ADS502 - Group Assignment - Fetal Classification

R Libraries

```
library(reshape2)
library(corrplot)
```

```
## corrplot 0.90 loaded
```

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──
```

```
## ✓ tibble  3.1.2     ✓ purrr   0.3.4
## ✓ tidyr   1.1.3     ✓ stringr 1.4.0
## ✓ readr   2.0.0     ✓ forcats 0.5.1
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x purrr::lift()   masks caret::lift()
```

```
library(arules)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
##
## Attaching package: 'arules'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following objects are masked from 'package:base':
##
##     abbreviate, write
```

```
library(e1071)
library(caret)
library(class)
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```
library(C50)
library(partykit)
```

```
## Loading required package: grid
```

```
## Loading required package: libcoin
```

```
## Loading required package: mvtnorm
```

```
library(nnet)
library(rpart)
library(rpart.plot)
library(caret)
library(randomForest)
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
library(ggcorrplot)
options(scipen=999)
```

Exploratory Data Analysis

In this section, we will explore our data and develop an understanding of the information available to us. Our overall goal is to determine which records in the set may be prone to higher chances of mortality.

Read data

```
fetal_df <- read.csv(file = 'fetal_health.csv')
head(fetal_df)
```

```
##   baseline.value accelerations fetal_movement uterine_contractions
## 1            120         0.000              0                 0.000
## 2            132         0.006              0                 0.006
## 3            133         0.003              0                 0.008
## 4            134         0.003              0                 0.008
## 5            132         0.007              0                 0.008
## 6            134         0.001              0                 0.010
##   light_decelerations severe_decelerations prolongued_decelerations
## 1               0.000                    0                     0.000
## 2               0.003                    0                     0.000
## 3               0.003                    0                     0.000
## 4               0.003                    0                     0.000
## 5               0.000                    0                     0.000
## 6               0.009                    0                     0.002
##   abnormal_short_term_variability mean_value_of_short_term_variability
## 1                              73                                  0.5
## 2                              17                                  2.1
## 3                              16                                  2.1
## 4                              16                                  2.4
## 5                              16                                  2.4
## 6                              26                                  5.9
##   percentage_of_time_with_abnormal_long_term_variability
## 1                                                     43
## 2                                                      0
## 3                                                      0
## 4                                                      0
## 5                                                      0
## 6                                                      0
##   mean_value_of_long_term_variability histogram_width histogram_min
## 1                                 2.4              64            62
## 2                                10.4             130            68
## 3                                13.4             130            68
## 4                                23.0             117            53
## 5                                19.9             117            53
## 6                                 0.0             150            50
##   histogram_max histogram_number_of_peaks histogram_number_of_zeroes
## 1           126                         2                          0
## 2           198                         6                          1
## 3           198                         5                          1
## 4           170                        11                          0
## 5           170                         9                          0
## 6           200                         5                          3
##   histogram_mode histogram_mean histogram_median histogram_variance
## 1            120            137              121                 73
## 2            141            136              140                 12
## 3            141            135              138                 13
## 4            137            134              137                 13
## 5            137            136              138                 11
## 6             76            107              107                170
##   histogram_tendency fetal_health
## 1                  1            2
## 2                  0            1
## 3                  0            1
## 4                  1            1
## 5                  1            1
## 6                  0            3
```

Missing Values

As we see below, there are no missing values in this dataset. Thus, no action will be taken in this regard.

```
## Count of missing values for each column.
sapply(fetal_df, function(x) sum(is.na(x)))
```

```
##                                          baseline.value
##                                                       0
##                                           accelerations
##                                                       0
##                                          fetal_movement
##                                                       0
##                                    uterine_contractions
##                                                       0
##                                     light_decelerations
##                                                       0
##                                    severe_decelerations
##                                                       0
##                                 prolongued_decelerations
##                                                       0
##                          abnormal_short_term_variability
##                                                       0
##                        mean_value_of_short_term_variability
##                                                       0
## percentage_of_time_with_abnormal_long_term_variability
##                                                       0
##                         mean_value_of_long_term_variability
##                                                       0
##                                       histogram_width
##                                                       0
##                                         histogram_min
##                                                       0
##                                         histogram_max
##                                                       0
##                                  histogram_number_of_peaks
##                                                       0
##                                histogram_number_of_zeroes
##                                                       0
##                                    histogram_mode
##                                                       0
##                                    histogram_mean
##                                                       0
##                                  histogram_median
##                                                       0
##                                 histogram_variance
##                                                       0
##                                 histogram_tendency
##                                                       0
##                                       fetal_health
##                                                       0
```

Correlation Analysis

Correlations will be used to reduce the feature set down initially to those that have more of a relation to fetal_health, the target variable.

Note that due to the large size of the correlation matrix, it has been output as a .png file and discussed further in our paper.

Further EDA will be conducted on the remaining feature set.

We see from the first visual below that there are no features that have a strong correlation to fetal_health; with the highest correlation being *prolonged_decelerations* (0.48).

```
fetalcor <- round(cor(fetal_df),2)
plot2 <- png(file="corr.png", res=300, width=4500, height=4500)
ggcorrplot(fetalcor, hc.order = TRUE, type = "lower",lab = TRUE,lab_size= 4, tl.cex=10,
           ggtheme = ggplot2::theme_gray,colors = c("#6D9EC1", "white", "#E46726"))
```

Based on the correlations of this dataset; if a minimum correlation of abs(0.20) were used; there would be 10 major predictor features of interest. These have been listed in order of absolute correlation below.

fetal_health - Target Variable

prolongued_decelerations - 0.485

abnormal_short_term_variability - 0.471

percentage_of_time_with_abnormal_long_term_variability - 0.426

accelerations - 0.364

histogram_mode - 0.250

histogram_mean - 0.227

mean_value_of_long_term_variability - 0.227

histogram_variance - 0.207

histogram_median - 0.205

uterine_contractions - 0.204

Of these 10 predictors, the second visual below will be used to ensure that the features are not highly correlated to one another, to avoid weighting the model to a particular direction. If variables are found to be highly correlated to each other, the variable with the higher correlation to fetal_health will be retained and the other removed.

histogram_mode is highly correlated to histogram_mean and histogram_median. The latter two features will be removed.

All other features will be retained.

Hence, the dataframe has been reduced to 8 features at this stage, which will be analyzed further;

Feature list after correlation analysis

fetal_health

prolongued_decelerations

abnormal_short_term_variability

percentage_of_time_with_abnormal_long_term_variability

accelerations

histogram_mode

mean_value_of_long_term_variability

histogram_variance

uterine_contractions

```
fetal_df <- fetal_df[, c('prolongued_decelerations', 'abnormal_short_term_variability', 'percentage_of_time_with_
abnormal_long_term_variability', 'accelerations', 'histogram_mode', 'mean_value_of_long_term_variability', 'histo
gram_variance', 'uterine_contractions','fetal_health')]
```

Removal of Outliers

The following function has been defined and used to remove outliers from the 8 columns above based on the analyses from section **Distributions and Outlier Analysis**.

Outliers have been defined as following:

First Quartile = Q1 Third Quartile = Q3 Interquartile Range = IQR

Outliers are any points < (Q1 - (1.5 * IQR)) or points > (Q3 + (1.5 * IQR))

Based on the boxplots in Appendix 1, we see most predictor show some tendency towards outliers. However, the majority of the distributions seen in the histograms of these columns also show that most values tend to 0, or close to it. Thus, any non-zero value may be important in the context of fetal_health (e.g. prolonged decelarations may only occur in circumstances where fetal health is compromised).

Predictor variables **abnormal_short_term_variability and histogram_mode** will have outliers removed, whilst the other predictor variables will not be transformed or reduced.

```
outliers <- function(x) {

  Q1 <- quantile(x, probs=.25)
  Q3 <- quantile(x, probs=.75)
  iqr = Q3-Q1

 upper_limit = Q3 + (iqr*1.5)
 lower_limit = Q1 - (iqr*1.5)

 x > upper_limit | x < lower_limit
}

remove_outliers <- function(df, cols = names(df)) {
  for (col in cols) {
    df <- df[!outliers(df[[col]]),]
  }
  df
}


fetal_df2 <- remove_outliers(fetal_df, c('abnormal_short_term_variability', 'histogram_mode'))
```

No transformation

As mentioned in the outlier removal step, many of the remaining predictor variables have a tendency of the value being close to zero.

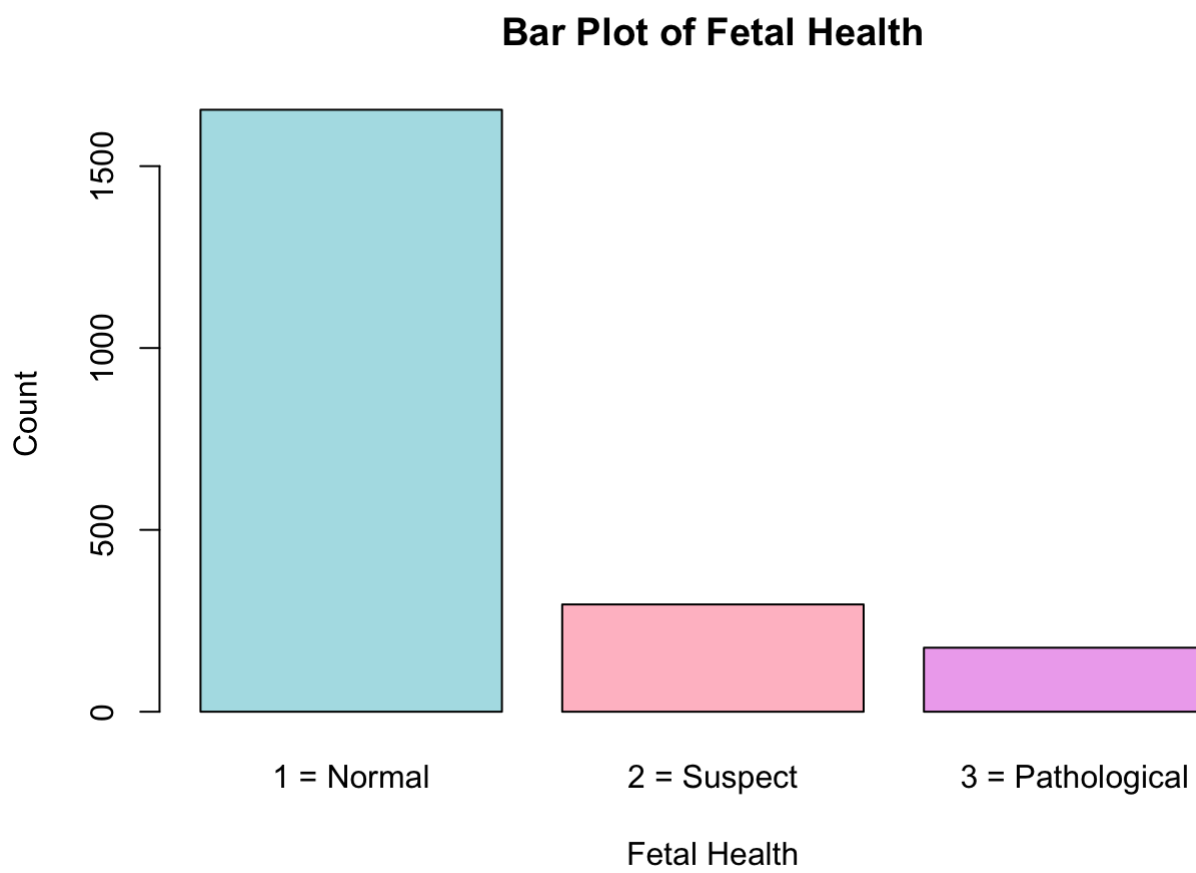
prolongued_decelerations, percentage_of_time_with_abnormal_long_term_variability, accelerations, mean_value_of_long_term_variability, histogram_variance, uterine_contractions and fetal_health have not been transformed and will be used as-is.

Target Variable

Before splitting our cleaned data into training/test sets for classification, a final exploration has been conducted on the target variable **fetal_health*.

We can see from below that the vast majority of records reside in category 1 (healthy). This may create bias in our model, and hence the data will be resampled prior to running through our algorithms.

