HW6_106022103

106022103

2021/4/11

- Helped by: 106070038, 106022113

Set up

import libary

```
library(ggplot2)
library(plyr)
require(qqplotr)
```

Random seed

```
SEED <- 1234
```

Read File

```
verizon <- read.csv("data/verizon.csv")
```

Useful function

```
# create bootstrap data
boot <- function(sample0){
  sample(x = sample0, size = length(sample0), replace = TRUE)
}

# bootstrap and return the statistic value
sample_boot <- function(sample0, hyp_mean) {
  resample <- boot(sample0)
  resample_se <- sd(resample) / sqrt(length(resample))
  boot_mean <- mean(resample)
  boot_diff <- mean(resample) - hyp_mean
  boot_t <- (mean(resample) - hyp_mean) / resample_se
  c(boot_mean, boot_diff, boot_t)
}

# bootstrap and return the hypothesis value
bootstrap_null_alt <- function(sample0, hyp_mean) {
  resample <- boot(sample0)
  resample_se <- sd(resample) / sqrt(length(resample))
  t_stat_alt <- (mean(resample) - hyp_mean)      / resample_se
  t_stat_null <- (mean(resample) - mean(sample0)) / resample_se
  c(t_stat_alt, t_stat_null)
```

```
}

# convert the bootstrap t-stastics into dataframe to return
boot_t_stat <- function(sample0, hyp_mean, repeat_times){
  t_stats <- replicate(repeat_times, bootstrap_null_alt(sample0, hyp_mean))
  # t_alts <-  t_stats[1,]
  # t_null <-  t_stats[2,]
  df_alts <- data.frame(Time=t_stats[1,], Group=rep("Alternative hypothesis", repeat_times))
  df_null <- data.frame(Time=t_stats[2,], Group=rep("Null hypothesis", repeat_times))
  rbind(df_alts, df_null)
}

# bootstrap the alternative and null distributions of F
sd_providers_test <- function(larger_sd_sample, smaller_sd_sample) {
  # foolproof:
  # if the sd of larger_sd_sample is smaller, exchange them
  if (sd(larger_sd_sample) < sd(smaller_sd_sample)){
    temp = smaller_sd_sample
    smaller_sd_sample = larger_sd_sample
    larger_sd_sample = temp
  }
  resample_larger_sd <- sample(larger_sd_sample, length(larger_sd_sample), replace=TRUE)
  resample_smaller_sd <- sample(smaller_sd_sample, length(smaller_sd_sample), replace=TRUE)
  f_alt <- var(resample_larger_sd) / var(resample_smaller_sd)
  f_null <- var(resample_larger_sd) / var(larger_sd_sample)
  c(f_alt, f_null)
}

# convert the bootstrap F-stastics into dataframe to return
boot_f_stat <- function(larger_sd_sample, smaller_sd_sample, repeat_times){
  f_stats <- replicate(repeat_times, sd_providers_test(larger_sd_sample, smaller_sd_sample))
  df_alts <- data.frame(value=f_stats[1,], Group=rep("Alternative hypothesis", repeat_times))
  df_null <- data.frame(value=f_stats[2,], Group=rep("Null hypothesis", repeat_times))
  rbind(df_alts, df_null)
}
```

Question 1

Recall the example from last week's HW about Verizon's customer response times. You might have noted that each response time was labeled as ILEC or CLEC in the 'Group' column of that data file. Here is the full story. Verizon was an Incumbent Local Exchange Carrier (ILEC), responsible for maintaining land-line phone service in certain areas. Other competing providers, termed Competitive Local Exchange Carriers (CLEC), could also sell long-distance phone services in Verizon's areas. When something went wrong, Verizon would be responsible to respond and repair services as quickly for CLEC long-distance customers as for its own ILEC customers. The New York Public Utilities Commission (PUC) monitored fairness by comparing Verizon's response times for its ILEC customers versus CLEC customers. In each case, a hypothesis test was performed at the 1% significance level, to determine whether response times for CLEC customers were significantly slower than for Verizon's customers. If Verizon failed to provide fair treatment for CLEC customers, then Verizon would pay large penalties.

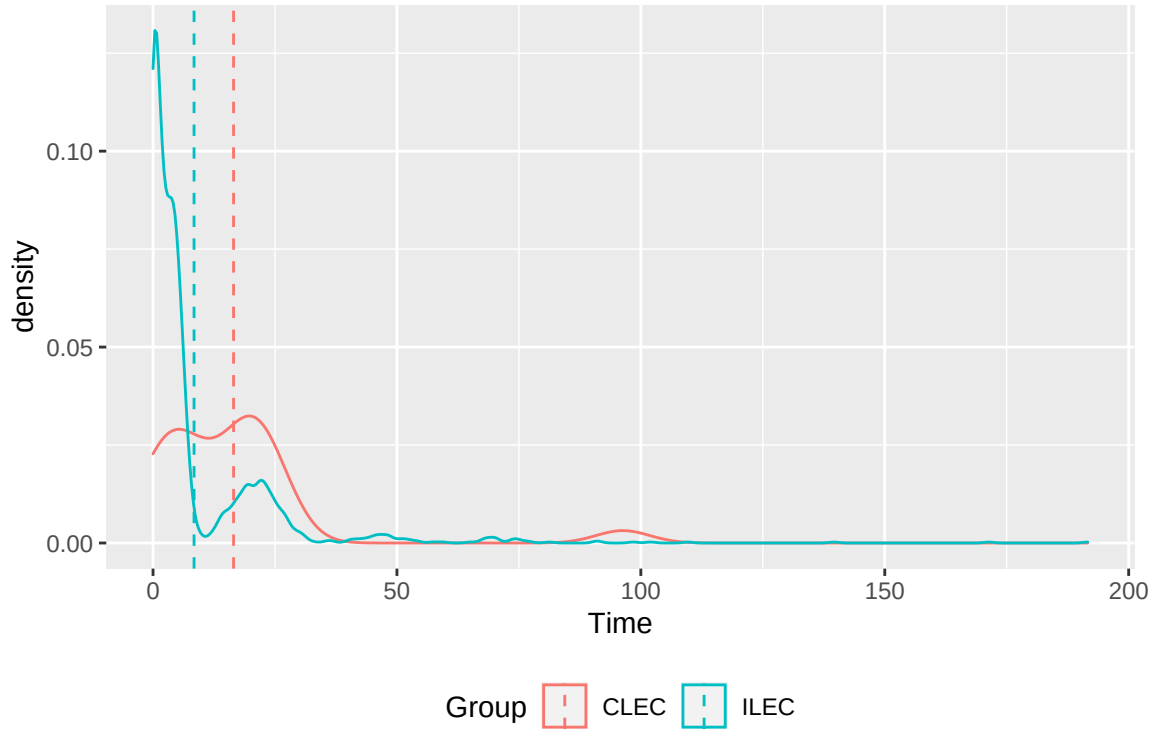
Verizon claims that mean response time for ILEC and CLEC customers are the same, but the PUC would like to test if CLEC customers were facing greater response times.

```
ILEC <- verizon[verizon$Group == "ILEC",]
CLEC <- verizon[verizon$Group == "CLEC",]
```

(a) Visualize

Visualize Verizon's response times for ILEC vs. CLEC customers

```
# get means of each group
verizon_time_mu <- ddply(verizon, "Group", summarise, grp.mean=mean(Time))
# Visualize the density plot and add mean lines.
p <- ggplot(verizon, aes(x=Time, color=Group)) +
  geom_density() +
  geom_vline(data=verizon_time_mu, aes(xintercept=grp.mean, color=Group),
             linetype="dashed") +
  theme(legend.position="bottom")
p
```



(b) t-test using `t.test()`

Use the appropriate form of the `t.test()` function to test the difference between the mean of ILEC sample response times versus the mean of CLEC sample response times. From the output of `t.test()`:

i.

