JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY NOIDA

MINOR PROJECT (2019) REPORT

TOPIC : HEART DISEASE PREDICTION AND RECOMMENDATION SYSTEM

SUBMITTED TO:

- Dr. MEGHA RATHI

SUBMITTED BY:

- AYUSH RAJ (16103281) (B8)
- RIZUL SINGH (16104017) (B12)
- AMOGH SANJEEV GUPTA (16104018) (B12)
- VISHRUT SACHETI (16104058) (B12)

ACKNOWLEDGEMENT

We have taken efforts in this project, Heart Disease Predictor. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

We are highly indebted to our mentor, Dr. Megha Rathi for her guidance and constant supervision as well as for providing necessary information regarding the project and also for her support in completing the project.

We would like to express our gratitude towards our parents and friends for their kind co-operation and encouragement which help us in completion of this project.

We would like to express our special gratitude and thanks to college persons for giving us such attention and time. Our thanks and appreciations also go to our colleague in developing the project and people who have willingly helped us out with their abilities.

ABSTRACT

There is a rapid growth of Machine Learning in the Healthcare domain. Articles and research papers have proven Machine Learning results beneficial, accurate and cost efficient across several domains. The project undertaken was to apply three machine learning models on datasets obtained from the UCI machine learning repository. Three datasets chosen were Cleveland, Hungarian Institute Of Cardiology and Switzerland Heart Disease datasets, all related to predicting presence or absence of heart diseases in patients. The models applied on these datasets were logistic regression, decision trees, support vector machine, and neural networks. The results were obtained by varying parameters for different models. The goal of the project was to check the model consistency on datasets and find the best suited algorithm for that particular dataset. Support Vector Machine worked best for our dataset when compared with logistic regression and decision trees.

OBJECTIVES

The main objectives of the project are as follows:

- Combine datasets of different regions into a single dataset based on area codes.
- Pre-processing of data to counter null and out of bound values.
- Training of data using different machine learning models.
- Choosing the best model based on accuracy and implementing the model using Shiny web based interface.
- Recommendation system to suggest lifestyle based on user inputs.

DATASET DESCRIPTION

There are 3 available databases concerning heart disease diagnosis. All attributes are numeric valued. The data was collected from the 3 following locations:

- Cleveland Clinic Foundation
- Hungarian Institute Of Cardiology
- University Hospital, Switzerland

A brief description of the dataset used: The dataset has 14 attributes (categorical, Integer and Real) and 720 instances. The 'num' attribute is the response variable, used for the prediction.

(Link: https://archive.ics.uci.edu/ml/datasets/heart+Disease)

CONTRIBUTION

The contributions are as follows :

Rizul Singh and Vishrut Sacheti worked on background research, research on data selection and implementation and project report. Amogh S. Gupta and Rizul Singh worked on tools, libraries and packages, and implementation of algorithms. Amogh S. Gupta worked on debugging of codes. Ayush Raj worked on research paper and project report. Vishrut Sacheti and Ayush Raj contributed to build recommendation system. All the members had almost the same contribution in the project.

DESCRIPTIVE FEATURES

Field	Description	Range and Values
Age	Age of the patient	0-100 in years
Sex	Gender of the patient	0-1 (1:Male 0:Female)
Chest Pain	Type of chest pain	1-4 (1: Typical Angina, 2: Atypical Angina, 3: Non-angina, 4: Asymptotic)
Resting Blood Pressure	Blood pressure during rest	mm Hg
Cholesterol	Serum Cholesterol	mg / dl
Fasting Blood Sugar	Blood sugar content before food intake if >120 mg/dl	0-1 (0: False, 1: True)
ECG	Resting Electrocardiographic results	0-1 (0: Normal, 1: Having ST-T wave)
Max Heart Rate	Maximum heart beat rate.	Beats/min
Exercise Induced Angina	Has pain been induced by exercise	0-1 (0: No, 1: Yes)
Old Peak	ST depression induced by exercise relative to rest	0-4
Slope of Peak Exercise	Slope of the peak exercise ST segment	1-3 (1: Up sloping, 2: Flat, 3: Down sloping)
Ca	Number of vessels colored by fluoroscopy	0-3
Thal		3- normal 6-Fixed Defect 7- Reversible Defect
Num	Diagnostics of Heart Disease	0-1 (0: <50% Narrowing 1: >50% Narrowing)

METHODOLOGY

We considered four classifiers –Decision tree, Logistic regression, Neural networks, Support vector machine. Each classifier was trained to make probability predictions so that we were able to adjust the prediction threshold to refine the performance. We split the full data set into a 70 % training set and 30 % test set. Each set resembled the full data by having the same proportion of target classes.