```
barplot(table(fetal_df$fetal_health), col = c("powderblue", "pink", "plum2"),xlab = "Fetal Health", ylab = "Coun
t", names.arg = c("1 = Normal", "2 = Suspect", "3 = Pathological"), main = "Bar Plot of Fetal Health")
```



For Modelling Phase; create training and test sets

```
## Create train and test sets; to be used later for modelling
set.seed(7)
sample_size = round(nrow(fetal_df2)*.80)
index <- sample(seq_len(nrow(fetal_df2)), size = sample_size)

fetal_train <- fetal_df2[index, ]
fetal_test <- fetal_df2[-index, ]


fetal_train_dim <- dim(fetal_train)
cat('Number of Rows in Student Training Dataset: ', fetal_train_dim[1])
```

```
## Number of Rows in Student Training Dataset:  1642
```

```
cat('Number of Variables in Student Training Dataset: ', fetal_train_dim[2])
```

```
## Number of Variables in Student Training Dataset:  9
```

```
#looking at dimensions of testing file
fetal_test_dim <- dim(fetal_test)
cat('Number of Rows in Student Testing Dataset: ', fetal_test_dim[1])
```

```
## Number of Rows in Student Testing Dataset:  411
```

```
cat('Number of Variables in Student Testing Dataset: ', fetal_test_dim[2])
```

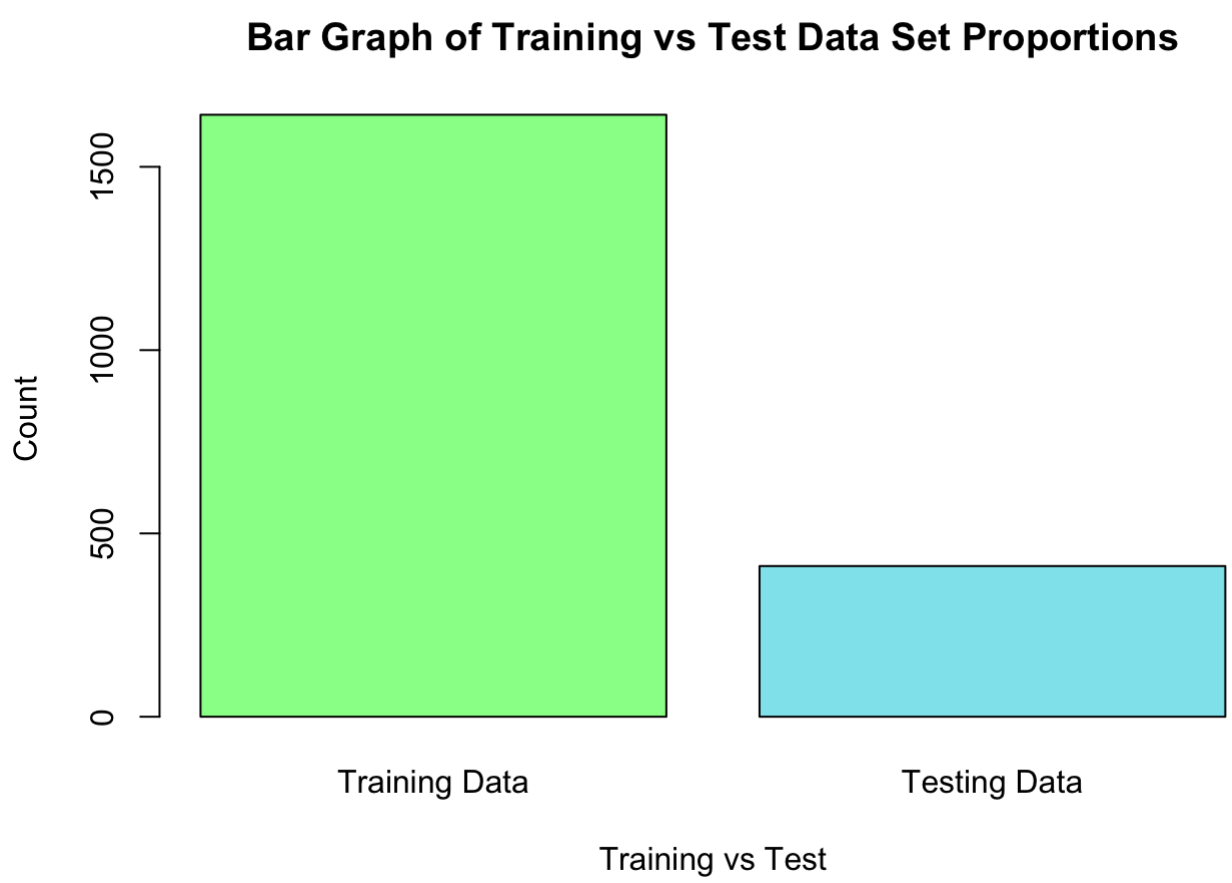
```
## Number of Variables in Student Testing Dataset:  9
```

```
#looking at fetal health data
t1 <- table(fetal_train$fetal_health)
rownames(t1) <- c("1 = Normal", "2 = Suspect", " 3 = Pathalogical")
t1
```

```
##
##        1 = Normal       2 = Suspect  3 = Pathalogical
##              1305               245                92
```

```
#bar graph to see training and testing dimensions
fetal_train_test_prop <- c(fetal_train_dim[1], fetal_test_dim[1])

barplot(fetal_train_test_prop, col = c("palegreen", "cadetblue2"), main = "Bar Graph of Training vs Test Data Set
Proportions", xlab = "Training vs Test", ylab = "Count", names.arg = c("Training Data", "Testing Data"), cex.name
s = 1)
```



Rebalancing the training set so that p(3) = .15

As mentioned earlier, most records in the dataset have fetal_health = 1. fetal_health = 3 had the lowest occurrence; which has been rebalanced to have a sample size closer to fetal_health = 2. The records are not perfectly balanced, but this new split ensures better sampling across the three classes.

```
table(fetal_train$fetal_health)
```

```
##
##    1    2    3
## 1305  245   92
```

```
#We can see that p(3) is currently .056 so let's increase it to .15
to.resample1 <- which(fetal_train$fetal_health == "3")
our.resample1 <- sample(x = to.resample1, size = 182, replace = TRUE)
our.resample1 <- fetal_train[our.resample1, ]
train_fetal_rebal <- rbind(fetal_train, our.resample1)
table(train_fetal_rebal$fetal_health)
```

```
##
##    1    2    3
## 1305  245  274
```

```
#chi square test for homogeneity of proportions
#first making a table of training count and testing count for each target var
table5.2 <- as.table(rbind(c(1390, 237, 278), c(329, 56, 26)))
dimnames(table5.2) <- list(Data.set = c("Training Set", "Test Set"), Status = c("1", "2", "3"))
Xsq_data <- chisq.test(table5.2)
#test statistic
Xsq_data$statistic
```

```
## X-squared
##  20.26222
```

```
#p-value
Xsq_data$p.value
```

```
## [1] 0.00003982115
```

```
#expected counts
Xsq_data$expected
```

```
##               Status
## Data.set               1         2         3
##   Training Set 1413.9443 241.00389 250.05181
##   Test Set      305.0557  51.99611  53.94819
```

```
write.csv(fetal_test,"/Users/bikramgill/Documents/GitHub/ADS502/fetal_test.csv")
```

Classification Models

Now that our features have been selected, data has been cleaned and train/test data has been prepared, we will evaluate the performance of various classification models on this dataset. The goal is to determine whether underlying data can be used to determine fetal health accurately, and if so, which model provides the best results.

This will be further explored in our overall paper.

Evaluation function

This function has been defined to compute evaluation metrics based on contingency tables. It will be used by our classification models further below.

```
## Define function for later use.
summaryStats <- function(cm) {
  #convert confusion matrix to matrix
  cm <- as.matrix(cm)
  #number of instances
  n = sum(cm)
  #numbers of classes
  nc = nrow(cm)
  #correctly classified instances in a class
  diag = diag(cm)
  #numbers of instances in a class
  rowsums = apply(cm, 1, sum)
  #number of predictions in a class
  colsums = apply(cm, 2, sum)
  accuracy = sum(diag)/n
  error_rate = 1 - accuracy
  # precision = diag/colsums
  # recall = diag/rowsums
  precision = diag/rowsums
  recall = diag/colsums
  f1 = 2* precision*recall / (precision + recall)
  f2 = 5*(precision*recall) / ((4*precision) + recall)
  f0.5 = 1.25*(precision*recall) / ((0.25*precision) + recall)
  fetal.health <- c(1,2,3)
  theRest <- data.frame(fetal.health,precision, recall, f1, f2, f0.5)
  theRest <- theRest %>% kbl(caption = sprintf("Accuracy: %f \\\\ Error Rate: %f", accuracy, error_rate) ) %>%  k
able_classic(full_width = F, html_font = "Cambria")
  return(theRest)

}
```

Logistic Regression

```
#train regression model
logreg02 <- multinom(fetal_health ~ prolongued_decelerations + abnormal_short_term_variability + percentage_of_ti
me_with_abnormal_long_term_variability + accelerations + histogram_mode + mean_value_of_long_term_variability + h
istogram_variance + uterine_contractions, data = train_fetal_rebal)
```

```
## # weights:  30 (18 variable)
## initial  value 2003.868815
## iter  10 value 1138.328807
## iter  20 value 671.552214
## iter  30 value 651.674352
## iter  40 value 562.464715
## iter  50 value 562.329588
## iter  60 value 545.532941
## iter  70 value 505.896392
## iter  80 value 495.832730
## iter  80 value 495.832728
## iter  80 value 495.832727
## final  value 495.832727
## converged
```

```
#prediction on test data
lpred <- predict(logreg02, fetal_test, type = 'class')
table(fetal_test$fetal_health, lpred)
```

```
##    lpred
##       1   2   3
##   1 319  10   4
##   2  16  23   9
##   3   4   6  20
```

```
#evalutation metrics
logreg_cm <- confusionMatrix(lpred, factor(fetal_test$fetal_health))
summaryStats(logreg_cm)
```

Accuracy: 0.880779 \ Error Rate: 0.119221

fetal.health	precision	recall	f1	f2	f0.5
1	0.9410029	0.9579580	0.9494048	0.9545183	0.9443458
2	0.5897436	0.4791667	0.5287356	0.4978355	0.5637255
3	0.6060606	0.6666667	0.6349206	0.6535948	0.6172840

K-NN

```
#normalize data
data_norm <- function(x) {((x - min(x)) / (max(x) - min(x)))}

#normalize train and test data
fetal_train_norm <- as.data.frame(lapply(fetal_train[ , c(1:8)], data_norm))
fetal_test_norm <- as.data.frame(lapply(fetal_test[ , c(1:8)], data_norm))

#getting our target variable
fetal_train_labels <- fetal_train[1:fetal_train_dim[1], 9]
fetal_test_labels <- fetal_test[1:fetal_test_dim[1], 9]


#k decided based on the squareroot of data points
#training dataset has 1642 variables. Therefore k is ~40

fetal_pred <- knn(fetal_train_norm, fetal_test_norm, fetal_train_labels, k = 40)
table_KNN <- table(fetal_pred, fetal_test_labels)
table_KNN <- addmargins(A = table_KNN, FUN = list(Total = sum), quiet = TRUE)
table_KNN
```

```
##           fetal_test_labels
## fetal_pred   1   2   3 Total
##      1     326  16   7   349
##      2       6  27   6    39
##      3       1   5  17    23
##      Total 333  48  30   411
```

```
#evaluation metrics
KNN_cm <-confusionMatrix(fetal_pred, factor(fetal_test$fetal_health))
KNN_cm
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   1   2   3
##          1 326  16   7
##          2   6  27   6
##          3   1   5  17
##
## Overall Statistics
##
##                Accuracy : 0.9002
##                  95% CI : (0.8671, 0.9275)
##     No Information Rate : 0.8102
##     P-Value [Acc > NIR] : 0.0000004042
##
##                   Kappa : 0.6639
##
##  Mcnemar's Test P-Value : 0.02753
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3
## Sensitivity            0.9790  0.56250  0.56667
## Specificity            0.7051  0.96694  0.98425
## Pos Pred Value         0.9341  0.69231  0.73913
## Neg Pred Value         0.8871  0.94355  0.96649
## Prevalence             0.8102  0.11679  0.07299
## Detection Rate         0.7932  0.06569  0.04136
## Detection Prevalence   0.8491  0.09489  0.05596
## Balanced Accuracy      0.8421  0.76472  0.77546
```

