Math 437 - Midterm Exam 1

Mason Huynh

Due March 13, 2023

Exam Rules

1. Solutions to this exam must be uploaded to Canvas by 11:59 PM on Monday, March 13. **NO EXTENSIONS WILL BE GRANTED**. Do not ask for them.
2. You should submit *both* this .Rmd file with your solutions included *and* the knitted pdf file. If you are having trouble knitting, knit to HTML and print/save the html file to pdf.
3. **Exams must be done individually**. Do not confer with anyone other than me about the exam.
4. You may consult any reference materials available to you, including the required and recommended textbooks, your course notes/code you have written, and anything on Canvas (such as my solutions). You are allowed to search the Internet or the library, but provide citations for outside references used.
5. Approach the exam from the point of view of a statistician, not a student. There may be multiple valid approaches on a problem, and any valid approach will be accepted so long as it is accompanied with appropriate justification and/or explanation.
6. I will provide a three-tiered hint system by e-mail:

- You may ask me to clarify something in a problem for no point penalty.
- You may give me pseudocode or buggy code, and ask me to implement/fix the code, for a partial point penalty (I will reply with how many points you will lose, then you confirm you want me to do it).
- You may ask me for my solution code, and I will provide it for a full point penalty (you will lose all Code points for that part). This may still be a good deal if you need the code to do the Explanation part or another part of the problem.

```r
# Packages you probably need
library(ggplot2)
library(dplyr)

# Please load any other packages you need either all in this chunk
# or in the chunk you need it for

# Data you need for the applied problem
SIS <- readr::read_csv("SIS.csv")
```

This exam is worth a total of **105** points, but is graded out of **100** points.

Conceptual Problem (16 points total)

Consider testing $H_0 : \mu_1 - \mu_2 = 0$. Make the following assumptions:

1. The response variable has been scaled such that $\sigma = 1$ in both groups.
2. We are performing a two-sided hypothesis test using $\alpha = 0.05$.

3. We are using the difference in sample means $\bar{x}_1 - \bar{x}_2$ as our test statistic, NOT a standardized t or z statistic.
4. The sample size in each group is large enough to assume that the distribution of $\bar{x}$ is normal in both groups.

Part a (2 pts)

Write a sentence explaining what the p-value represents in the context of this test, using the assumptions (1)-(4) above. You should start your sentence, "The p-value is the probability…".

The p-value is the probability of observing data in favor of the alternative hypothesis if the null hypothesis is true.

Part b (2 pts)

Suppose that we obtain a sample of 32 observations in each group and find that $\bar{x}_1 - \bar{x}_2 = 0.8$. In the chunk below, write code to compute (and print/output) the p-value.

The following hints may be useful:

1. The function for computing areas under the normal distribution curve is `pnorm`.
2. The first argument to `pnorm` should indicate the boundary value (e.g., test statistic value) that you are converting to an area under curve. The second and third arguments are the `mean` and `sd` of the normal distribution. The argument `lower.tail` takes the value `TRUE` if computing the area to the left of the boundary value or `FALSE` if computing the area to the right of the value.
3. Correctly explaining *what* you want R to do in order to compute the p-value will earn at least 1 pt even if you code it completely wrong.
4. If you get stuck, I would recommend reviewing assumptions (1)-(4) above, as well as your Math 338 notes.

```r
a <- 0
s <- 1
n <- 32
xbar <- 0.8
2*(1-pnorm(xbar, mean=a, sd=s/sqrt(n)))
```

```
## [1] 6.025761e-06
```

Part c (2 pts)

Based on the p-value you computed in Part (b), is it possible to make a Type I Error? Is it possible to make a Type II Error? Explain your reasoning.

The following hints may be useful:

1. You can still earn full credit even if you computed the wrong p-value in Part (b).
2. If your code in Part (b) does not run, just pick an arbitrary p-value (tell me what it is!) and answer the question assuming that p-value is the right one.

Although the p-value is extremely small, it is still possible to make a Type I error. Since our significance level alpha = 0.05, we have a 5% chance of making a Type I error. It is also possible to make a Type II error (failing to reject the null hypothesis when it is actually false) if we have a small effect size (if the difference between the groups is small), small sample size, large variability, or small significance level.

Part d (2 pts)

Suppose we obtained a second sample from the same population with 32 observations in each group, but this time $\bar{x}_1 - \bar{x}_2 = 0.3$. Would the power of our test applied to the second sample be lower, higher, or the same compared to the original sample with $\bar{x}_1 - \bar{x}_2 = 0.8$? Explain your reasoning.

For $\bar{x}_1 - \bar{x}_2 = 0.3$, the power of our test applied to second sample would be lower compared to the original sample with $\bar{x}_1 - \bar{x}_2 = 0.8$ since the effect size is smaller. That is, the difference between the two groups is smaller, so it is more difficult to detect a significant difference between the groups.

Part e (2 pts)

Suppose that Group 1 and Group 2 were instead two of six groups we wish to compare the population means between. Using a Bonferroni correction to control the FWER, what significance level should we compare the p-value to and why?

Since we're adjusting the significance level and not the p-value, then using the Bonferroni correction to control the FWER, we compare the adjusted significance level $0.05/15 = 0.0033$ because there are now 15 or (6 choose 2) different pair-wise comparisons being made.

Part f (2 pts)

Under the conditions from part (e), suggest a better alternative to the Bonferroni correction for controlling the FWER and explain why it is better.

A better alternative would be the Holm step-down procedure because the p-values are arranged in ascending order. Then starting from the lowest p-value, each one is compared to the significance level until we get a non-significant one. Then all the p-values following the non-significant one will also be non-significant since they'll all be greater than $\alpha/(m + j - 1)$ for j=3,4,. . . ,m.

Part g (2 pts)

Suppose that we only obtained 8 observations in each group, so that assumption 4 - the normal distribution of $\bar{x}$ - may or may not hold. What would you do instead to get an accurate p-value for testing $H_0 : \mu_1 = \mu_2 = 0$? (You may assume we only have two groups again.) What additional assumptions (if any) would you have to make?

Since we don't have a normal distribution and we don't know what the distribution is, we can instead use a non-parametric test to get an accurate p-value. For non-parametric tests, we assume independence, random sampling, constant variance, and ranking or order of data.