What are the appropriate null and alternative hypotheses in this case?

ANSWER:

- $H_0 : \mu_{ILEC} = \mu_{CLEC}$
- $H_1 : \mu_{ILEC} \neq \mu_{CLEC}$

ii.

Based on output of the t.test(), would you reject the null hypothesis or not?

```
t.test(ILEC$Time, CLEC$Time,conf.level = 0.99)
```

```
##
##  Welch Two Sample t-test
##
## data:  ILEC$Time and CLEC$Time
## t = -1.9834, df = 22.346, p-value = 0.05975
## alternative hypothesis: true difference in means is not equal to 0
## 99 percent confidence interval:
##  -19.588967   3.393927
## sample estimates:
## mean of x mean of y
##  8.411611 16.509130
```

ANSWER: As the result of `t.test()`, the t-statistic ≈ -1.9834 and $p - value \approx 0.05975$. Since $p > 0.01$, we will **NOT reject** H_0.

(c) Boostrap

Let's try this using bootstrapping: Estimate bootstrapped null and alternative values of t by using the same `t.test()` function to compare: bootstrapped samples of ILEC against bootstrapped samples of CLEC (alt t-values); and bootstrapped samples of ILEC against the original ILEC sample (null t-values).

```
set.seed(SEED)
t.test(boot(ILEC$Time), boot(CLEC$Time), conf.level = 0.99)
```

```
##
##  Welch Two Sample t-test
##
## data:  boot(ILEC$Time) and boot(CLEC$Time)
## t = -4.1432, df = 24.378, p-value = 0.0003573
## alternative hypothesis: true difference in means is not equal to 0
## 99 percent confidence interval:
##  -10.535803  -2.050392
## sample estimates:
## mean of x mean of y
##  7.768642 14.061739
```

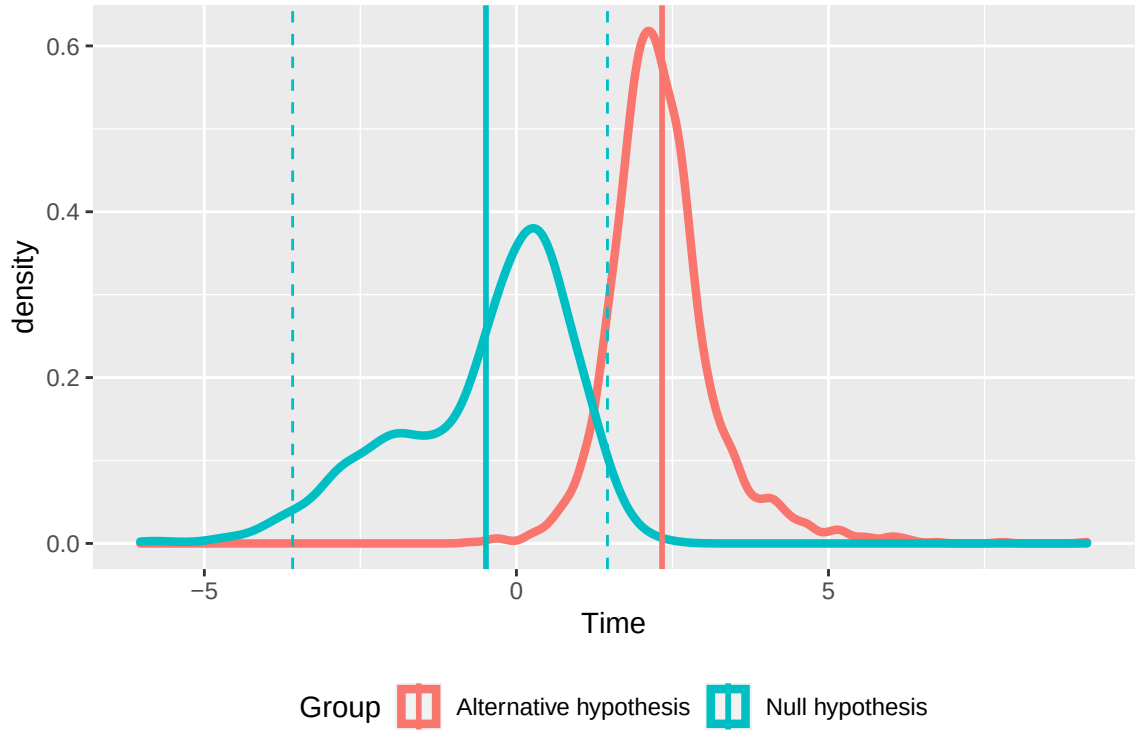
i.

Plot a distribution of the bootstrapped null t-values and alternative t-values, adding vertical lines to show the 5% rejection zone of the null distribution (use the same one-vs-two tail logic as 1b).

```
# bootstrap the alt and null t-values of CLEC
hyp_mean <- mean(boot(ILEC$Time))
t_boots_ILEC <- boot_t_stat(sample0 = CLEC$Time, hyp_mean = hyp_mean, repeat_times = 2000)
t_boots_ILEC_null <- t_boots_ILEC[t_boots_ILEC$Group == "Null hypothesis",]
# visulize
verizon_time_mu <- ddply(t_boots_ILEC, "Group", summarise, grp.mean=mean(Time))
p <- ggplot(t_boots_ILEC, aes(x=Time, color=Group)) +
  geom_density(size=1.5) +
```