```
summaryStats(KNN_cm)
```

Accuracy: 0.900243 \ Error Rate: 0.099757

fetal.health	precision	recall	f1	f2	f0.5
1	0.9340974	0.9789790	0.9560117	0.9696609	0.9427415
2	0.6923077	0.5625000	0.6206897	0.5844156	0.6617647
3	0.7391304	0.5666667	0.6415094	0.5944056	0.6967213

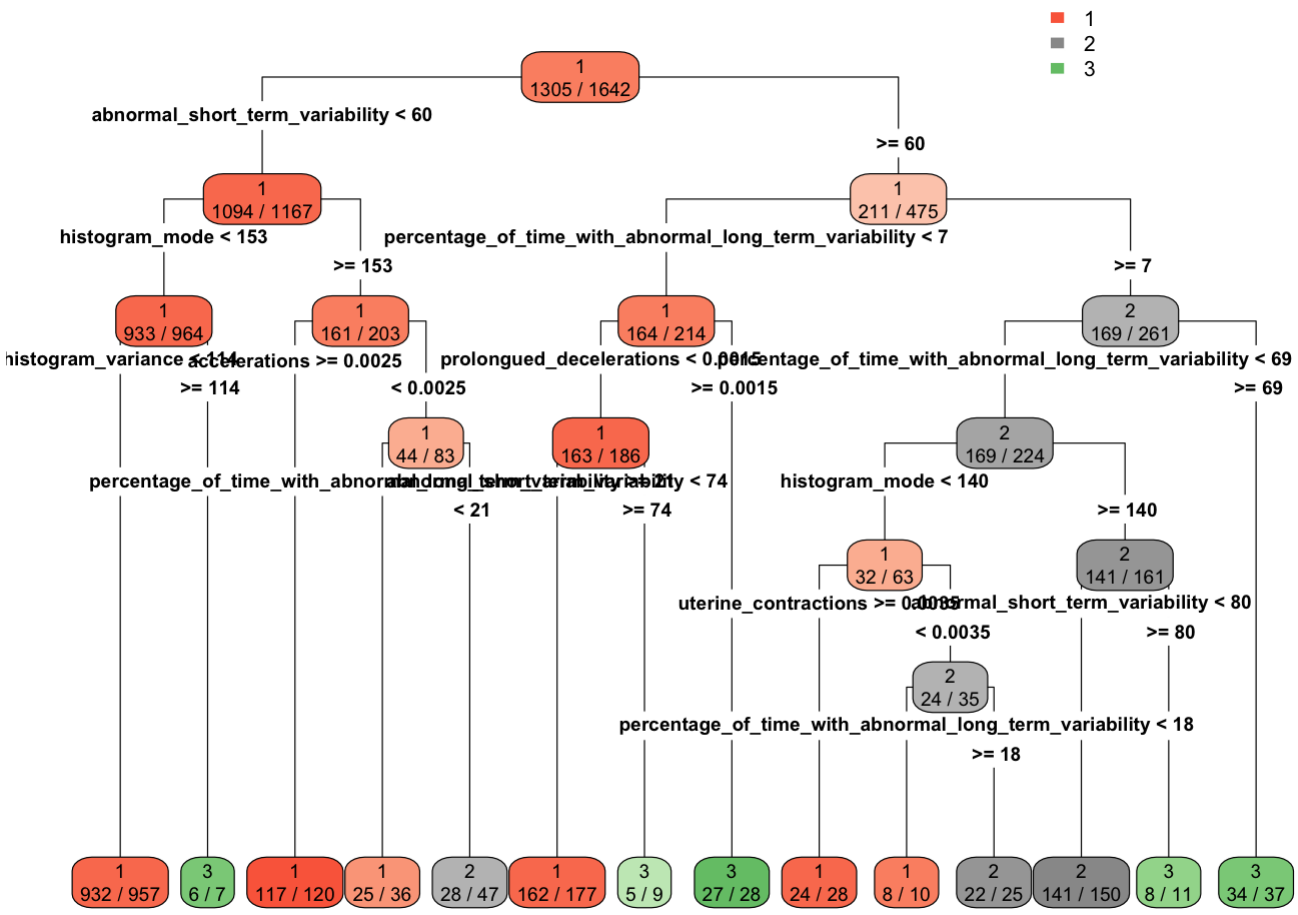
Decision Tree: CART

```
#normalize data
fetal_train$fetal_health <- factor(fetal_train$fetal_health)
fetal_test$fetal_health <- factor(fetal_test$fetal_health)

#train CART model
cart01_fetal_train <- rpart(formula = fetal_health ~ prolongued_decelerations + abnormal_short_term_variability +
percentage_of_time_with_abnormal_long_term_variability + accelerations + histogram_mode + mean_value_of_long_term
_variability + histogram_variance +uterine_contractions, data = fetal_train)

rpart.plot(cart01_fetal_train, type = 4, extra = 2, cex = 0.6)
```

```
## Warning: labs do not fit even at cex 0.15, there may be some overplotting
```




```
#prediction on test data
X_fetal_test_cart <- data.frame(prolongued_decelerations = fetal_test$prolongued_decelerations, abnormal_short_te
rm_variability = fetal_test$abnormal_short_term_variability,percentage_of_time_with_abnormal_long_term_variabilit
y = fetal_test$percentage_of_time_with_abnormal_long_term_variability, accelerations = fetal_test$accelerations,
 histogram_mode = fetal_test$histogram_mode, mean_value_of_long_term_variability = fetal_test$mean_value_of_long_
term_variability, histogram_variance = fetal_test$histogram_variance, uterine_contractions = fetal_test$uterine_c
ontractions, fetal_health =fetal_test$fetal_health )

fetal_predCart <- predict(object = cart01_fetal_train, newdata = X_fetal_test_cart, type = "class")

#evaluation metrics
table_CART <- table(fetal_test$fetal_health, fetal_predCart)
table_CART <- addmargins(A = table_CART, FUN = list(Total=sum), quiet = TRUE)
table_CART
```

```
##        fetal_predCart
##           1   2   3 Total
##   1     322   6   5   333
##   2       9  38   1    48
##   3       7   2  21    30
##   Total 338  46  27   411
```

```
CART_cm <- confusionMatrix(fetal_predCart, fetal_test$fetal_health)
CART_cm
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   1   2   3
##          1 322   9   7
##          2   6  38   2
##          3   5   1  21
##
## Overall Statistics
##
##                Accuracy : 0.927
##                  95% CI : (0.8974, 0.9502)
##     No Information Rate : 0.8102
##     P-Value [Acc > NIR] : 0.00000000001657
##
##                   Kappa : 0.7689
##
##  Mcnemar's Test P-Value : 0.7371
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3
## Sensitivity            0.9670  0.79167  0.70000
## Specificity            0.7949  0.97796  0.98425
## Pos Pred Value         0.9527  0.82609  0.77778
## Neg Pred Value         0.8493  0.97260  0.97656
## Prevalence             0.8102  0.11679  0.07299
## Detection Rate         0.7835  0.09246  0.05109
## Detection Prevalence   0.8224  0.11192  0.06569
## Balanced Accuracy      0.8809  0.88481  0.84213
```

```
summaryStats(CART_cm)
```

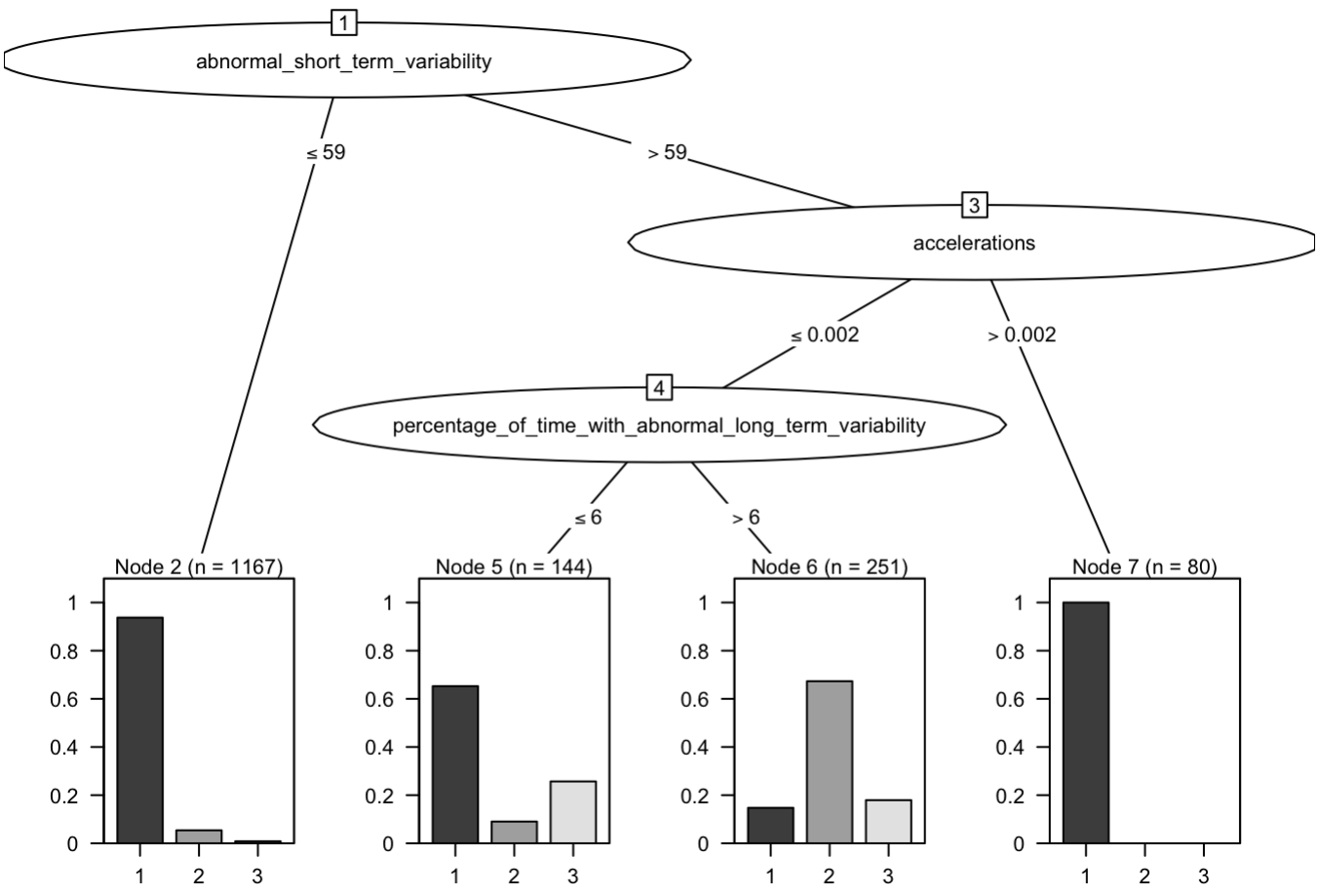
Accuracy: 0.927007 \ Error Rate: 0.072993

fetal.health	precision	recall	f1	f2	f0.5
1	0.9526627	0.9669670	0.9597615	0.9640719	0.9554896
2	0.8260870	0.7916667	0.8085106	0.7983193	0.8189655
3	0.7777778	0.7000000	0.7368421	0.7142857	0.7608696

Decision Tree: C5

```
#train C5 model
C5_fetal <- C5.0(formula = fetal_health ~ prolongued_decelerations + abnormal_short_term_variability + percentage
_of_time_with_abnormal_long_term_variability + accelerations + histogram_mode + mean_value_of_long_term_variabili
ty + histogram_variance +uterine_contractions, data = fetal_train, control = C5.0Control(minCases = 75))

plot(C5_fetal, gp = gpar(fontsize = 8))
```



```
#prediction on test data
fetal_pred_C5 <- predict(object = C5_fetal, newdata = X_fetal_test_cart)

#evaluation metrics
table_C5 <- table(fetal_test$fetal_health, fetal_pred_C5 )
table_C5 <- addmargins(A = table_C5, FUN = list(Total=sum), quiet = TRUE)
table_C5
```

```
##         fetal_pred_C5
##            1   2   3 Total
##   1      325   8   0   333
##   2       18  30   0    48
##   3       16  14   0    30
##   Total  359  52   0   411
```

```
C5_cm <- confusionMatrix(fetal_pred_C5, fetal_test$fetal_health)
C5_cm
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction   1   2   3
##          1 325  18  16
##          2   8  30  14
##          3   0   0   0
##
## Overall Statistics
##
##                Accuracy : 0.8637
##                  95% CI : (0.8267, 0.8954)
##     No Information Rate : 0.8102
##     P-Value [Acc > NIR] : 0.002554
##
##                   Kappa : 0.509
##
##  Mcnemar's Test P-Value : 0.0000002135
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3
## Sensitivity            0.9760  0.62500  0.00000
## Specificity            0.5641  0.93939  1.00000
## Pos Pred Value         0.9053  0.57692      NaN
## Neg Pred Value         0.8462  0.94986  0.92701
## Prevalence             0.8102  0.11679  0.07299
## Detection Rate         0.7908  0.07299  0.00000
## Detection Prevalence   0.8735  0.12652  0.00000
## Balanced Accuracy      0.7700  0.78220  0.50000
```

```
summaryStats(C5_cm)
```

Accuracy: 0.863747 \ Error Rate: 0.136253

fetal.health	precision	recall	f1	f2	f0.5
1	0.9052925	0.975976	0.9393064	0.9609698	0.9185981
2	0.5769231	0.625000	0.6000000	0.6147541	0.5859375
3	NaN	0.000000	NaN	NaN	NaN

Random Forest

```
#train random forest model
random_Fetal <- randomForest(formula = fetal_health ~ prolongued_decelerations + abnormal_short_term_variability
 + percentage_of_time_with_abnormal_long_term_variability + accelerations + histogram_mode + mean_value_of_long_t
erm_variability + histogram_variance +uterine_contractions, data = fetal_train, ntree = 100, type = "classificati
on")

#prediction on test data
fetal_random_pred <- predict(object = random_Fetal, X_fetal_test_cart)

#evaluation metrics
table_RF <- table(fetal_test$fetal_health, fetal_random_pred)
table_RF <- addmargins(A = table_RF, FUN = list(Total=sum), quiet = TRUE)
table_RF
```

```
##        fetal_random_pred
##           1   2   3 Total
##   1     330   2   1   333
##   2      11  37   0    48
##   3       3   5  22    30
##   Total 344  44  23   411
```