PREPROCESSING

As there were null values in the dataset we needed to preprocess the data. Null values for attributes which had continuous values were replaced by their means and for those attributes which had discretized range of values were replaced by the modal value of the attribute.

ALGORITHMS

1. DECISION TREE

A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes). In decision analysis, a decision tree and the closely related influence diagram are used as a visual and analytical decision support tool, where the expected values (or expected utility) of competing alternatives are calculated.

A decision tree consists of three types of nodes:

- Decision nodes –represented by squares
- Chance nodes –represented by circles
- End nodes –represented by triangles

2. LOGISTIC REGRESSION

Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. The logistic function, also called the sigmoid function was developed by statisticians to describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment. It's an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

1 / (1 + e^-value)

Where e is the base of the natural logarithms and value is the actual numerical value that you want to transform.

3. NEURAL NETWORK

A neural network is a network or circuit of neurons, or in a modern sense, an artificial neural network, composed of artificial neurons or nodes. Thus a neural network is either a biological neural network, made up of real biological neurons, or an artificial neural network, for solving artificial intelligence (AI) problems. All inputs are modified by a weight and summed. This activity is referred as a linear combination. Finally, an activation function controls the amplitude of the output. For example, an acceptable range of output is usually between 0 and 1, or it could be −1 and 1.

4. SUPPORT VECTOR MACHINE

Support vector machine (SVM) are supervised learning method that analyse data used for classification and regression analysis. It is given a set of training data, marked as belonging to either one of two categories; an SVM training algorithm then builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

RECOMMENDATION SYSTEM

The dataset for recommending lifestyle consists of edibles items and exercises with their respective id. We implemented item-based collaborative filtering based on cosine similarity. Then the similarity values are used to recommend the items for user-item pairs not present in the dataset.