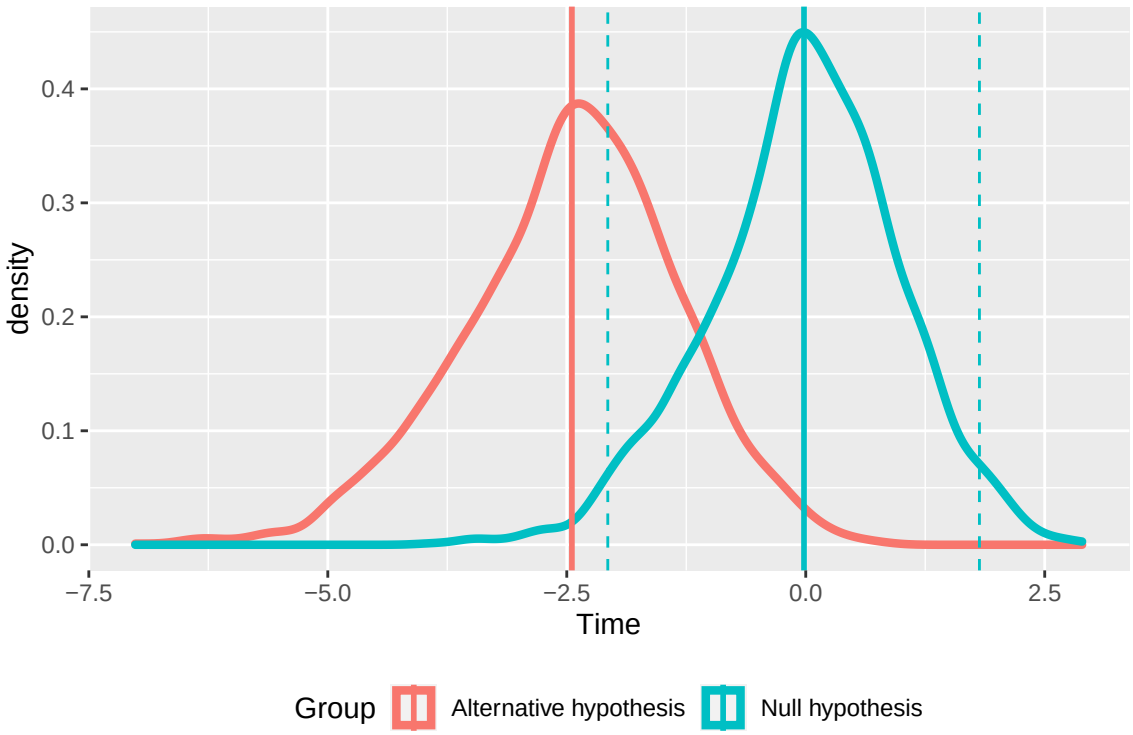
```
  geom_vline(data=verizon_time_mu, aes(xintercept=grp.mean, color=Group),size=1) +
  geom_vline(xintercept=quantile(t_boots_ILEC_null$Time, c(0.025,0.975)), color=rgb(0/255,191/255,196/25
  theme(legend.position="bottom")
p
```

Using mean of ILEC as hypothesis mean, plot the null/alteanative hypothesis of CLEC.



Using mean of CLEC as hypothesis mean, plot the null/alteanative hypothesis of ILEC.

```
# bootstrap the alt and null t-values of ILEC.
hyp_mean2 <- mean(boot(CLEC$Time))
t_boots_CLEC <- boot_t_stat(sample0 = ILEC$Time, hyp_mean = hyp_mean2, repeat_times = 2000)
t_boots_CLEC_null <- t_boots_CLEC[t_boots_CLEC$Group == "Null hypothesis",]
# visulize
verizon_time_mu2 <- ddply(t_boots_CLEC, "Group", summarise, grp.mean=mean(Time))
p <- ggplot(t_boots_CLEC, aes(x=Time, color=Group)) +
  geom_density(size=1.5) +
  geom_vline(data=verizon_time_mu2, aes(xintercept=grp.mean, color=Group),size=1) +
  geom_vline(xintercept=quantile(t_boots_CLEC_null$Time, c(0.025,0.975)), color=rgb(0/255,191/255,196/25
  theme(legend.position="bottom")
p
```



ii.

Based on these bootstrapped results, should we reject the null hypothesis?

ANSWER: Since the 95% CI of null hypothesis conatains 0, we shoud **NOT reject** the H_0.

Question 2

We also wish to test whether the variance of ILEC response times is different than the variance of CLEC response times.

(a) null and alternative hypotheses

What is the null and alternative hypotheses in this case? (Start by identifying which group likely has the higher variance from the sample data at hand)

ANSWER:

- $H_0 : \sigma_{ILEC} = \sigma_{CLEC}$
- $H_1 : \sigma_{ILEC} \neq \sigma_{CLEC}$

(b) traditional statistic

Let's try traditional statistical methods first:

i.

What is the F-statistic of the ratio of variances?

```
var.test(ILEC$Time, CLEC$Time,conf.level = 0.99)
```

```
##
```

```
##  F test to compare two variances
##
## data:  ILEC$Time and CLEC$Time
## F = 0.56731, num df = 1663, denom df = 22, p-value = 0.03165
## alternative hypothesis: true ratio of variances is not equal to 1
## 99 percent confidence interval:
##  0.2221099 1.1111329
## sample estimates:
## ratio of variances
##          0.5673061
```

```
var.test(ILEC$Time, CLEC$Time,conf.level = 0.95)
```

```
##
##  F test to compare two variances
##
## data:  ILEC$Time and CLEC$Time
## F = 0.56731, num df = 1663, denom df = 22, p-value = 0.03165
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.2824333 0.9532391
## sample estimates:
## ratio of variances
##          0.5673061
```

ANSWER: The F-statistic of the ratio of variances is $F \approx 0.5673$.

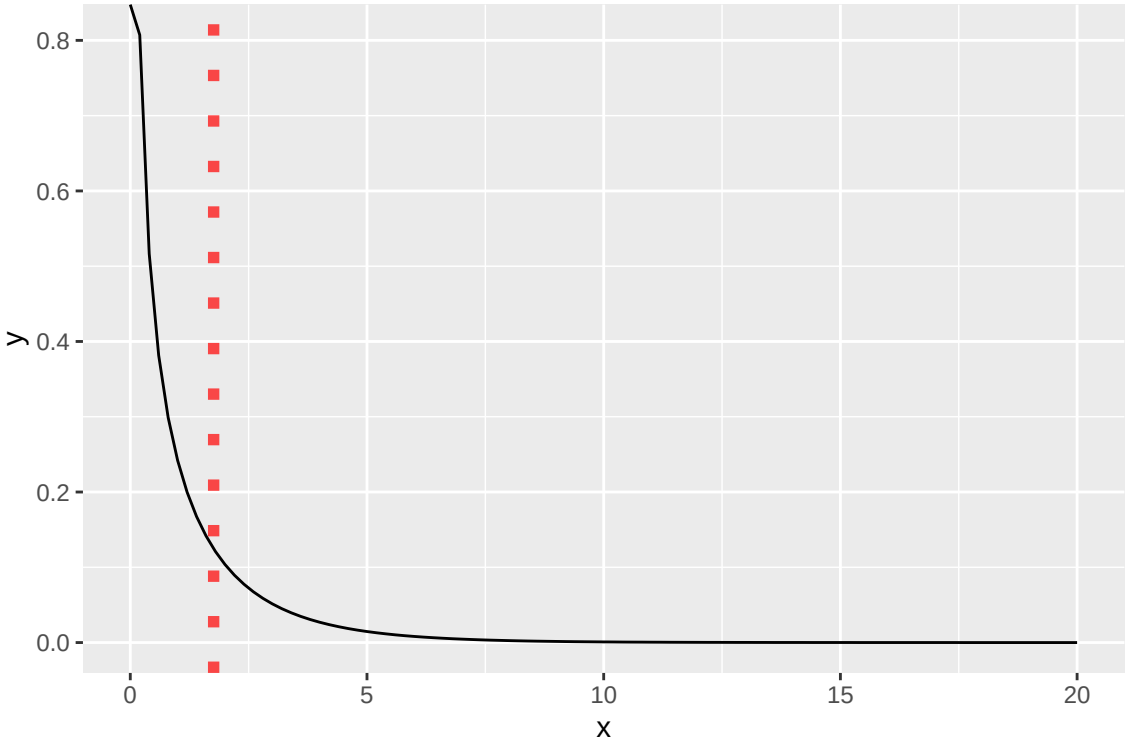
ii.