```
randforest_cm <- confusionMatrix(fetal_random_pred, fetal_test$fetal_health)
randforest_cm
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   1   2   3
##          1 330  11   3
##          2   2  37   5
##          3   1   0  22
##
## Overall Statistics
##
##                Accuracy : 0.9465
##                  95% CI : (0.9201, 0.9662)
##     No Information Rate : 0.8102
##     P-Value [Acc > NIR] : 0.0000000000000007778
##
##                   Kappa : 0.8247
##
##  Mcnemar's Test P-Value : 0.006633
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3
## Sensitivity            0.9910  0.77083  0.73333
## Specificity            0.8205  0.98072  0.99738
## Pos Pred Value         0.9593  0.84091  0.95652
## Neg Pred Value         0.9552  0.97003  0.97938
## Prevalence             0.8102  0.11679  0.07299
## Detection Rate         0.8029  0.09002  0.05353
## Detection Prevalence   0.8370  0.10706  0.05596
## Balanced Accuracy      0.9058  0.87577  0.86535
```

```
summaryStats(randforest_cm)
```

Accuracy: 0.946472 \ Error Rate: 0.053528

fetal.health	precision	recall	f1	f2	f0.5
1	0.9593023	0.9909910	0.9748892	0.9844869	0.9654769
2	0.8409091	0.7708333	0.8043478	0.7838983	0.8258929
3	0.9565217	0.7333333	0.8301887	0.7692308	0.9016393

Naive Bayes

```
#normalize data
cols = c('prolongued_decelerations', 'abnormal_short_term_variability', 'percentage_of_time_with_abnormal_long_te
rm_variability', 'accelerations', 'histogram_mode', 'mean_value_of_long_term_variability', 'histogram_variance',
'uterine_contractions','fetal_health')
train_fetal_rebal[, cols] <- lapply(train_fetal_rebal[, cols], as.factor)

#train NB model
nb01 <- naiveBayes(formula = fetal_health ~ prolongued_decelerations + abnormal_short_term_variability + percenta
ge_of_time_with_abnormal_long_term_variability + accelerations + histogram_mode + mean_value_of_long_term_variabi
lity + histogram_variance + uterine_contractions, data = train_fetal_rebal)

#prediction on test data
fetal_test[, cols] <- lapply(fetal_test[, cols], as.factor)
ypred <- predict(object = nb01, newdata = fetal_test)
```

##The A-priori probabilities are the values of p(Y) ### p(1) = .715 ### p(2) = .134 ### p(3) = .15

```
#evaluation metrics
t.preds <- table(fetal_test$fetal_health, ypred)
rownames(t.preds) <- c("Actual: 1", "Actual: 2", "Actual: 3")
colnames(t.preds) <- c("Predicted: 1", "Predicted: 2", "Predicted: 3   ")
addmargins(A = t.preds, FUN = list(Total = sum), quiet = TRUE)
```

```
##            ypred
##             Predicted: 1 Predicted: 2 Predicted: 3    Total
##   Actual: 1          307           19                7   333
##   Actual: 2           11           35                2    48
##   Actual: 3            2            7               21    30
##   Total              320           61               30   411
```

```
fetal_test[, cols] <- lapply(fetal_test[, cols], as.factor)
predictions <- predict(nb01, fetal_test)

nb_cm <- confusionMatrix(predictions, fetal_test[, 'fetal_health'], positive='yes')
nb_cm
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   1   2   3
##          1 307  11   2
##          2  19  35   7
##          3   7   2  21
##
## Overall Statistics
##
##                Accuracy : 0.8832
##                  95% CI : (0.8482, 0.9126)
##     No Information Rate : 0.8102
##     P-Value [Acc > NIR] : 0.00004501
##
##                   Kappa : 0.663
##
##  Mcnemar's Test P-Value : 0.0529
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3
## Sensitivity            0.9219  0.72917  0.70000
## Specificity            0.8333  0.92837  0.97638
## Pos Pred Value         0.9594  0.57377  0.70000
## Neg Pred Value         0.7143  0.96286  0.97638
## Prevalence             0.8102  0.11679  0.07299
## Detection Rate         0.7470  0.08516  0.05109
## Detection Prevalence   0.7786  0.14842  0.07299
## Balanced Accuracy      0.8776  0.82877  0.83819
```

```
summaryStats(nb_cm)
```

Accuracy: 0.883212 \ Error Rate: 0.116788

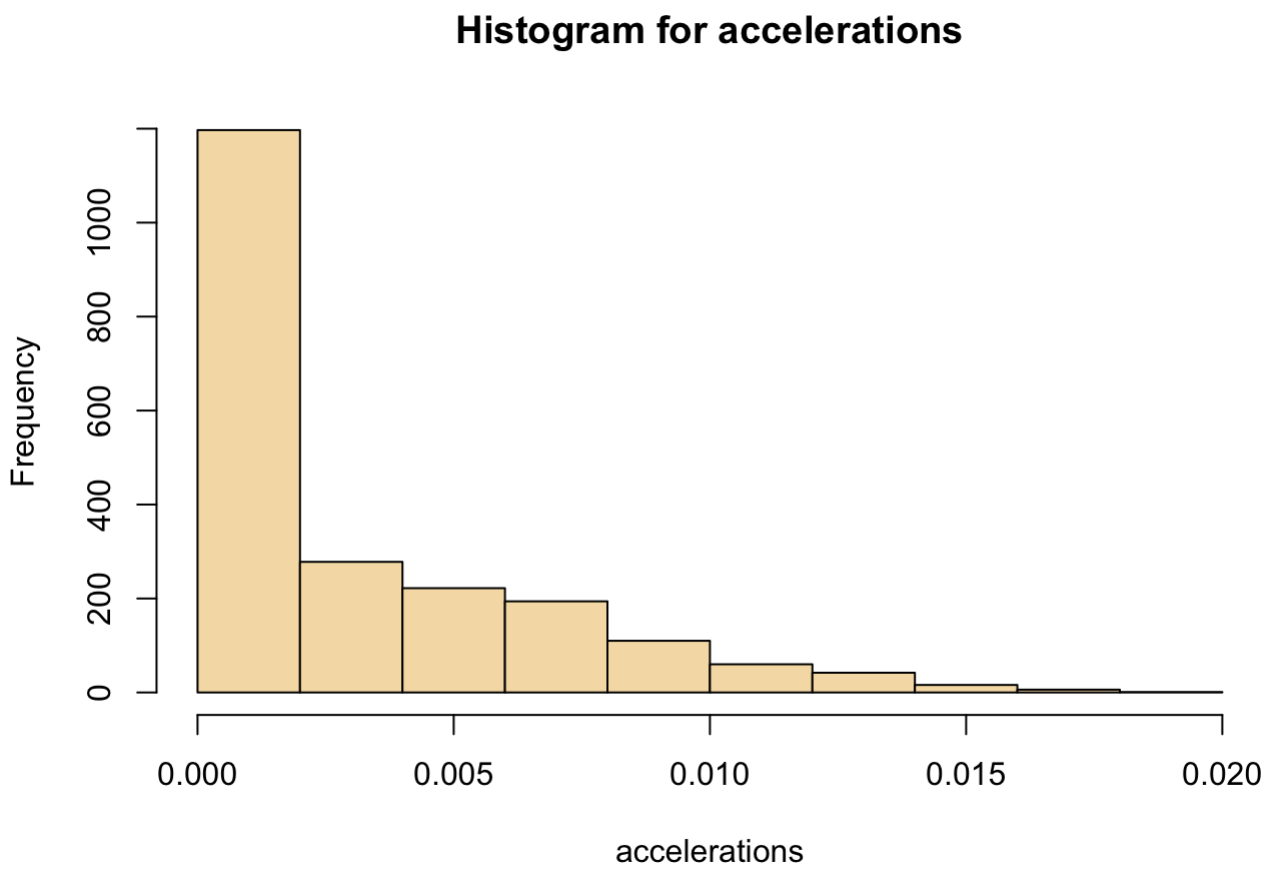
fetal.health	precision	recall	f1	f2	f0.5
1	0.9593750	0.9219219	0.9402757	0.9291768	0.9516429
2	0.5737705	0.7291667	0.6422018	0.6916996	0.5993151
3	0.7000000	0.7000000	0.7000000	0.7000000	0.7000000

Code Appendix

Code Appendix 1: Boxplots and Histograms for each Feature

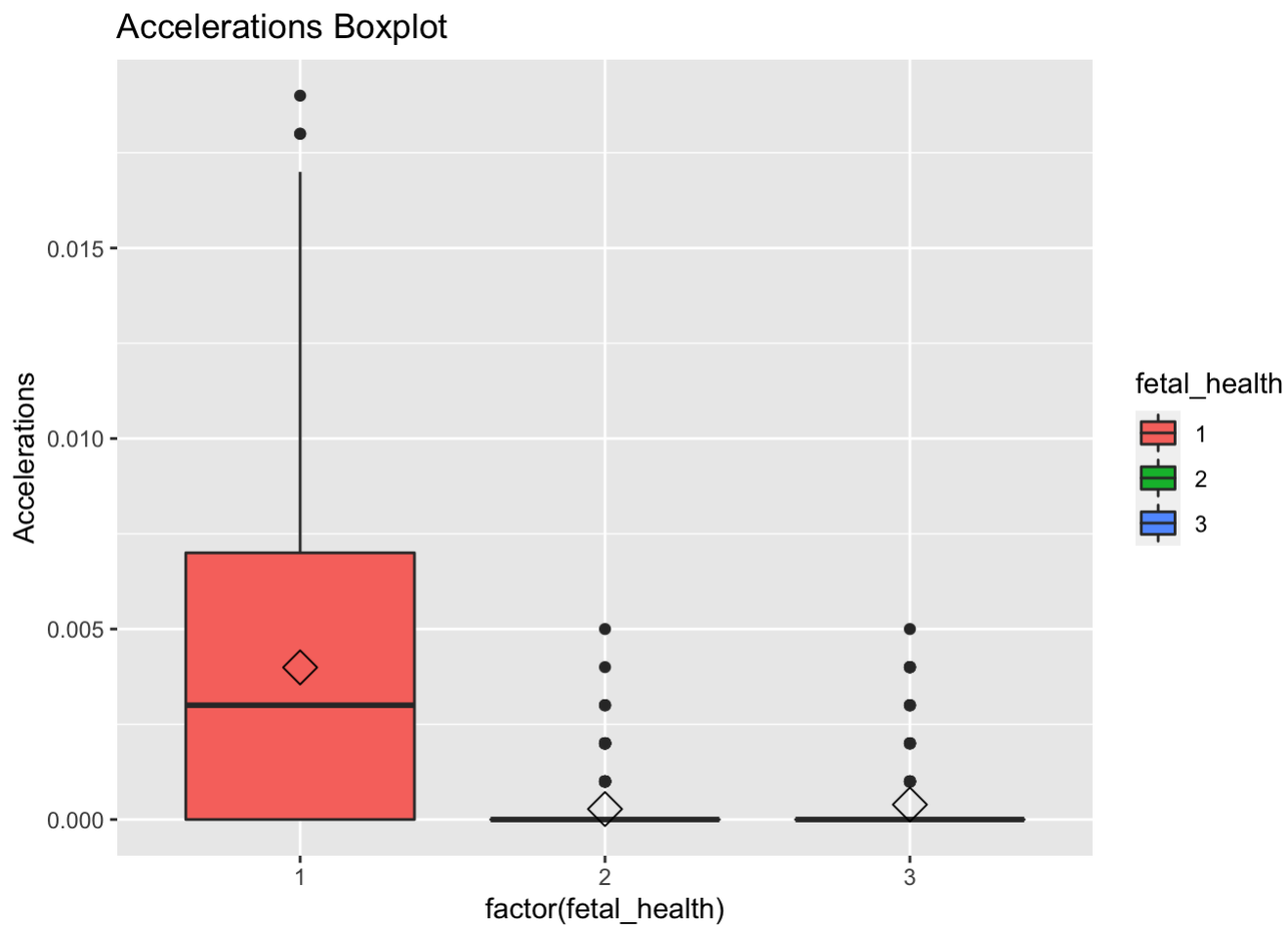
accelerations

