Homework 3

Denis Ostroushko

2022-10-04

```
# packages for HW

library(tidyverse)
library(kableExtra)
library(readxl)
library(MASS)
```

Problem 6.3

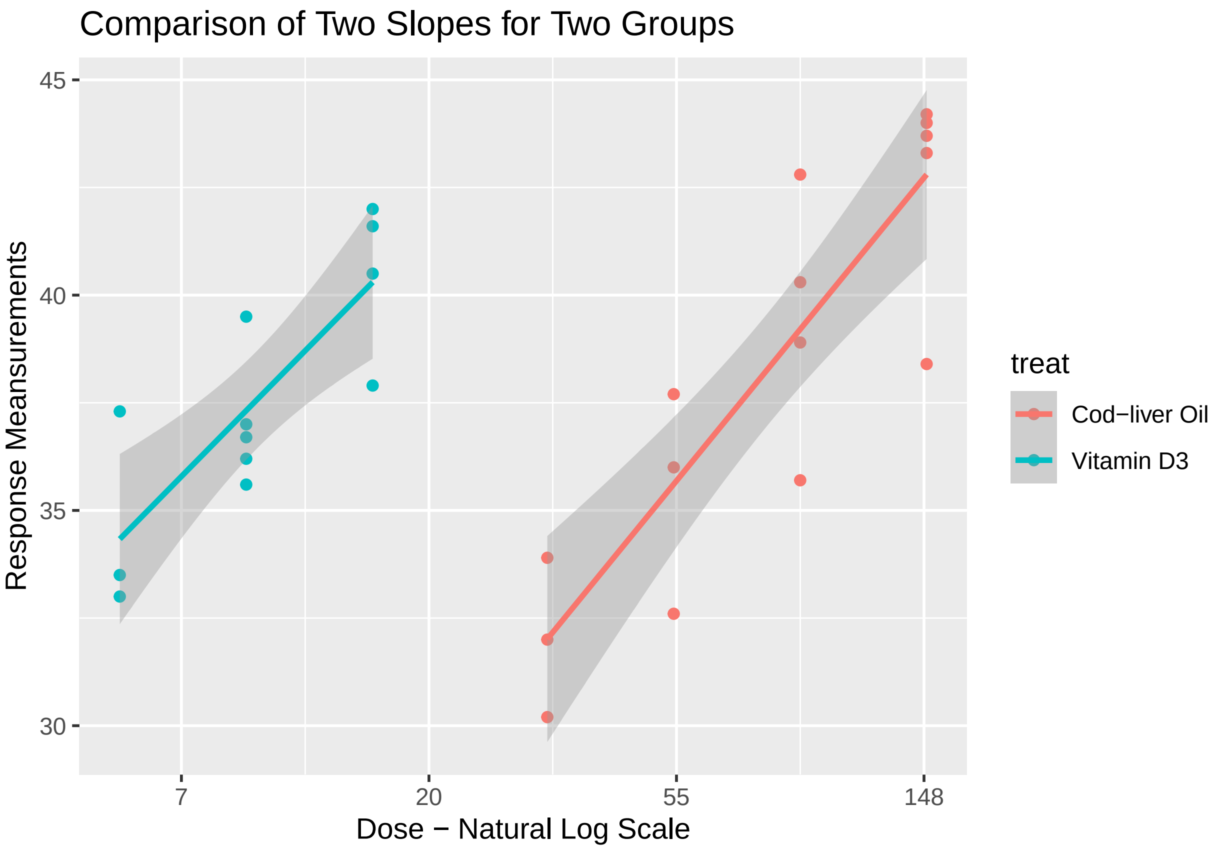
We enter the data below.

```
#put in the data

dose <- c(rep(5.76,3),
          rep(9.6, 5),
          rep(16, 4),
          rep(32.4, 3),
          rep(54, 3),
          rep(90, 4),
          rep(150, 5))

treat <- c(rep("Vitamin D3", 12),
           rep("Cod-liver Oil", 15))

response <- c(33.5, 37.3, 33,
              36.2, 35.6, 36.7, 37, 39.5,
              41.6, 37.9, 40.5, 42,
              32, 33.9, 30.2,
              32.6, 37.7, 36,
              35.7, 42.8, 38.9, 40.3,
              44, 43.3, 38.4, 44.2, 43.7)

vit_data <- data.frame(dose, response, treat)
```

6.3 - A

We should look at the slopes first. Scope of dose values is quite different between the two types of treatments. To assess if the lines are indeed parallel, i.e. the slopes are the same, we need to work with the predictor variable, dose, on the logarithmic scale. I chose natural logarithm scale for this assignment.

```r
ggplot(data = vit_data,
       aes(x = log(dose),
           y = response,
           color = treat)) + geom_point() + stat_smooth(method = "lm") +
  scale_x_continuous(labels = function(x){round(exp(x))},
                     breaks = seq(from = 1, to = 5, by = 1)) +
  ggtitle("Comparison of Two Slopes for Two Groups ") +
  ylab("Response Meansurements") +
  xlab("Dose - Natural Log Scale")
```