CODE

SVM

```
#install.packages("dplyr")
library(dplyr)

svm1 <- function(region){

 Mode <- function(x){
  x <- na.omit(x)
  ux <- unique(x)
  ux[which.max(tabulate(match(x,ux)))]
 }

 dataset <- read.csv(file="Dataset_Heart.csv", header = T)

 # Preprocessing
 a <- mean(dataset$trestbps,na.rm = TRUE)

 for(i in 1:NROW(dataset))
 {
  if(is.na(dataset$trestbps[i])==TRUE)
  {
   dataset$trestbps[i] <- a
  }
 }
 a <- mean(dataset$chol,na.rm=TRUE)

 for(i in 1:NROW(dataset))
 {
  if(is.na(dataset$chol[i])==TRUE)
  {
   dataset$chol[i] <- a
  }
 }
 a <- Mode(dataset$fbs)

 for(i in 1:NROW(dataset))
```

```
{
  if(is.na(dataset$fbs[i])==TRUE)
  {
    dataset$fbs[i] <- a
  }
}
a <- Mode(dataset$restecg)

for(i in 1:NROW(dataset))
{
  if(is.na(dataset$restecg[i])==TRUE)
  {
    dataset$restecg[i] <- a
  }
}
a <- Mode(dataset$thalach)

for(i in 1:NROW(dataset))
{
  if(is.na(dataset$thalach[i])==TRUE)
  {
    dataset$thalach[i] <- a
  }
}
a <- Mode(dataset$exang)

for(i in 1:NROW(dataset))
{
  if(is.na(dataset$exang[i])==TRUE)
  {
    dataset$exang[i] <- a
  }
}
a <- mean(dataset$oldpeak,na.rm=TRUE)

for(i in 1:NROW(dataset))
{
  if(is.na(dataset$oldpeak[i])==TRUE)
  {
    dataset$oldpeak[i] <- a
  }
}
a <- Mode(dataset$slope)

for(i in 1:NROW(dataset))
{
  if(is.na(dataset$slope[i])==TRUE)
  {
    dataset$slope[i] <- a
  }
}
a <- Mode(dataset$ca)

for(i in 1:NROW(dataset))
{
```

```
  if(is.na(dataset$ca[i])==TRUE)
  {
    dataset$ca[i] <- a
  }
}
a <- Mode(dataset$thal)

for(i in 1:NROW(dataset))
{
  if(is.na(dataset$thal[i])==TRUE)
  {
    dataset$thal[i] <- a
  }
}
# Preprocessing ends


dataset <- dataset[dataset[, "region"] == region,]
dataset<-dataset[,-ncol(dataset)]
#print(dataset)
library(caret)
dataset$num<-as.factor(dataset$num)
levels(dataset$num) <- c("NotDisease","Disease")

# Cross validation
fitControl <- trainControl(method = "repeatedcv",
                           number = 10,
                           repeats = 10,
                           # Estimate class probabilities
                           classProbs = TRUE,
                           summaryFunction = twoClassSummary)
svmModel <- train(num ~ ., data = dataset,
                  method = "svmRadial",
                  trControl = fitControl,
                  tuneLength = 8,
                  metric = "ROC")


newtest <- read.csv(file.choose(), header = T)
newtest=newtest[,-1]
names(newtest) <- c("age", "sex","cp","trestbps","chol","fbs",
                    "restecg","thalach","exang","oldpeak","slope","ca","thal")
newtest<-tail(newtest,1)
#print(newtest)

svmPrediction <<- predict(svmModel,newtest)
svmPredictionprob <- predict(svmModel,newtest, type='prob')[2]

result=svmPredictionprob[1, "Disease"]*100
result=round(result,digits = 2)
print(paste0(result,"%"))

}
```

```
#a,d,t represents asthama, diabetes and thyroid values respectively which are passed from shiny ui
recommend <- function(a,d,t){

 options(digits=4)
 data <- read.csv(file="newlifestyle.csv", header = T)
 datanew <- select (data,-c(id,items))

 h=ifelse(svmPrediction=="NotDisease",0,1)
 i=ifelse(a==0,0,1)
 j=ifelse(d==0,0,1)
 k=ifelse(t==0,0,1)
 tes<-c(h,i,j,k)

 Cosinefun <- function(x,y)
 {
  this.cosine <- sum(x*y) / (sqrt(sum(x*x)) * sqrt(sum(y*y)))
  return(this.cosine)
 }

 similarity<-matrix(NA, nrow = 84, ncol = 2)

 for(i in 1:nrow(datanew)) {
  similarity[i,]= Cosinefun(tes,datanew[i,])
 }

 similarity[,1]<-c(1:84)
 similarity<-similarity[order(similarity[,2],decreasing = TRUE),]
 index<-t(c(similarity[1:8,1]))
 for(k in 1:ncol(index))
 {
  for (l in 1:nrow(data))
  {
   if(index[1,k]==data$id[l])
   {
    print(data$items[l], max.levels = 0)
   }
  }
 }

}
```