```
hist(fetal_df$accelerations,
     main="Histogram for accelerations",
     xlab="accelerations",
     border="black",
     col="wheat")
```



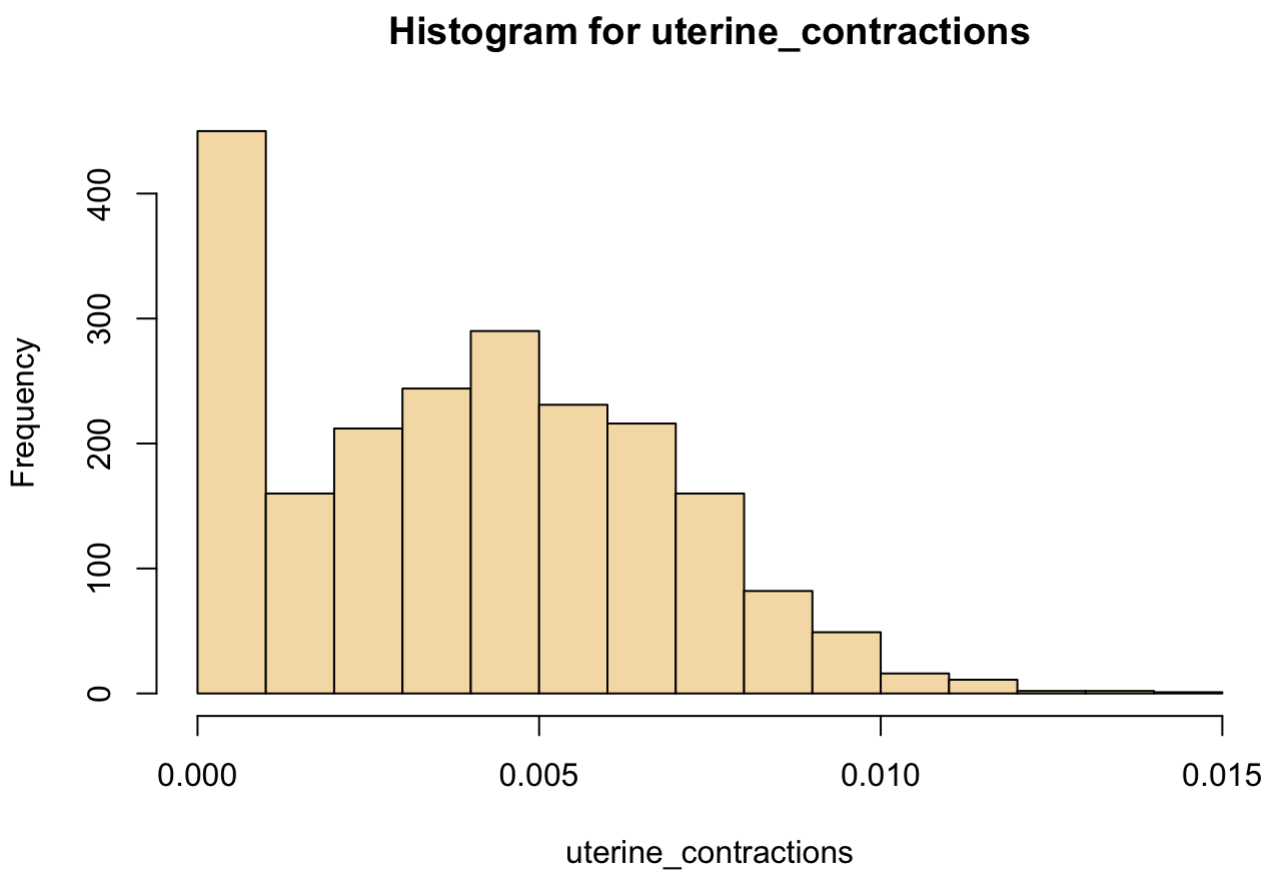

```
temp_fetal_df <- fetal_df
temp_fetal_df$fetal_health <- factor(temp_fetal_df$fetal_health)

accelerations_boxplot <- ggplot(data = temp_fetal_df, aes(x = factor(fetal_health), y = accelerations))
accelerations_boxplot + geom_boxplot(aes(fill = fetal_health)) + ylab("Accelerations") + ggtitle("Accelerations B
oxplot") + stat_summary(fun=mean, geom = "point", shape = 5, size = 4)
```



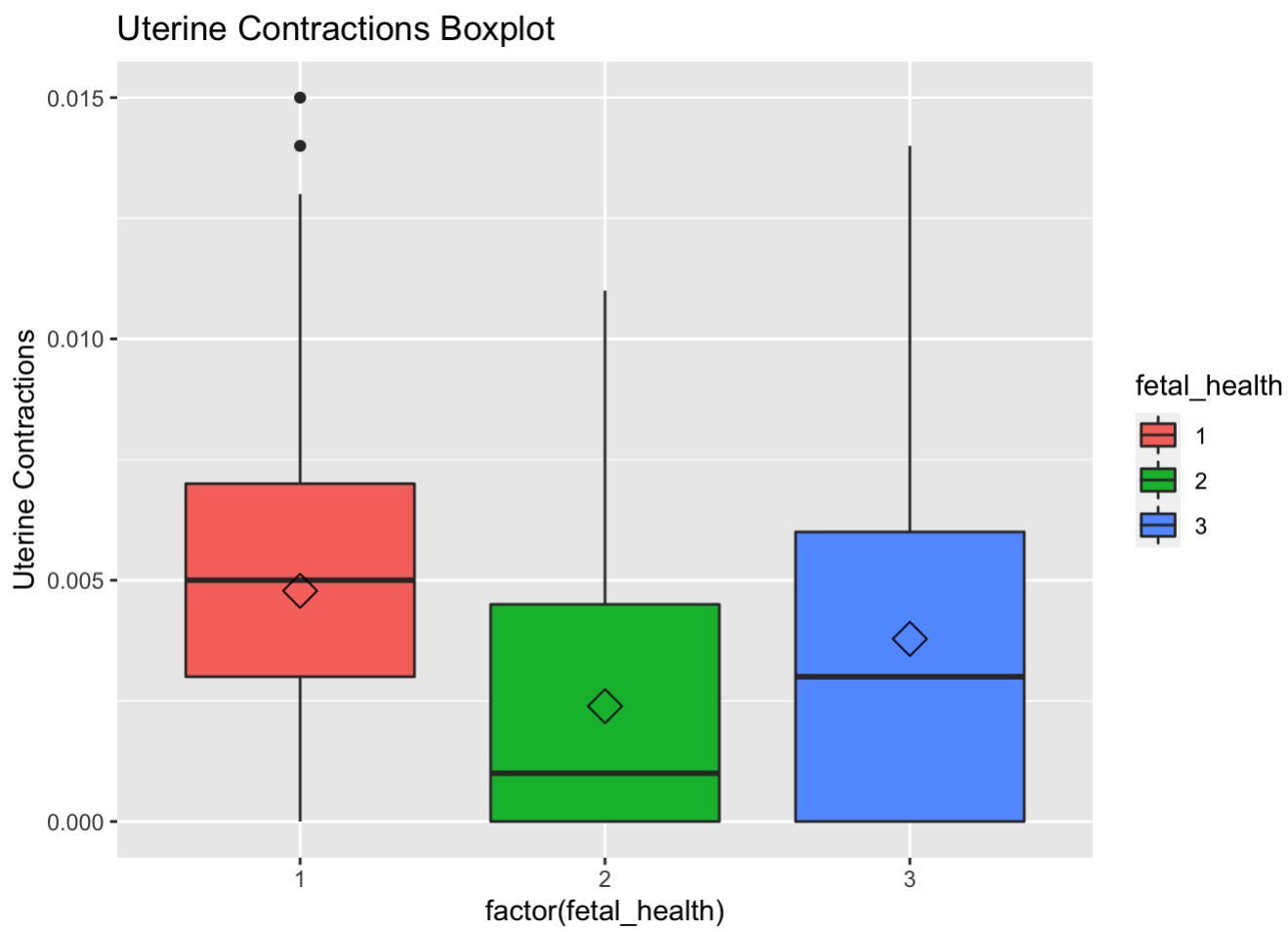
uterine_contractions

```
hist(fetal_df$uterine_contractions,
     main="Histogram for uterine_contractions",
     xlab="uterine_contractions",
     border="black",
     col="wheat")
```



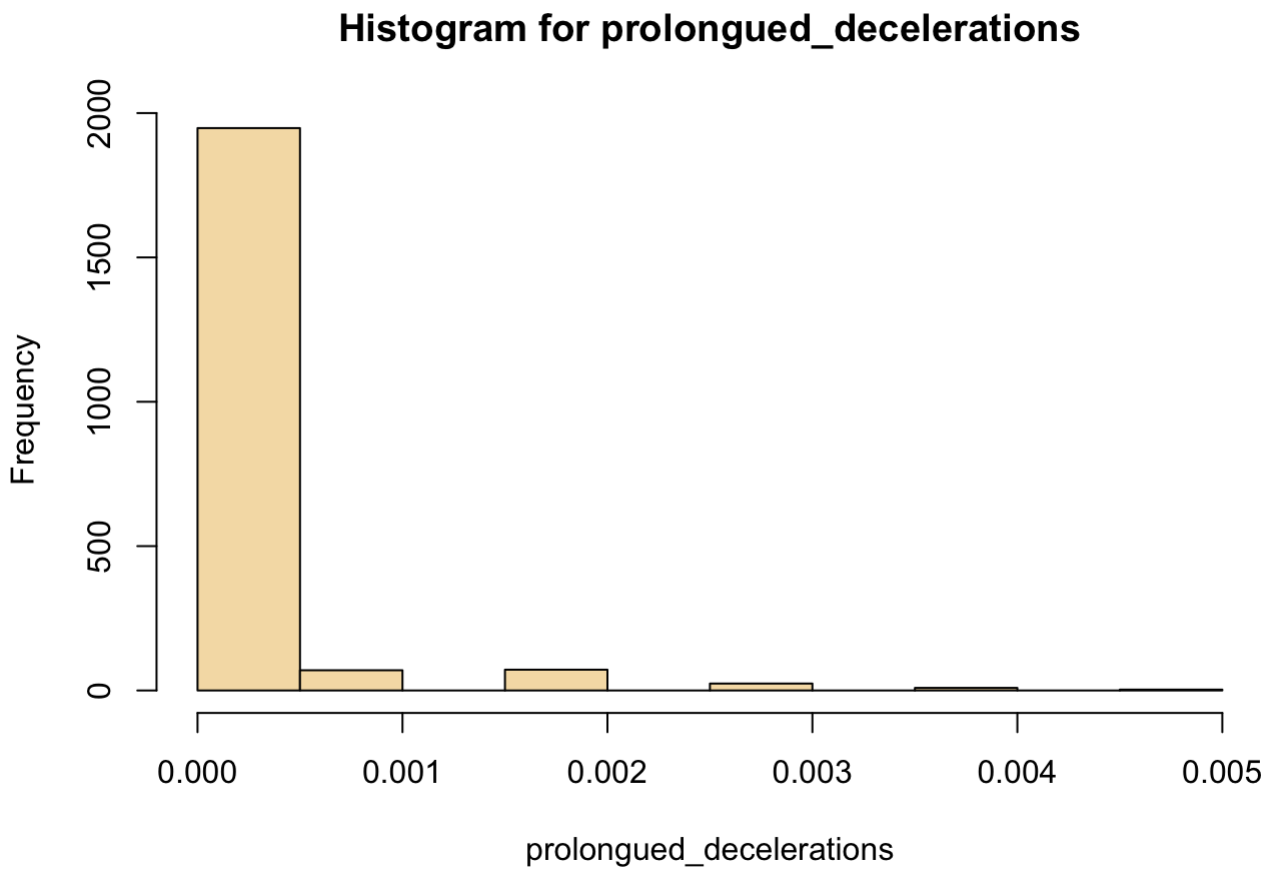

```
temp_fetal_df <- fetal_df
temp_fetal_df$fetal_health <- factor(temp_fetal_df$fetal_health)

uterine_contractions_boxplot <- ggplot(data = temp_fetal_df, aes(x = factor(fetal_health), y = uterine_contractio
ns))
uterine_contractions_boxplot + geom_boxplot(aes(fill = fetal_health)) + ylab("Uterine Contractions") + ggtitle("U
terine Contractions Boxplot") + stat_summary(fun=mean, geom = "point", shape = 5, size = 4)
```



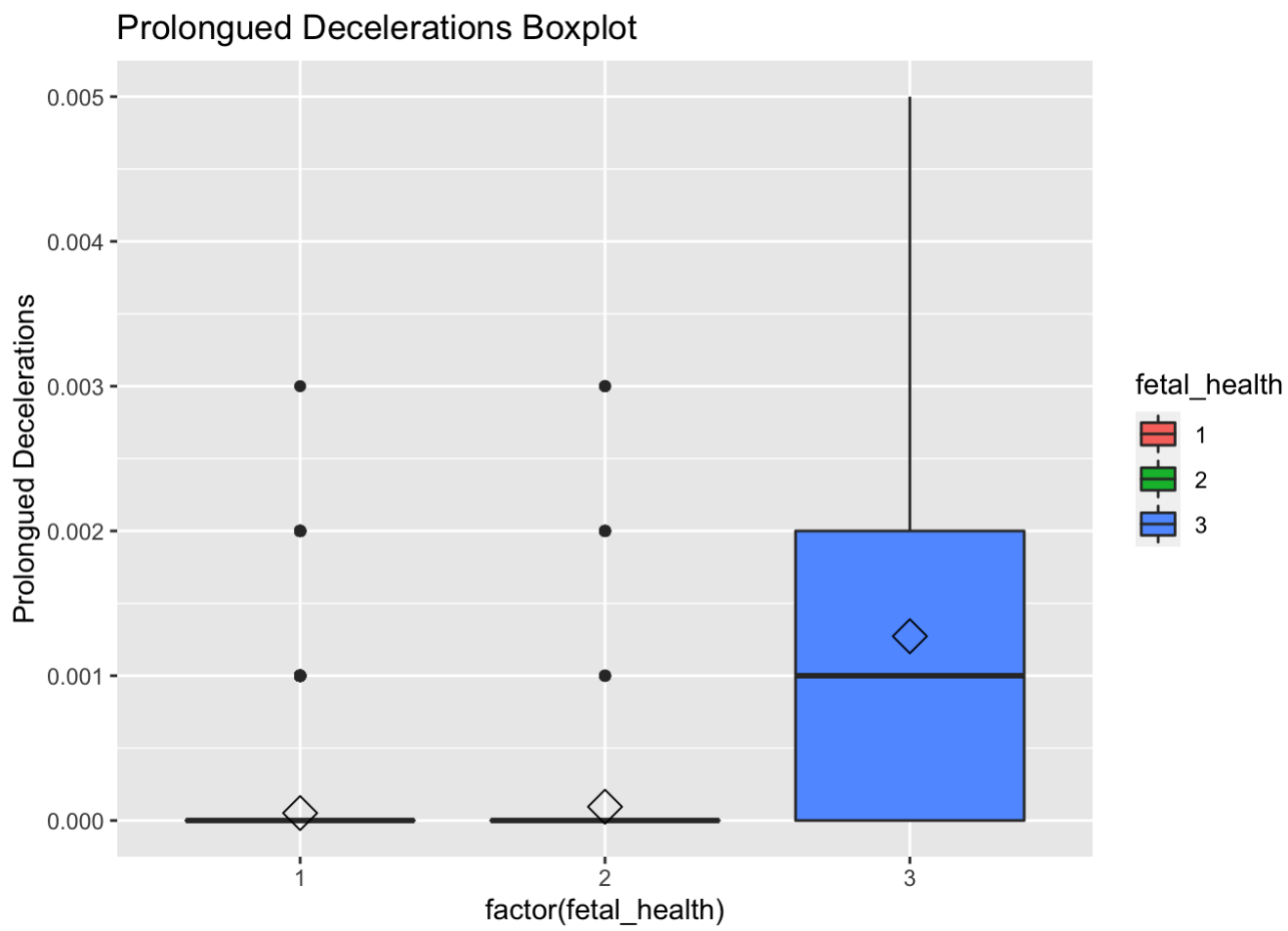
prolongued_decelerations