In order to compare two slopes we will obtain them from simple linear regression models. To get each slope I partition the data set based on treatment and extract estimates for slope and standard errors. All values are saved into variables below.

```r
d3_lm <- lm(response ~ log(dose), data = vit_data %>% filter(treat == "Vitamin D3"))
cod_lm <- lm(response ~ log(dose), data = vit_data %>% filter(treat == "Cod-liver Oil"))

sum_d3 <- summary(d3_lm)
sum_cod <- summary(cod_lm)

d3_dose_int <- coefficients(d3_lm)[1]
cod_dose_int <- coefficients(cod_lm)[1]

d3_dose_slope <- coefficients(d3_lm)[2]
cod_dose_slope <- coefficients(cod_lm)[2]

d3_slope_s2 <- sum_d3$coefficients[,2][2]^2
```

```r
cod_dose_s2 <- sum_cod$coefficients[,2][2]^2

# Page 10 of slides provides test for slopes
```

For each slope we need a weight, which is the inverse of the slope's estimated variance. All formulas are taken from slides 10-12 for parts (a) and (b).

- We have weight for each group i defined as

$$weight_i = \frac{1}{se(\hat{\beta}_i)^2}$$

```r
# since we have just two weights we will not rely on vectors.
weight_d3 <- 1/d3_slope_s2
weight_cod <- 1/cod_dose_s2
```

Additionally, we need to obtain a weighted average slope. I present a formula for our case, where we have just two groups:

$$\bar{b} = \frac{weight_1 * \hat{\beta}_1 + weight_2 * \hat{\beta}_2}{weight_1 + weight_2}$$

```r
weighted_average <-
  (weight_d3 * d3_dose_slope + weight_cod * cod_dose_slope)/
  (weight_d3 + weight_cod)
```

And finally, we get a G statistic. Once again, I will present a formula for G statistic that is applied to our case with two groups

$$G = weight_1 * (\hat{\beta}_1 - \hat{b})^2 + weight_2 * (\hat{\beta}_2 - \hat{b})^2$$

```r
G_stat <-
  weight_d3 * (d3_dose_slope - weighted_average)^2 +
    weight_cod * (cod_dose_slope - weighted_average) ^ 2

#chi squared on k-1 degrees of freedom, where K is in the number of groups, so DF = 1

G_cutoff <- qchisq(1-.05/2, 1)
```

Now we can set up a test to compare the two slopes:

- Null Hypothesis: $H_0 : \hat{\beta}_1 = \hat{\beta}_2$
- Alternative Hypothesis: $H_a : \hat{\beta}_1 \neq \hat{\beta}_2$
- Test statistic $G : 0.5099$
- Cutoff value from χ^2 distribution with 1 degree of freedom is 5.0239
  - Usually degree of freedom for this test is $k - 1$, where k is the number of groups. We have 2 groups, so we have 1 degree of freedom
- Obtained G statistic does not exceed the threshold, so we can not reject the null hypothesis. Therefore, we do not have enough evidence to conclude that 1 unit increase in dose on the natural logarithmic scale has difference impact on the response between the two different treatments.

  We will obtain a weighted average for the two treatments now

6.3 - B

we have the slope, now we need the standard error

Weighted average slope is 6.572. We calculated this value as a part of G statistic calculation.

Standard error for our case:

$$se(\bar{b}) = \frac{1}{weight_1 + weight_2}$$

```r
weighted_s2 <- 1/(weight_d3 + weight_cod)
```

So, the standard error for the weighted average slope is 0.82.

6.3 - C

To find relative potency we refer to Slide #72. We will use estimates for the intercept, slope, standard error and weight from parts (a) and (b).

We assign Vitamin D3 treatment as Standard treatment, and Cod-liver Oil as Test treatment.

- Let $log(r)$ be the potency, then

$$log(r) = \frac{Intercept_{D3} - Intercept_{Cod}}{\frac{weight_{D3} * \hat{\beta}_{D3} + weight_{COD} * \hat{\beta}_{COD}}{weight_{D3} + weight_{COD}}}$$

- Note that previously we already established weights

```r
log_potency <- (cod_dose_int - d3_dose_int )/
  (
    (weight_d3 * d3_dose_slope + weight_cod * cod_dose_slope)/
      (weight_d3 + weight_cod)
  )
```

- $log(r)$ = relative potency = -2.5232.
- When we convert relative potency from the logarithmic scale using $e^{log(r)} = r$ we obtain a point estimate of 0.0802.
- So, the potency of Cod-liver treatment relevant to Vitamin D3 treatment is 0.0802. So, we need to administer way more test(Cod-liver) drug to obtain the same response as standard(D3) drug.