Part h (2 pts)

Under the conditions from part (g), a t-based confidence interval may not be appropriate. How would you obtain a 95% bootstrap percentile confidence interval for $\mu_1 - \mu_2$? You do *not* have to write any code to obtain the interval, but pseudocode may be useful. Think very carefully about the fact that the data consists of 8 observations in each group!

We would collect the data and calculate the difference in sample means. Resample the data with replacement and calculate the difference in sample means a large number of times since we only have 8 observations. Then we calculate the 2.5% and 97.5% percentiles to get the 95% percentile confidence interval. Finally, we check if $\mu_1 - \mu_2$ is in the confidence interval.

Simulation Problem (16 points total)

Part a (Code: 5 pts)

Simulate a single sample under $H_a : \mu_1 - \mu_2 = 0.25$; that is, the sample should contain 32 observations from $N(0.25, 1)$ in Group 1 and 32 observations from $N(0, 1)$ in Group 2. You can either use separate variables `x1` and `x2` for the values from the two groups or create a 64×2 data frame to store the values and groups in.

```
x1 <- rnorm(n=32, mean=0.25, sd=1)
x2 <- rnorm(n=32, mean=0, sd=1)
```

Obtain the (simulated) value of $\bar{x}_1 - \bar{x}_2$ and the corresponding p-value for testing $H_0 : \mu_1 - \mu_2 = 0$ against a two-sided alternative.

```
sim_xbar <- mean(x1) - mean(x2)
sim_pvalue <- 2 * (1 - pnorm(sim_xbar, mean=0, sd=1 / sqrt(32)))
```

You should make the same assumptions as in the Conceptual Problem:

1. The response variable has been scaled such that $\sigma = 1$ in both groups.
2. We are performing a two-sided hypothesis test using $\alpha = 0.05$.
3. We are using the difference in sample means $\bar{x}_1 - \bar{x}_2$ as our test statistic, NOT a standardized t or z statistic. *Do not* use the `t.test` function to get a p-value.
4. The sample size in each group is large enough to assume that the distribution of $\bar{x}$ is normal in both groups.

The following hints may be useful:

1. Think carefully about the distribution of $\bar{x}_1 - \bar{x}_2$ under H_0. Your solution to Conceptual Problem part (b) may be informative.
2. The `abs` function takes the absolute value of each element in a vector. You don't necessarily *need* to use it, but it may be helpful.
3. Correctly explaining what you want R to do in order to get the p-value will earn at least 1 point even if you code it completely wrong.

Part b (Pseudocode and Code: 3 pts)

Using a `for` loop and your code from part (a), obtain 1000 samples of x1 and x2 under H_a and the corresponding test statistics and p-values. Using $\alpha = 0.05$, estimate the power based on these 1000 samples.

```
set.seed(427)
sim_test_stat <- numeric(1000) # empty vector to store sample means in
sim_pvalue1 <- numeric(1000)
power <- 0

for (i in 1:1000) {
  x1 <- rnorm(n=32, mean=0.25, sd=1)
  x2 <- rnorm(n=32, mean=0.25, sd=1)
  sim_test_stat[i] <- mean(x1) - mean(x2)
  sim_pvalue1[i] <- 2 * (1 - pnorm(sim_test_stat[i], mean=0, sd=1 / sqrt(32)))
  if (sim_pvalue1[i] < 0.05) {
    power <- power + 1
  }
}

power <- power / 1000
power
```

```
## [1] 0.083
```

The power of the test in this simulation is 0.083.

The following hints may be useful:

1. Boolean operations in R include `==` $(a = b)$, `!=` $(a \neq b)$, `<` $(a < b)$, `>` $(a > b)$, `<=` $(a \leq b)$, and `>=` $(a \geq b)$.
2. Remember what the `sum` and `mean` functions do with Boolean (TRUE/FALSE) variables.

3. Correctly explaining what you want R to do in order to estimate the power after you do the simulation will earn at least 1 point even if you code it completely wrong.

Part c (Pseudocode and Code: 3 pts)

Perform another simulation as in part (b), but this time simulate 1000 samples under H_0; that is, each sample should contain 32 observations from $N(0, 1)$ in Group 1 and 32 observations from $N(0, 1)$ in Group 2. Estimate the Type I Error Rate in this simulation.

```
set.seed(427)
sim_test_stat <- numeric(1000) # empty vector to store sample means in
sim_pvalue2 <- numeric(1000)

for (i in 1:1000) {
  x1 <- rnorm(n=32, mean=0, sd=1)
  x2 <- rnorm(n=32, mean=0, sd=1)
  sim_test_stat[i] <- mean(x1) - mean(x2)
  sim_pvalue2[i] <- 2 * (1 - pnorm(sim_test_stat[i], mean=0, sd=1 / sqrt(32)))
}

alpha <- 0.05
sum(sim_pvalue2 < alpha/2 | sim_pvalue2 > 1 - alpha/2)
```

```
## [1] 588
```

The Type I error rate is 588/1000 = 0.588.

The following hints may be useful:

1. In the next part you will need to use both simulations, so do not overwrite the vectors containing the test statistics and p-values from part (b).
2. This simulation is almost the exact same as in part (b). If you did that simulation correctly, you should be able to copy the code and make minor changes.
3. Correctly explaining what you want R to do in order to estimate the Type I Error Rate after you do the simulation will earn at least 1 point even if you code it completely wrong.

Part d (Code: 1 pt; Explanation: 2 pts)

Combine the 2000 p-values from parts (b) and (c) into a single vector. Using this vector, produce a table like the ones in Lab 13.6.1 and estimate the false discovery rate at $\alpha = 0.05$ for this set of 2000 tests.

```
p.values <- c(sim_pvalue1, sim_pvalue2)
decision <- rep("Do not reject H0", 2000)
decision[p.values <= 0.05] <- "Reject H0"
table(decision,
      c(rep("H0 is false", 1000), rep("H0 is true", 1000))
      )
```

```
##
## decision           H0 is false H0 is true
##   Do not reject H0         917        917
##   Reject H0                 83         83
```

False Discovery Rate is the proportion of rejected H0's that were incorrectly rejected. The False Discovery Rate in this simulation is 83/166 = 0.5.