```
hist(fetal_df$prolongued_decelerations,
     main="Histogram for prolongued_decelerations",
     xlab="prolongued_decelerations",
     border="black",
     col="wheat")
```



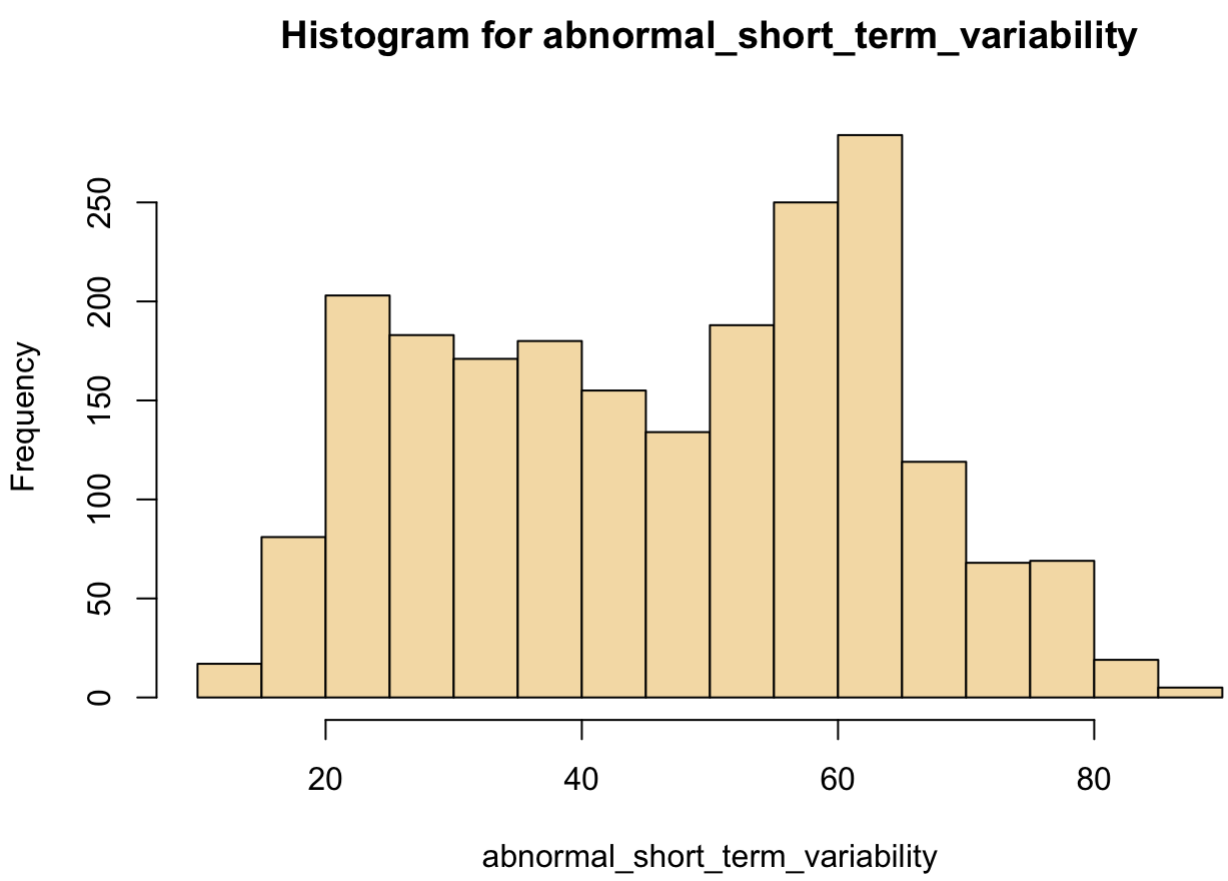

```
temp_fetal_df <- fetal_df
temp_fetal_df$fetal_health <- factor(temp_fetal_df$fetal_health)

prolongued_decelerations_boxplot <- ggplot(data = temp_fetal_df, aes(x = factor(fetal_health), y = prolongued_dec
elerations))
prolongued_decelerations_boxplot + geom_boxplot(aes(fill = fetal_health)) + ylab("Prolongued Decelerations") + gg
title("Prolongued Decelerations Boxplot") + stat_summary(fun=mean, geom = "point", shape = 5, size = 4)
```



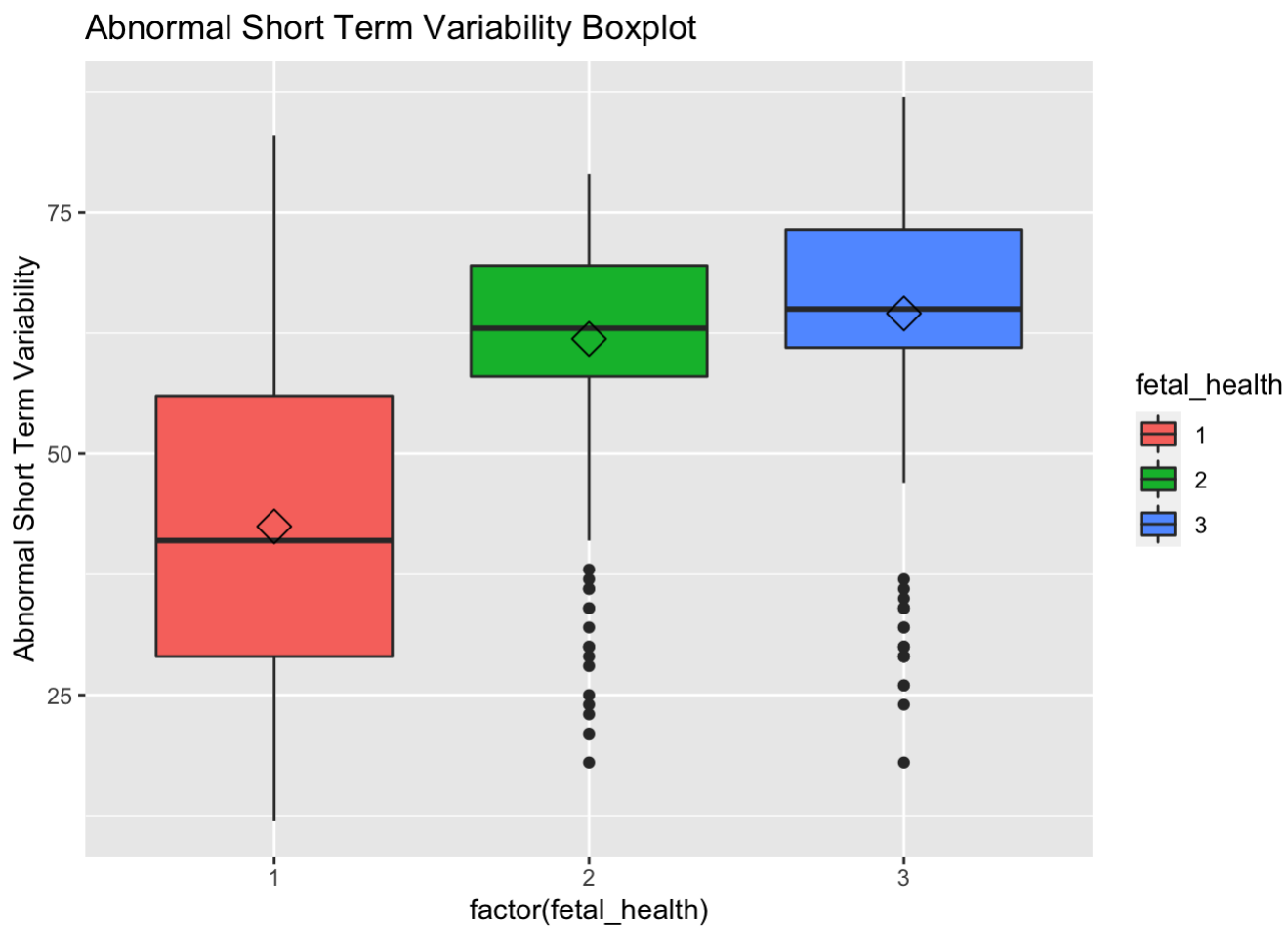
abnormal_short_term_variability

```
hist(fetal_df$abnormal_short_term_variability,
     main="Histogram for abnormal_short_term_variability",
     xlab="abnormal_short_term_variability",
     border="black",
     col="wheat")
```



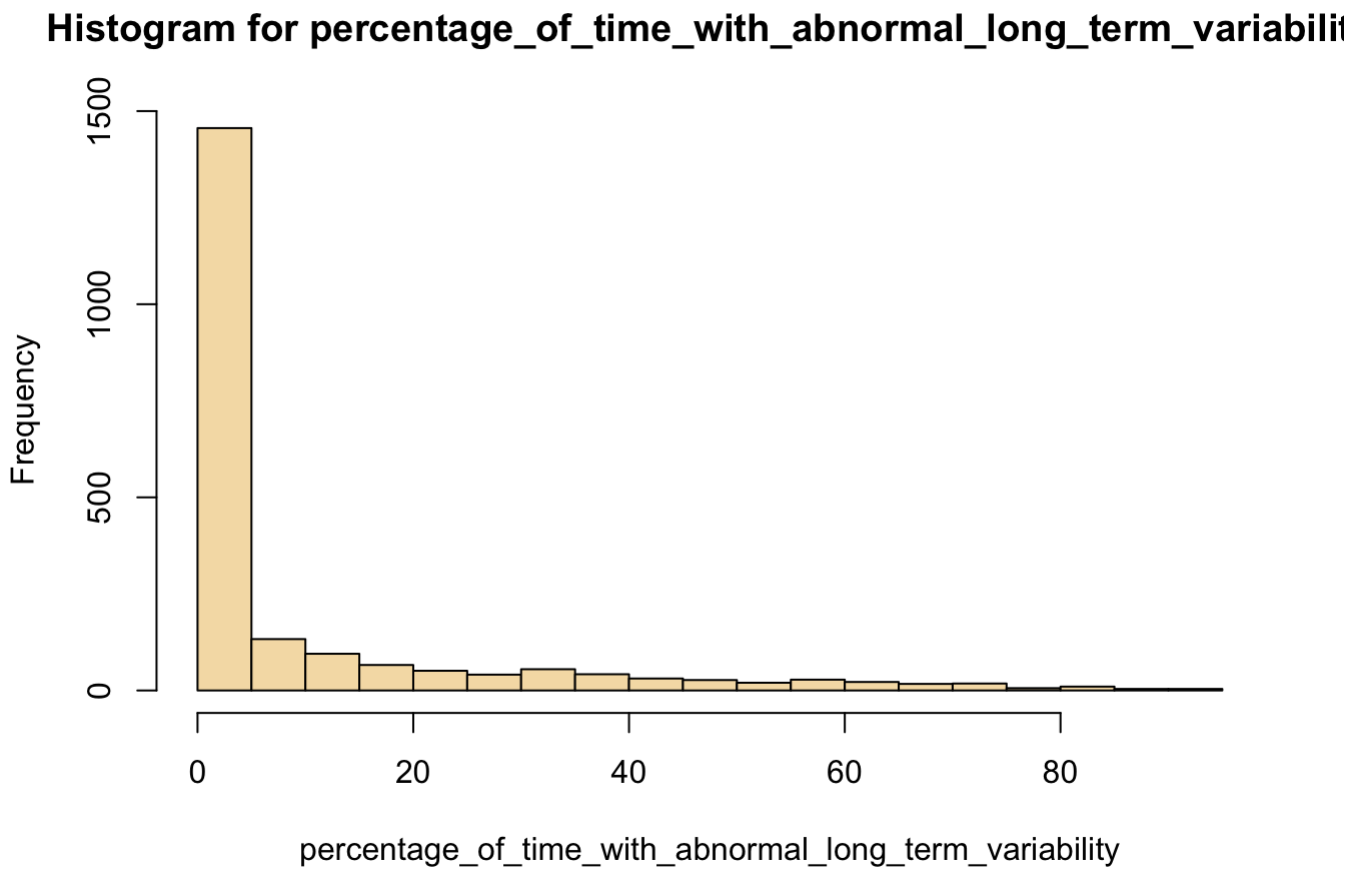