What is the cut-off value of F, such that we want to reject the 5% most extreme F-values? Use the `qf()` function in R to determine the cutoff.

```
CI95 <- qf(p=0.95, df1=length(ILEC$Time), df2 = length(CLEC$Time))
CI95
```

```
## [1] 1.761161
```

```
ggplot(data.frame(x = c(0, 20)), aes(x = x)) +
    stat_function(fun = dchisq, args = list(df=length(verizon)-1)) +
    geom_vline(xintercept=CI95, color=rgb(1,0,0,0.7),lwd=2,linetype="dotted")
```




```
    # geom_vline(xintercept=CI99, color=rgb(0,0.7,1,0.9),lwd=1.5,linetype="dashed")
```

iii.

Can we reject the null hypothesis?

ANSWER: Since the result above, we would **NOT reject** the null hypothesis.

(c) boostrapping

i.

Create bootstrapped values of the F-statistic, **for both null and alternative hypotheses**.

```
set.seed(SEED)
f_boots <- boot_f_stat(ILEC$Time, CLEC$Time, 2000)
f_boots_null <- f_boots[f_boots$Group == "Null hypothesis",]
f_boots_alts <- f_boots[f_boots$Group == "Alternative hypothesis",]
head(f_boots)
```

```
##       value                  Group
## 1 1.8574408 Alternative hypothesis
## 2 4.1739173 Alternative hypothesis
## 3 0.3928725 Alternative hypothesis
## 4 2.8302181 Alternative hypothesis
## 5 0.5163622 Alternative hypothesis
## 6 0.4116104 Alternative hypothesis
```

ii.

What is the 95% cutoff value according to the bootstrapped null values of F?

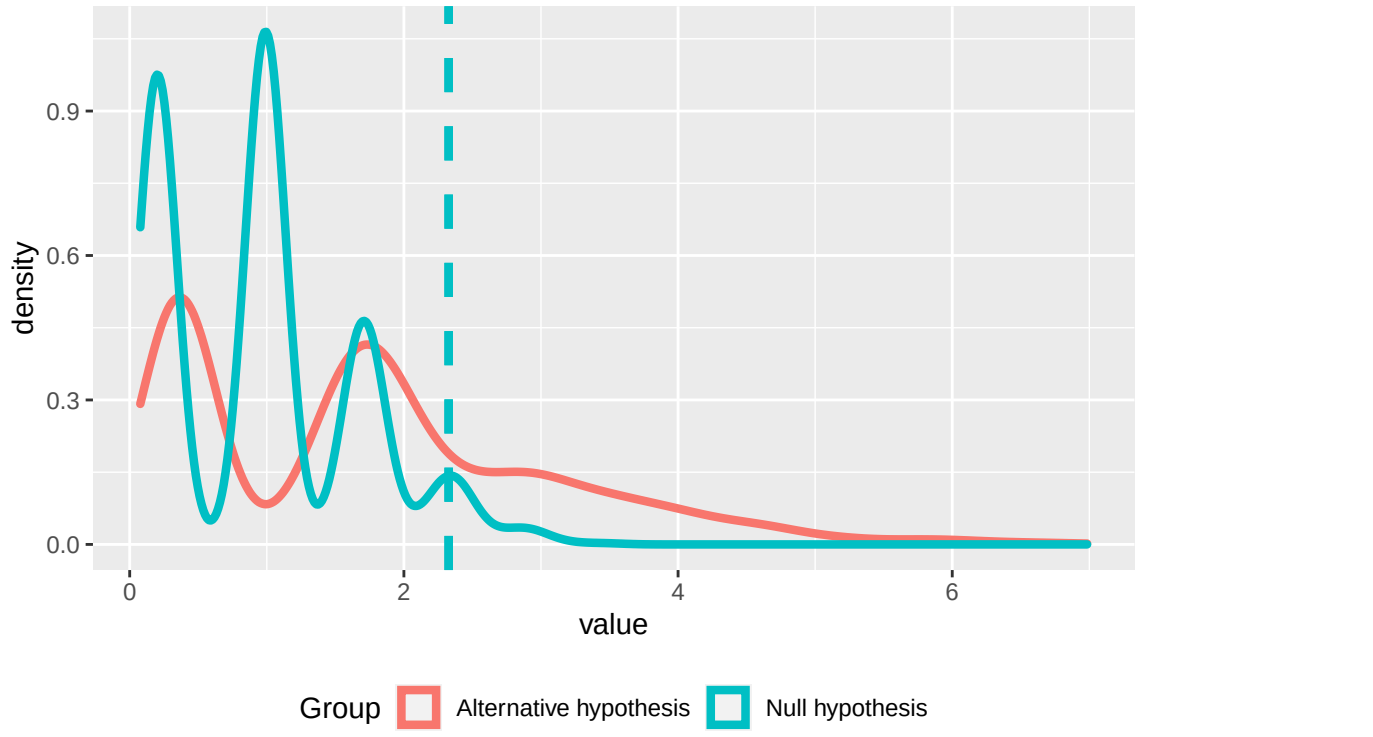
```r
boot_F_CI95 <- quantile(f_boots_null$value, probs=0.95)
boot_F_CI95
```

```
##      95%
## 2.326755
```

iii.