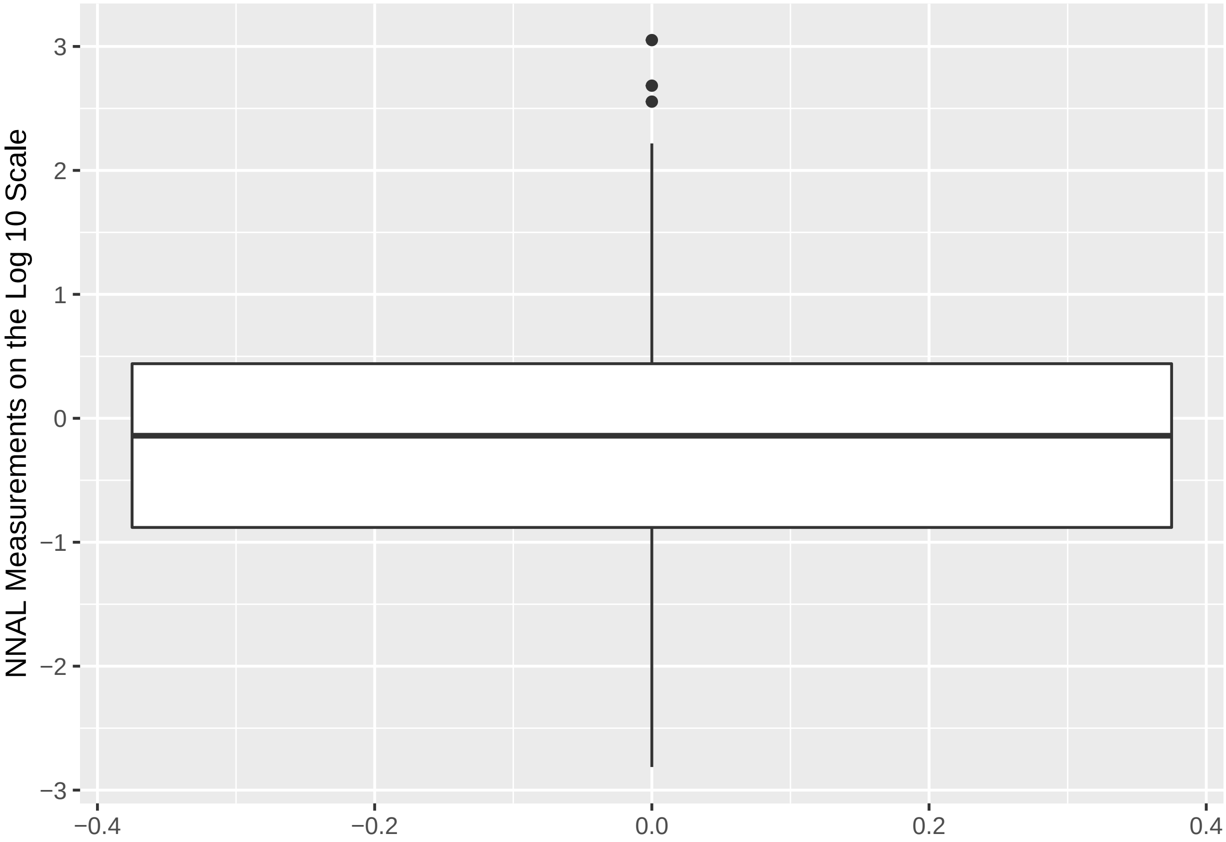
Problem 7.1

```
cig <- read_xls("/Users/denisostroushko/Desktop/UofM MS/MS Fall 2022/Puhb 7405/Data Sets/Cigarettes.xls

original_names <- colnames(cig)

colnames(cig) <- c("age", "gender", "cpd", "carbon_mono", "cotinine", "nnal")

cig$log_nnal <- log(cig$nnal)
```

7.1 - A

Simple box plot of NNAL measurements on the natural base logarithmic scale is given below:

```
ggplot(data = cig,
       aes(y = log_nnal)) + geom_boxplot() +

  ylab("NNAL Measurements on the Log 10 Scale")
```