Part e (Code: 1 pt; Explanation: 1 pt)

Adjust your vector of 2000 p-values from part (d) using the Benjamini-Hochberg method. Comment on how the new false discovery proportion when controlling $q = 0.05$ compares to the false discovery proportion from part (d).

```
new.p.values <- p.adjust(p.values, method="BH", n=length(p.values))
decision <- rep("Do not reject H0", 2000)
decision[new.p.values <= 0.05] <- "Reject H0"
table(decision,
      c(rep("H0 is false", 1000), rep("H0 is true", 1000))
      )
```

```
##
## decision           H0 is false H0 is true
##   Do not reject H0         999        999
##   Reject H0                  1          1
```

The new false discovery rate is still 0.5. However, compared to the false discovery rate in part (d), only 1 H0 was incorrectly rejected this time.

Applied Problem (63 points total)

In this problem you will be replicating and expanding on the results of Bissig and DeCarli (2019). Bissig and DeCarli recorded a number of variables on 100 patients who visited the emergency department at University of California, Davis hospital and who required a neurology consultation. This data can be found in the SIS file on Canvas.

Please see the *SIS_dictionary* file for a description of each of the variables in the dataset.

Part a (Code: 3 points)

The researchers defined their primary numerical response variable of interest as the total number of tests ordered and their primary explanatory variable of interest as whether the patient passed (combined score of 4 or higher on the Orientation and Free Recall tasks) or failed (combined score of 3 or lower) the SIS. Unfortunately, neither of these variables is present in the dataset.

Using the dplyr package, add two new variables to the dataset, Quantity_Total_Orders and Pass, that represent the two variables of interest. Pass may be a numerical, character, factor, or logical variable.

```
#Quantity_Total_Orders should take the sum of Quantity_Lab_Orders and Quantity_Image_Orders
SIS_new <- SIS %>% mutate(
  Quantity_Total_Orders = Quantity_Lab_Orders + Quantity_Imaging_Orders,
  Pass = Orientation_Score + Free_Recall_Score
)
#Pass should sum Orientation_Score and Free_Recall_Score
View(SIS_new)
```

Part b (Code: 8 points; Explanation: 12 points)

Perform an exploratory data analysis on the two new variables you created as well as Admission and Duration_of_Admission. In particular, I will be grading on how well you can answer the following questions and support your answer with numerical, tabular, and/or graphical summaries:

- What data is missing? Is there an obvious explanation for any of the missingness?

There are 26 out of 100 missing observations for `Duration_of_Admission`. These same 26 observations also were not admitted into the hospital after the neurology consult, so they couldn't have spent any time in the hospital if they weren't admitted into the hospital.

```
SIS_new %>% select(Admission, Duration_of_Admission) %>% arrange(Duration_of_Admission)
```

```
## # A tibble: 100 x 2
##    Admission Duration_of_Admission
##    <chr>                     <dbl>
##  1 Y                             1
##  2 Y                             1
##  3 Y                             1
##  4 Y                             1
##  5 Y                             1
##  6 Y                             1
##  7 Y                             1
##  8 Y                             1
##  9 Y                             1
## 10 Y                             1
## # ... with 90 more rows
```

- What proportion of patients passed the SIS? What proportion of patients were admitted to the hospital after consultation? Does there appear to be a relationship between these two variables?

73% of the patients passed the SIS. 74% of the patients were admitted to the hospital after consultation.

```
with(SIS_new, sum(Pass >= 4)/100)
```

```
## [1] 0.73
```

```
with(SIS_new, sum(Admission == "Y")/100)
```

```
## [1] 0.74
```

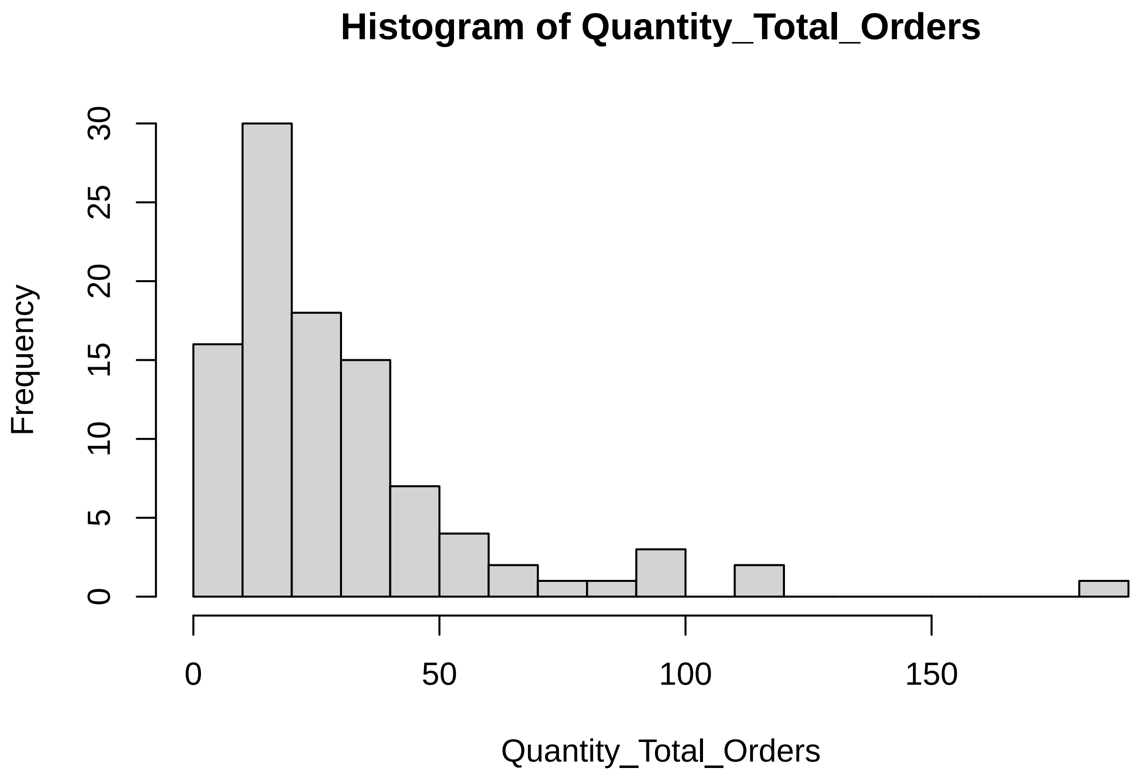
```
#ggplot(SIS_new, aes(x=Admission, y=Pass)) + geom_()
SIS_new %>% select(Admission, Pass) %>% arrange(Pass)
```

```
## # A tibble: 100 x 2
##    Admission  Pass
##    <chr>     <dbl>
##  1 Y             0
##  2 Y             0
##  3 Y             1
##  4 Y             1
##  5 Y             1
##  6 Y             1
##  7 Y             1
##  8 Y             1
##  9 Y             1
## 10 Y             1
## # ... with 90 more rows
```

- What is the distribution of `Quantity_Total_Orders`? Does there appear to be a relationship between this response variable and any of the other three variables (`Pass`, `Admission`, `Duration of Admission`) under investigation? If so, describe the relationship.