```
temp_fetal_df <- fetal_df
temp_fetal_df$fetal_health <- factor(temp_fetal_df$fetal_health)

abnormal_short_term_variability_boxplot <- ggplot(data = temp_fetal_df, aes(x = factor(fetal_health), y = abnorma
l_short_term_variability))
abnormal_short_term_variability_boxplot + geom_boxplot(aes(fill = fetal_health)) + ylab("Abnormal Short Term Vari
ability") + ggtitle("Abnormal Short Term Variability Boxplot") + stat_summary(fun=mean, geom = "point", shape = 5
, size = 4)
```



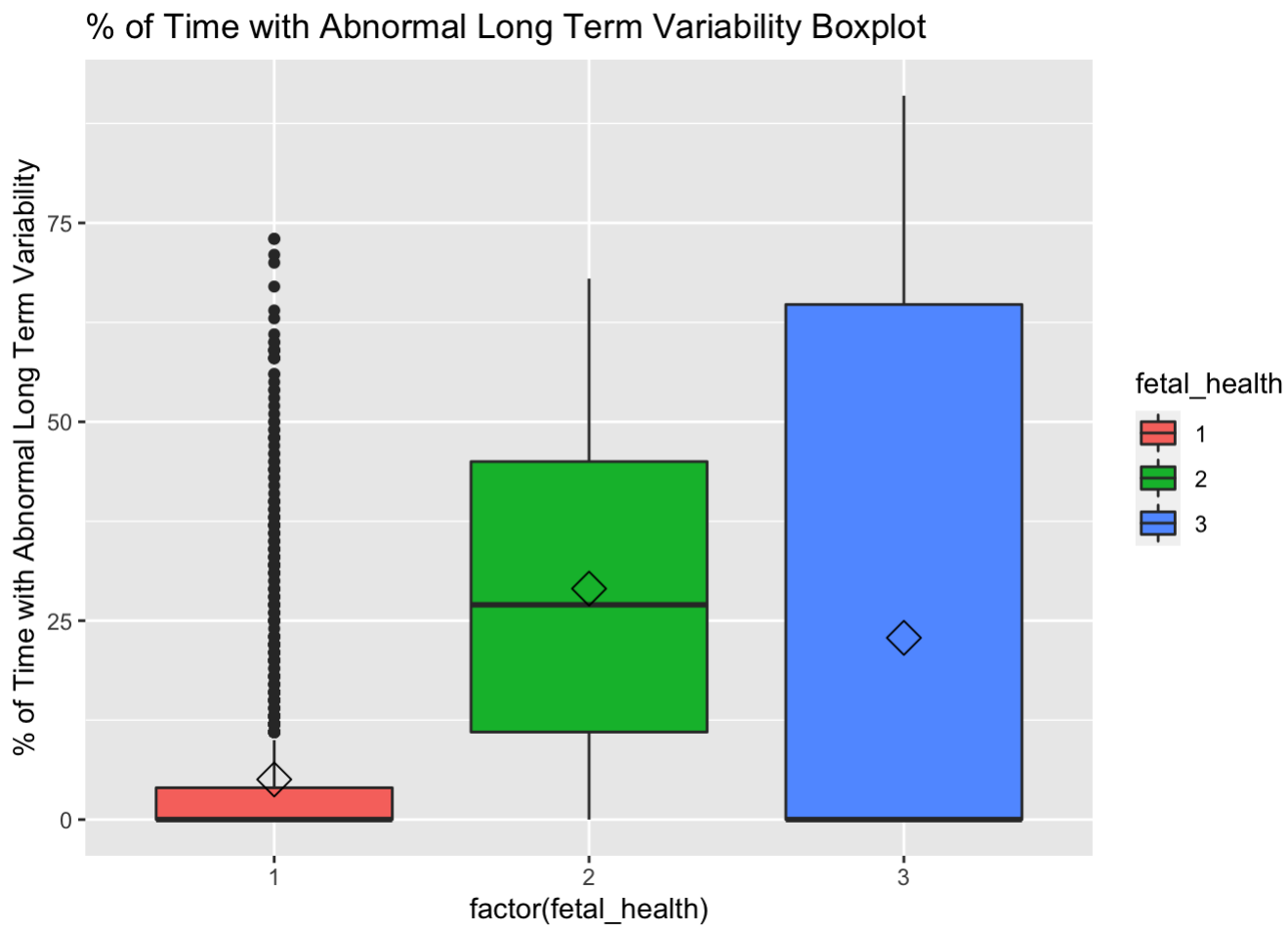
percentage_of_time_with_abnormal_long_term_variability

```
hist(fetal_df$percentage_of_time_with_abnormal_long_term_variability,
     main="Histogram for percentage_of_time_with_abnormal_long_term_variability",
     xlab="percentage_of_time_with_abnormal_long_term_variability",
     border="black",
     col="wheat")
```



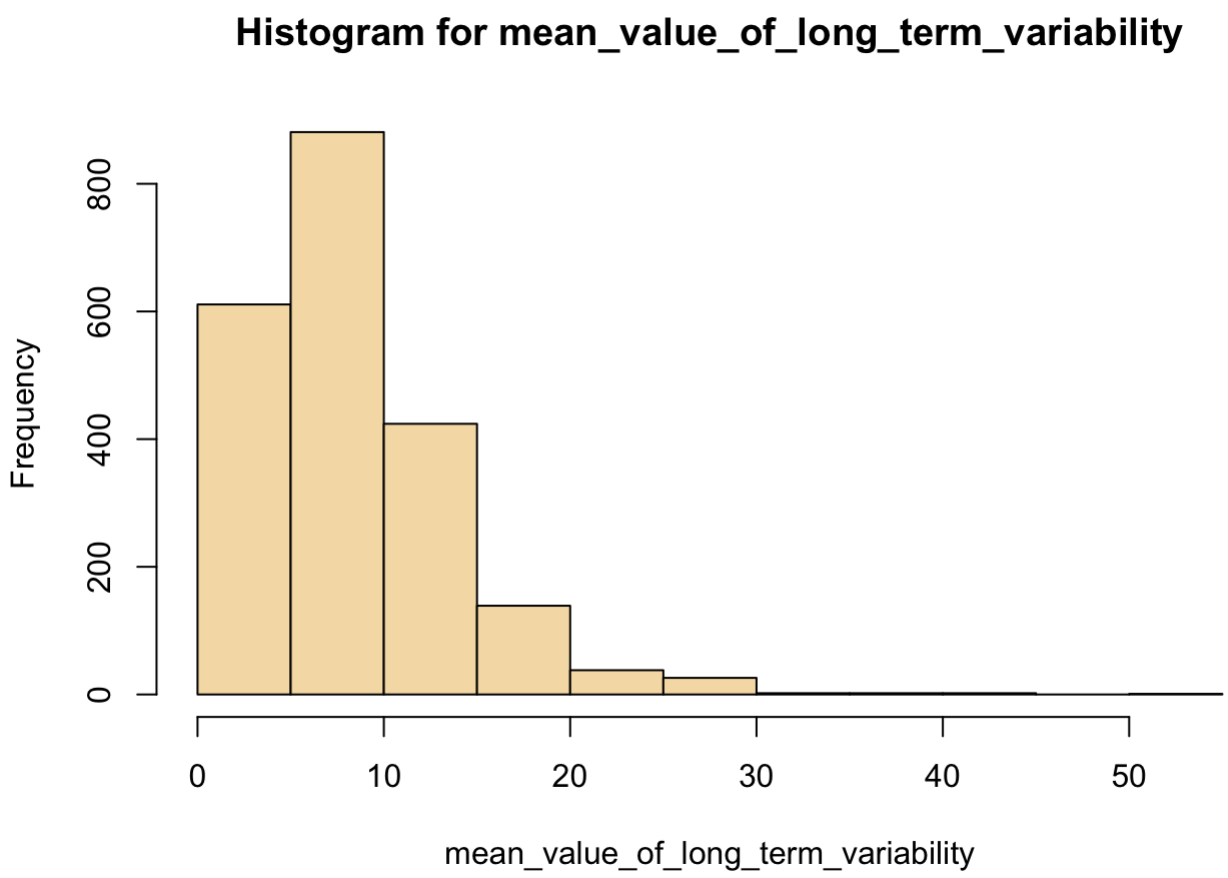

```
temp_fetal_df <- fetal_df
temp_fetal_df$fetal_health <- factor(temp_fetal_df$fetal_health)

percentage_of_time_with_abnormal_long_term_variability_boxplot <- ggplot(data = temp_fetal_df, aes(x = factor(fet
al_health), y = percentage_of_time_with_abnormal_long_term_variability))
percentage_of_time_with_abnormal_long_term_variability_boxplot + geom_boxplot(aes(fill = fetal_health)) + ylab("%
of Time with Abnormal Long Term Variability") + ggtitle("% of Time with Abnormal Long Term Variability Boxplot")
 + stat_summary(fun=mean, geom = "point", shape = 5, size = 4)
```



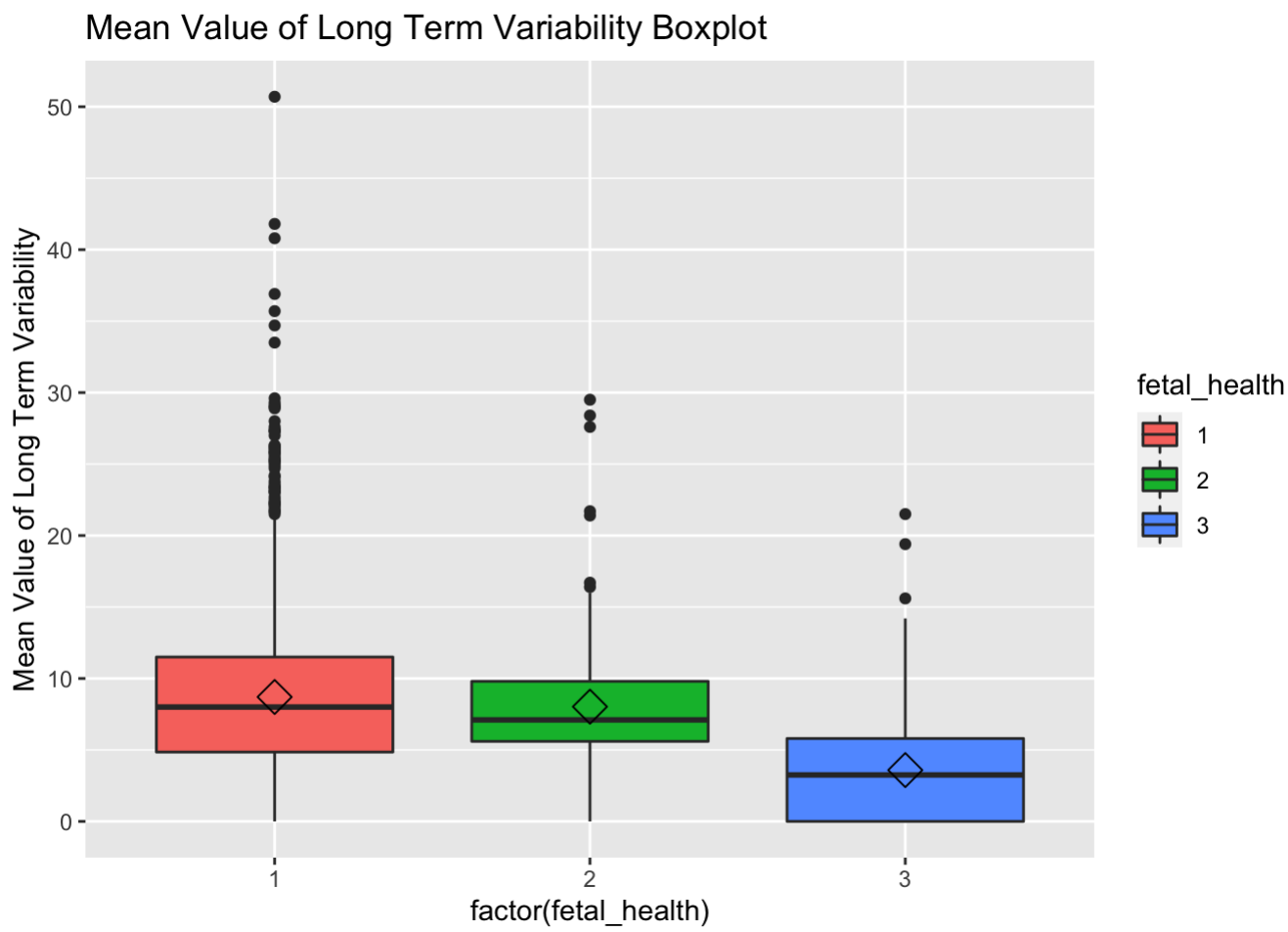
mean_value_of_long_term_variability

