Predicting the Chance of 10-Year Coronary Heart Disease

David Oniani, Madeline Pope, Zachary Sturgeon

December 11, 2019

Abstract

We will analyze the dataset (posted on Kaggle) describing Heart Disease Factors. It consists of 15 predictor variables. The dataset is diverse, featuring both numerical and categorical variables. Our goal is to build a model that accurately predicts the likelihood of a patient having a high risk of a coronary disease in the next 10 years. We are also interested in determining the most influential factors contributing to the increased risks of the disease.

Contents

Data and Preparation

The dataset was obtained from https://www.kaggle.com and features 15 predictor variables. The description of the variables are provided below.

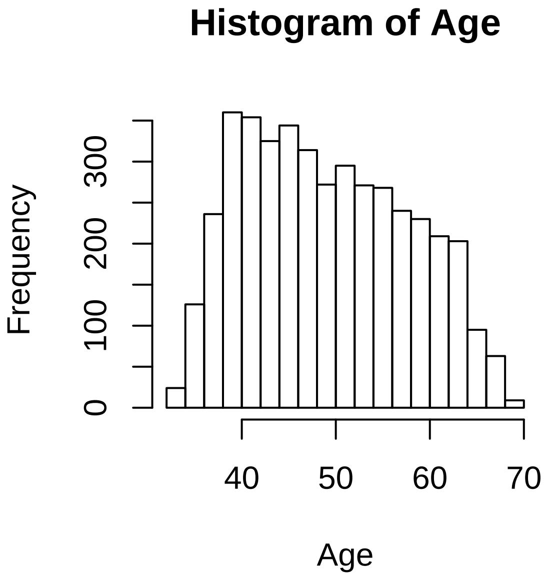
- **Male** - Categorical, this variable describes binarily as to whether the patient is a male or female (Male = 1, Female = 0).
- **Age** - Numerical, this variable describes the patient's age numerically.
- **Education** - Categorical, describes the level of education patient has received(1 = Some High School, 2 = High school/GED, 3 = Some College/Vocational School, 4 =College, NA = Not available)
- **CurrentSmoker** - Categorical, this variable describes binarily as to whether the patient smokes or not (Smoker = 1, Non-smoker = 0)
- **cigsPerDay** - Numerical, this variable describes numerically how many cigarettes the patient smokes per day.
- **BPMeds** - Categorical, whether or not the patient was on blood pressure medication. (Doesn't take medication = 0, takes medication = 1)
- **PrevalentStroke** - Categorical, whether or not the patient had previously had a stroke. (Hasn't had a stroke = 0, Has had a stroke = 1)
- **PrevalentHyp** - Categorical, whether or not the patient was hypertensive. (Patient isn't hypertensive = 0, patient is hypertensive = 1)
- **Diabetes** - Categorical, whether or not the patient has diabetes (Patient doesn't have diabetes = 0, patient does have diabetes = 1)
- **totChol** - Numerical, total cholesterol level.
- **sysBP** - Numerical, systolic blood pressure
- **diaBP** - Numerical, Diastolic blood pressure
- **BMI** - Numerical, patient's body mass index
- **HeartRate** - Numerical, patient's heart rate
- **Glucose** - Numerical, the glucose level in the patient.

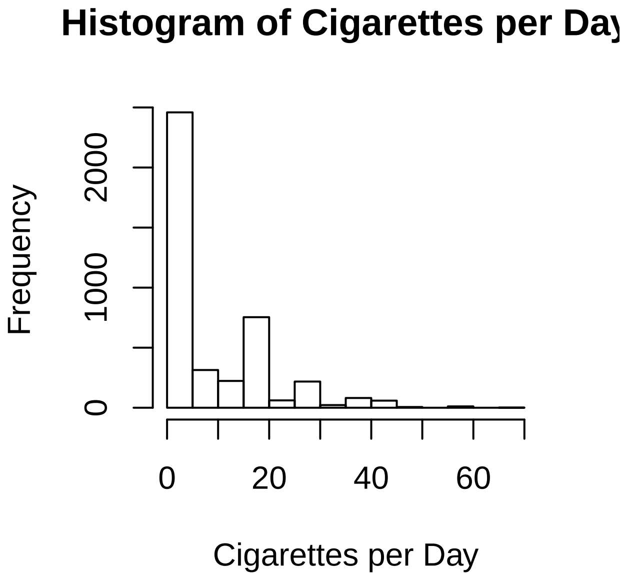
```r
# Read the CSV data
dataset <- read.csv("./framingham.csv", header=TRUE, sep=",")

# Get rid of rows with NA values
cleanData = na.omit(dataset)
```

Analyzing Distributions

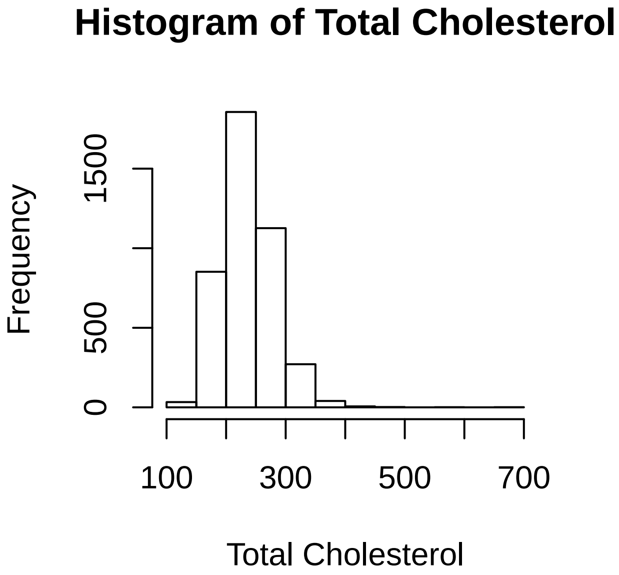
Analyzing Numerical Variables

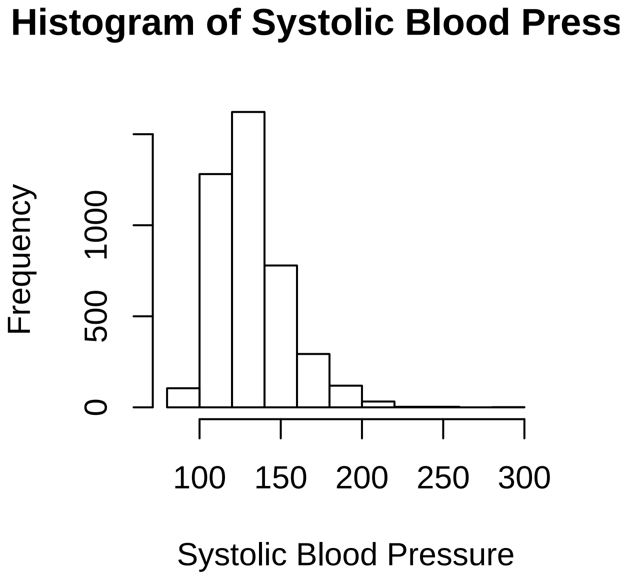
```r
# Four plots side-by-side
par(mfrow=c(1,2))

# Histograms of all numerical variables
hist(dataset$age,
     xlab="Age",
     main="Histogram of Age")

hist(dataset$cigsPerDay,
     xlab="Cigarettes per Day",
     main="Histogram of Cigarettes per Day")
```





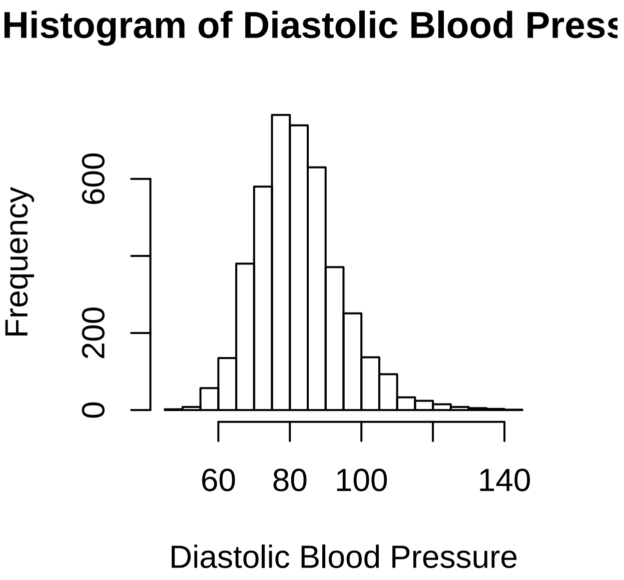
```
hist(dataset$totChol,
     xlab="Total Cholesterol",
     main="Histogram of Total Cholesterol")

hist(dataset$sysBP,
     xlab="Systolic Blood Pressure",
     main="Histogram of Systolic Blood Pressure")
```