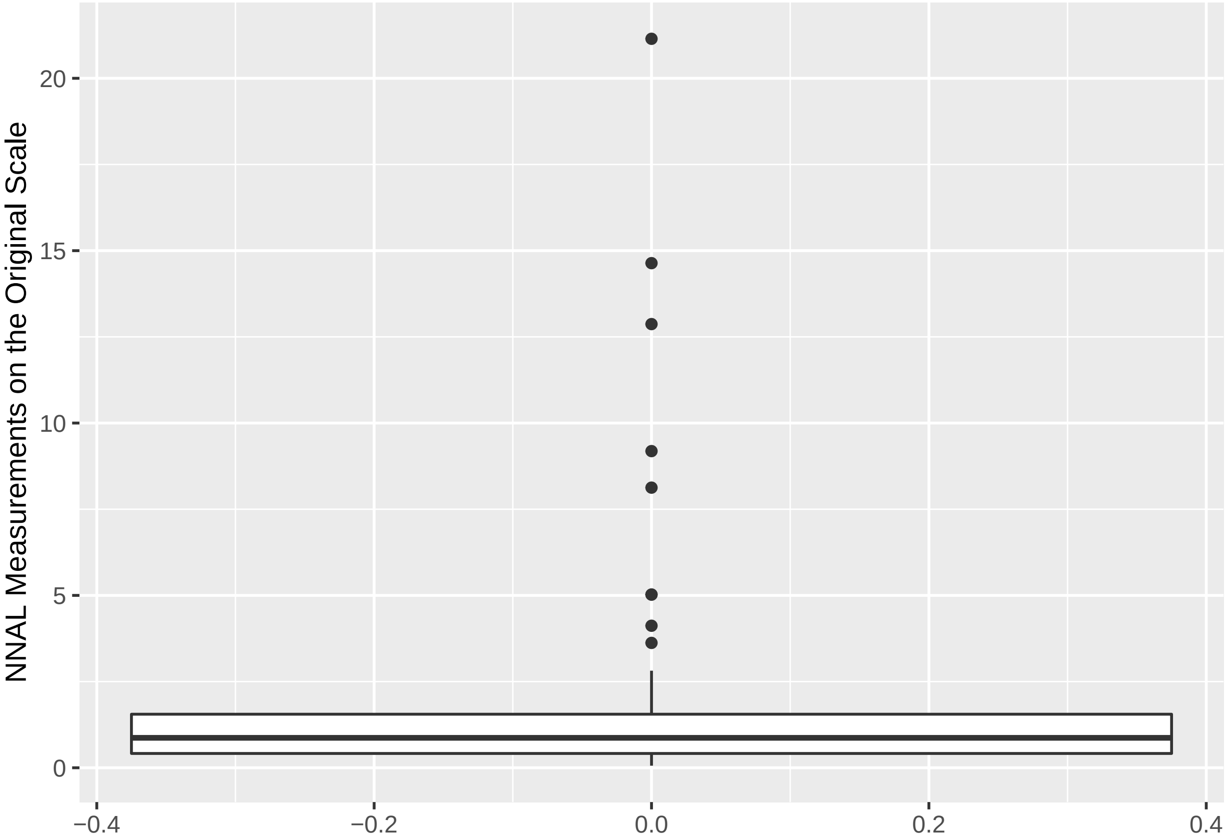
Section (i) There are extreme measurements on NNAL on the natural base logarithmic scale. They are located on the higher end of the distribution. These values are identified as potential outliers in the data.

There are 5 values of Log NNAL measurements that are above that range.

Section (ii) This plot is symmetric. The main body of the box is almost evenly distributed around zero, with a slight bias towards negative values. Same applies to maximum values withing the 1.5 IQR range.

If the get a box plot for values of NNAL on the original scale, we can why we need a transformation.

```
ggplot(data = cig,
       aes(y = nnal)) + geom_boxplot() +

  ylab("NNAL Measurements on the Original Scale")
```