UI

```
library(shiny)
library(shinydashboard)
d<-getwd()
setwd(d)
source("Svm.R")
options(warn = -1)

shinyUI(dashboardPage(
```

```
 dashboardHeader(title = "Predictions"),

 dashboardSidebar(
  sidebarMenu(
   menuItem("Dashboard",
          tabName = "dashboard",
          icon = icon("dashboard")))),

 dashboardBody(
  tabItems(

   tabItem( tabName = "dashboard",
          h1("Enter The Details for prediction "),
          fluidRow(
           column(width = 6,
                box( title = h4("Enter Personal Details"),
                    width = 12 ,solidHeader = T,
                    status = "primary",

                    numericInput("age","Enter Your Age",35),

                    selectInput("gender", label="Enter Your Gender", selectize = TRUE, choices =
c("0", "1"), selected = "1"),helpText("1 = male, 0 = female"),

                    selectInput("region", label="Enter Your Region Code", selectize = TRUE, choices
= c("101", "102","103"), selected = "101"),helpText("101 = Cleveland, 102 = Hungarian, 103 =
Switzerland"),

                    selectInput("cp",label = "Chest Pain Type:",selectize = TRUE, choices =
c("1","2","3","4"), selected = "2"), helpText("1 = Typical Angina, 2 = Atypical Angina, 3 = Non
Anginal, 4 = Asymptotic"),

                    textInput("restingbp", label =("Resting Blood Pressure"), value =
"111"),helpText("Range=100-189 in mm/Hg"),

                    textInput("cholestrol", label =("Cholestrol"), value = "222"), helpText("Range=130-
410 mg/dL"),

                    textInput("fastingbp", label =("Fasting Blood Sugar"), value = "1"), helpText("
(120>)-1 = true, (120<)-0 = false"),

                    textInput("restcg", label =("Resting ECG result"), value = "1"), helpText("0 =
normal, 1 = having ST-T wave abnormality, 2 = showing ventricular hypertrophy "),

                    textInput("maxheartrate", label =("Max Heart Rate Achieved"),  value = "170"),

                    textInput("exang", label =("Exercise induced angina"),  value = "1"), helpText("0 =
no, 1 = yes"),

                    textInput("oldpeak", label =("ST depression induced due to exercise"),  value =
"1"),

                    textInput("slope", label =("Slope of peak ST segment"),  value = "1"), helpText("1
= upsloping, 2 = flat, 3 = downsloping"),
```

```
                    textInput("ca", label =("Number of major vessels coloured by fluroscopy"), value =
"1"), helpText("0 - 3"),

                    textInput("thal", label =("Thal value"),value = "3"), helpText("3 = normal, 6 = fixed
defect, 7 = reversible defect"),

                    selectInput("asthama",label = "Asthama:",selectize = TRUE, choices = c("0","1"),
selected = "0"), helpText("0 = Not present, 1 = Present"),

                    selectInput("diabetes",label = "Diabetes:",selectize = TRUE, choices = c("0","1"),
selected = "0"), helpText("0 = Not present, 1 = Present")
                    selectInput("thyroid",label = "Thyroid:",selectize = TRUE, choices = c("0","1"),
selected = "0"), helpText("0 = Not present, 1 = Present")
                  )
            ),

            #tableOutput("table"),
            actionButton("Action", "Submit!"),
            #Button to save the file
            downloadButton('Data', 'Download!'),
            #Button to predict
            actionButton("Pred","Predict!"),
            textOutput("Prediction"),
            #Button to recommend lifestyle
            actionButton("Recommend","Recommendations!"),
            textOutput("Result")


          ))
   ))
))
```

SERVER

```
library(shiny)
library(shinydashboard)
d<-getwd()
setwd(d)
source("Svm.R")
options(warn = -1)

shinyServer<-function(input, output){
  Data <- data.frame()
  Results <- reactive(data.frame(input$age, input$gender, input$cp, input$restingbp, input$cholestrol,
                          input$fastingbp, input$restcg, input$maxheartrate, input$exang,
                          input$oldpeak, input$slope,input$ca, input$thal))

  observeEvent(input$Action,{
    Data <<- rbind(Data,Results())
    output$table <- renderTable(Data)
```

```
  })

  output$Data <- downloadHandler(

    filename = function() {
      paste("data-", Sys.Date(), ".csv", sep="")
    },
    content = function(file) {
      write.csv(Data, file)})

  observeEvent(input$Pred,{output$Prediction <- renderText(svm1(input$region)
observeEvent(input$Recommend,{output$Result <- renderPrint(recommend(input$asthama,
input$diabetes, input$thyroid))})
}
```