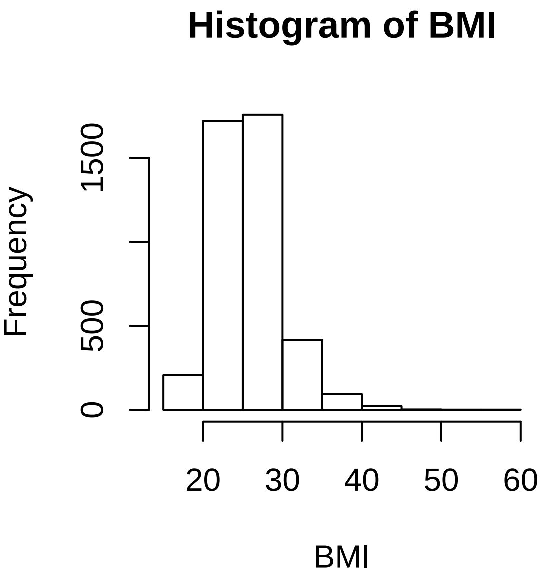




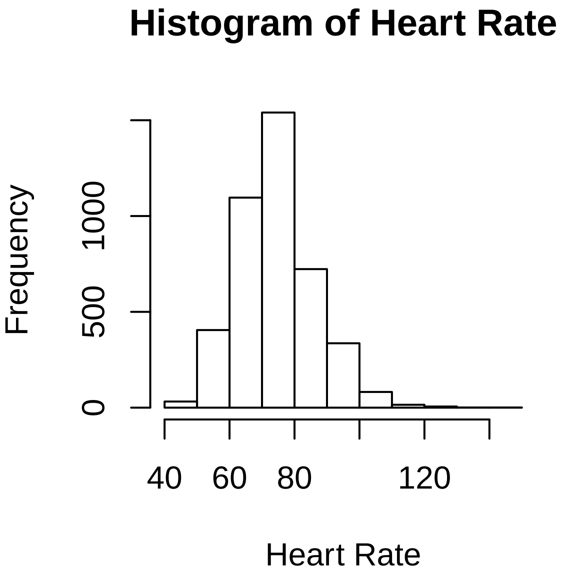
```
hist(dataset$diaBP,
     xlab="Diastolic Blood Pressure",
     main="Histogram of Diastolic Blood Pressure")
```

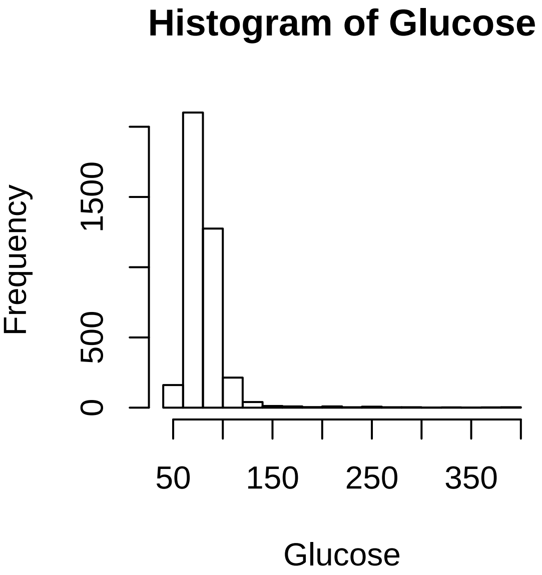
```r
hist(dataset$BMI,
     xlab="BMI",
     main="Histogram of BMI")
```





```r
hist(dataset$heartRate,
     xlab="Heart Rate",
     main="Histogram of Heart Rate")

hist(dataset$glucose,
     xlab="Glucose",
     main="Histogram of Glucose")
```

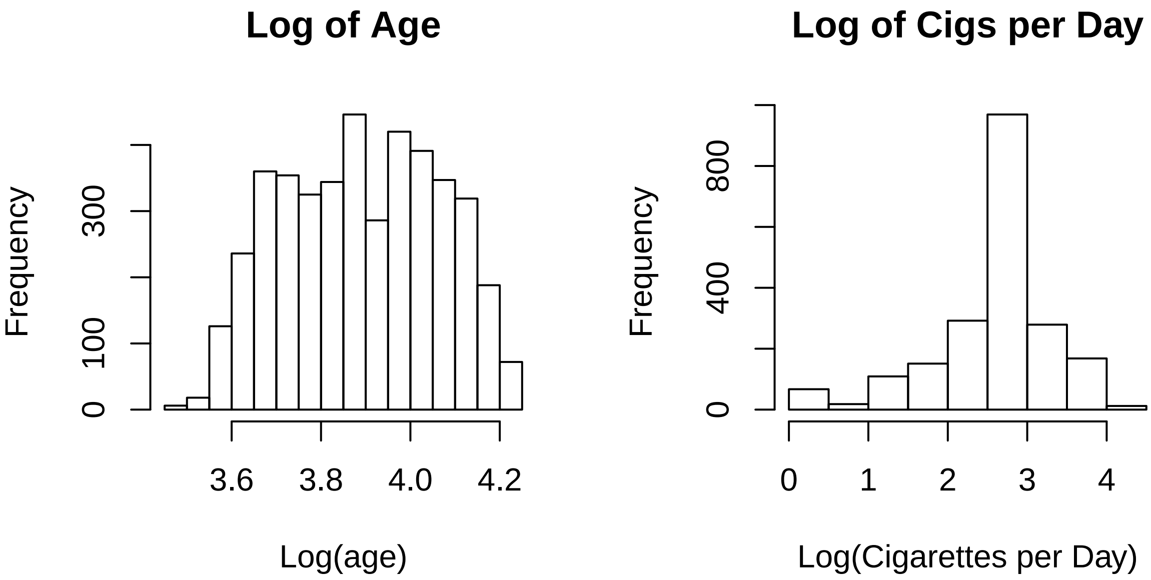




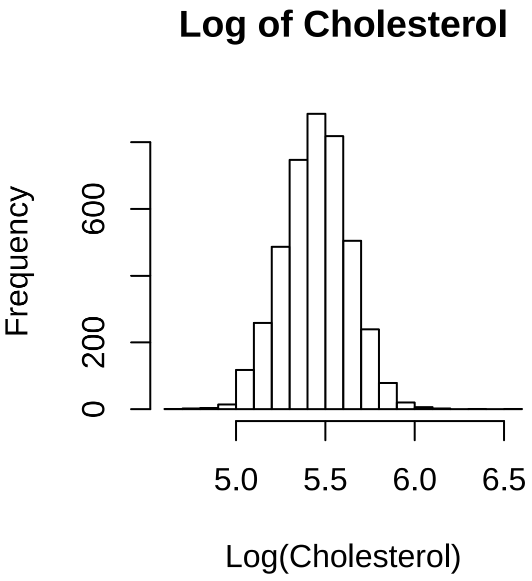
For numerical variables (age, cigarettes per day, total cholesterol, systolic blood pressure, diastolic blood pressure, BMI, heart rate, and glucose), we used a histogram to look at the distributions. Distributions of the variables diaBP and heartRate are approximately normal. Apart from these two, all other variables are skewed and need a transformation. Therefore, we proceed performing log-transformations.