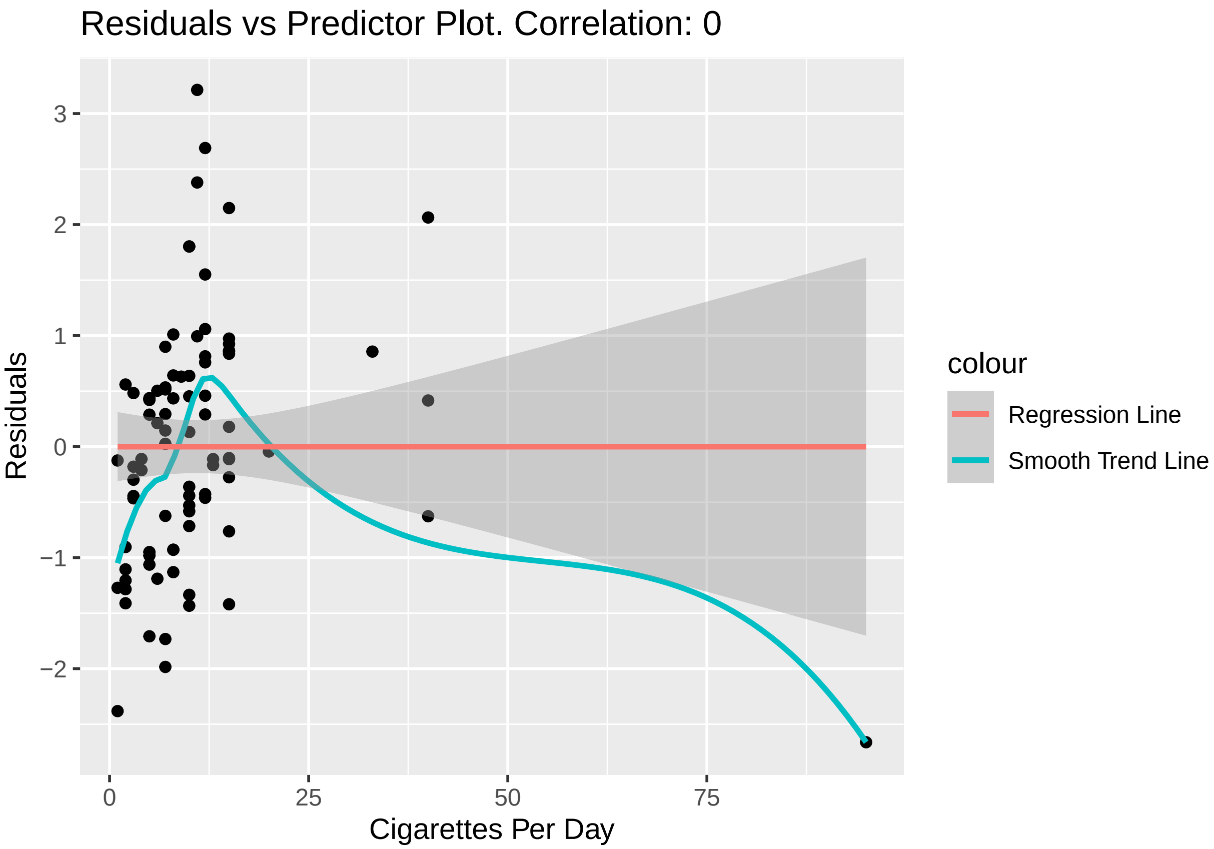


This type of plot reveal almost no information about the variance of NNAL values, other than there are heavy outliers on the upper end. Usually, this type of distribution suggests that the measurements are in fact log-normally distributed, which is why we performed a transformation in the first place.

7.1 - B

We showed how to fit a linear model and obtain $\hat{\beta}_i$ estimates using least squares method in the previous homework, so we will skip manual calculation this time. We will use R function to fit the model and obtain fitted, or predicted, values as well as residuals. Code below obtains all needed data, and shows the plot of residuals against corresponding values of predictor variable - the number of cigarettes per day.

```
lm_7_1 <- lm(log_nnal ~ cpd, data = cig)

cig$residuals <- lm_7_1$residuals

# obtain studentized residuals
cig$student_residuals <- studres(lm_7_1)

cig$y_hat <- lm_7_1$fitted.values

ggplot(data = cig,
       aes(x = cpd,
           y = residuals)) + geom_point() +
  geom_smooth(aes(colour = "Smooth Trend Line"), se = F) +
  geom_smooth(aes(colour = "Regression Line"), method = "lm", se = T)  +

  ylab("Residuals")+
  xlab("Cigarettes Per Day") +
  ggtitle(paste0("Residuals vs Predictor Plot. Correlation: ", round(cor(cig$cpd, cig$residuals),4)))
```



There is a number of departures from the normal error regression that we can study.

- **Non-constant variance**

Right away we can see that the variance of residuals may not be constant. However, it is hard to tell if that is true, because we have some outliers in terms of cigarette consumption. However, due to the shape of the cluster of residuals and values of predictor variable where residuals are concentrated, more tests are required to say more about the constant variance assumption.

- **Linearity of Regression Function and Residual Independence**

We can see that the regression line against residuals in flat at zero, so residuals are independent from the predictor variable.

The model clearly does not fit the outliers very well. We can see that a person who consumes more than 75 cigarettes per day has values of NNAL much lower than expected under this mode, while one person who consumes around 35 cigarettes per day has values of NNAL that are much larger than expected.

- **Outliers and Important Variables**

The reason for outliers, as Chap explained, might be the proportion of a cigarette that these people consume. A person who consumers over 75 cigarettes per day might start it, have some of it, and throw most of it away. While someone who smokes around 35 per day consumer all 35 in full. This is, of course, just a speculation, but if we include more data and construct a multiple linear regression model we may be able to explain these outliers better.

- **Normality of Residual Distribution**

In order to test outliers for normality we plot the residuals against expected values of residuals in a normally distributed random sample.

We can calculate these expected values using the formula:

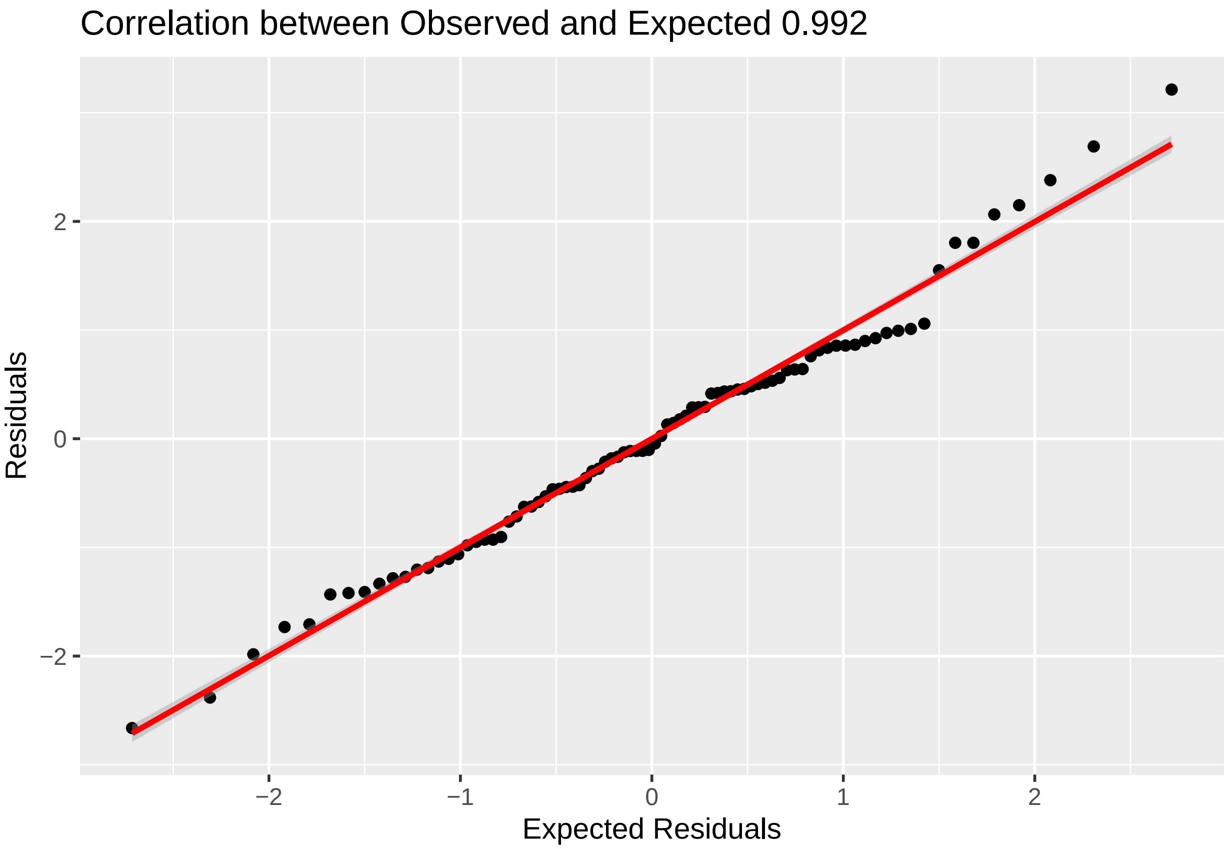
$$\sqrt{MSE} \times z(\frac{Residual - .375}{N + .25})$$

Where $z()$ is the quartile of the standard normal distribution.

We plot this relationship below:

```
mse <- sum(lm_7_1$residuals^2)/lm_7_1$df.residual

cig <- cig %>% arrange(residuals)

cig$resid_rank <- as.numeric(rownames(cig))

N <- nrow(cig)

cig$expected_resid <- sqrt(mse) * qnorm((cig$resid_rank - .375)/(N + .25))

corr <- cor(cig$residuals, cig$expected_resid)

crit_value_80 <- .985
crit_value_90 <- .986
```

```
ggplot(data= cig,
       aes(x = expected_resid, y = residuals)) + geom_point() +
  geom_smooth(method = "lm", color = "red") +
  ylab("Residuals") +
  xlab("Expected Residuals") +
  ggtitle(paste("Correlation between Observed and Expected", round(cor(cig$residuals, cig$expected_resi
```



We can see that observed and expected residuals are very strongly linearly correlated. So, we do not have heavy tails and residuals do not deviate from normality. Correlation coefficient is 0.992. Our sample has 86 observations, so we can find a critical value for 95% confidence level for this correlation coefficient. With 80 observations, the critical level is 0.985, while with 90 observations the critical level is 0.986. Estimated correlation coefficient is above both of those values, so out residuals are normally distributed.

7.1 - C

In order to perform a Brown-Forsythe test we need to complete a number of data transformation steps. We begin by saving residuals into its own data frame, and splitting them into two groups. We have a total of 86 observations, so we will assign 43 lowest values of residuals into “Lower” groups, and the rest into “Upper” group.

```
resid_df <- data.frame(residuals = cig$residuals)

# sort data by residuals and split into two groups.
resid_df <- resid_df %>% arrange(residuals)

resid_df$rank <- seq(from = 1, to = nrow(resid_df), by  = 1)

resid_df$group <-
  as.factor(
    case_when(
      resid_df$rank <= nrow(resid_df)/2 ~ "Lower",
      TRUE ~ "Upper"
    )
  )
paste("Observation in each groups")
```