Plot a visualization of the null and alternative distributions of the bootstrapped F-statistic, with vertical lines at the cutoff value of F nulls.

```r
# # bootstrap the alt and null t-values of ILEC
# hyp_mean <- mean(boot(ILEC$Time))
# t_boots_ILEC <- boot_t_stat(sample0 = CLEC$Time, hyp_mean = hyp_mean, repeat_times = 2000)
# t_boots_ILEC_null <- t_boots_ILEC[t_boots_ILEC$Group == "Null hypothesis",]
# visulize
f_boots_mu <- ddply(f_boots, "Group", summarise, grp.mean=mean(value))
p <- ggplot(f_boots, aes(x=value, color=Group)) +
  geom_density(size=1.5) +
  # geom_vline(data=verizon_time_mu, aes(xintercept=grp.mean, color=Group),size=1) +
  geom_vline(xintercept=boot_F_CI95, color=rgb(0/255,191/255,196/255),lwd=1.5,linetype="dashed") +
  geom_vline(xintercept=boot_F_CI95, color=rgb(0/255,191/255,196/255),lwd=1.5,linetype="dashed") +
  theme(legend.position="bottom")
p
```



iv.

What do the bootstrap results suggest about the null hypothesis?

ANSWER: Since $2.32 \geq 1.76$, we should **reject** the null hypothesis.

Question 3

Let's try to see when we should use the non-parametric bootstrap and when we might be better off with traditional statistical approaches.

(a)

Let's create a function to see if key statistics/assumptions of normality are met in our distributions. We will do it by comparing the distributions of our values to a perfect normal distribution. The ellipses (. . .) in the steps below indicate where you should write your own code.

Make a function called `norm_qq_plot()` that takes a set of values):
`norm_qq_plot <- function(values) { ... }`
Within the function body, create zix lines of code as follows.

i.

Create a sequence of probability numbers from 0 to 1, with ~1000 probabilities in between
`probs1000 <- seq(0, 1, 0.001)`

ii.

Calculate ~1000 quantiles of our values (you can use probs=probs1000), and name it `q_vals`
`q_vals <- quantile(...)`

iii.

Calculate ~1000 quantiles of a perfectly normal distribution with the same mean and standard deviation as our values; name this vector of normal quantiles `q_norm`
`q_norm <- qnorm(...)`

iv.

Create a scatterplot comparing the quantiles of a normal distribution versus quantiles of `values`
`plot(q_norm, q_vals, xlab="normal quantiles", ylab="values quantiles")`

v.

Finally, draw a red line with intercept of 0 and slope of 1, comparing these two sets of quantiles
`abline( ... , col="red", lwd=2)`

You have now created a function that draws a "normal quantile-quantile plot" or Normal Q-Q plot (please show code for the whole function in your HW report)

```
norm_qq_plot <- function(values){
  probs1000 <- seq(0, 1, 0.001)
  q_vals <- quantile(values, probs = probs1000)
  q_norm <- qnorm(probs1000, mean = mean(values), sd = sd(values))
  plot(q_norm, q_vals, xlab="normal quantiles", ylab="values quantiles")
  abline( a = 0, b = 1 , col="red", lwd=2)
}

norm_qq_ggplot <- function(values){
  df <- data.frame(value=values)
  gg <- ggplot(data = df, mapping = aes(sample = value)) +
      stat_qq_band() +
      stat_qq_line() +
      stat_qq_point() +
```

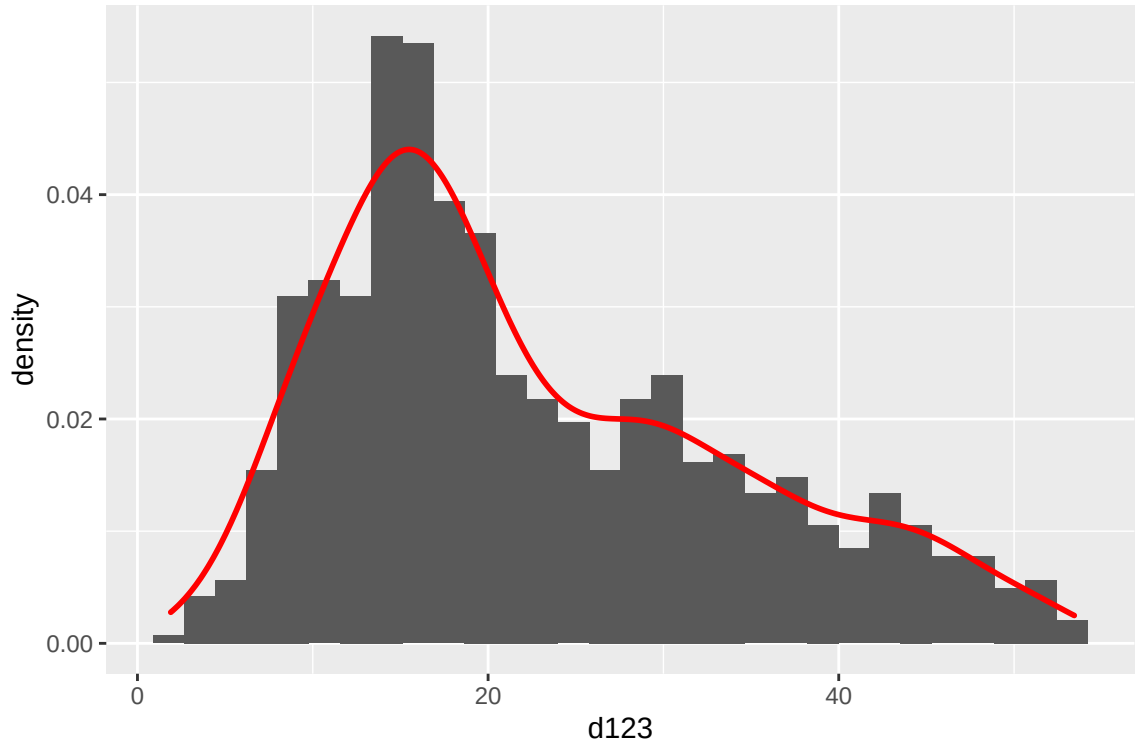
```
      labs(x = "Theoretical Quantiles", y = "Sample Quantiles")
  gg
}
```

(b)