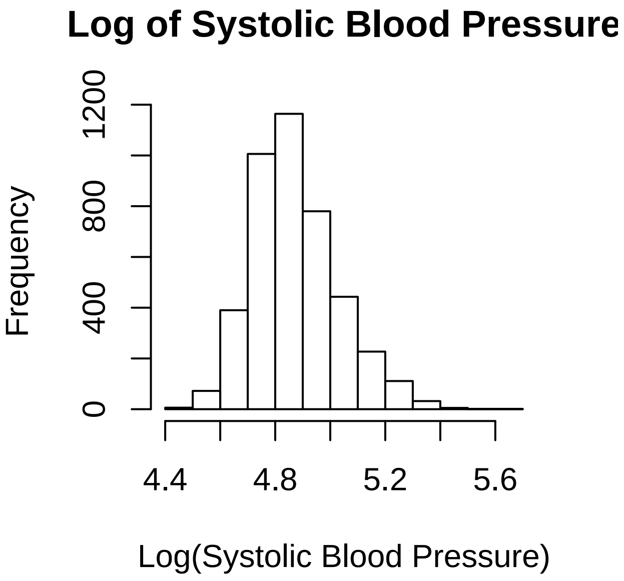
Log transformations

```r
# Plots side-by-side
par(mfrow=c(1,2))

# Histograms of all log-transformed numerical variables
hist(log(dataset$age),
     xlab="Log(age)",
     main="Log of Age")

hist(log(dataset$cigsPerDay),
     xlab="Log(Cigarettes per Day)",
     main = "Log of Cigs per Day")
```

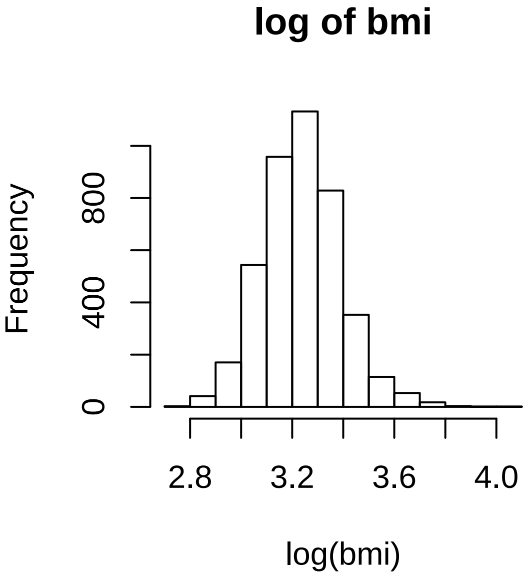


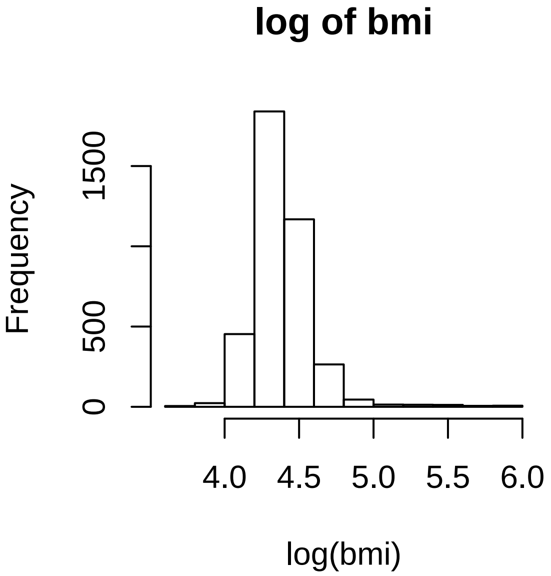
```r
hist(log(dataset$totChol),
     xlab="Log(Cholesterol)",
     main= "Log of Cholesterol")

hist(log(dataset$sysBP),
     xlab="Log(Systolic Blood Pressure)",
     main="Log of Systolic Blood Pressure")
```





```r
hist(log(dataset$BMI),
     xlab="log(bmi)",
     main="log of bmi")

hist(log(dataset$glucose),
     xlab="log(bmi)",
     main="log of bmi")
```



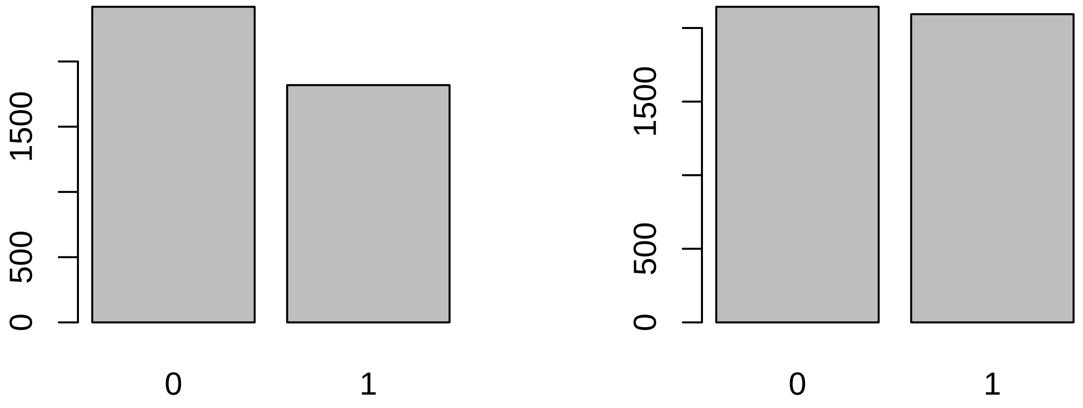


After comparing the initial distributions against their log-transformed alternatives, we have decided to keep the log-transformations of the following variables: age, cigsPerDay, totChol, diaBP, sysBP, BMI, glucose. Log-transforming variable glucose did not completely normalize the distribution. The approach we took is described in the section “Additional Transformations."

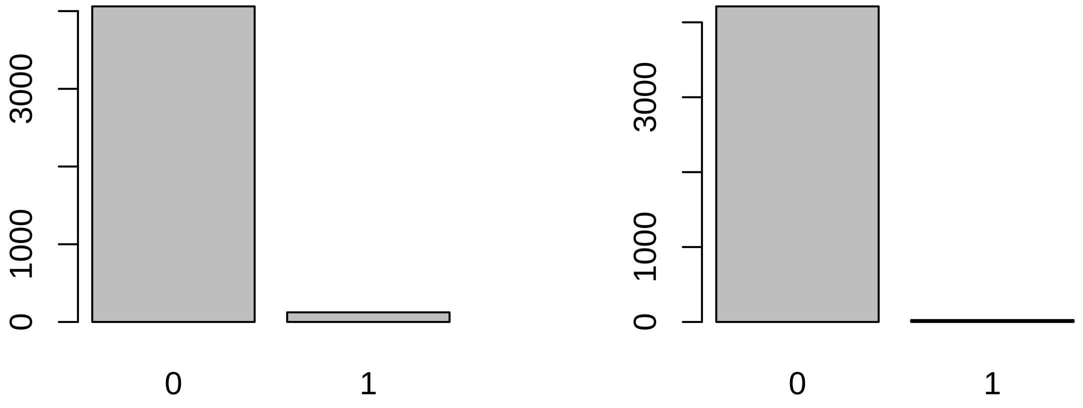
Analyzing Categorical Variables

For exploratory purposes, we have included the distributions of the categorical variables as well.

