Pima Diabetes Analysis

Isabella Chittumuri

10/08/2020

```
# Install packages
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(faraway)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.0     v stringr 1.4.0
## v tidyr   1.1.3     v forcats 0.5.1
## v readr   1.4.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Chapter 2 Exercise 2

The National Institute of Diabetes and Digestive and Kidney Diseases conducted a study on 768 adult female Pima Indians living near Phoenix. The purpose of the study was to investigate factors related to diabetes. The data may be found in the dataset pima.

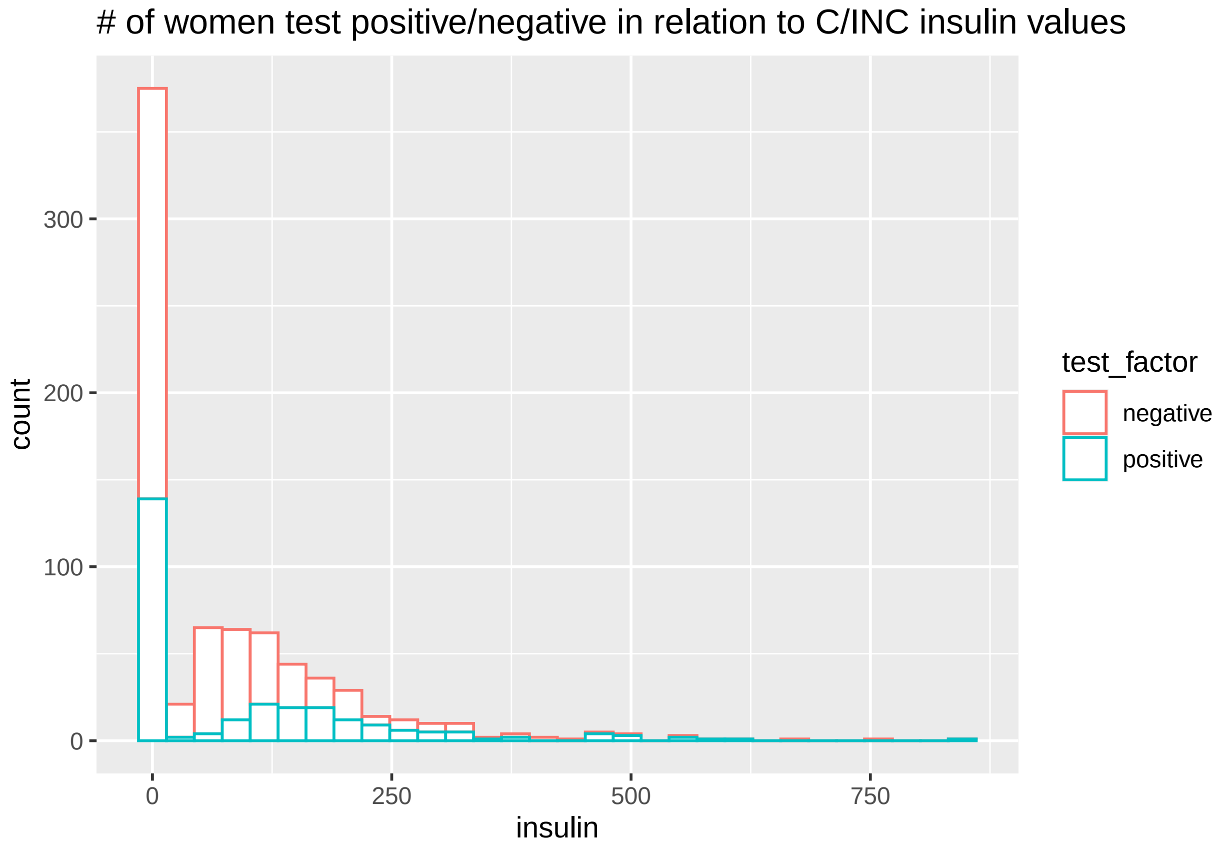
```
# Get dataset
data("pima")
?pima
summary(pima)
```