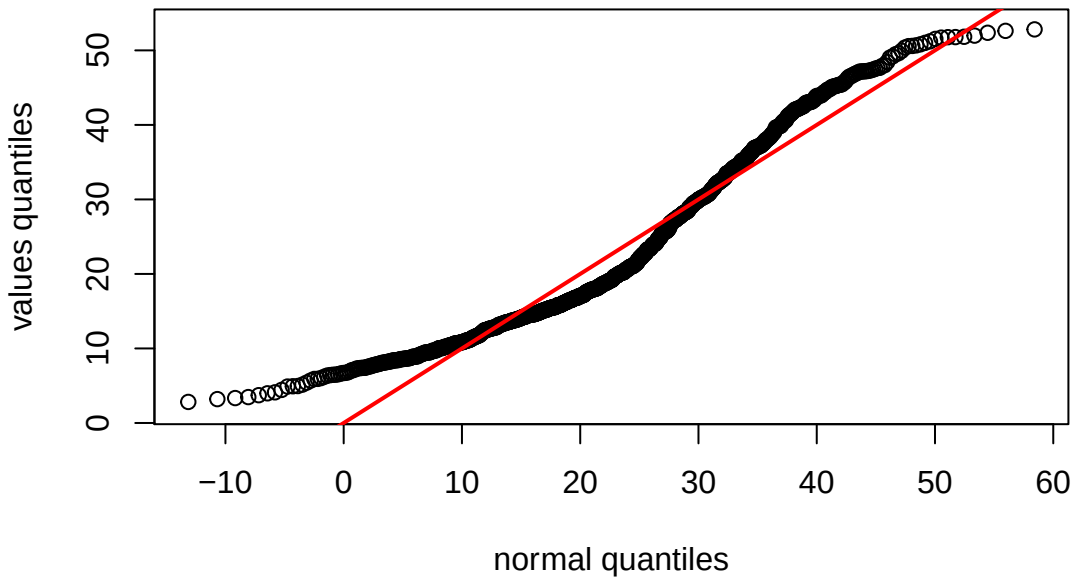
Confirm that your function works by running it against the values of our d123 distribution from week 3 and checking that it looks like the plot on the right:

Interpret the plot you produced (see this article on how to interpret normal Q-Q plots) and tell us if it suggests whether d123 is normally distributed or not.

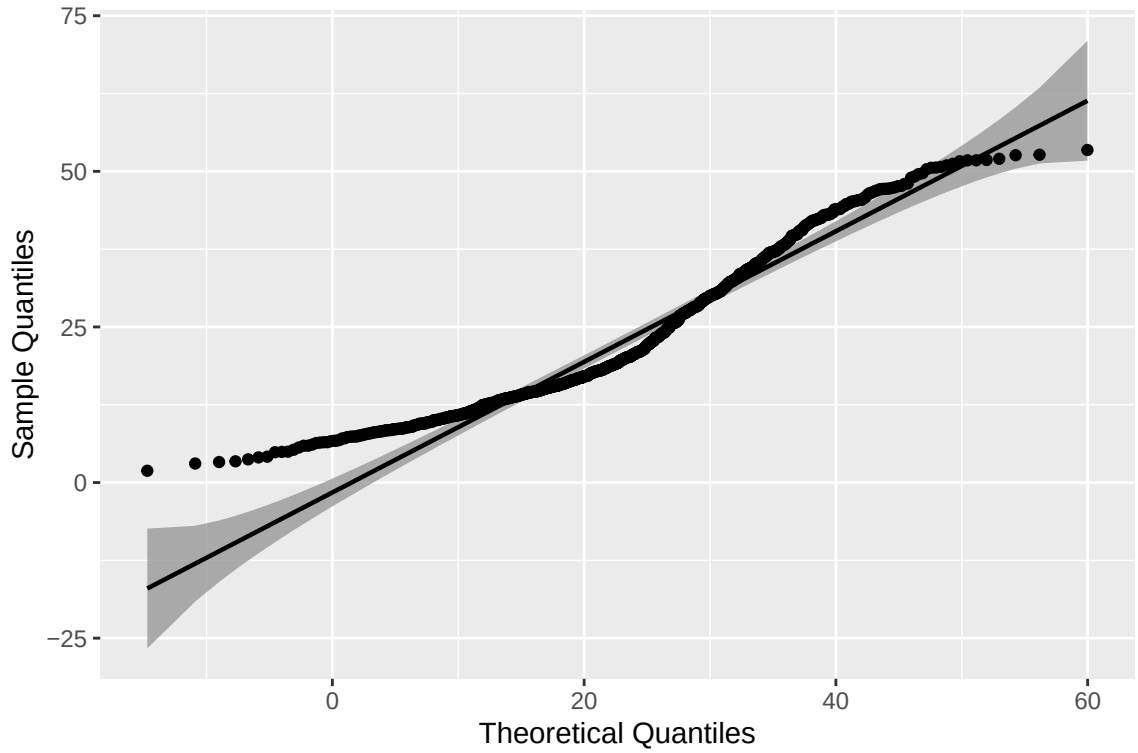
```
d1 <- rnorm(n=500, mean=15, sd=5)
d2 <- rnorm(n=200, mean=30, sd=5)
d3 <- rnorm(n=100, mean=45, sd=5)
d123 <- c(d1, d2, d3)
p <- ggplot(mapping = aes(d123)) + geom_histogram(mapping = aes(y = stat(density))) +
  geom_density(color = "red", size = 1)
p
```




```
norm_qq_plot(d123)
```




```
norm_qq_ggplot(d123)
```

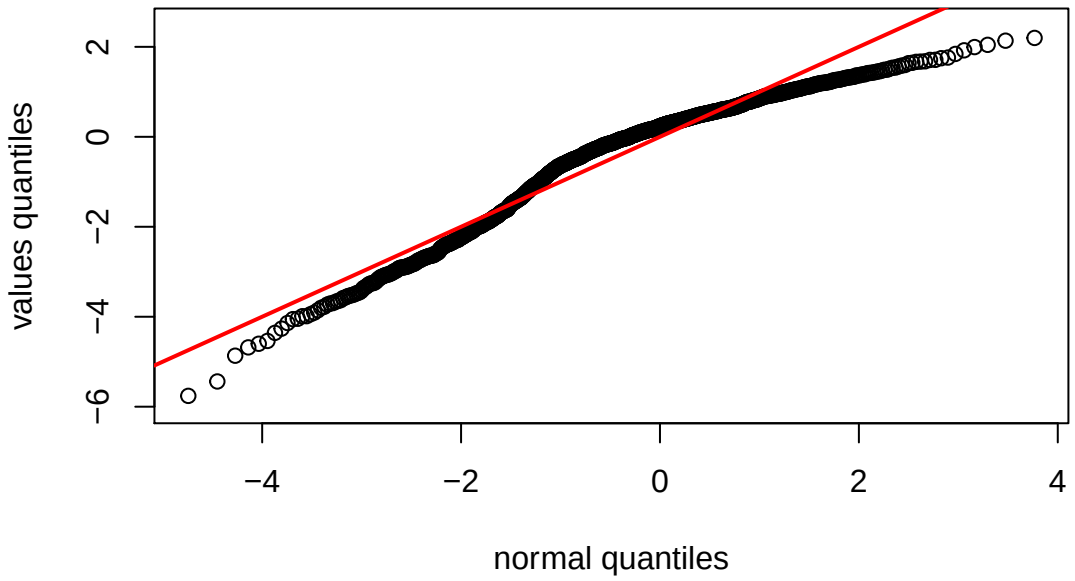


ANSWER:: Because the black points are basically along the red line (y=x), the sample quantiles of d123 would be normally distributed. Moreover, it is also a slightly right-skewed distribution.