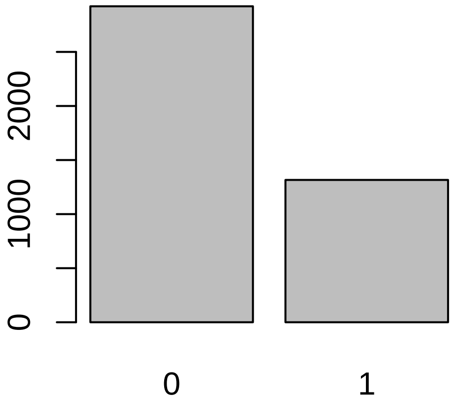
```r
# Two plots side-by-side
par(mfrow=c(1,2))

# Tables of all categorical variables
tableMale = table(dataset$male)
tableSmoker = table(dataset$currentSmoker)
tableBPMeds = table(dataset$BPMeds)
tableStroke = table(dataset$prevalentStroke)
tableHypertension = table(dataset$prevalentHyp)
tableDiabetes = table(dataset$diabetes)
tableEducation = table(dataset$education)
table10Year = table(dataset$TenYearCHD)

# Histograms of all categorical variables
barplot(tableMale)
barplot(tableSmoker)
```

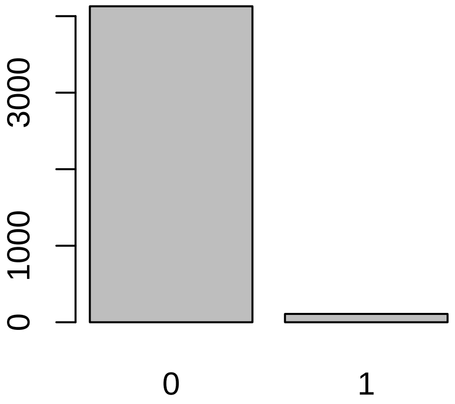


```r
barplot(tableBPMeds)
barplot(tableStroke)
```

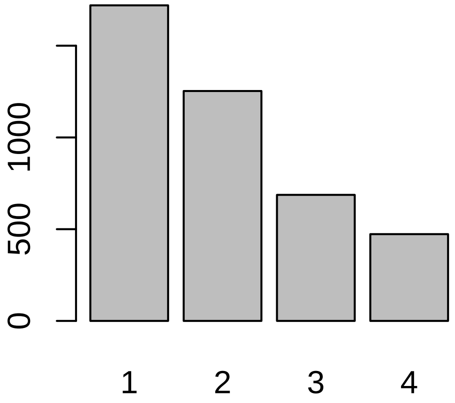


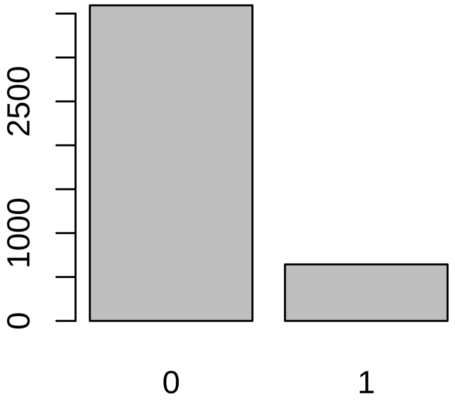
```r
barplot(tableHypertension)
barplot(tableDiabetes)
```





```
barplot(tableEducation)
barplot(table10Year)
```





```
# Percentages
tableMale["0"] / sum(tableMale) * 100
```

```
##        0
## 57.07881
```

```
tableMale["1"] / sum(tableMale) * 100
```

```
##        1
## 42.92119
```

```
tableSmoker["0"] / sum(tableSmoker) * 100
```

```
##       0
## 50.5899
```

```
tableSmoker["1"] / sum(tableSmoker) * 100
```

```
##       1
## 49.4101
```

```
tableBPMeds["0"] / sum(tableBPMeds) * 100
```

```
##        0
## 97.03704
```

```
tableBPMeds["1"] / sum(tableBPMeds) * 100
```

```
##        1
## 2.962963
```

```r
tableStroke["0"] / sum(tableStroke) * 100
```

```
##       0
## 99.4101
```

```r
tableStroke["1"] / sum(tableStroke) * 100
```

```
##         1
## 0.5899009
```

```r
tableHypertension["0"] / sum(tableHypertension) * 100
```

```
##        0
## 68.94762
```

```r
tableHypertension["1"] / sum(tableHypertension) * 100
```

```
##        1
## 31.05238
```

```r
tableDiabetes["0"] / sum(tableDiabetes) * 100
```

```
##        0
## 97.42803
```

```r
tableDiabetes["1"] / sum(tableDiabetes) * 100
```

```
##        1
## 2.571968
```

```r
tableEducation["1"] / sum(tableEducation) * 100
```

```
##        1
## 41.61626
```

```r
tableEducation["2"] / sum(tableEducation) * 100
```

```
##        2
## 30.31696
```

```r
tableEducation["3"] / sum(tableEducation) * 100
```

```
##        3
## 16.62231
```

```r
tableEducation["4"] / sum(tableEducation) * 100
```

```
##        4
## 11.44447
```

```r
table10Year["0"] / sum(table10Year) * 100
```

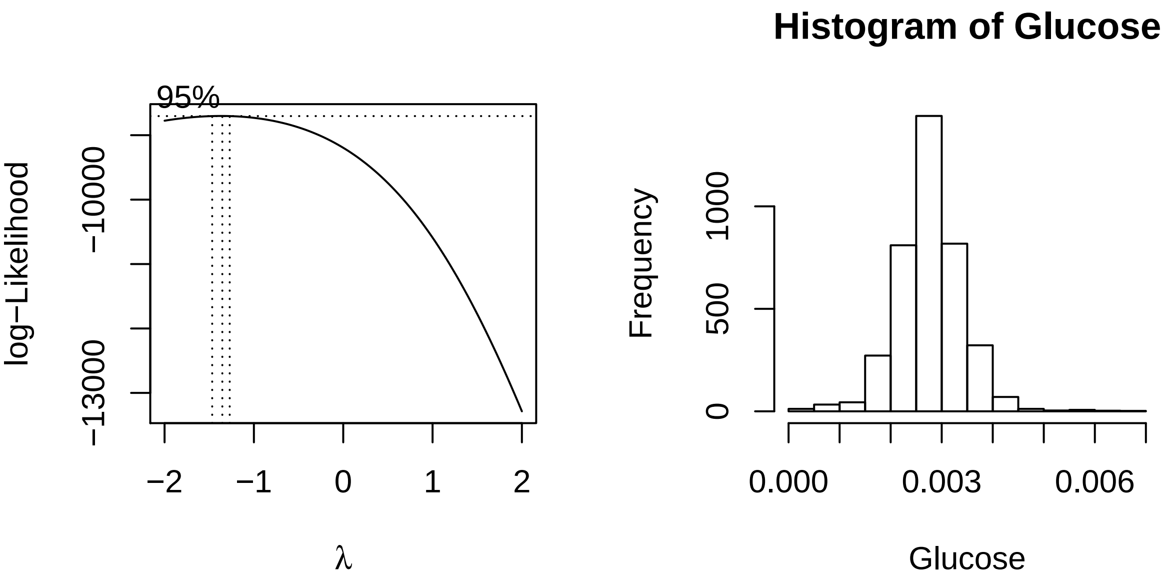
```
##        0
## 84.80415
```

```r
table10Year["1"] / sum(table10Year) * 100
```

```
##        1
## 15.19585
```

Additional Transformations

```
# Two plots side-by-side
par(mfrow=c(1,2))

# Histograms of all numerical variables
MASS::boxcox(lm(glucose ~ 1, data=cleanData))
hist(dataset$glucose^(-1.35),
     xlab="Glucose",
     main="Histogram of Glucose")
```



In order to normalize the distribution for glucose, we used a Box-Cox analysis and decided that a power transformation was needed. In this case, we raise the values to (-1.35) power.

Logistic Regression Models

First-order Model

```
# Getting rid of N/A values
cleanData$glucoseClean = cleanData$glucose^(-1.35)

# The first order model
firstOrder <- glm(TenYearCHD ~ log(age) +
                               log(cigsPerDay + 0.5) +
                               log(totChol) +
                               log(sysBP) +
                               log(BMI) +
                               diaBP +
                               glucoseClean +
                               education +
                               male +
                               currentSmoker +
                               BPMeds +
                               prevalentStroke +
                               prevalentHyp +
                               diabetes +
                               heartRate,
                               family=binomial,
                               data=cleanData)
```

We have also added 0.5 to the variable cigsPerDay, the primary reason having a significant number of values that were zero.