OUTPUT

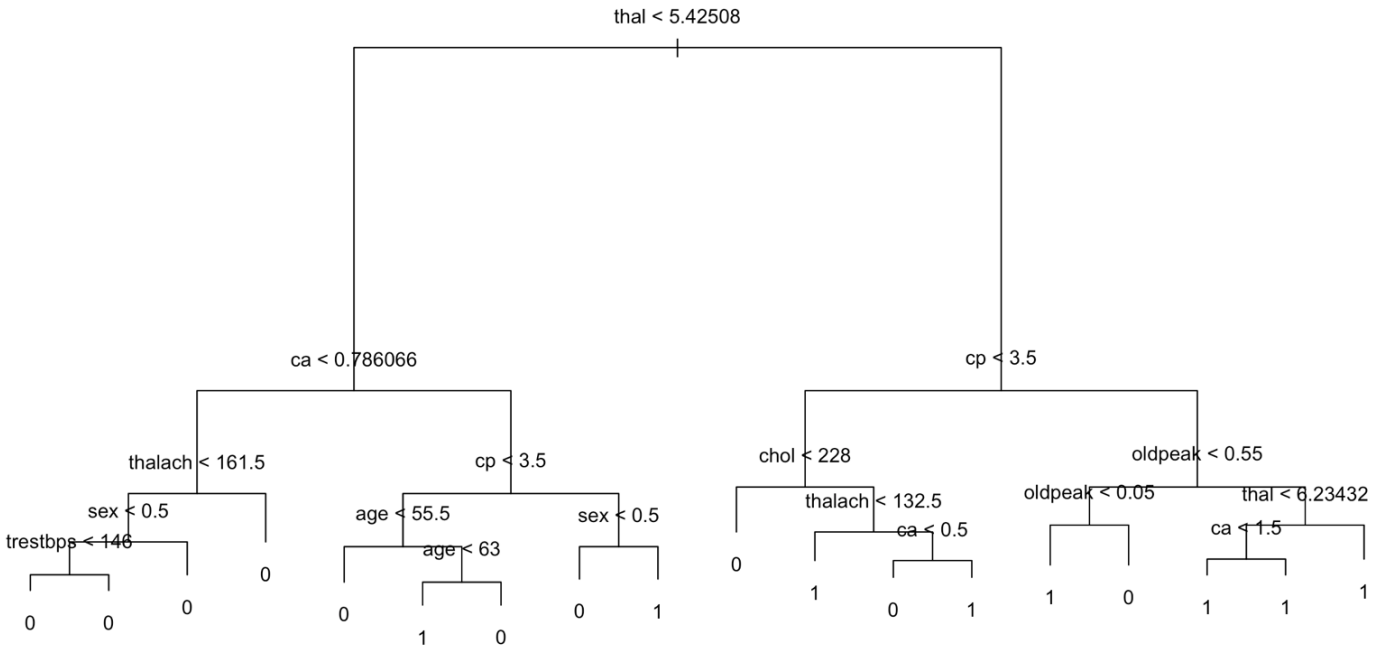

Fig 1: Decision Tree (Without Pruning)