```
##     pregnant         glucose        diastolic        triceps     
##  Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00  
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00  
##  Median : 3.000   Median :117.0   Median : 72.00   Median :23.00  
##  Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54  
##  3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00  
##  Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00  
##     insulin          bmi           diabetes           age       
##  Min.   :  0.0   Min.   : 0.00   Min.   :0.0780   Min.   :21.00  
##  1st Qu.:  0.0   1st Qu.:27.30   1st Qu.:0.2437   1st Qu.:24.00  
##  Median : 30.5   Median :32.00   Median :0.3725   Median :29.00  
##  Mean   : 79.8   Mean   :31.99   Mean   :0.4719   Mean   :33.24  
##  3rd Qu.:127.2   3rd Qu.:36.60   3rd Qu.:0.6262   3rd Qu.:41.00  
##  Max.   :846.0   Max.   :67.10   Max.   :2.4200   Max.   :81.00  
##       test      
##  Min.   :0.000  
##  1st Qu.:0.000  
##  Median :0.000  
##  Mean   :0.349  
##  3rd Qu.:1.000  
##  Max.   :1.000  
```

(a) Create a factor version of the test results and use this to produce an interleaved histogram to show how the distribution of insulin differs between those testing positive and negative. Do you notice anything unbelievable about the plot?

```
# Change test 0,1 values into negative and positive
pima$test_factor <- ifelse(pima$test == 0, "negative", "positive")

# Interleaved histogram
pima %>%
  ggplot(aes(x = insulin, color = test_factor)) +
  geom_histogram(fill = "white") + ggtitle("# of women test positive/negative in relation to C/INC insul
```



This graph displays the number of women who tested positive/negative for diabetes in relation to their insulin levels, with complete and incomplete cases. This distribution is right-skewed for both positive and negative test results.

The reference range for normal 2-Hour serum insulin levels is 16-166 (mu U/ml). Anything above or below that range could be an indictor to problems with insulin production in the body, which can result in diabetes. It’s unbelievable to see that there are over 300 females who have insulin levels of 0 (mu U/ml) tested negative for diabetes.

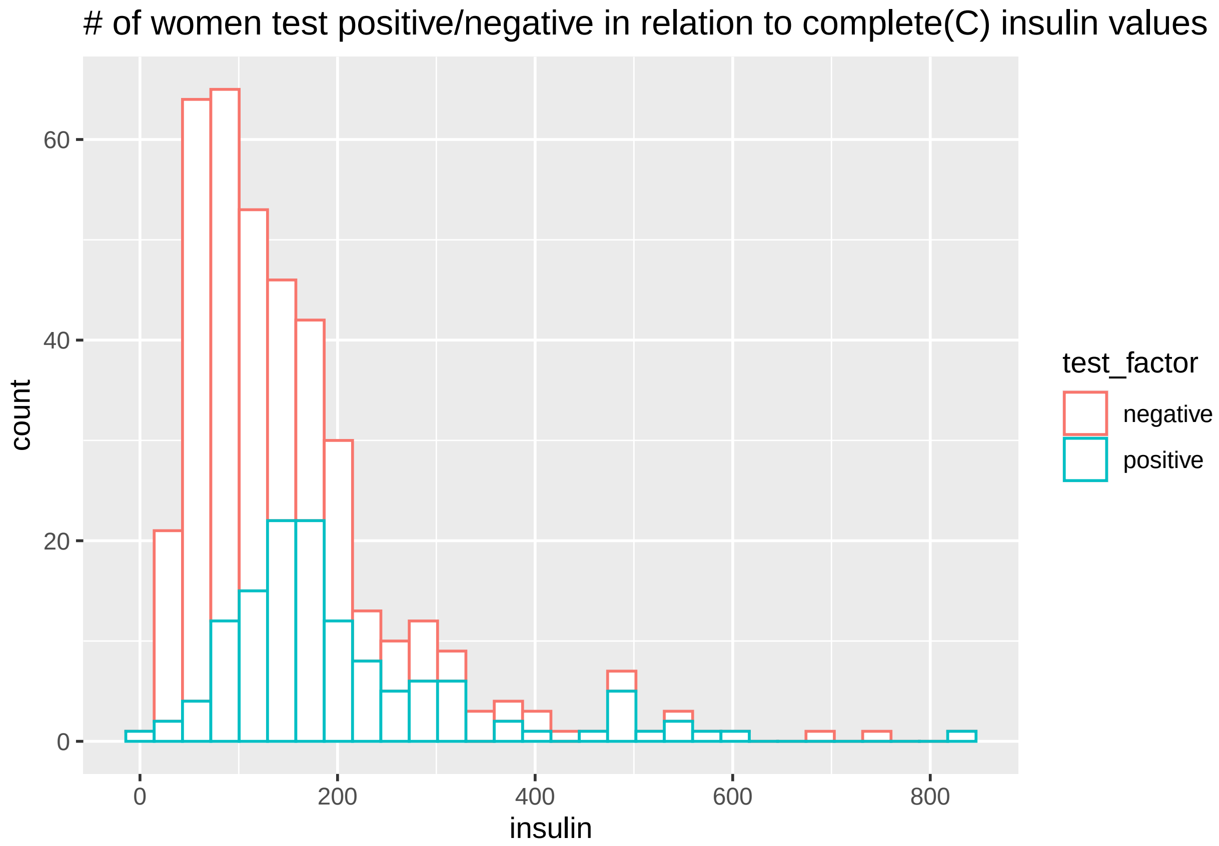
(b) Replace the zero values of insulin with the missing value code NA. Recreate the interleaved histogram plot and comment on the distribution.