Numerical Summary

```
summary(firstOrder)
```

```
##
## Call:
## glm(formula = TenYearCHD ~ log(age) + log(cigsPerDay + 0.5) +
##     log(totChol) + log(sysBP) + log(BMI) + diaBP + glucoseClean +
##     education + male + currentSmoker + BPMeds + prevalentStroke +
##     prevalentHyp + diabetes + heartRate, family = binomial, data = cleanData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3689  -0.6018  -0.4300  -0.2723   3.0422
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -2.799e+01  2.999e+00  -9.331  < 2e-16 ***
## log(age)               3.323e+00  3.478e-01   9.555  < 2e-16 ***
## log(cigsPerDay + 0.5)  3.091e-01  9.877e-02   3.129  0.00175 **
## log(totChol)           4.635e-01  2.808e-01   1.651  0.09883 .
## log(sysBP)             2.213e+00  5.688e-01   3.890  0.00010 ***
## log(BMI)               1.164e-01  3.488e-01   0.334  0.73850
## diaBP                 -3.505e-03  6.413e-03  -0.546  0.58473
## glucoseClean          -1.198e+02  7.880e+01  -1.520  0.12850
## education             -4.811e-02  4.921e-02  -0.978  0.32831
## male                   5.320e-01  1.085e-01   4.903 9.44e-07 ***
## currentSmoker         -6.513e-01  3.532e-01  -1.844  0.06516 .
## BPMeds                 2.119e-01  2.308e-01   0.918  0.35837
## prevalentStroke        6.649e-01  4.885e-01   1.361  0.17353
## prevalentHyp           2.249e-01  1.395e-01   1.613  0.10682
## diabetes               5.321e-01  2.526e-01   2.107  0.03513 *
## heartRate             -2.832e-03  4.207e-03  -0.673  0.50086
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3120.5  on 3655  degrees of freedom
## Residual deviance: 2762.4  on 3640  degrees of freedom
## AIC: 2794.4
##
## Number of Fisher Scoring iterations: 5
```

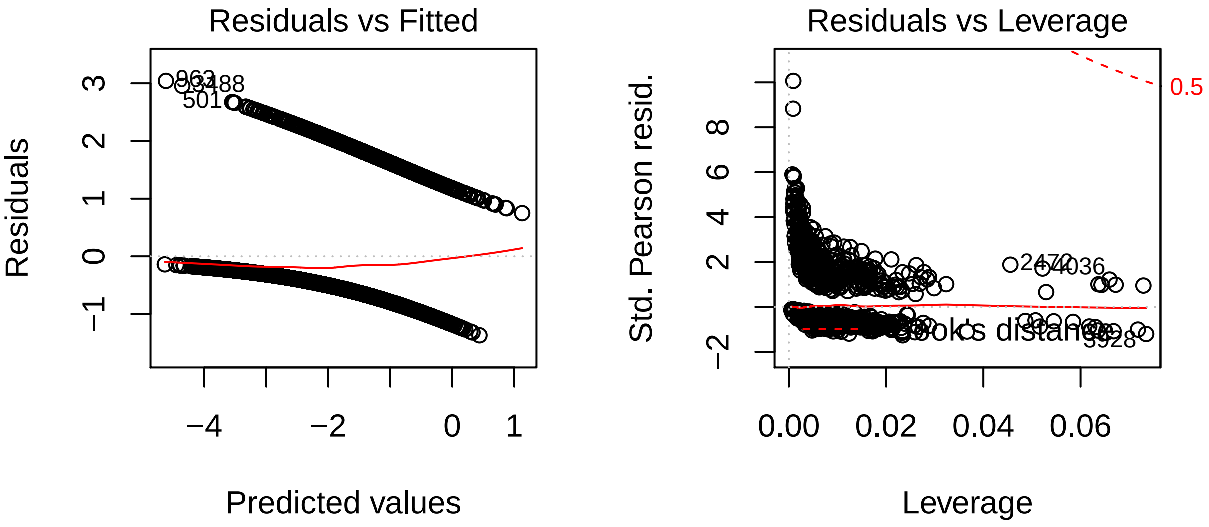
```
confint(firstOrder)
```

```
## Waiting for profiling to be done...

##                              2.5 %        97.5 %
## (Intercept)             -33.90406508 -22.141893322
## log(age)                  2.64652048   4.010310583
## log(cigsPerDay + 0.5)     0.11923074   0.506843859
## log(totChol)             -0.08663639   1.014512208
## log(sysBP)                1.09860138   3.329227831
## log(BMI)                 -0.56823616   0.799664725
## diaBP                    -0.01605176   0.009101456
## glucoseClean           -275.60632779  33.442664052
## education                -0.14535100   0.047657628
## male                      0.31979129   0.745331928
## currentSmoker            -1.36225377   0.024054379
## BPMeds                   -0.24885602   0.657783586
## prevalentStroke          -0.32942754   1.610513930
## prevalentHyp             -0.04906419   0.497912846
## diabetes                  0.02906357   1.021269485
## heartRate                -0.01113002   0.005366116
```

The summary shows that out of all regressor variables, only log(age), log(cigsPerDay + 0.5), log(sysBP), and male are significant with diabetes being marginally significant. The confidence intervals for these variables are also provided above.

Visual Summary and Diagnostics

```
par(mfrow=c(1,2))
plot(firstOrder, which=c(1,5))
```



The first plot (Residuals VS Fitted Values) shows a slight curvature and that the spread is approximately constant. There is slight evidence of heteroscedasticity.

The second plot (Residuals VS Leverage) plot shows no significant outliers.

Improvements - Stepwise Regression and Interaction Effects

For improving the model, we first do a stepwise regression without the interaction effects and then proceed by introducing interactions to the “filtered-out" model.

```
# Centering numerical variables
cleanData$centeredLogAge = log(cleanData$age) - mean(log(cleanData$age))

cleanData$centeredLogCigsPerDay = log(cleanData$cigsPerDay + 0.5) -
                                  mean(log(cleanData$cigsPerDay + 0.5))

cleanData$centeredLogTotChol = log(cleanData$totChol) - mean(log(cleanData$totChol))
cleanData$centeredLogSysBP = log(cleanData$sysBP) - mean(log(cleanData$sysBP))
cleanData$centeredLogDiaBP = log(cleanData$diaBP) - mean(log(cleanData$diaBP))
cleanData$centeredLogBMI = log(cleanData$BMI) - mean(log(cleanData$BMI))

cleanData$centeredLogGlucose = log(cleanData$glucoseClean) -
                               mean(log(cleanData$glucoseClean))

# Interaction effects
modelWithoutInteractions = glm(TenYearCHD ~ centeredLogAge +
                                            centeredLogCigsPerDay +
                                            centeredLogTotChol +
                                            centeredLogSysBP +
                                            centeredLogDiaBP +
                                            centeredLogBMI +
                                            centeredLogGlucose +
                                            male +
                                            currentSmoker +
                                            BPMeds +
                                            prevalentStroke +
                                            prevalentHyp +
                                            diabetes,
                                            family="binomial",
                                            data=cleanData)

# Stepwise regression with no interactions using BIC criterion
# NOTE: `trace=0` disables the traceback functionality of the `step` function
stepwiseWithoutInteractions = step(modelWithoutInteractions,
                                   direction="both",
                                   k=log(dim(cleanData)[1]),
                                   trace=0)

stepwiseWithoutInteractions
```

```
##
## Call:  glm(formula = TenYearCHD ~ centeredLogAge + centeredLogCigsPerDay +
##     centeredLogSysBP + centeredLogGlucose + male, family = "binomial",
##     data = cleanData)
##
## Coefficients:
##           (Intercept)         centeredLogAge  centeredLogCigsPerDay
##               -2.2331                 3.4869                 0.1308
##      centeredLogSysBP     centeredLogGlucose                   male
##                2.6211                -0.5856                 0.5480
```

```
##
## Degrees of Freedom: 3655 Total (i.e. Null);  3650 Residual
## Null Deviance:       3121
## Residual Deviance: 2773  AIC: 2785
```

Due to having 15 predictor variables, we first apply the stepwise procedure using the BIC criterion (to eliminate insignificant ones) and then proceed by introducing interaction effects. The retained predictors are centeredLogAge, centeredLogCigsPerDay, centeredLogSysBP, centeredLogGlucose, and male.