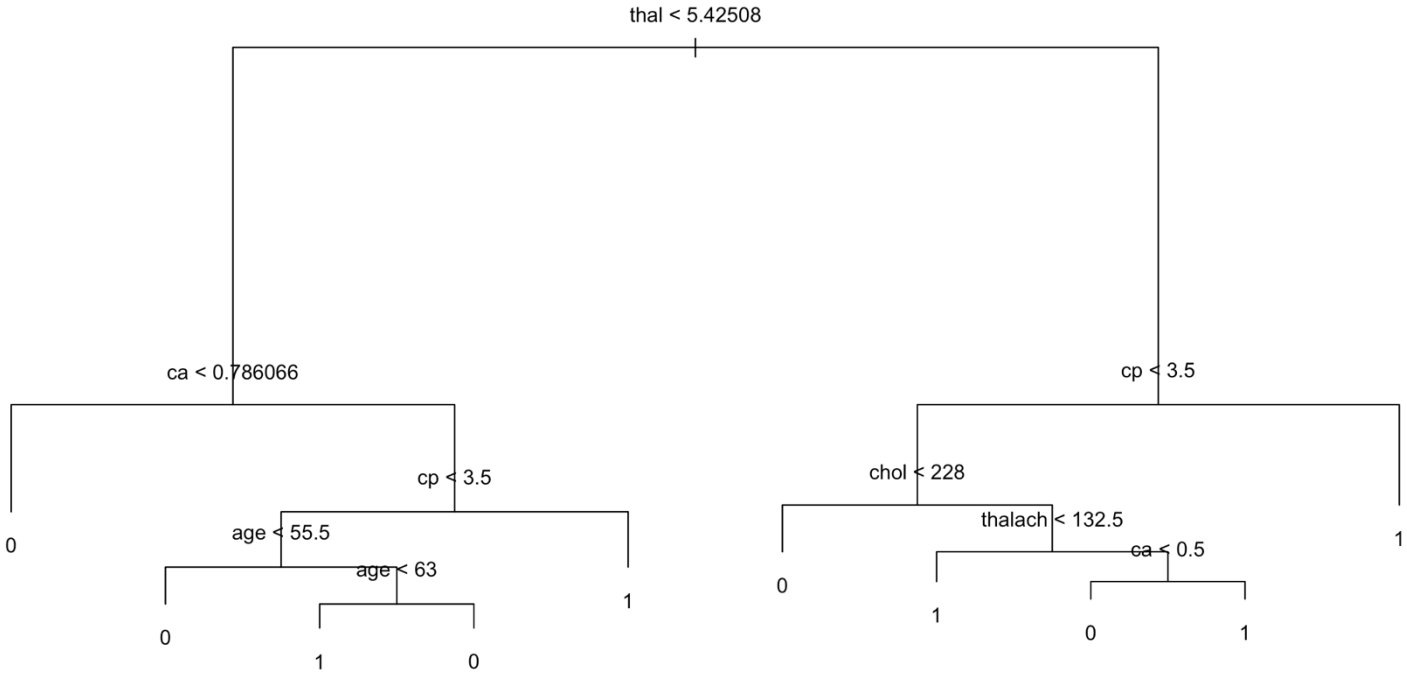

Fig 2: Decision Tree (After Pruned)

