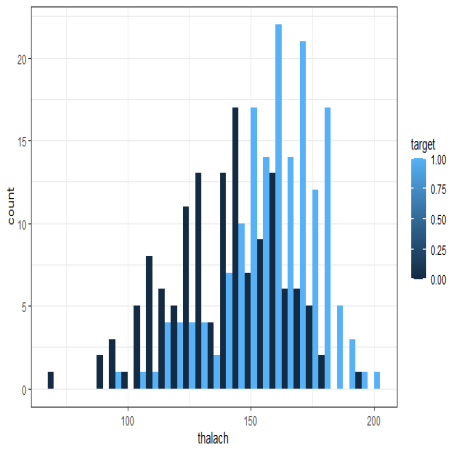
Data Visualization is best way to represent the data to get useful knowledge. We plot the data points through histograms which breaks the data into bins and shows the distribution of the data.If we want to know the count of patients on basis of their target columns, then we can plot histogram using continuous variable age,chol, cp etc as shown below.



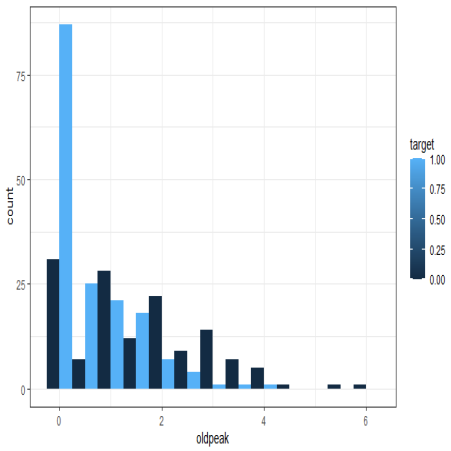
(a) Age Distribution



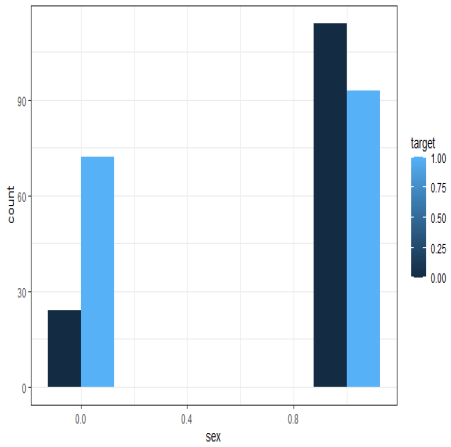
(b) Cholestrol



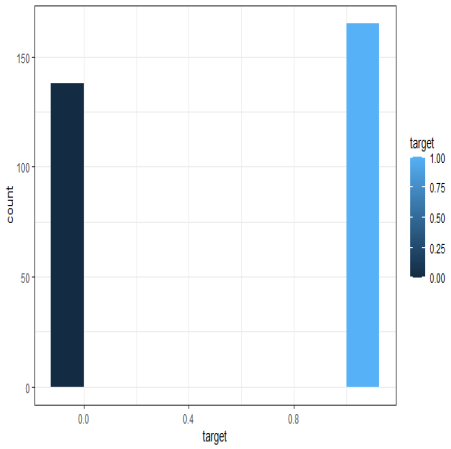
(a) Maximum Heart Rate



(b) oldpeak



(a) Gender Distribution



(b) Target

From the Visualization we observe that the data distribution for Age and Cholesterol seems to be normal distribution.

From the Age Distribution plot we observe that the count of patients with Age 60 with No Heart Disease is more than the patients of same Age with Heart disease.

From the Target distribution we observe that the count of patients with "No Heart Disease" is higher than the count of patients with "Heart Disease."

3.2 Correlation Matrix

We compute a correlation matrix from the data. This matrix provides the correlation coefficients for each variable X_i with respect to other variables X_j.

We plot a heatmap for this correlation matrix as shown in the figure. From the figure we observe that the main diagonal has highest value 1, which shows that each variable is perfectly correlated with itself.We also observe that this matrix is symmetrical, as the same correlation is shown above the main diagonal being a mirror image of those below the main diagonal.