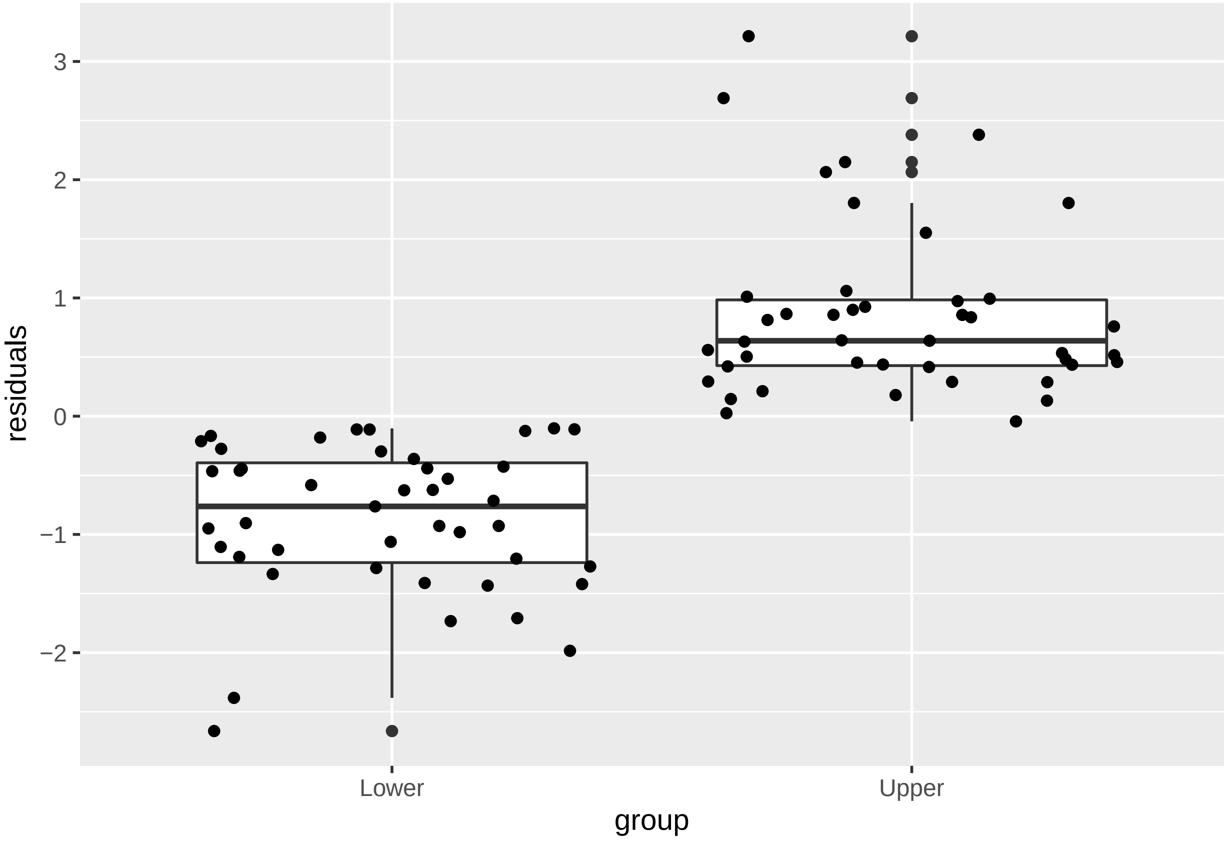
```
## [1] "Observation in each groups"
```

```
summary(resid_df$group)
```

```
## Lower Upper
##    43    43
```

Let's examine variance of residuals in each group using boxplots

```
ggplot(
  data = resid_df,
  aes(x = group,
      y = residuals)
) + geom_boxplot() + geom_jitter()
```



So far, we can see that the median and values of residuals in the upper group are greater than those in the lower group, however, their variance might not differ that much. While residuals in the lower group tend to be below the median of their group, and residuals in the upper group tend t be above the median in their respective group, absolute deviations do not appear so different in the two groups, at least visually.

So, we continue to perform transformations of residual data, mainly we find the median residual value in each group, and calculate absolute deviations. Median residual values for two groups is given below:

```
medians <-
  resid_df %>%
  group_by(group) %>%
  summarize(
    median = median(residuals))

medians
```

