

Project 1

William Davis

6/11/2022

Background

- In America, 1 in every 4 males suffer from heart disease related deaths each year.
- This attributed to 357,761 deaths in 2019. I chose the Heart Attack Analysis and Prediction Dataset to help scientist and medical professional prevent future events of heart attack.
- This dataset will predict the likelyhood of a paitient going into cardiac arrest.

```
heart_data = read.csv("heart.csv", sep = ",")  
heart_data$cp <- as.factor(heart_data$cp)  
heart_data$sex <- as.factor(heart_data$sex)  
heart_data$age <- as.factor(heart_data$age)  
heart_data$restecg <- as.factor(heart_data$restecg)  
heart_data$exng <- as.factor(heart_data$exng)  
heart_data$fbs <- as.factor(heart_data$fbs)  
heart_data$output <- as.factor(heart_data$output)  
heart_data$thall <- as.factor(heart_data$thall)
```

The heart attack analysis and prediction dataset, used many fields that needed to cleaned and converted to factors. In this dataset the following values were converted into factors.

- age
- sex
- cp (chest pain)
- restecg (resting electrocardiographic results)
- exng (exercise induced angina)
- fbs (fasting blood sugar > 120 mg/dl)
- thall
- output (likely-hood of having a heart attack)

Feature Engineering

To validate the data values for thalach (maximum heart rate acheived) and trtbps (resting blood pressure), The values have been normalized and used in the predictive model.

```
meanMaxHeartRateAtt = mean(heart_data$thalachh)
s = sd(heart_data$thalachh)
heart_data$MaxHeartNormal = (heart_data$thalachh - meanMaxHeartRateAtt)/s

meanRestBlPress = mean(heart_data$trtbps)
s = sd(heart_data$trtbps)
heart_data$RestBloodPressNormal = (heart_data$trtbps - meanRestBlPress)/s

diabetes = heart_data$fbs

diabetesChance <- character()

for(i in 1:length(diabetes)){
  if(diabetes[i] == 1){
    diabetesChance[i] <- 1L
  }else if(diabetes[i] == 0){
    diabetesChance[i] <- 0L
  }
}

heart_data$twiceAsLikely = diabetesChance
```

According to the CDC, men with diabetes are twice as likely to enter cardiac arrest. With this in mind, a patient with a fbs above 120 is likely to have diabetes. The feature “twiceAsLikely” has been created and added to the dataset to contribute to the predictive model with 1 = twice as likely, and 2 = not twice as likely.

Data Sampling

```
heart_dataSelect <- select(heart_data, age, sex, caa, cp, chol, trtbps, fbs, thall, thalachh, exng, output, twiceAsLikely)

samples <- createDataPartition(heart_data$output,
                             times = 1,
                             p = 0.80,
                             list = F)

heart_dataSelectTrain <- heart_dataSelect[samples,]
heart_dataSelectTest <- heart_dataSelect[-samples,]

control <- trainControl(method = "repeatedcv",
                         number = 10,
                         repeats = 2,
                         search = "grid")

cl <- makeCluster(3, type = "SOCK")
registerDoSNOW(cl)
set.seed(1234)
caret.cv <- train(output ~.,
                    data = heart_dataSelectTrain,
                    method = "multinom",
                    preProcess = ("center"),
                    trControl = control)

## # weights: 58 (57 variable)
## initial value 168.434765
## iter 10 value 81.322395
## iter 20 value 74.522022
## final value 74.521050
## converged
stopCluster(cl)

pred <- predict(caret.cv, heart_dataSelectTest)

confusionMatrix(pred, heart_dataSelectTest$output)

## Confusion Matrix and Statistics
##
##             Reference
## Prediction   0   1
##           0 20  6
##           1  7 27
```

```

##          Accuracy : 0.7833
##  95% CI : (0.658, 0.8793)
##  No Information Rate : 0.55
##  P-Value [Acc > NIR] : 0.0001472
##
##          Kappa : 0.5608
##
##  Mcnemar's Test P-Value : 1.0000000
##
##          Sensitivity : 0.7407
##          Specificity : 0.8182
##  Pos Pred Value : 0.7692
##  Neg Pred Value : 0.7941
##          Prevalence : 0.4500
##          Detection Rate : 0.3333
##  Detection Prevalence : 0.4333
##  Balanced Accuracy : 0.7795
##
##  'Positive' Class : 0
##
View(heart_data)

```

Embedded Application

It's also possible to embed an entire Shiny application within an R Markdown document using the `shinyAppDir` function. This example embeds a Shiny application located in another directory:

Note the use of the `height` parameter to determine how much vertical space the embedded application should occupy.

You can also use the `shinyApp` function to define an application inline rather than in an external directory.

In all of R code chunks above the `echo = FALSE` attribute is used. This is to prevent the R code within the chunk from rendering in the document alongside the Shiny components.