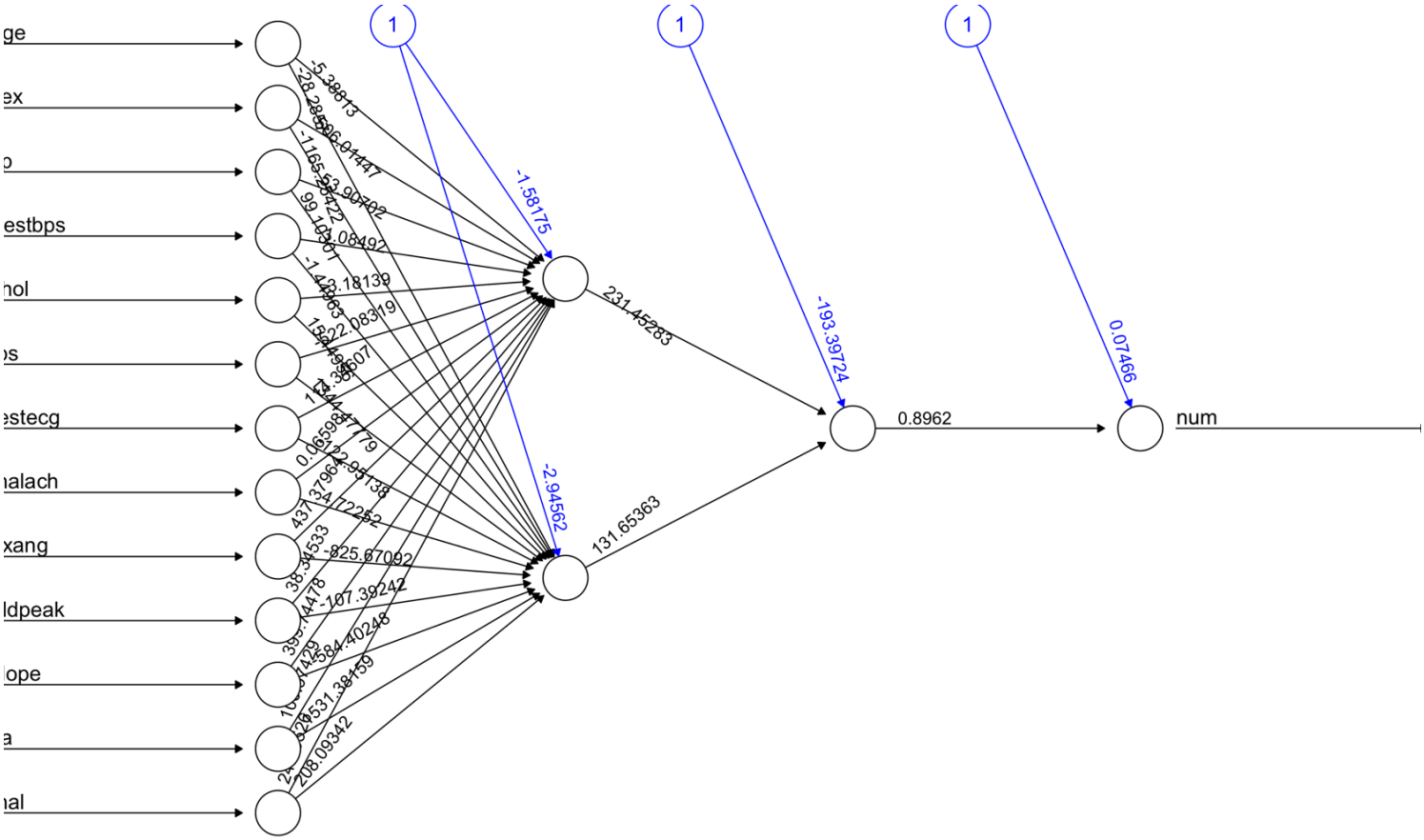

Fig 3: Neural Network

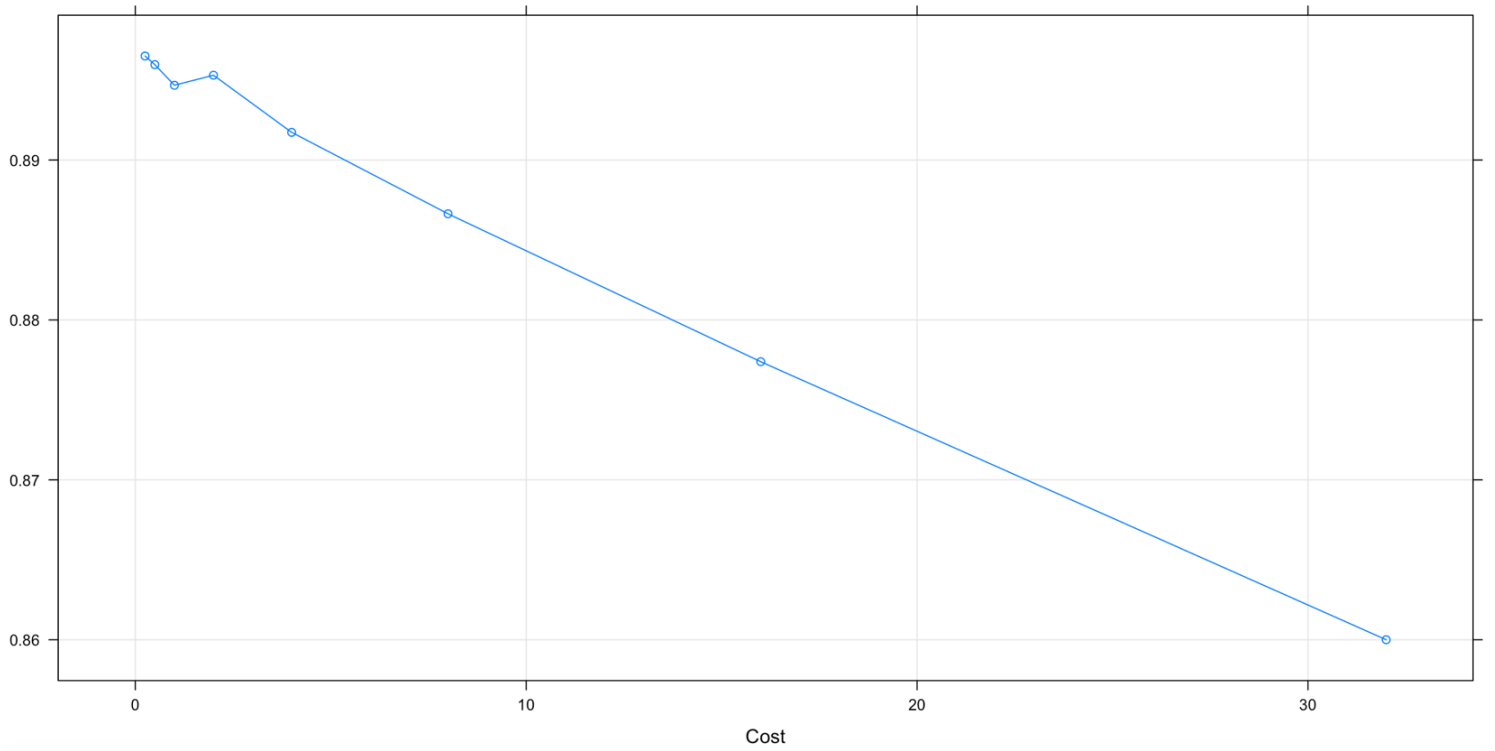

Fig 4: SVM (ROC vs Cost curve)