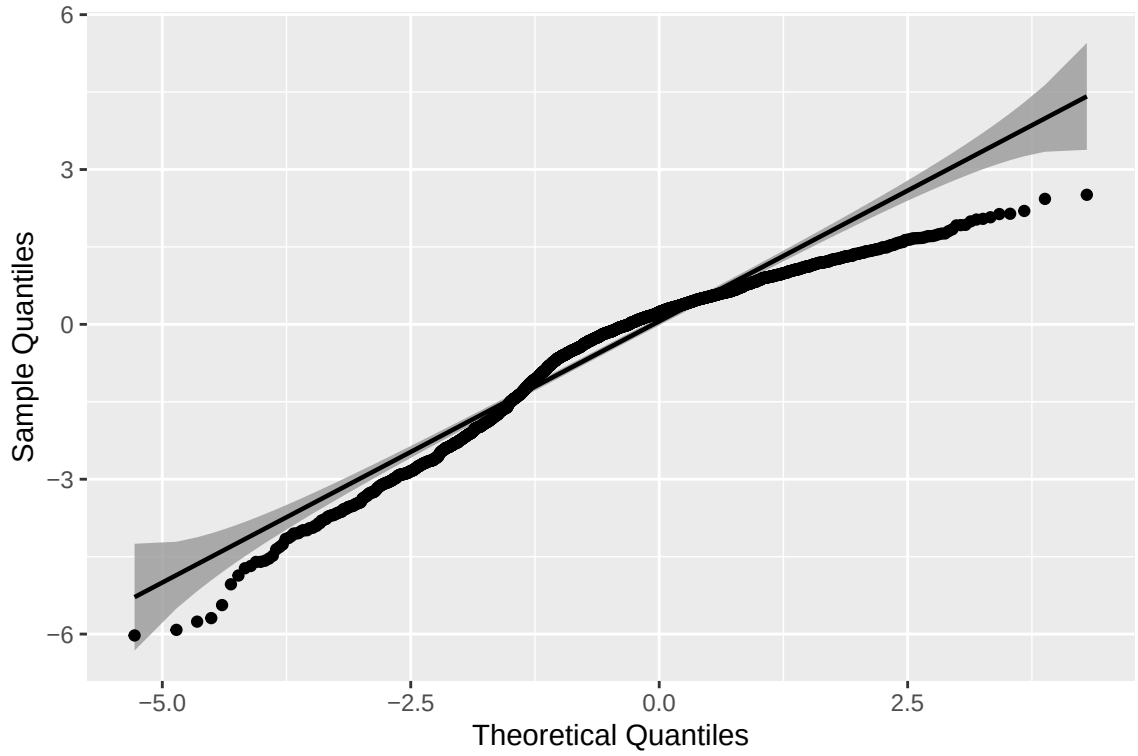
(c)

We generally don't need to use bootstrapping for hypothesis tests of the mean (t-tests) if the null distribution of the t-statistic follows a normal distribution (traditional statistics measures would work fine). Use your normal Q-Q plot function to check if the bootstrapped distribution of null t-values in question 1c was normally distributed. What's your conclusion?

```
norm_qq_plot(t_boots_ILEC_null$Time)
```




```
norm_qq_ggplot(t_boots_ILEC_null$Time)
```

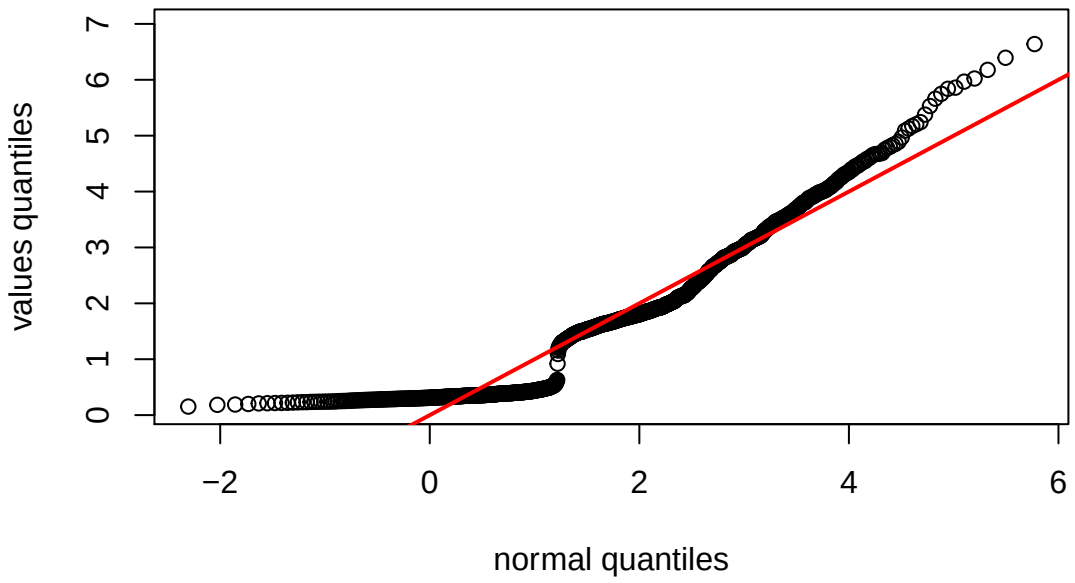


ANSWER:: Because the black points are basically along the red line (y=x), the sample quantiles of d123 would be normally distributed. Moreover, it is also a slightly left-skewed distribution.

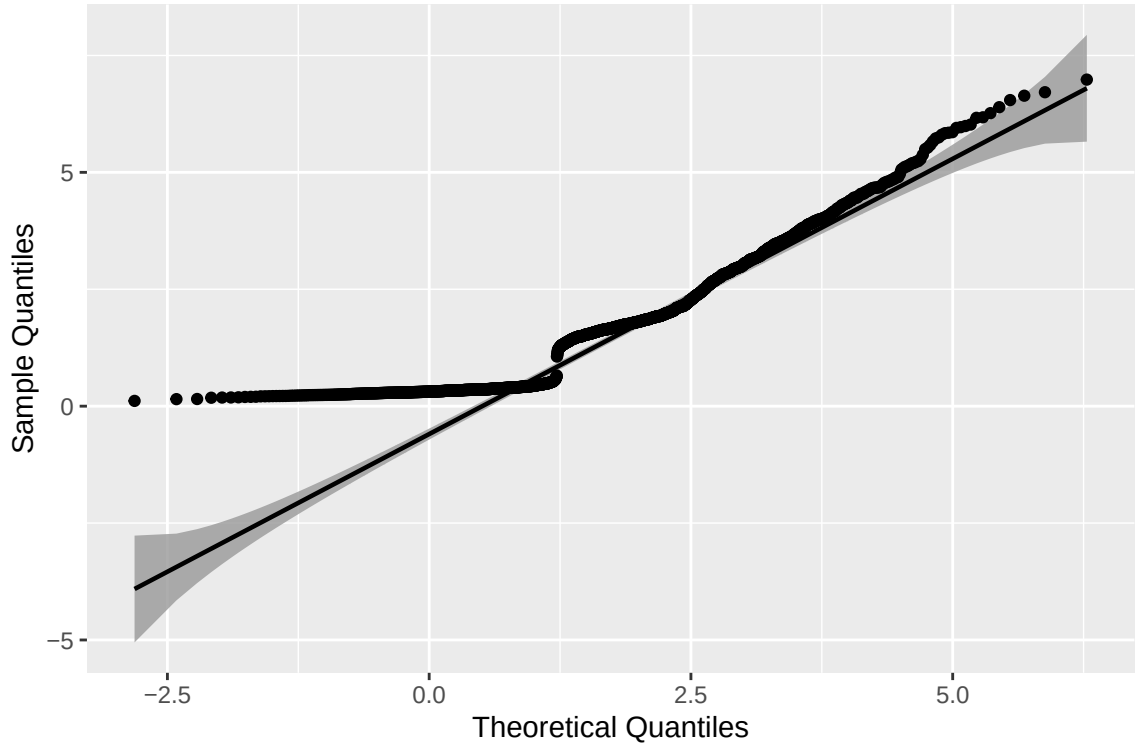
(d)