```
# Change insulin 0 values into NA's
pima2 <- pima %>%
  mutate(insulin = na_if(insulin, "0"))

summary(pima2)
```

```
##     pregnant         glucose        diastolic         triceps
##  Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00
##  Median : 3.000   Median :117.0   Median : 72.00   Median :23.00
##  Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54
##  3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00
##  Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00
##
##     insulin            bmi           diabetes           age
##  Min.   : 14.00   Min.   : 0.00   Min.   :0.0780   Min.   :21.00
##  1st Qu.: 76.25   1st Qu.:27.30   1st Qu.:0.2437   1st Qu.:24.00
##  Median :125.00   Median :32.00   Median :0.3725   Median :29.00
##  Mean   :155.55   Mean   :31.99   Mean   :0.4719   Mean   :33.24
##  3rd Qu.:190.00   3rd Qu.:36.60   3rd Qu.:0.6262   3rd Qu.:41.00
##  Max.   :846.00   Max.   :67.10   Max.   :2.4200   Max.   :81.00
##  NA's   :374
##       test        test_factor
##  Min.   :0.000   Length:768
##  1st Qu.:0.000   Class :character
##  Median :0.000   Mode  :character
##  Mean   :0.349
##  3rd Qu.:1.000
##  Max.   :1.000
##
```

```
# Interleaved histogram w/o insulin missing values (NA)
pima2 %>%
  ggplot(aes(x = insulin, color = test_factor)) +
  geom_histogram(fill = "white") + ggtitle("# of women test positive/negative in relation to complete(C
```



This graph shows the relationship between women who tested positive/negative for diabetes and their insulin levels, with only complete cases (without NA's). The distribution is still right-skewed, suggesting that the median is a better measure of spread than the mean.

Looking at the summary output, dataset pima2 (w/o NA's) has a higher insulin median compared to that of dataset pima (w/ NA's), with a difference of 94.5 (mu U/ml).

(c) Replace the incredible zeros in other variables with the missing value code. Fit a model with the result of the diabetes test as the response and all the other variables as predictors. How many observations were used in the model fitting? Why is this less than the number of observations in the data frame.