```
hist(fetal_df$mean_value_of_long_term_variability,
     main="Histogram for mean_value_of_long_term_variability",
     xlab="mean_value_of_long_term_variability",
     border="black",
     col="wheat")
```



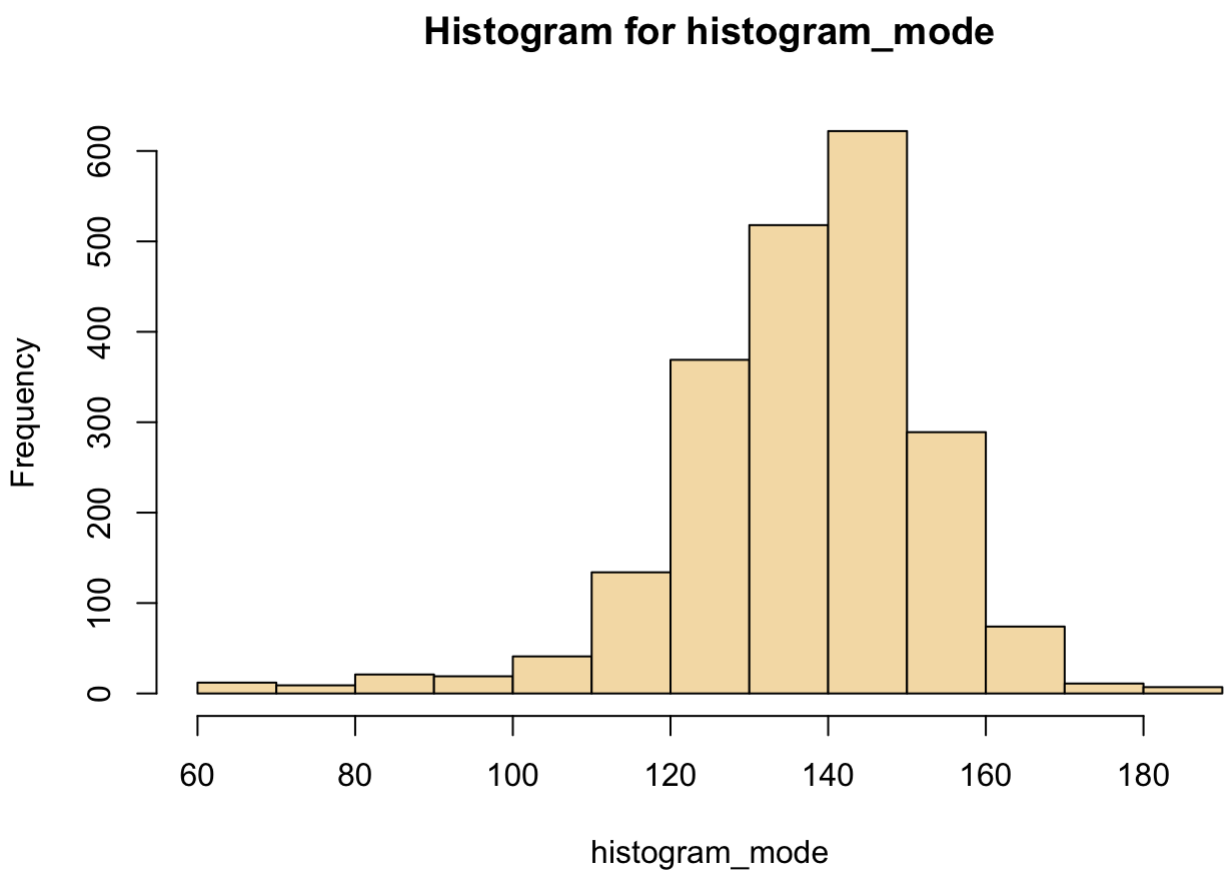

```
temp_fetal_df <- fetal_df
temp_fetal_df$fetal_health <- factor(temp_fetal_df$fetal_health)

mean_value_of_long_term_variability_boxplot <- ggplot(data = temp_fetal_df, aes(x = factor(fetal_health), y = mea
n_value_of_long_term_variability))
mean_value_of_long_term_variability_boxplot + geom_boxplot(aes(fill = fetal_health)) + ylab("Mean Value of Long T
erm Variability") + ggtitle("Mean Value of Long Term Variability Boxplot") + stat_summary(fun=mean, geom = "poin
t", shape = 5, size = 4)
```



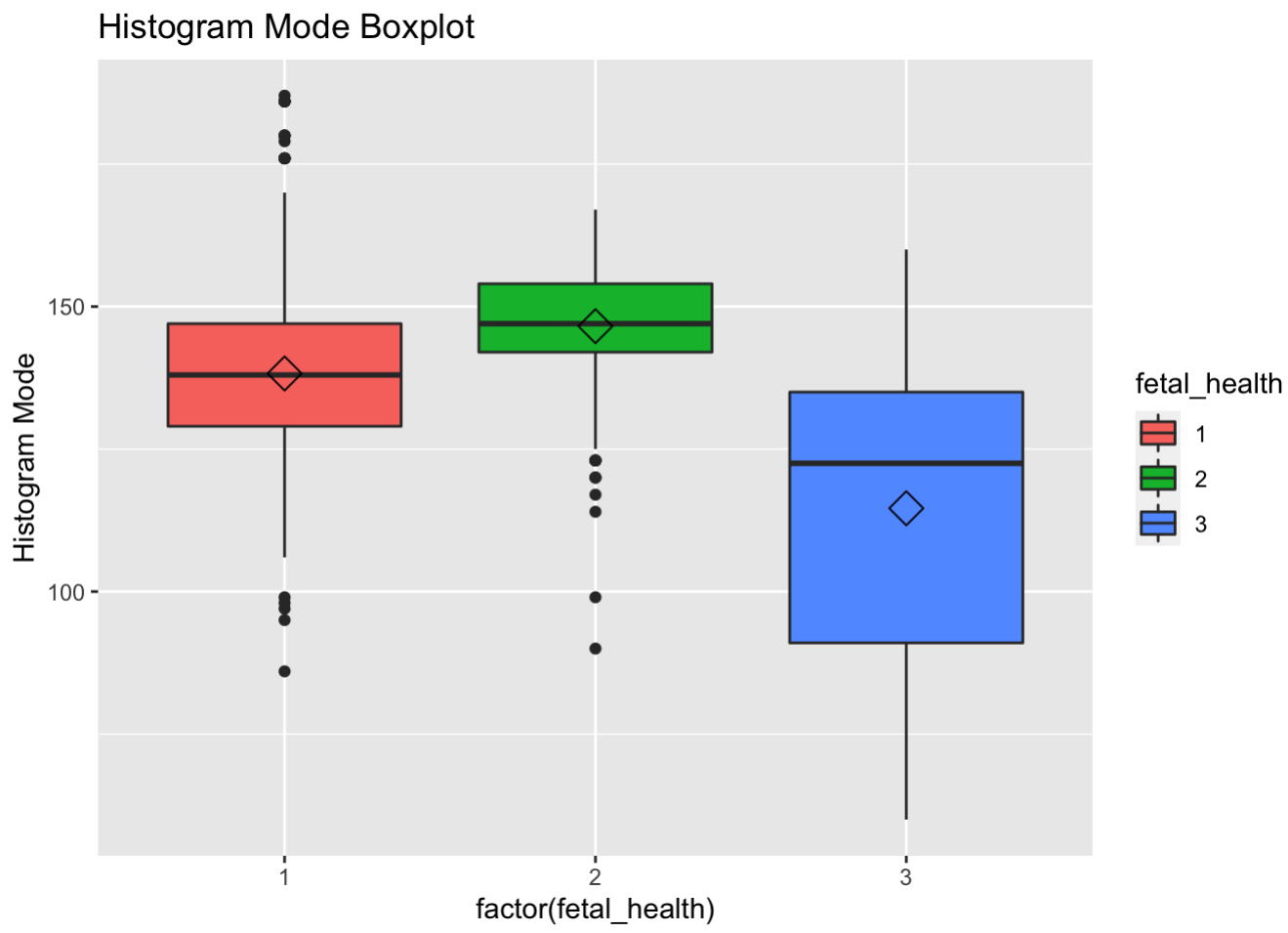
histogram_mode

```
hist(fetal_df$histogram_mode,
     main="Histogram for histogram_mode",
     xlab="histogram_mode",
     border="black",
     col="wheat")
```



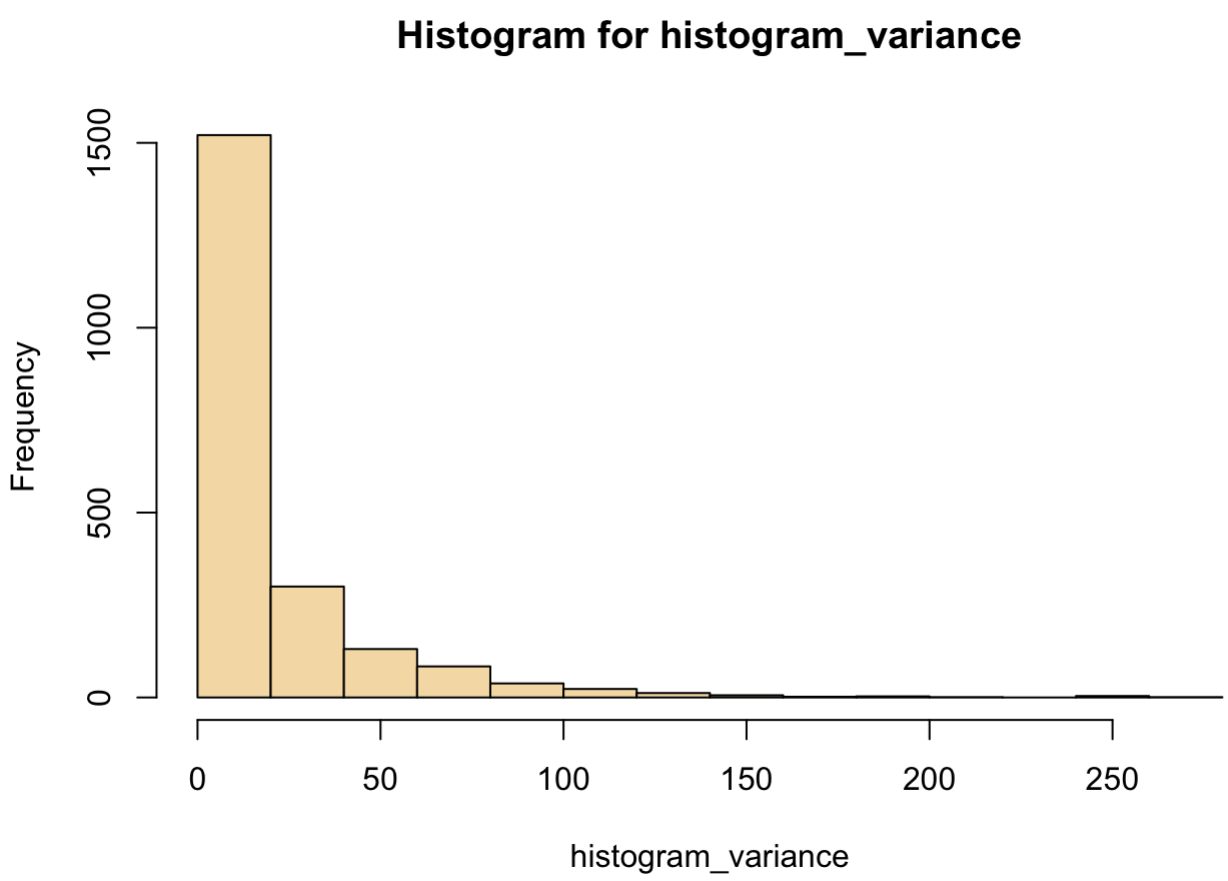

```
temp_fetal_df <- fetal_df
temp_fetal_df$fetal_health <- factor(temp_fetal_df$fetal_health)

histogram_mode_boxplot <- ggplot(data = temp_fetal_df, aes(x = factor(fetal_health), y = histogram_mode))
histogram_mode_boxplot + geom_boxplot(aes(fill = fetal_health)) + ylab("Histogram Mode") + ggtitle("Histogram Mod
e Boxplot") + stat_summary(fun=mean, geom = "point", shape = 5, size = 4)
```



histogram_variance

```
hist(fetal_df$histogram_variance,
     main="Histogram for histogram_variance",
     xlab="histogram_variance",
     border="black",
     col="wheat")
```




```
temp_fetal_df <- fetal_df
temp_fetal_df$fetal_health <- factor(temp_fetal_df$fetal_health)

histogram_variance_boxplot <- ggplot(data = temp_fetal_df, aes(x = factor(fetal_health), y = histogram_variance))
histogram_variance_boxplot + geom_boxplot(aes(fill = fetal_health)) + ylab("Histogram Variance") + ggtitle("Histo
gram Variance Boxplot") + stat_summary(fun=mean, geom = "point", shape = 5, size = 4)
```