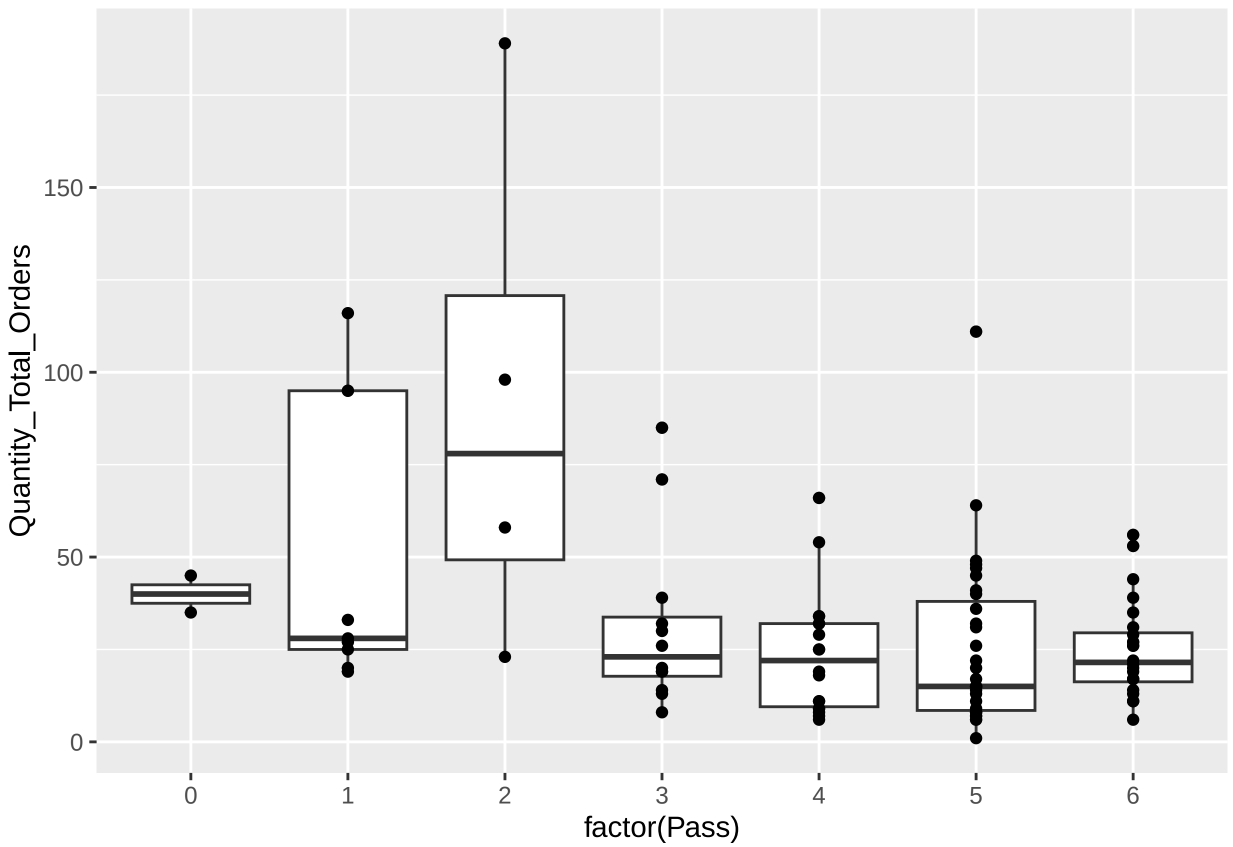
`Quantity_Total_Orders` has a right skewed distribution. Most people had less than 50 tests ordered, and fewer people had more than 50 tests ordered.

```
with(SIS_new, hist(Quantity_Total_Orders, breaks=15))
```