```
# Change all incredible 0 predictor values into NA's
# Do not include insulin because already done^
# Do not include pregnant because 0 means never been pregnant, valid
pima2$glucose[pima2$glucose==0] <- NA
pima2$diastolic[pima2$diastolic==0] <- NA
pima2$triceps[pima2$triceps==0] <- NA
pima2$bmi[pima2$bmi==0] <- NA
pima2$diabetes[pima2$diabetes==0] <- NA
pima2$age[pima2$age==0] <- NA

# Take out all NA's from dataset
pima3 <- na.omit(pima2)

# Generalized linear model (GLM) w/ link 'logit' because data is skewed
lmod <- glm(test ~ pregnant + glucose + diastolic + triceps + insulin + bmi + diabetes + age, family = 

# Summary of lmod
summary(lmod)
```

```
##
## Call:
## glm(formula = test ~ pregnant + glucose + diastolic + triceps +
##     insulin + bmi + diabetes + age, family = binomial(link = "logit"),
##     data = pima3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7823  -0.6603  -0.3642   0.6409   2.5612
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.004e+01  1.218e+00  -8.246  < 2e-16 ***
## pregnant     8.216e-02  5.543e-02   1.482  0.13825
## glucose      3.827e-02  5.768e-03   6.635 3.24e-11 ***
## diastolic   -1.420e-03  1.183e-02  -0.120  0.90446
## triceps      1.122e-02  1.708e-02   0.657  0.51128
## insulin     -8.253e-04  1.306e-03  -0.632  0.52757
## bmi          7.054e-02  2.734e-02   2.580  0.00989 **
## diabetes     1.141e+00  4.274e-01   2.669  0.00760 **
## age          3.395e-02  1.838e-02   1.847  0.06474 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 344.02  on 383  degrees of freedom
## AIC: 362.02
##
## Number of Fisher Scoring iterations: 5
```

```
# Coefficients of lmod
lmod %>% broom::tidy()
```

```
## # A tibble: 9 x 5
##   term          estimate std.error statistic  p.value
##   <chr>            <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -10.0        1.22       -8.25  1.64e-16
## 2 pregnant      0.0822     0.0554      1.48  1.38e- 1
## 3 glucose       0.0383     0.00577     6.64  3.24e-11
## 4 diastolic    -0.00142    0.0118     -0.120 9.04e- 1
## 5 triceps       0.0112     0.0171      0.657 5.11e- 1
## 6 insulin      -0.000825   0.00131    -0.632 5.28e- 1
## 7 bmi           0.0705     0.0273      2.58  9.89e- 3
## 8 diabetes      1.14       0.427       2.67  7.60e- 3
## 9 age           0.0340     0.0184      1.85  6.47e- 2
```

```
# Number of observations in lmod
nobs(lmod, use.fallback=F)
```

```
## [1] 392
```

There are 392 observations used in this model. This is less than the number of observations in the original dataset because we omitted all missing values from each predictor, except pregnant.

(d) Refit the model but now without the insulin and triceps predictors. How many observations were used in fitting this model? Devise a test to compare this model with that in the previous question.

```
# GLM w/o insulin and triceps
lmod2 <- glm(test ~ pregnant + glucose + diastolic + bmi + diabetes + age, family = binomial(link = "lo

# Summary of lmod2
summary(lmod2)
```

```
##
## Call:
## glm(formula = test ~ pregnant + glucose + diastolic + bmi + diabetes +
##     age, family = binomial(link = "logit"), data = pima3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8832  -0.6537  -0.3704   0.6502   2.5763
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.9604796  1.1818764  -8.428  < 2e-16 ***
## pregnant     0.0840497  0.0550728   1.526 0.126971
## glucose      0.0364863  0.0049973   7.301 2.85e-13 ***
## diastolic   -0.0008002  0.0118034  -0.068 0.945949
## bmi          0.0785728  0.0215674   3.643 0.000269 ***
## diabetes     1.1492368  0.4250340   2.704 0.006854 **
## age          0.0346079  0.0181919   1.902 0.057121 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 344.88  on 385  degrees of freedom
## AIC: 358.88
##
## Number of Fisher Scoring iterations: 5
```