Hypothesis tests of variances (f-tests) assume the two samples we are comparing come from normally distributed populations. Use your normal Q-Q plot function to check if the two samples we compared in question 2 could have been normally distributed. What's your conclusion?

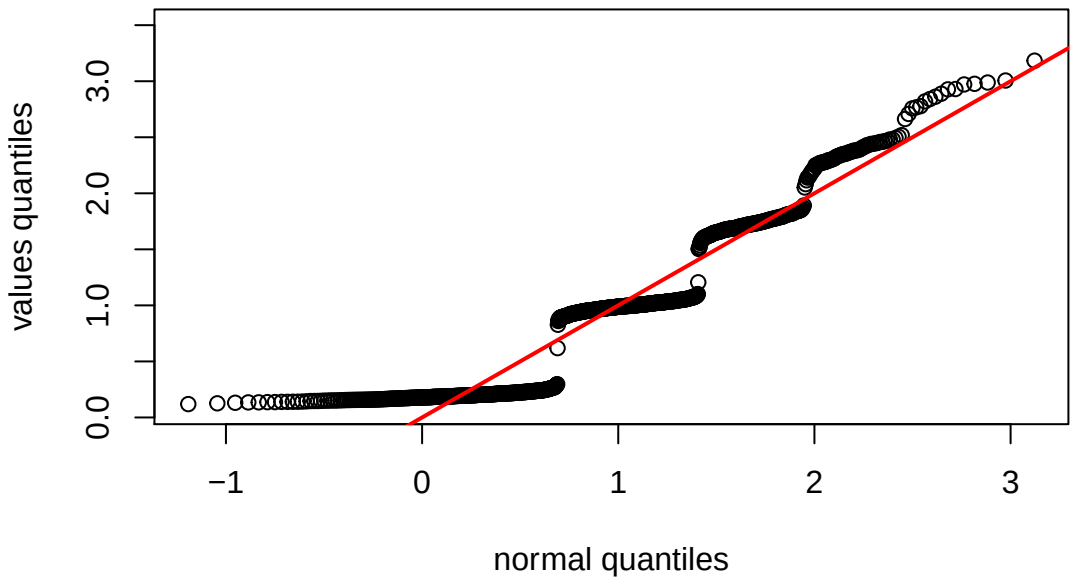
```
norm_qq_plot(f_boots_alts$value)
```



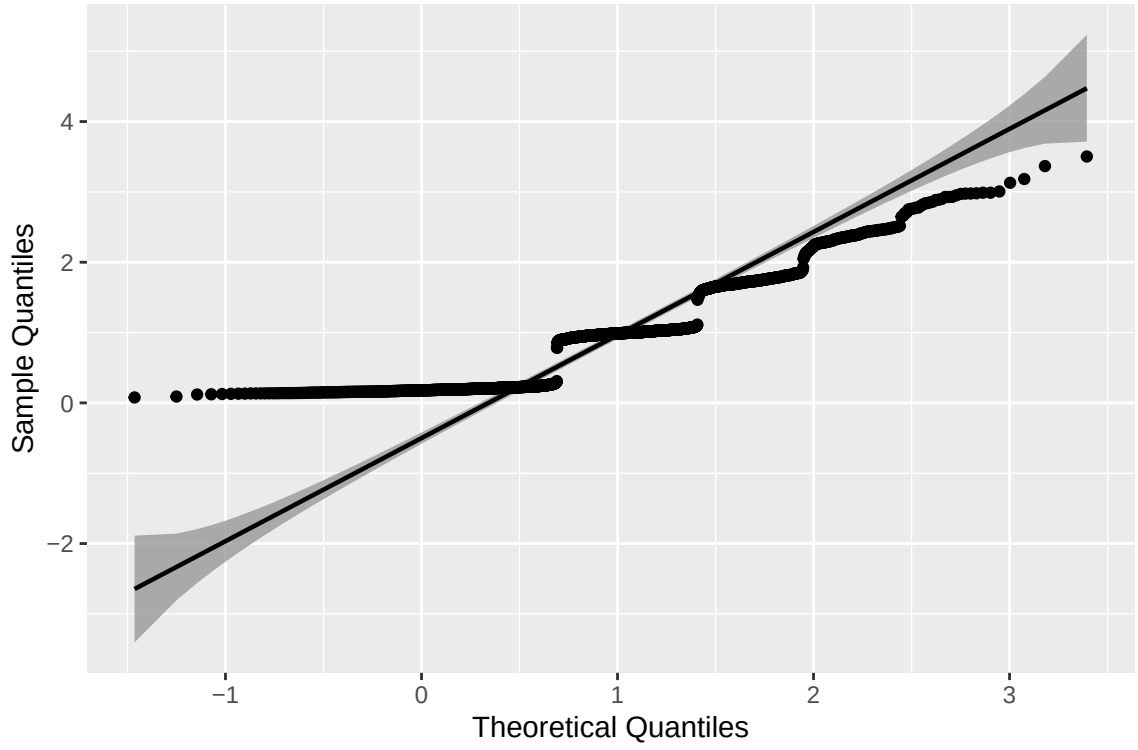

```
norm_qq_ggplot(f_boots_alts$value)
```




```
norm_qq_plot(f_boots_null$value)
```




```
norm_qq_ggplot(f_boots_null$value)
```



AN-

SWER:: Since both qq quantiles are not follow the $y = x$, both are not normally distributed. Moreover, because there is a turn at `f_boots_alts`, it shows a bimodal distribution. Similarly, `f_boots_null` is a multimodal distribution.

Reference

- ggplot2 density plot
- qqplotr
- How to interpret a QQ plot