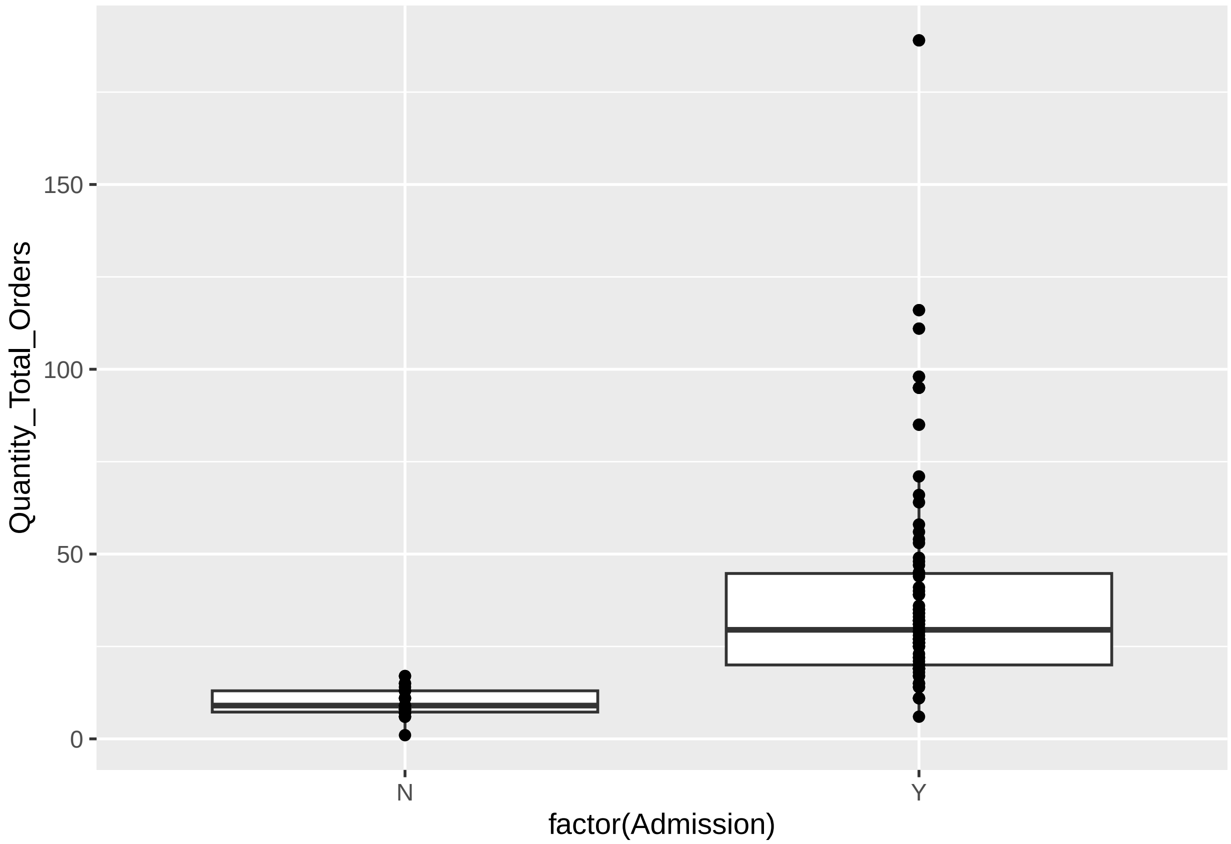
Since there are more people who passed (people who got a combined score of 4 or higher) than people who didn't pass (people who got a combined score of 3 or lower), then the following boxplots are heavily skewed for people who didn't pass. However, on average, it looks like people who passed had less tests ordered than people who didn't pass.

```
#ggplot(SIS_new, aes(x=Quantity_Total_Orders)) + geom_histogram()
ggplot(SIS_new, aes(x=factor(Pass), y=Quantity_Total_Orders)) + geom_boxplot() + geom_point()
```



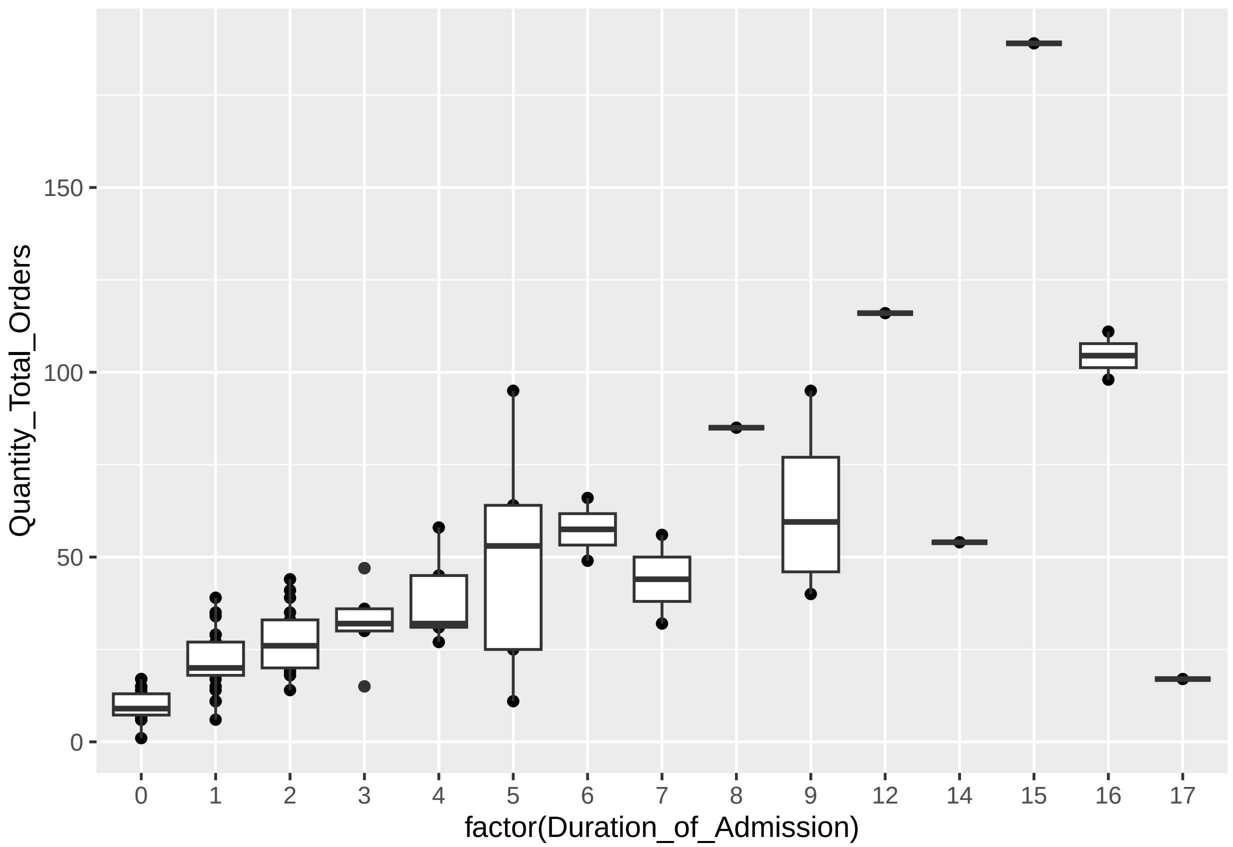
Judging by the boxplots, we can see that people who were admitted to the hospital had more tests ordered than people who were not admitted to the hospital. Since `Quantity_Total_Orders` includes the number of tests ordered before and after admission to the hospital, then people who were admitted to the hospital were likely to have more tests ordered than people who were not admitted to the hospital.

```
ggplot(SIS_new, aes(x=factor(Admission), y=Quantity_Total_Orders)) + geom_boxplot() + geom_point()
```