```
# Coefficients of lmod2
lmod %>% broom::tidy()
```

```
## # A tibble: 9 x 5
##   term         estimate std.error statistic  p.value
##   <chr>           <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -10.0       1.22       -8.25  1.64e-16
## 2 pregnant      0.0822    0.0554      1.48  1.38e- 1
## 3 glucose       0.0383    0.00577     6.64  3.24e-11
## 4 diastolic    -0.00142   0.0118     -0.120 9.04e- 1
## 5 triceps       0.0112    0.0171      0.657 5.11e- 1
## 6 insulin      -0.000825  0.00131    -0.632 5.28e- 1
## 7 bmi           0.0705    0.0273      2.58  9.89e- 3
## 8 diabetes      1.14      0.427       2.67  7.60e- 3
## 9 age           0.0340    0.0184      1.85  6.47e- 2
```

```
# Number of observations in lmod2
nobs(lmod2, use.fallback=F)
```

```
## [1] 392
```

```
# Anova test
anova(lmod, lmod2, test = "Chi")
```

```
## Analysis of Deviance Table
##
## Model 1: test ~ pregnant + glucose + diastolic + triceps + insulin + bmi +
##     diabetes + age
## Model 2: test ~ pregnant + glucose + diastolic + bmi + diabetes + age
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       383     344.02
## 2       385     344.88 -2 -0.85931   0.6507
```

There are 392 observations used in this model (lmod2), same as the previous model (lmod). Using the ANOVA test, we see that the p-value is greater than 0.05. This suggests that we can drop insulin and triceps as predictors during the fitting of the model.

(e) Use AIC to select a model. You will need to take account of the missing values. Which predictors are selected? How many cases are used in your selected model?

```
# AIC
AIC(lmod)
```

```
## [1] 362.0212
```

```
AIC(lmod2)
```

```
## [1] 358.8805
```

We selected the second model, with the lowest AIC value of 358.8805. This value means that the second model generated a better fit to the data than the first one. The predictors used were pregnant, glucose, diastolic, bmi, diabetes, and age. In our selected model, we have 392 cases.

(f) Create a variable that indicates whether the case contains a missing value. Use this variable as a predictor of the test result. Is missingness associated with the test result? Refit the selected model, but now using as much of the data as reasonable. Explain why it is appropriate to do this.

```
# Any column w/ NA will return 0, complete cases will return 1
# Pima2 is the dataset b4 omitted NA's
pima2$check <-  as.integer(complete.cases(pima2))

# Check 0/1 will return FALSE/TRUE
pima2$complete <- factor(pima2$check)
levels(pima2$complete) <- c("FALSE", "TRUE"); levels(pima2$complete)
```

```
## [1] "FALSE" "TRUE"
```

```
# GLM w/ test as response; check as predictor
lmod_check <- glm(test ~ check, family = binomial(link = "logit"), data = pima2)

# Summary of lmod_check
summary(lmod_check)
```

```
##
## Call:
## glm(formula = test ~ check, family = binomial(link = "logit"),
##     data = pima2)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9564  -0.9564  -0.8977   1.4159   1.4857
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.5450     0.1070  -5.094 3.51e-07 ***
## check        -0.1558     0.1515  -1.028    0.304
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 993.48  on 767  degrees of freedom
## Residual deviance: 992.43  on 766  degrees of freedom
## AIC: 996.43
##
## Number of Fisher Scoring iterations: 4
```

```
# Number of observations in lmod_check
nobs(lmod_check, use.fallback=F)
```

```
## [1] 768
```

```
# Coefficients of lmod_check
lmod_check %>% broom::tidy()
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic     p.value
##   <chr>          <dbl>     <dbl>     <dbl>       <dbl>
## 1 (Intercept)   -0.545     0.107     -5.09 0.000000351
## 2 check         -0.156     0.152     -1.03 0.304
```

The variable check is a column created in pima2 that only have values of 0 (incomplete cases) and 1 (complete cases). Looking at the summary of this model, the intercept has a very significant p-value of 3.51e-07, while check has an insignificant p-value of 0.304. This outcome shows that incomplete cases, or missingness, are not associated with the test result.