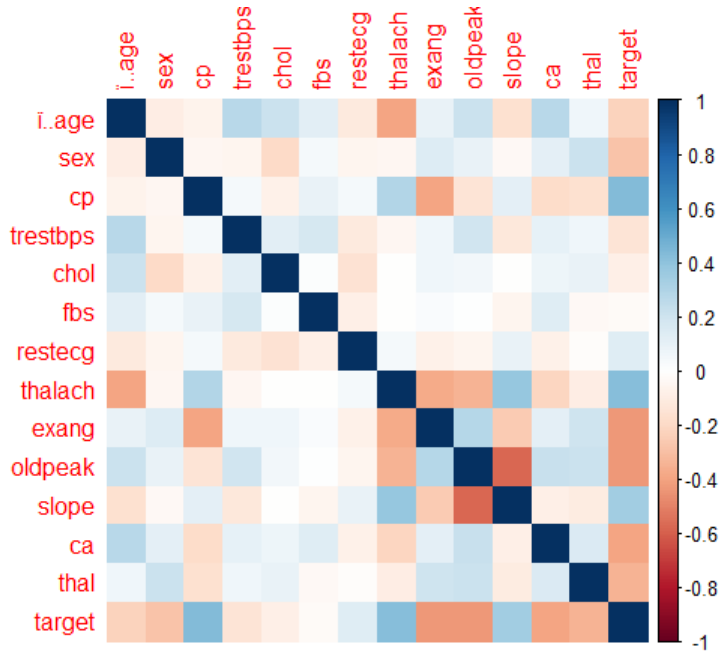


Figure 5: Correlation Between Variables

4 Results and Discussion

4.1 Calculate Mean Vector

After the data analysis we implement the Maximum Likelihood function. Initially split the independent variables and target variable. Now we have two sets of data X for each label

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} & X_{1,3} \cdots X_{1,13} \\ X_{2,1} & X_{2,2} & X_{2,3} \cdots X_{2,13} \\ \vdots & \vdots & \ddots \\ X_{303,1} & X_{303,2} & X_{303,3} \cdots X_{303,13} \end{bmatrix} \quad \vdots \tag{6}$$

For each class label we calculate a mean vector

$$\mu_{y=1} = (E[X_1], E[X_2].....E[X_13]) \tag{7}$$

$$\mu_{y=0} = (E[X_1], E[X_2].....E[X_13]) \tag{8}$$

4.2 Calculate Covariance Matrix

Calculate the covariance matrix for X for both the labels. We will receive a 13x13 covariance matrix for each of the label.

$$sigma = (Cov(X_{y=1}), Cov(X_{y=0})) \tag{9}$$

Check if the Eigen values of the Covariance matrix for both the labels is positive or not. If its not positive then throw an error stating "Covariance matrix for is not definite positive for label".

Calculate the inverse Covariance matrix and determinant of the Covraince matrix for each of the label.

$$sigma\_inv = (inv(sigma_{y=1}), inv(sigma_{y=0})) \tag{10}$$

$$det = (det(sigma_{y=1}), det(sigma_{y=0})) \tag{11}$$

4.3 Calculate Likelihood

Now calculate the Scalar value of the pdf function $\frac{1}{\sqrt{(2\pi)^d det\Sigma}}$ for each label.

$$scalars = (\frac{1}{\sqrt{(2\pi)^d det\Sigma_{y=1}}}, \frac{1}{\sqrt{(2\pi)^d det\Sigma_{y=0}}}) \tag{12}$$

Now calculate the likelihood of f(x) for both both the labels.

$$likelihood = (scalar_{y=1} * e^{\frac{-1}{2}(x-\mu_{y=1})^T \Sigma_{y=1}^{-1}(x-\mu_{y=1})}, scalar_{y=0} * e^{\frac{-1}{2}(x-\mu_{y=0})^T \Sigma_{y=0}^{-1}(x-\mu_{y=0})}) \tag{13}$$

4.4 Predict the Label

Now to predict the target label for the input X. For this we consider the argmax of the likelihood for y=0 and y=1 and assign that label to the input.

$$Predict_{X[i]} = Argmax(Likelihood) \tag{14}$$

This is the way we predict the target label for the input X using the Maximumm likelihood function. After computing the target label for all the records using Maximum likelihood function we comapre the predicted result with original target label and calculate the accuracy score.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ records} \tag{15}$$

Through this Maximum likelihood function we acheieved an accuracy of 86.79

5 Conclusion

In conclusion, using Maximum Likelihood function we were able to predict the Heart Disease of a patient if he has a heart disease or not. We performed the prediction based on Maximum Likelihood function for multi variate data. After prediction of all the data points we have achieved and accuracy of 86.7% Though we were not able to product very high accuracy it is an interesting way to achieve reasonable results using Maximum Likelihood function for its simplicity.