Final Model

```
modelWithInteractions = glm(TenYearCHD ~
                            (centeredLogAge +
                             centeredLogCigsPerDay +
                             centeredLogSysBP +
                             centeredLogGlucose +
                             male)^2,
                            family="binomial",
                            data=cleanData)

# Stepwise regression with interactions
stepwiseWithInteractions = step(modelWithInteractions,
                                direction="both",
                                trace=0)

stepwiseWithInteractions
```

```
##
## Call:  glm(formula = TenYearCHD ~ centeredLogAge + centeredLogCigsPerDay +
##     centeredLogSysBP + centeredLogGlucose + male, family = "binomial",
##     data = cleanData)
##
## Coefficients:
##           (Intercept)         centeredLogAge  centeredLogCigsPerDay
##               -2.2331                 3.4869                 0.1308
##      centeredLogSysBP     centeredLogGlucose                   male
##                2.6211                -0.5856                 0.5480
##
## Degrees of Freedom: 3655 Total (i.e. Null);  3650 Residual
## Null Deviance:       3121
## Residual Deviance: 2773  AIC: 2785
```

In this case, we have included the interaction effects. That said, the stepwise regression could not find any significant interactions and therefore, has not included any in the final model. Thus, our optimal model does not contain interaction effects. The model parameters are centeredLogAge, centeredLogCigsPerDay, centeredLogSysBP, centeredLogGlucose, and male.

NOTE: Interestingly, the dataset has something known as a perfect separation. It is also known as a Hauck-Donner phenomenon.

```
# Same coefficients
stepwiseWithoutInteractions$coefficients
```

```
##           (Intercept)        centeredLogAge centeredLogCigsPerDay
##            -2.2331318             3.4869423             0.1307796
##      centeredLogSysBP    centeredLogGlucose                  male
##             2.6211308            -0.5855547             0.5480394
```

```
stepwiseWithInteractions$coefficients
```

```
##           (Intercept)        centeredLogAge centeredLogCigsPerDay
##            -2.2331318             3.4869423             0.1307796
##      centeredLogSysBP    centeredLogGlucose                  male
##             2.6211308            -0.5855547             0.5480394
```

As shown above, the model without interactions is the same as the model with interactions (since the stepwise regression did not add any interaction effects).

Numerical Summary

```
summary(stepwiseWithInteractions)
```

```
##
## Call:
## glm(formula = TenYearCHD ~ centeredLogAge + centeredLogCigsPerDay +
##     centeredLogSysBP + centeredLogGlucose + male, family = "binomial",
##     data = cleanData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6358  -0.6016  -0.4326  -0.2829   2.9994
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -2.23313    0.07925 -28.180  < 2e-16 ***
## centeredLogAge         3.48694    0.33194  10.505  < 2e-16 ***
## centeredLogCigsPerDay  0.13078    0.02928   4.466 7.95e-06 ***
## centeredLogSysBP       2.62113    0.31492   8.323  < 2e-16 ***
## centeredLogGlucose    -0.58555    0.15366  -3.811 0.000139 ***
## male                   0.54804    0.10352   5.294 1.20e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3120.5  on 3655  degrees of freedom
## Residual deviance: 2773.3  on 3650  degrees of freedom
## AIC: 2785.3
##
## Number of Fisher Scoring iterations: 5
```

```
confint(stepwiseWithInteractions)
```

```
## Waiting for profiling to be done...
```

```
##                            2.5 %     97.5 %
## (Intercept)           -2.3914913 -2.0807276
## centeredLogAge         2.8415882  4.1432867
## centeredLogCigsPerDay  0.0735197  0.1883424
## centeredLogSysBP       2.0059865  3.2409840
## centeredLogGlucose    -0.8860464 -0.2827265
## male                   0.3456040  0.7515869
```

All of the predictor variables are significant. The final AIC value is 2785.3. There were no significant interactions. That said, we have reduced the number of predictor variables from 15 to 5. This is a big improvement as the complexity of the model went down significantly.

P-value for centeredLogAge is less than $2 * 10-16$. The standard error is 0.332.

P-value for centeredLogCigsPerDay is $7.95 * 10^{-6}$. The standard error is 0.029.

P-value for centeredLogSysBP is less than $2 * 10^{-16}$. The standard error is 0.315.

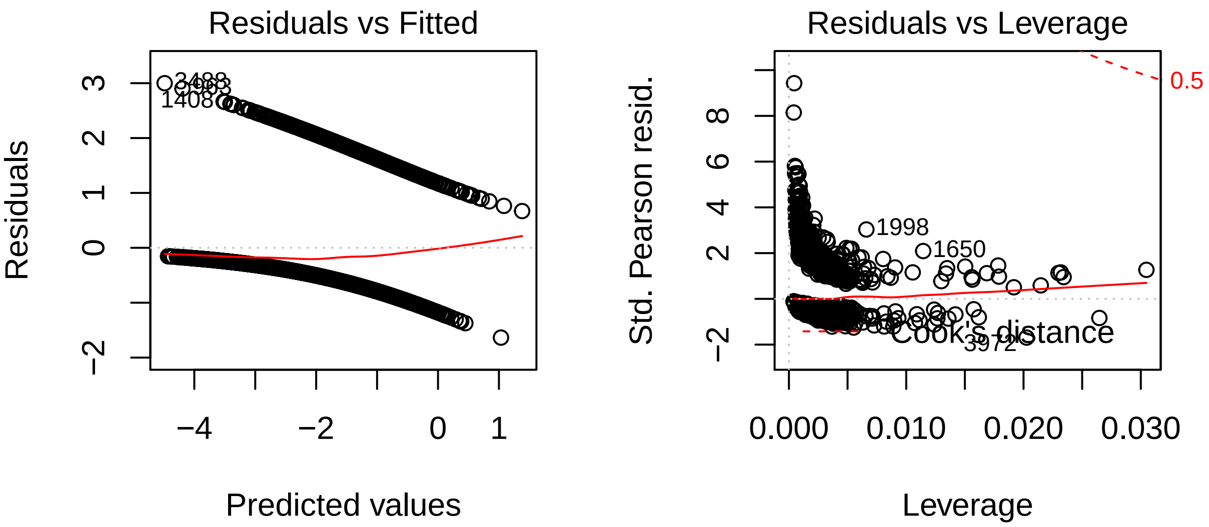
P-value for centeredLogGlucose is 0.000139. The standard error is 0.154..

P-value for male is $1.20 * 10^{-7}$. The standard error is 0.104.

The confidence intervals are shown above.

Visual Summary and Diagnostics

```r
par(mfrow=c(1,2))
plot(stepwiseWithInteractions, which = c(1,5))
```



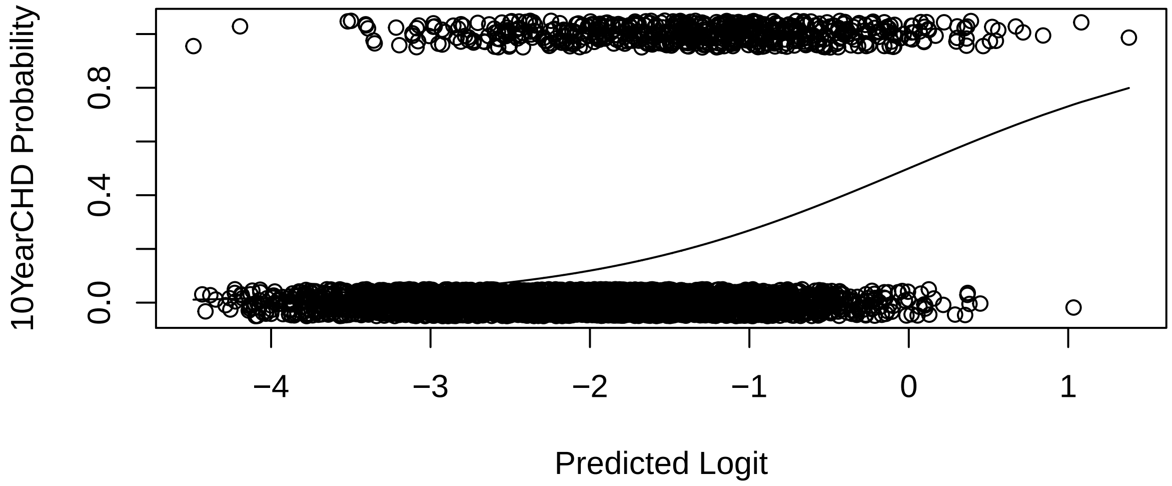
The first plot (Residuals VS Fitted Values) shows slight curvature and that the spread is approximately constant. There is slight evidence of heteroscedasticity.