```
# Refit selected model (lmod2)
lmod2_step <- step(lmod2, trace=1)
```

```
## Start:  AIC=358.88
## test ~ pregnant + glucose + diastolic + bmi + diabetes + age
##
##             Df Deviance    AIC
## - diastolic  1   344.89 356.89
## <none>           344.88 358.88
## - pregnant   1   347.23 359.23
## - age        1   348.63 360.63
## - diabetes   1   352.65 364.65
## - bmi        1   358.88 370.88
## - glucose    1   411.55 423.55
```

```
##
## Step:  AIC=356.89
## test ~ pregnant + glucose + bmi + diabetes + age
##
##            Df Deviance    AIC
## <none>          344.89 356.89
## - pregnant  1   347.23 357.23
## - age       1   348.72 358.72
## - diabetes  1   352.72 362.72
## - bmi       1   360.44 370.44
## - glucose   1   411.85 421.85
```

```
# GLM w/ test as response; pregnant, glucose, bmi, diabetes, & age as predictors
lmod3 <- glm(test ~ pregnant + glucose + bmi + diabetes + age, family = binomial(link = "logit"), data =

# Summary of lmod3
summary(lmod3)
```

```
##
## Call:
## glm(formula = test ~ pregnant + glucose + bmi + diabetes + age,
##     family = binomial(link = "logit"), data = pima3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8827  -0.6535  -0.3694   0.6521   2.5814
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.992080   1.086866  -9.193  < 2e-16 ***
## pregnant     0.083953   0.055031   1.526 0.127117
## glucose      0.036458   0.004978   7.324 2.41e-13 ***
## bmi          0.078139   0.020605   3.792 0.000149 ***
## diabetes     1.150913   0.424242   2.713 0.006670 **
## age          0.034360   0.017810   1.929 0.053692 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 344.89  on 386  degrees of freedom
## AIC: 356.89
##
## Number of Fisher Scoring iterations: 5
```

```
# Number of observations in lmod3
nobs(lmod3, use.fallback=F)
```

```
## [1] 392
```

```
# Coefficients of lmod3
lmod3 %>% broom::tidy()
```

```
## # A tibble: 6 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  -9.99     1.09        -9.19 3.80e-20
## 2 pregnant      0.0840   0.0550       1.53 1.27e- 1
## 3 glucose       0.0365   0.00498      7.32 2.41e-13
## 4 bmi           0.0781   0.0206       3.79 1.49e- 4
## 5 diabetes      1.15     0.424        2.71 6.67e- 3
## 6 age           0.0344   0.0178       1.93 5.37e- 2
```

After using the step function, we see that the best linear model, with the lowest AIC value of 356.89, uses pregnant, glucose, bmi, diabetes, and age as predictors of the test result. This is appropriate because all the other predictors have little to no significance to the model given the ones already included.

(g) Using the last fitted model of the previous question, what is the difference in the odds of testing positive for diabetes for a woman with a BMI at the first quartile compared with a woman at the third quartile, assuming that all other factors are held constant? Give a confidence interval for this difference.