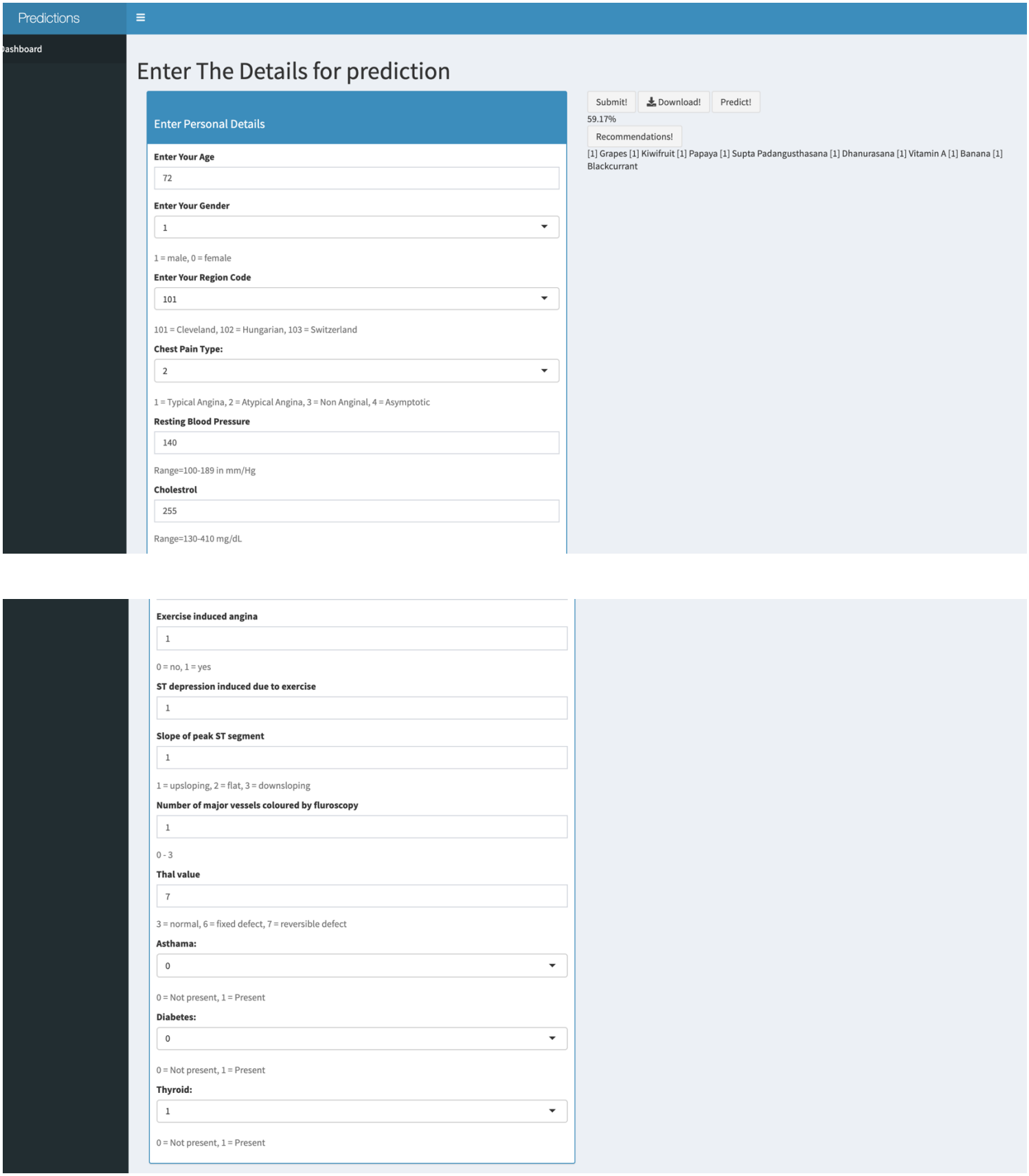

Fig 5: Shiny Interface

RESULT

Accuracy was compared for the four algorithms and the result were observed as follows :

- Decision Tree : 82.02% (Without Prunung)
  83.14% (With Pruning)
- Logistic Regression : 84.52%
  90.47% (With Significant Features)
- Neural Network : 79.78%
- SVM : 87.64%

Hence, we conclude that SVM model is best suited for the dataset and therefore we implemented a probabilistic model in Shiny web application and built a recommendation system for user lifestyle based on their personalized details and other diseases information.

REFERENCES

[1] https://rpubs.com/cgalea/396365

[2] http://csjournals.com/IJCSC/PDF7-1/18.%20Tejpal.pdf

[3] https://archive.ics.uci.edu/ml/datasets/heart+Disease

[4] https://www.who.int/en/news-room/fact-sheets/detail/the-top-10-causes-of-death