The second plot (Residuals VS Leverage) plot shows no significant outliers.

General Plot of Response VS Predicted

```r
predpr = predict(stepwiseWithInteractions, type="response")
predlogit = predict(stepwiseWithInteractions)

plot(jitter(cleanData$TenYearCHD, 0.25) ~
            predlogit,
            xlab="Predicted Logit",
            ylab="10YearCHD Probability")
```

```
predOrd = order(predlogit)
lines (predlogit[predOrd], predpr[predOrd])
```



In the plot above, we see that the probability of the risk of a heart disease in the next 10 year increases with age, cigsPerDay, sysBP, glucose, and changes depending on the gender of a person.

Deviance Test of Lack of Fit

```
# First model
pchisq(deviance(firstOrder),
       df.residual(firstOrder),
       lower=F)
```

```
## [1] 1
```

```
# Final model
pchisq(deviance(stepwiseWithInteractions),
       df.residual(stepwiseWithInteractions),
       lower=F)
```

```
## [1] 1
```

There is no significant lack of fit in either the first model (firstOrder) or the final model (stepwiseWithInteractions) as p > 0.05 in both cases.

Likelihood Ratio Test

```
# First model
1 - pchisq(firstOrder$null.deviance - firstOrder$deviance,
           firstOrder$df.null - firstOrder$df.residual)
```

```
## [1] 0
```

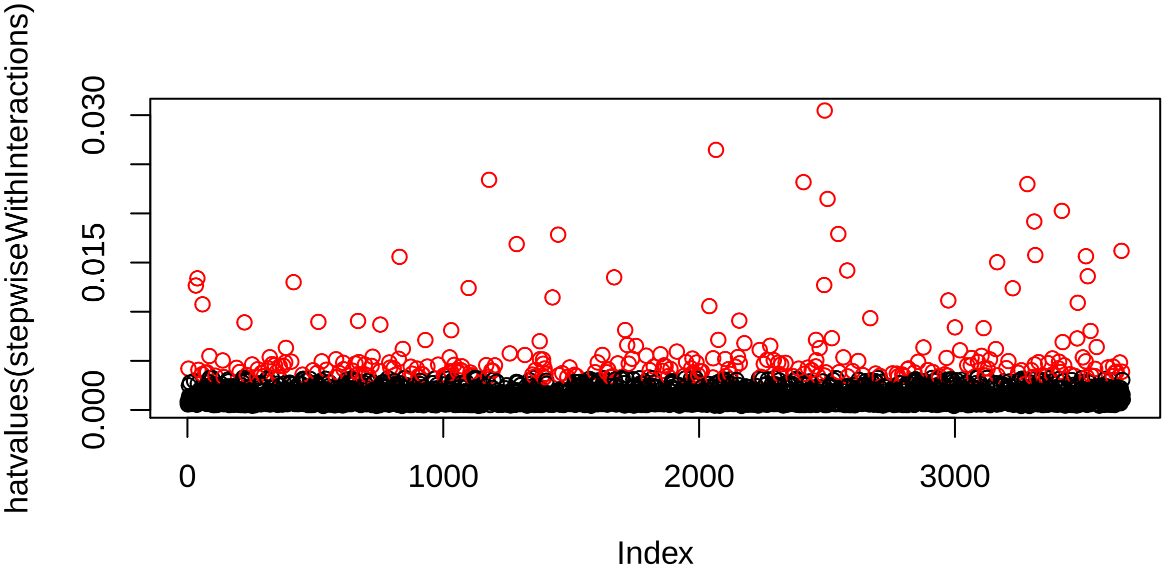
```
# Final model
1 - pchisq(modelWithInteractions$null.deviance - stepwiseWithInteractions$deviance,
           stepwiseWithInteractions$df.null - stepwiseWithInteractions$df.residual)
```

```
## [1] 0
```

Both first and final models have significant effects on TenYearCHD as the p-value is less than 0.05.

Hat Matrix Diagonals

```
thresholdHatMatrix = 2 * length(stepwiseWithInteractions$coefficients) /
                         length(cleanData$diabetes)

plot(hatvalues(stepwiseWithInteractions),
     col=(hatvalues(stepwiseWithInteractions) > thresholdHatMatrix) + 1)
```



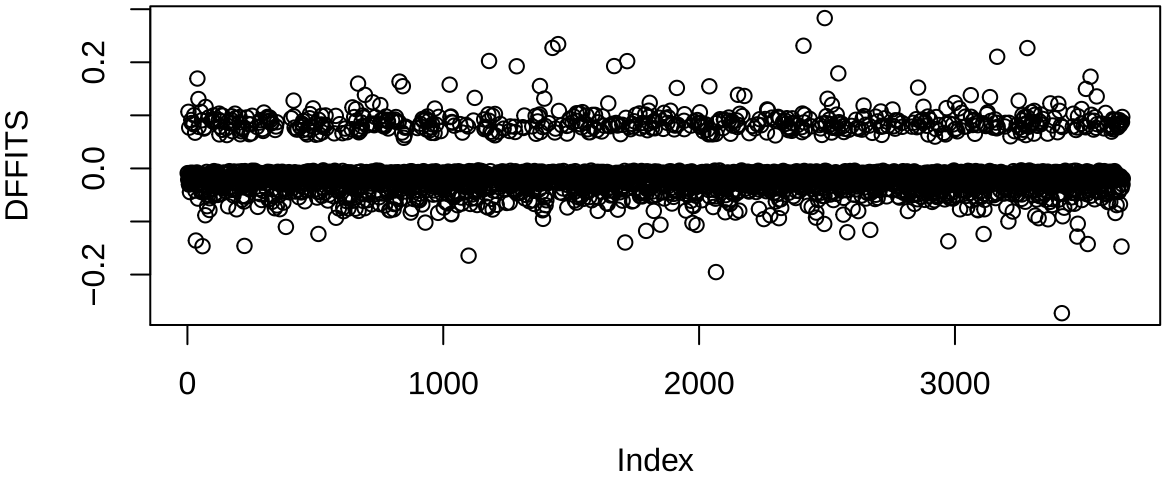
```
# Number of the data points of concern
length(hatvalues(stepwiseWithInteractions)
       [hatvalues(stepwiseWithInteractions) > thresholdHatMatrix])
```

```
## [1] 291
```

There are 291 data points of concern which is approximately 8% of the total data and can be considered relatively negligible when compared to the overall size of the data.

DFFITS

```
thresholdDFFITS <- 2 * sqrt(length(stepwiseWithInteractions$coefficients) /
                            length(stepwiseWithInteractions$diabetes))

DFFITS <- dffits(stepwiseWithInteractions)

plot(DFFITS, col=(DFFITS > thresholdDFFITS) + 1)
```



```
# Data points of concern
length(DFFITS[DFFITS > thresholdDFFITS])
```

```
## [1] 0
```

According to DFFITS criterion, the dataset contains no outliers.