```
# Coefficients of lmod3
coef(lmod3)
```

```
## (Intercept)    pregnant     glucose         bmi    diabetes         age
## -9.99207971  0.08395301  0.03645776  0.07813866  1.15091285  0.03436036
```

```
# Quartile values
quantile(pima3$bmi, c(.25, .75))
```

```
##  25%  75%
## 28.4 37.1
```

```
quant_diff <- 37.1 - 28.4; quant_diff
```

```
## [1] 8.7
```

```
# Logodds
x1 <- 1
x2 <- 1
x3 <- 1
x4 <- 1
x5 <- 1

logodd_1 = -9.99207971 + 0.08395301 * x1 + 0.03645776 * x2 + 0.07813866 * x3 + 1.15091285 * x4 + 0.0343
```

```
## [1] -8.608257
```

```
x3_d <- x3 + quant_diff

logodd_2 = -9.99207971 + 0.08395301 * x1 + 0.03645776 * x2 + 0.07813866 * x3_d + 1.15091285 * x4 + 0.03
```

```
## [1] -7.928451
```

```
# Odds
odd1 <- exp(logodd_1); odd1
```

```
## [1] 0.0001825919
```

```
odd2 <- exp(logodd_2); odd2
```

```
## [1] 0.0003603443
```

```
# Odds ratio
odds_ratio <- odd2/odd1; odds_ratio
```

```
## [1] 1.973496
```

```
# check
odds_ratio_check <- exp(0.07813866 * quant_diff); odds_ratio_check
```

```
## [1] 1.973496
```

We calculated logodd_1 by adding the estimates of our predictors from the last fitted model. Then we exponentiated this value to get odd1, the odds of a positive test result for diabetes. The value for odd1 is 1.826e-4

The difference between the first quartile BMI value and the third quartile BMI value is 8.25. We used this number to alter the BMI estimate of logodd_2. Then we exponentiated logodd_2 to get odd2, which is 3.603e-4. Then to calculate the odds ratio, we divided odd2 by odd1, giving us a value of 1.974. This value is the increase in the odds of getting a positive test result going from the first BMI quartile to the third BMI quartile.

```
# 95% Confidence Interval for odds ratio

# Bmi standard error (se) of lmod3
se <- coef(summary(lmod3))[, 2]
bmi_se <- se[4]

# Calculate log(OR) because confidence interval (CI) is calculated in a linear scale
log_odds_ratio <- log(odds_ratio); log_odds_ratio
```

```
## [1] 0.6798063
```

```
# Calculate exp(CI) to get CI in an odds ratio scale
CI_lower <- exp(log_odds_ratio - (1.96 * bmi_se * quant_diff)); CI_lower
```