The average of `Quantity_Total_Orders` increases `Duration_of_Admission` increases but begins to deviate after 6 days because there are fewer patients who stayed longer than 6 days in the hospital.

```
SIS_new1 <- SIS_new %>% replace(is.na(SIS_new),0)
#ggplot(SIS_new1, aes(x=Duration_of_Admission, y=Quantity_Total_Orders)) + geom_point()
#lm(Quantity_Total_Orders ~ Duration_of_Admission, data = SIS_new1)
ggplot(SIS_new1, aes(x=factor(Duration_of_Admission), y=Quantity_Total_Orders)) + geom_point() + geom_bo
```

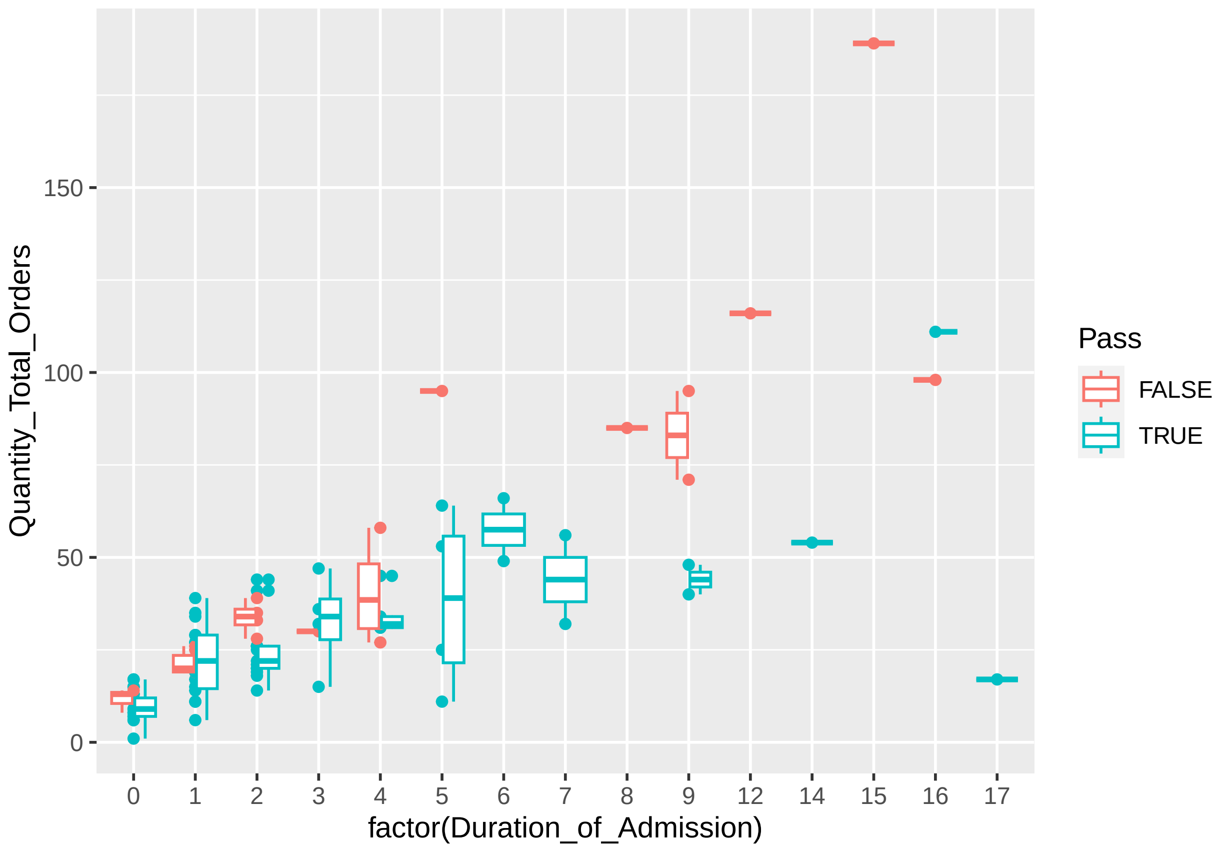


```
#all the lines of code that are commented out were used to fit a line
```

- Does the relationship between `Quantity_Total_Orders` and `Duration_of_Admission` appear to change depending on whether the patient passed the SIS or not?

The relationship between `Quantity_Total_Orders` and `Duration_of_Admission` doesn't appear to change depending on whether the patient passed the SIS or not.

```
#lm(Quantity_Total_Orders ~ Duration_of_Admission + Pass, data = SIS_new1)
SIS_new2 <- SIS_new1 %>% mutate(Pass = Pass >= 4)
#lm(Quantity_Total_Orders ~ Duration_of_Admission + Pass, data = SIS_new2)
ggplot(SIS_new2, aes(x=factor(Duration_of_Admission), y=Quantity_Total_Orders, color=Pass)) + geom_poin
```



Part c (Code: 4 pts; Explanation: 9 pts)

The authors report the following about a least-squares regression of the natural log of `Quantity_Total_Orders` ("ln[studies]" in the paper) against the natural log of `Duration_of_Admission` ("ln[LOS]" in the paper):

- The least-squares regression equation for the "good performers" (those who passed the SIS) is ln[studies] = 0.36 * ln[LOS] + 3.0
- The least-squares regression equation for the "poor performers" (those who did not pass the SIS) is ln[studies] = 0.65 * ln[LOS] + 3.0
- "The full model of ln(LOS) + SIS performance + the interaction explains 55% of the variance in ln(studies)"
- The p-value for testing H_0: in the population model, the interaction has no effect on the number of total orders against H_a: in the population model, the interaction does affect the number of total orders was 0.012
- "Based on the best-fit lines, about 40% more studies will be ordered on a poor performer than on a good performer during a 3-day hospitalization."