VIF (Variance Inflation Factor)

```
car::vif(stepwiseWithInteractions)
```

```
##        centeredLogAge centeredLogCigsPerDay      centeredLogSysBP 
##              1.176449              1.198061              1.141355 
##    centeredLogGlucose                  male 
##              1.018349              1.130991
```

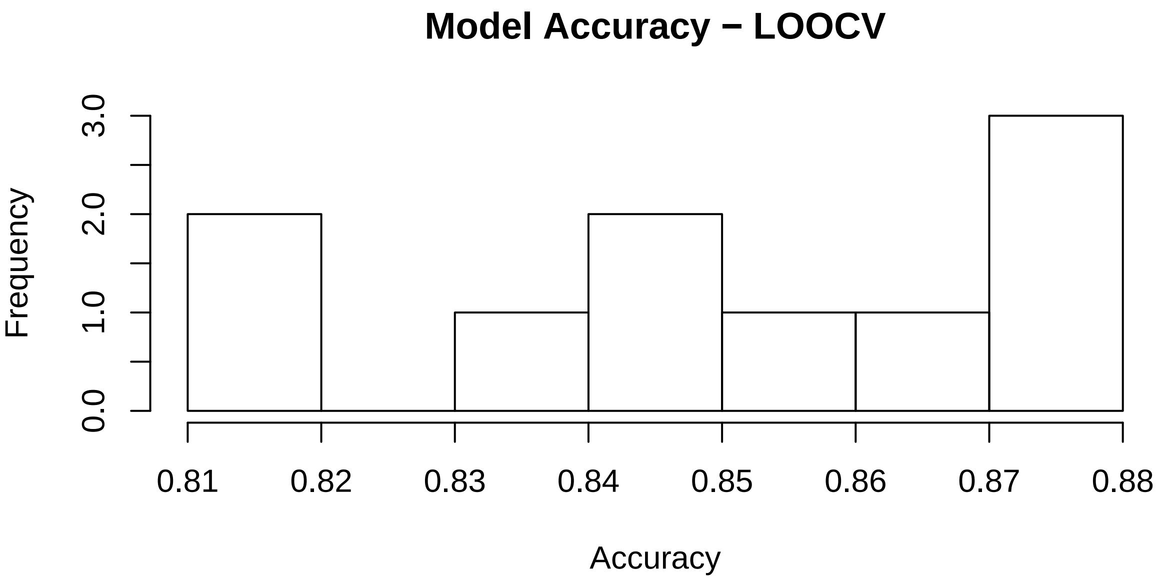
The VIF values for all parameters(the second column in the output) are less than 5 meaning that there is no indication of a problematic amount of collinearity.

10-Fold Cross-Validation

```
df = cleanData[sample(nrow(cleanData)),]
folds = cut(seq(1, nrow(df)), breaks=10, labels=FALSE)
accuracy = NULL

for(i in 1:10) {
  testIndices = which(folds==i, arr.ind=TRUE)

  testData = df[testIndices, ]
  trainData = df[-testIndices, ]

  # Initialize the model with the train data
  model = glm(TenYearCHD ~ centeredLogAge +
                          centeredLogCigsPerDay +
                          centeredLogSysBP +
                          centeredLogGlucose +
                          male,
                          family="binomial",
                              data=trainData)

  # If prob > 0.5 then 1, else 0
  results = ifelse(predict(model, testData) > 0.5, 1, 0)

  # Actual answers
  answers = testData$TenYearCHD

  # Calculate average accuracy
  accuracy[i] = mean(answers == results)
}

# Average accuracy
mean(accuracy)
```

```
## [1] 0.8490149
```

```
# Histogram of the model accuracy
hist(accuracy,
     xlab="Accuracy",
     ylab="Frequency",
     main="Model Accuracy - LOOCV")
```



The 10-fold cross-validation of the model yields 85% percent accuracy which, once again, tells us that the model has a high accuracy when predicting 10-year risk of a coronary heart disease.

The histogram of model accuracy is shown above.

ROC Curve

```
par (mfrow=c(1,1))
library(ROCR)
```

```
## Loading required package: gplots

##
```

```
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

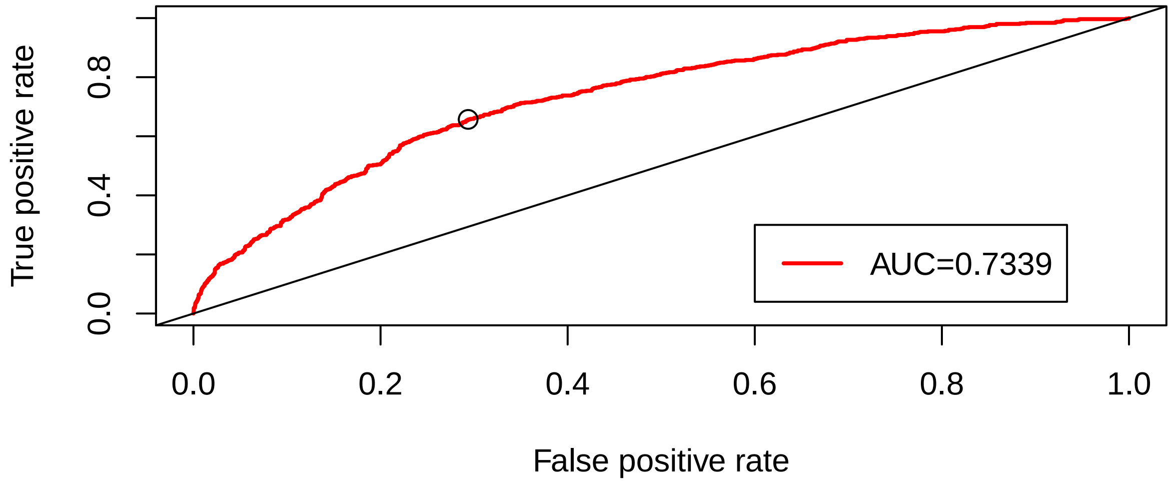
```
pred1 <- prediction(stepwiseWithInteractions$fitted.values, stepwiseWithInteractions$y)
perf1 <- performance(pred1,"tpr","fpr")
auc1 <- performance(pred1,"auc")@y.values[[1]]
auc1
```

```
## [1] 0.7338946
```

```
plot(perf1, lwd=2, col=2)
abline(0,1)
legend(0.6,
       0.3,
       c(paste ("AUC=", round (auc1, 4), sep="")),
       lwd=2,
       col=2)

# Extract the X and Y values from the ROC plot, as well as the probability cutoffs
roc.x = slot (perf1, "x.values") [[1]]
roc.y = slot (perf1, "y.values") [[1]]
cutoffs = slot (perf1, "alpha.values") [[1]]

auc.table = cbind.data.frame(cutoff=pred1@cutoffs,
                             tp=pred1@tp,
                             fp=pred1@fp,
                             tn=pred1@tn,
                             fn=pred1@fn)

names (auc.table) = c("Cutoff", "TP", "FP", "TN", "FN")
auc.table$sensitivity = auc.table$TP / (auc.table$TP + auc.table$FN)
auc.table$specificity = auc.table$TN / (auc.table$TN + auc.table$FP)
auc.table$FalsePosRate = 1 - auc.table$specificity
auc.table$sens_spec = auc.table$sensitivity + auc.table$specificity

# Find the row(s) in the AUC table where sensitivity + specificity is maximized
auc.best = auc.table [auc.table$sens_spec == max (auc.table$sens_spec),]
auc.best
```

```
##        Cutoff  TP  FP   TN  FN sensitivity specificity FalsePosRate
## 936 0.1656945 366 910 2189 191   0.6570916   0.7063569    0.2936431
##     sens_spec
## 936  1.363448
```

```
# Plot the maximum point(s) on the ROC plot
points (auc.best$FalsePosRate, auc.best$sensitivity, cex=1.3)
```



The ROC curve suggests the predictive ability of this model is better than random guessing since the AUC Value(0.734) is larger than 0.5. The optimal cutoff for classification is a fitted probability of 0.166, which has a false positive rate (1 - specificity) of 0.294, and a true positive rate (sensitivity) of 0.657 (the point is shown as a black circle on the ROC curve).

Conclusions

We have analyzed the dataset with over 4000 data points (3656 without NA values) with the goal of determining the best model for predicting the risk of coronary heart disease within the following 10 years of examination. Our initial model had 15 regressor variables with the response (`10YearCHD`). After applying stepwise regression, the final model retained 5 regressor variables which has drastically reduced the complexity of the model while maintaining accurate predictive properties. A number of diagnostic tests including DFFITS, Hat Matrix Diagonals, and 10-Fold Cross-Validation which were performed and have successfully verified the high accuracy of prediction for this model. The ROC curve also suggests that the model has high accuracy of prediction.

Additional improvements could potentially be made by further analysis of the dataset. Some of the questions that have not been addressed in the analysis and the routes of exploration could include:

- Why are there no significant interactions between the variables?
- Would the accuracy of the model improve if one refits the model without the outliers highlighted by the Hat Matrix diagnostic test?
- Why is there a perfect separation in the dataset?