```
##      bmi
## 1.388806
```

```
CI_upper <- exp(log_odds_ratio + (1.96 * bmi_se * quant_diff)); CI_upper
```

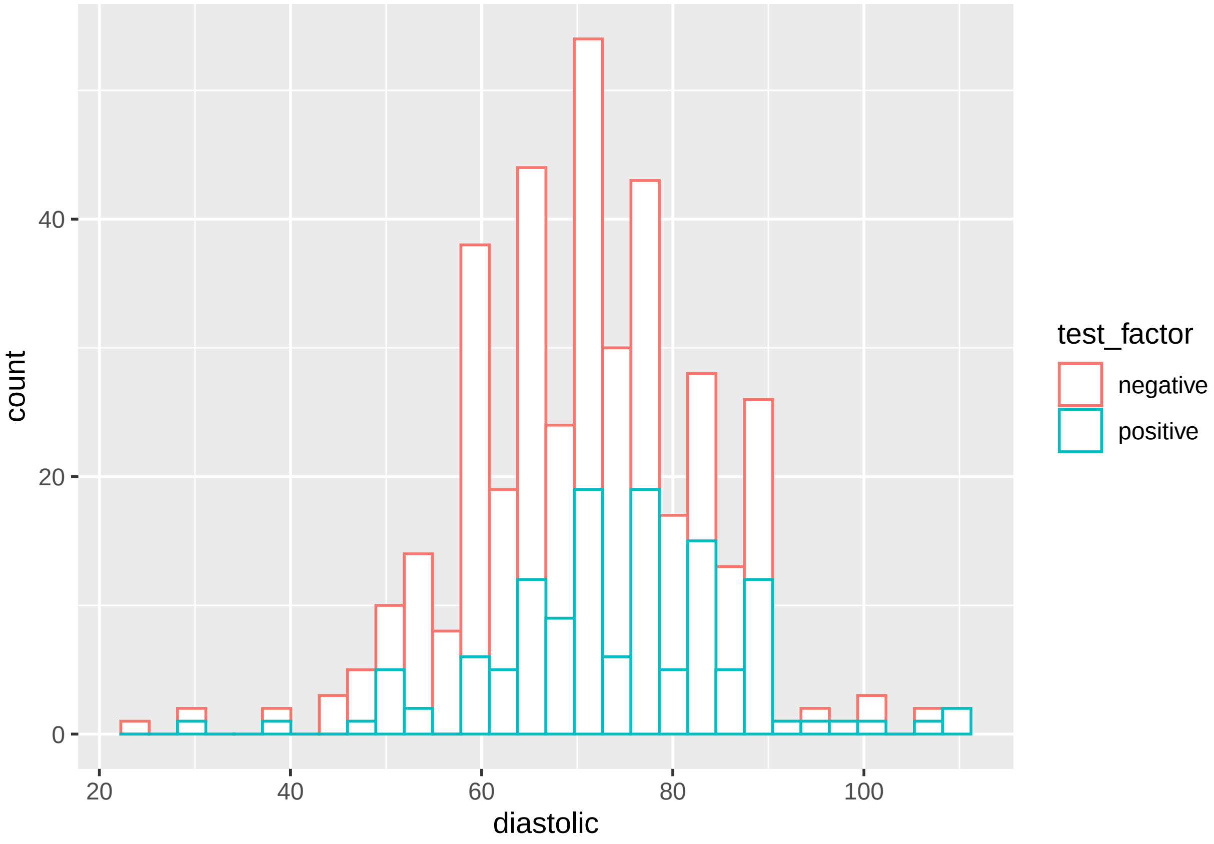
```
##      bmi
## 2.804341
```

If we repeat the experiment, we expect, on average, that 95% of the Confidence Interval (CI) contains the true value of the odds ratio. The lower CI limit is 1.389, while the upper CI limit is 2.804. This interval is very close in range to the calculated odds ratio, which was 1.974. This is all to say that 95% of the CI contains the true value of the increase in the odds ratio of getting a positive test result, going from the first BMI quartile to the third BMI quartile.

(h) Do women who test positive have higher diastolic blood pressures? Is the diastolic blood pressure significant in the regression model? Explain the distinction between the two questions and discuss why the answers are only apparently contradictory.

```
# Interleaved histogram for diastolic and test_factor
pima3 %>%
  ggplot(aes(x = diastolic, color = test_factor)) +
  geom_histogram(fill = "white")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
# Create two datasets, one only positive test results and the other only negative test results
pima3_pos <- subset(pima3, pima3$test == 1)
pima3_neg <- subset(pima3, pima3$test == 0)

summary(pima3_pos$diastolic)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   30.00   66.50   74.00   74.08   82.00  110.00
```

```
summary(pima3_neg$diastolic)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   24.00   60.00   70.00   68.97   76.00  106.00
```

Women who test positive have slightly higher diastolic blood pressure than women who test negative. Between the two, the absolute difference in mean value is 5.11. Diastolic blood pressure isn't significant to the regression model, because there is very little difference in statistical summary values between positive and negative test results.

The distinction between the two questions is that the first one is only looking for if women who test positive have higher diastolic blood pressure than those who test negative, which is yes. But it doesn't take into account what the second question does, which is if this difference is substantial enough to have an impact on the regression model, which is no.

The answers are only apparently contradictory if answered in a yes or no format. However, when you look at the minor numerical differences resulting from the first question, it actually supports the answer in the second.