Using your `Quantity_Total_Orders`, `Duration_of_Admission`, and `Pass` variables, fit the full model indicated by the authors (note that the R function to take natural logarithms is `log`; you may do this transformation either before fitting the model or in the model-fitting code itself). Determine whether each statement above is correct or incorrect, and justify your answer based on the model output.

```
Quantity_Total_Orders_log <- with(SIS_new2, log(Quantity_Total_Orders))
Duration_of_Admission_log <- with(SIS_new2, log(Duration_of_Admission+1))
lm_log <- lm(Quantity_Total_Orders_log ~ Duration_of_Admission_log * Pass, data = SIS_new2)
summary(lm_log)
```

```
##
```

```
## Call:
## lm(formula = Quantity_Total_Orders_log ~ Duration_of_Admission_log *
##     Pass, data = SIS_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.29556 -0.21612  0.03009  0.26160  0.87201
##
## Coefficients:
##                                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           2.4409     0.1684  14.493  < 2e-16 ***
## Duration_of_Admission_log             0.8772     0.1112   7.885 4.96e-12 ***
## PassTRUE                             -0.1453     0.1879  -0.773    0.441
## Duration_of_Admission_log:PassTRUE  -0.1616     0.1308  -1.235    0.220
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4684 on 96 degrees of freedom
## Multiple R-squared:  0.6802, Adjusted R-squared:  0.6702
## F-statistic: 68.05 on 3 and 96 DF,  p-value: < 2.2e-16
```

The least-squares regression for "good performers" is ln[studies] = 2.5 + (0.88)(ln[LOS]) + (-0.15)(Pass=1) + (-0.16)(ln[LOS])(Pass=1) ln[studies] = 2.35 + (0.72)(ln[LOS])

The least-squares regression for "poor performers" is ln[studies] = 2.5 + (0.88)(ln[LOS]) + (-0.15)(Pass=0) + (-0.16)(ln[LOS])(Pass=0) ln[studies] = 2.5 + (0.88)(ln[LOS])

Based on our full model, the least-squares regression for "good performers" and "poor performers" are not correct. Our full model explains 67% of the variance in ln(studies), not 55%. The p-value for interaction effect obtained from our full model was 0.220, not 0.012. Based on the best-fit lines of our model, about 10% more studies will be ordered on a poor-performer than a good-performer during a 3-day hospitalization.

```
ln_studies_good = 2.35 + 0.72*log(3)
ln_studies_good
```

```
## [1] 3.141001
```

```
ln_studies_bad = 2.5 + 0.88*log(3)
ln_studies_bad
```

```
## [1] 3.466779
```

```
ln_studies_bad/ln_studies_good
```

```
## [1] 1.103718
```

Part d (Explanation: 2 pts)

Would it be appropriate to remove the variable `Pass` from this model, thus leaving only `log(Duration_of_Admission)` and the interaction term? Why or why not?

Without the interaction effect, `Pass` is a significant variable and we wouldn't remove `Pass` from the model. However, with the interaction effect, `Pass` isn't a significant predictor and neither is the interaction term. Since neither `Pass` nor the interaction term is significant, then we can remove `Pass` from this model.

```
summary(lm(Quantity_Total_Orders_log ~ Duration_of_Admission_log + Pass, data = SIS_new2))
```

```
##
## Call:
```

```
## lm(formula = Quantity_Total_Orders_log ~ Duration_of_Admission_log +
##     Pass, data = SIS_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.25477 -0.17533 -0.00319  0.25350  0.88177
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)                2.5903     0.1175  22.044  < 2e-16 ***
## Duration_of_Admission_log   0.7603     0.0587  12.952  < 2e-16 ***
## PassTRUE                  -0.3356     0.1080  -3.108  0.00248 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4697 on 97 degrees of freedom
## Multiple R-squared:  0.6751, Adjusted R-squared:  0.6684
## F-statistic: 100.8 on 2 and 97 DF,  p-value: < 2.2e-16
```

Part e (Code: 2 pts; Explanation: 2 pts)

Create a *new* linear regression model that predicts `Quantity_Total_Orders` from `Duration_of_Admission`, `Pass`, and the interaction term (i.e., no log-transformations). Obtain the residual plot and q-q plot for both models (i.e., with and without the log-transformations). Why do you think the researchers chose to use a log-transformation?

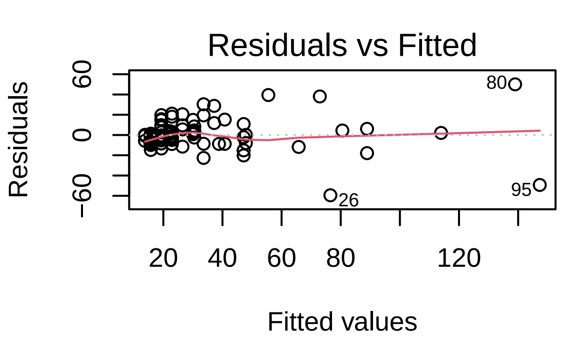
I think the researchers chose to use a log-transformation because the log-transformation more closely follows a normal distribution and is less right-skewed.

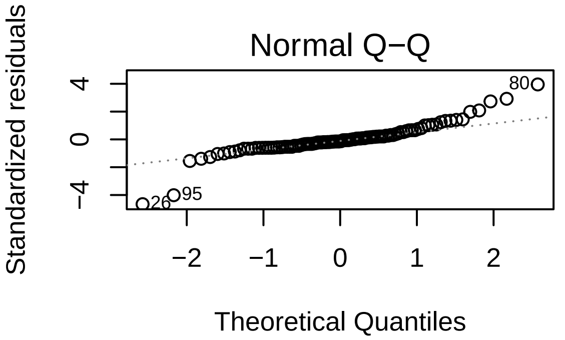
```r
lm_no_log <- lm(Quantity_Total_Orders ~ Duration_of_Admission * Pass, data = SIS_new2)
summary(lm_no_log)
```

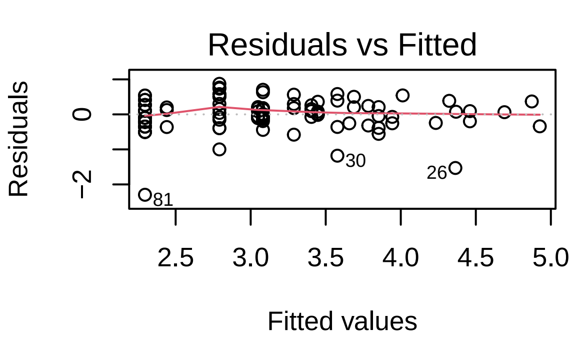
```
##
## Call:
## lm(formula = Quantity_Total_Orders ~ Duration_of_Admission *
##     Pass, data = SIS_new2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -59.501  -7.761  -2.022   4.465  50.034
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   13.7928     3.8331   3.598 0.000509 ***
## Duration_of_Admission          8.3449     0.6385  13.069  < 2e-16 ***
## PassTRUE                       1.9685     4.3836   0.449 0.654401
## Duration_of_Admission:PassTRUE -4.7719     0.8070  -5.913 5.13e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.74 on 96 degrees of freedom
## Multiple R-squared:  0.7377, Adjusted R-squared:  0.7295
## F-statistic: 90.01 on 3 and 96 DF,  p-value: < 2.2e-16
```