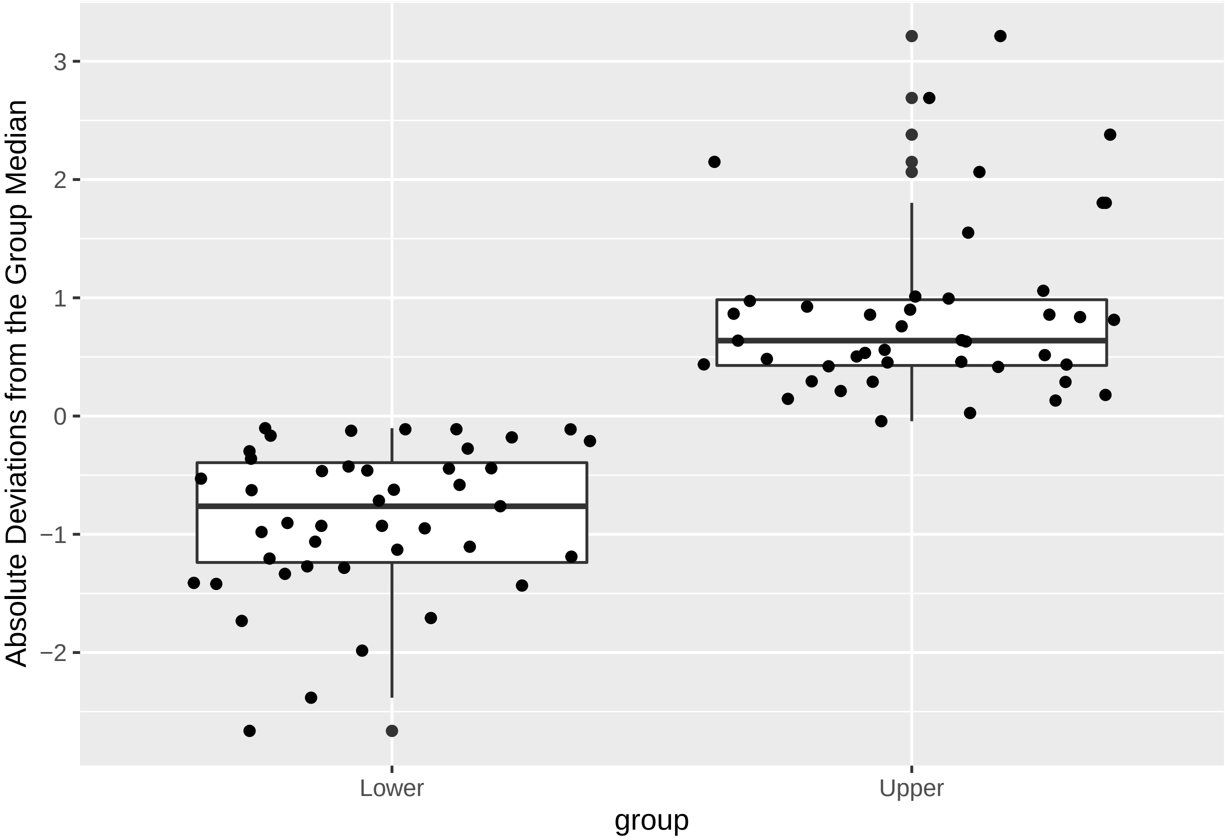
```
## # A tibble: 2 x 2
##   group median
##   <fct>  <dbl>
## 1 Lower -0.763
## 2 Upper  0.638
```

Now we take these values and apply them to resiuals data frame:

```
resid_df$absolute_deviation <-
  case_when(
    resid_df$group == "Lower" ~ abs(resid_df$residuals - medians$median[1]),
    TRUE ~ abs(resid_df$residuals - medians$median[2])
  )
```

Let's visually examine the absolute deviations:

```
ggplot(
  data = resid_df,
  aes(x = group,
      y = residuals)
) + geom_boxplot() + geom_jitter() +
  ylab("Absolute Deviations from the Group Median")
```



Same conclusion as before, while the values themselves are quite different for the two groups, dispersion of values does not seem to differ between the two groups.

Next we find the T statistic. We need average deviations, sample sizes, and pooled variance. All calculations are given in code below, we formally defined all equations in previous homework assignments.

```r
n_lower <- nrow(resid_df[resid_df$group == "Lower", ])
n_upper <- nrow(resid_df[resid_df$group == "Upper", ])

avg_lower <- mean(resid_df[resid_df$group == "Lower", ]$absolute_deviation)
avg_upper <- mean(resid_df[resid_df$group == "Upper", ]$absolute_deviation)

pooled_var <-
  (sum((resid_df[resid_df$group == "Lower", ]$absolute_deviation  - avg_lower)^2 ) +
      sum((resid_df[resid_df$group == "Upper", ]$absolute_deviation  - avg_upper)^2 ) ) /
  (nrow(resid_df) - 2)

t_stat <-
  abs(
  (avg_lower - avg_upper)/
  (sqrt(pooled_var) * sqrt(1/n_lower + 1/n_upper) )
    )
```

We now have all needed data to state the hypotheses and give the conclusion.

- Null Hypothesis: $H_0 : \bar{d}_1 = \bar{d}_2$
- Alternative Hypothesis: $H_0 : \bar{d}_1 \neq \bar{d}_2$
- Test T statistic: 0.032
- Cutoff T statistic value under $n - 2 = 84$ degrees of freedom is 1.99
- Test statistic does not exceed cutoff so we do not have enough statistical evidence to reject the null hypothesis. Therefore, we can not conclude that the variance of residuals differs between the two groups. Therefore, there must be no deviation from the assumption of constant variance.