6 Reference

Dr. Jing Zhang's lecture notes

7 R code

Listing 1: R Code

```
library(matlib)
library(corrplot)
data = read.csv("heart.csv")
summary(data)

data_cor=cor(data)
corrplot(data_cor, method = "color")

#Data Visualizations
```

```
ggplot(data,aes(x=  ..age,group=target,fill=target))+
  geom_histogram(position="dodge",binwidth=5)+theme_bw()
ggplot(data,aes(x=chol,group=target,fill=target))+
  geom_histogram(position="dodge",binwidth=15)+theme_bw()
ggplot(data,aes(x=thalach,group=target,fill=target))+
  geom_histogram(position="dodge",binwidth=5)+theme_bw()
ggplot(data,aes(x=oldpeak,group=target,fill=target))+
  geom_histogram(position="dodge",binwidth=0.5)+theme_bw()
ggplot(data,aes(x=sex,group=target,fill=target))+
  geom_histogram(position="dodge",binwidth=0.25)+theme_bw()
ggplot(data,aes(x=target,group=target,fill=target))+
  geom_histogram(position="dodge",binwidth=0.25)+theme_bw()

X=data[,1:13] #Feature Variables
y=data[,14]   #Label variable
d <- dim(X)[2]  #Number of columns
n <- dim(X)[1]  #Number of Rows
nclasses=length(unique(y))  #Number of classes in target Variable


##Calculate number of zeros and ones in the "target" variable
z=c()
o=c()
tot=c()
for(j in 2:n){
  if(y[j]==1){
    z=c(z,list(X[j,1:13]))
  }else if(y[j]==0){
    o=c(o,list(X[j,]))
  }
}


zlen=length(z)  ##Number of 1's in "target" variable
olen=length(o)  ##Number of 0's in "target" variable

#calculate mean vector for label 1 and 0
z_mean=c()
o_mean=c()
for(j in 1:d){
  z1=c()
  o1=c()
  for(i in 1:zlen){
    z1=c(z1,z[[i]][j])
  }
  for(i in 1:olen){
```

```
    o1=c(o1,o[[i]][j])
  }
  z_mean=rbind(z_mean,mean(sapply(z1, mean)))
  o_mean=rbind(o_mean,mean(sapply(o1, mean)))
}

tot_mean=list(z_mean,o_mean)

\textbf{#Subset of records for label 0 and 1}
cls_z=c()
cls_o=c()
for(i in 1:zlen){
  z2=c()
  for(j in 1:13){
    z2=c(z2,z[[i]][j])
  }
  cls_z=rbind(cls_z,as.numeric(z2))
}
for(i in 1:olen){
  o2=c()
  for(j in 1:13){
    o2=c(o2,o[[i]][j])
  }
  cls_o=rbind(cls_o,as.numeric(o2))
}
cls=list(cls_z,cls_o)

#Calculate covariance matrix for both the labels
sigma=list(cov(cls[[1]]),cov(cls[[2]]))

#Check if the eigen value is negative
for(e in 1:2){
  A=cov(cls[[e]])
  ev=eigen(A)

  if(sum(all(ev$values<=0))!=0){
    print("Covariance matrix for is not definite positive for label")
    print(e-1)
  }
}

##Calculate inverse covariance matrix for both labels
sigma_inv=list(inv(sigma[[1]]),inv(sigma[[2]]))

##Calculate the det of covariance matrix
dt=list(det(sigma[[1]]),det(sigma[[2]]))
```

```
##Compute 1/sqrt(2pi^d det(Sigma))
scalars=list(1/sqrt(dt[[1]]*((2*pi)^d)),1/sqrt(dt[[2]]*((2*pi)^d)))



#Predict the label
predict_func=function(nclasses,x,d){
  li=c()
  for(i in 1:nclasses){
    mu=tot_mean[[i]]
    sig_inv = sigma_inv[[i]]
    sc = scalars[[i]]
    m4=as.matrix(x-mu)
    m5=m4%*%sig_inv
    m6=m5%*%t(m4)
    exp1=m6*(-1/2)
    e_p=exp(exp1)*(-1/2)
    li=c(li,sc*e_p) # Calculate likelihood of x under the assumption that class
  }
  return(which.max(li)) #Calculate Argmax of likelihood
}


#Calculate the acuuracy of the prediction
pred_y=c()
x1=0
for(i in 1:n){
  pred_y[i]=predict_func(nclasses,X[i,],d)-1
  if(pred_y[i]==y[i]){
    x1=x1+1
  }
}

#Calculate Accuracy
accuracy=x1/n
print("Accuracy_is")
accuracy
```