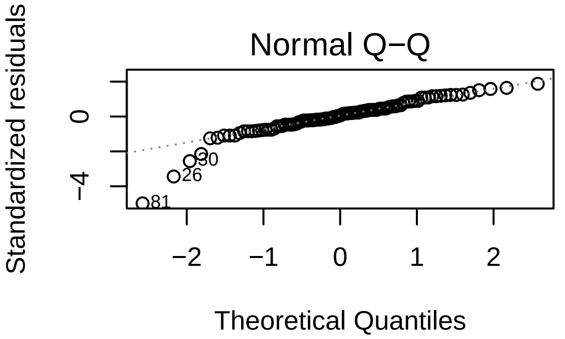
```r
#residual and q-q plot for model with no log-transformation
par(mfrow=(c(2,2)))
```

```
plot(lm_no_log, which = 1)
plot(lm_no_log, which = 2)
#residual and q-q plot for model with log-transformation
plot(lm_log, which = 1)
plot(lm_log, which = 2)
```









Part f (Explanation: 3 pts)

Write a sentence to interpret the slope corresponding to the variable `Pass` in the model from part (e) (i.e., the one without any log-transformations).

If a patient passed the SIS, then the patient is predicted number of studies ordered for them will increase by 2.

Part g (Code: 1 pt; Explanation: 4 pts)

The "False-Positive Psychology" paper refers to *researcher degrees of freedom* - small, seemingly insignificant decisions that can have large impacts on the ultimate inferential results.

In this paper, the researchers made several decisions about how they would represent the number of tests ordered (response), the duration of hospital stay (explanatory), and the performance on the SIS (explanatory) in their model.

Other than the log transformations, identify one alternative choice the researchers *could have* made but didn't when representing these variables in the model. Then, modify your linear regression model from part (c) to incorporate this change (you may need to create a new variable in the dataset before fitting the model). How did this change affect the slope estimates and their corresponding p-values, compared to the model you fit in part (c)?

```
Quantity_Total_Orders_sqrt <- with(SIS_new2, sqrt(Quantity_Total_Orders))
Duration_of_Admission_sqrt <- with(SIS_new2, sqrt(Duration_of_Admission))
lm_sqrt <- lm(Quantity_Total_Orders_sqrt ~ Duration_of_Admission_sqrt * Pass, data = SIS_new2)
summary(lm_sqrt)
```

```
##
## Call:
## lm(formula = Quantity_Total_Orders_sqrt ~ Duration_of_Admission_sqrt *
##     Pass, data = SIS_new2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2089 -0.5394 -0.0367  0.6615  2.5275
##
## Coefficients:
##                                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           2.6252     0.3875   6.774  1.0e-09 ***
## Duration_of_Admission_sqrt            2.2192     0.1929  11.506  < 2e-16 ***
## PassTRUE                              0.5600     0.4326   1.295    0.199
## Duration_of_Admission_sqrt:PassTRUE  -0.9709     0.2277  -4.264  4.7e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.08 on 96 degrees of freedom
## Multiple R-squared:  0.7501, Adjusted R-squared:  0.7423
## F-statistic: 96.07 on 3 and 96 DF,  p-value: < 2.2e-16
```

An alternative choice the researchers could have made was to use square-root transformation instead of log-transformation. Using the square-root transformation, the new intercept changed from 2.45 to 2.63, Duration_of_Admission_log changed from 0.88 to 2.22, Pass changed from -0.15 to 0.56, and the interaction term changed from -0.16 to -0.97. From the log-transformation to the square-root transformation, the p-value of Pass still isn't significant, but the p-value of the interaction term is significant now. This model explain 75% of the variance in Quantity_Total_Orders_sqrt instead of 67% of the variance.

Part h (Pseudocode and/or Explanation: 8 pts)

How would you conduct a permutation test to determine whether patients who can correctly draw a clock (`Clock_Draw` in the `SIS` dataset) have higher SIS scores, on average, than patients who cannot?

Be explicit about the null and alternative hypothesis (in context) and the steps you would take to obtain a p-value. *You do not need to actually write any code.*

H0: Patients who can correctly draw a clock have the same SIS scores, on average, as patients who cannot. Ha: Patients who can correctly draw a clock have higher SIS score, on average, than patients who cannot. Randomly assign the patients into two groups. Calculate our sample statistic, the difference in sample means between the two groups. Collect the data from both groups and randomly reassign the patients into two new groups of the same size. Repeat the previous steps a large number of times. Calculate the p-value. Compare the p-value to the significance level to make a decision whether to reject or fail to reject the null hypothesis.

Part i (Code: 5 pts)

Obtain a 99% confidence interval for the proportion of emergency room patients undergoing neurological consult that have an ischemic etiology (i.e., the proportion of patients for which `Ischemic == Y`). Use the bootstrap method of your choice with 10,000 bootstrap resamples.

```r
set.seed(427)
x <- SIS$Ischemic
x_yes <- sum(x == "Y")/length(x)
x_sample_yes <- numeric(10000)

for(i in 1:10000){
  x_sample <- sample(x, length(x), replace=TRUE)
  x_sample_yes[i] <- sum(x_sample == "Y")/length(x_sample)
}
quantile(x_sample_yes, c(0.005, 0.995))
```

```
##  0.5% 99.5%
##  0.19  0.43
```

We estimate with 99% confidence that the estimated proportion of patients for which `Ischemic == Y` is between the interval of 0.19 and 0.43.

Oral Exam (10 points)

Sign up on Canvas for a 15-minute window in which we will talk for about 10 minutes about your exam solutions.

You will receive 5 points for showing up and up to 5 points of extra credit based on the clarity of your explanations.