BACS-hw06-107070004

Question 1)

The Verizon dataset this week is provided as a “wide” data frame. Let’s practice reshaping it to a “long” data frame. You may use either shape (wide or long) for your analyses in later questions.

```
verizon <- read.csv("verizon_wide.csv", header=TRUE)
```

(a) Pick a reshaping package (we discussed two in class) – research them online and tell us why you picked it over others (provide any helpful links that supported your decision).

- https://jtr13.github.io/spring19/hx2259_qz2351.html
  - The gather() function only uses the first column to create the key-value pair, therefore we should comsider more.
  - melt() function treat “variable(host)” as an “id variable” and always produces a correct long form dataset.

(b) Show the code to reshape the versizon_wide.csv data

```
install.packages("reshape",repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/mf/14vfr0p53ps7vvtlfbghnsbr0000gn/T//Rtmpw9QcdY/downloaded_packages
```

```
library(reshape)
loads_long <- melt(verizon, na.rm = TRUE,
                    variable.names = "host",
                    value.names = "load_time")
```

```
## Using  as id variables
```

(c) Show us the “head” and “tail” of the data to show that the reshaping worked

```
head(loads_long)
```

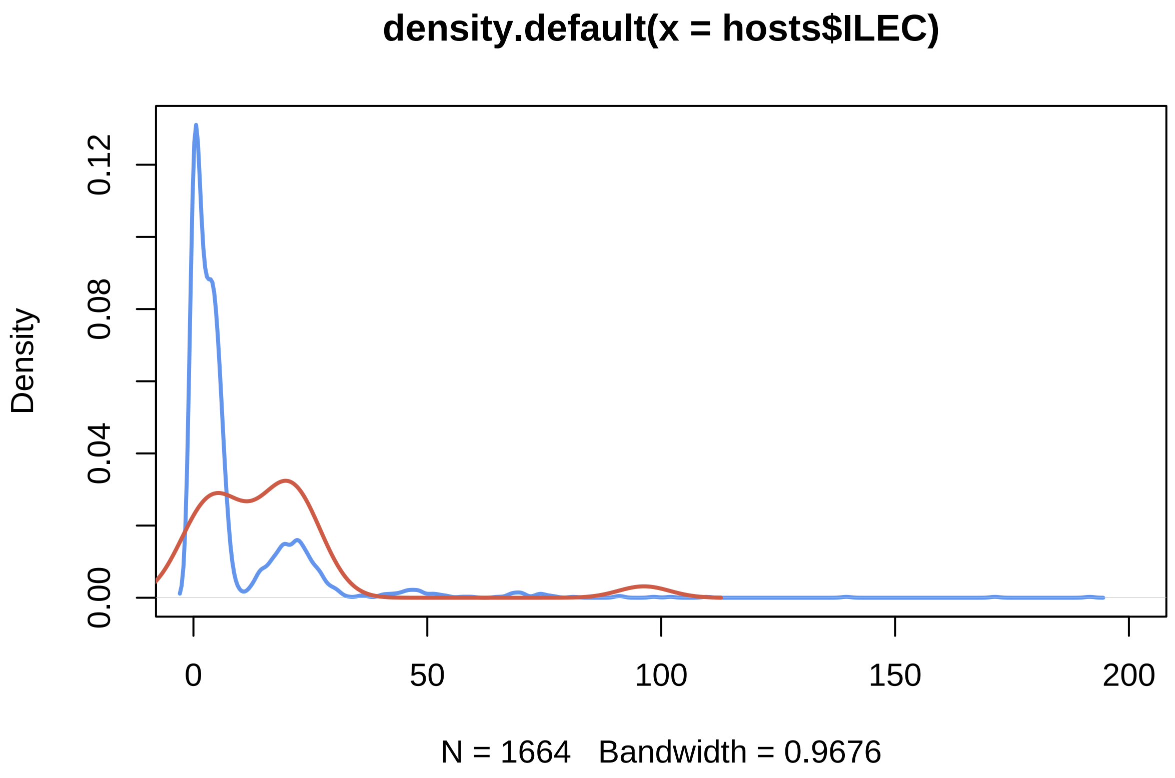
```
##   variable value
## 1     ILEC 17.50
## 2     ILEC  2.40
## 3     ILEC  0.00
## 4     ILEC  0.65
## 5     ILEC 22.23
## 6     ILEC  1.20
```

```
tail(loads_long)
```

```
##      variable value
## 1682     CLEC 24.20
## 1683     CLEC 22.13
## 1684     CLEC 18.57
## 1685     CLEC 20.00
## 1686     CLEC 14.13
## 1687     CLEC  5.80
```

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

```
hosts <- split(x=loads_long$value, f=loads_long$variable)
plot(density(hosts$ILEC), col="cornflowerblue", lwd=2, xlim=c(0, 200))
lines(density(hosts$CLEC), col="coral3", lwd=2)
legend(300, 0.5, lty=1, c("ILEC", "CLEC"), col=c("cornflowerblue", "coral3"))
```



Question 2)

Let’s test if the mean of response times for CLEC customers is greater than for ILEC customers.

(a) State the appropriate null and alternative hypotheses (one-tailed)

H0 : ILEC = CLEC H1 : ILEC < CLEC

(b) Use the appropriate form of the t.test() function to test the difference between the mean of ILEC versus CLEC response times at 1% significance. For each of the following tests, show us the results and tell us whether you would reject the null hypothesis.

(i) Conduct the test assuming variances of the two populations are equal

```r
t.test(hosts$ILEC, hosts$CLEC, conf.level = 0.99, alternative = "two.sided", var.equal=TRUE)
```

```
##
##  Two Sample t-test
##
## data:  hosts$ILEC and hosts$CLEC
## t = -2.6125, df = 1685, p-value = 0.009068
## alternative hypothesis: true difference in means is not equal to 0
## 99 percent confidence interval:
##  -16.0903564  -0.1046833
## sample estimates:
## mean of x mean of y
##  8.411611 16.509130
```

Since p-value(0.09068) < significant level(0.01), reject H0.

(ii) Conduct the test assuming variances of the two populations are not equal

```r
t.test(hosts$ILEC, hosts$CLEC, conf.level = 0.99, alternative = "two.sided", var.equal=FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  hosts$ILEC and hosts$CLEC
## 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
```

Since p-value(0.05975) > significant level(0.01), don‘t reject H0.

(c) Use a permutation test to compare the means of ILEC vs. CLEC response times

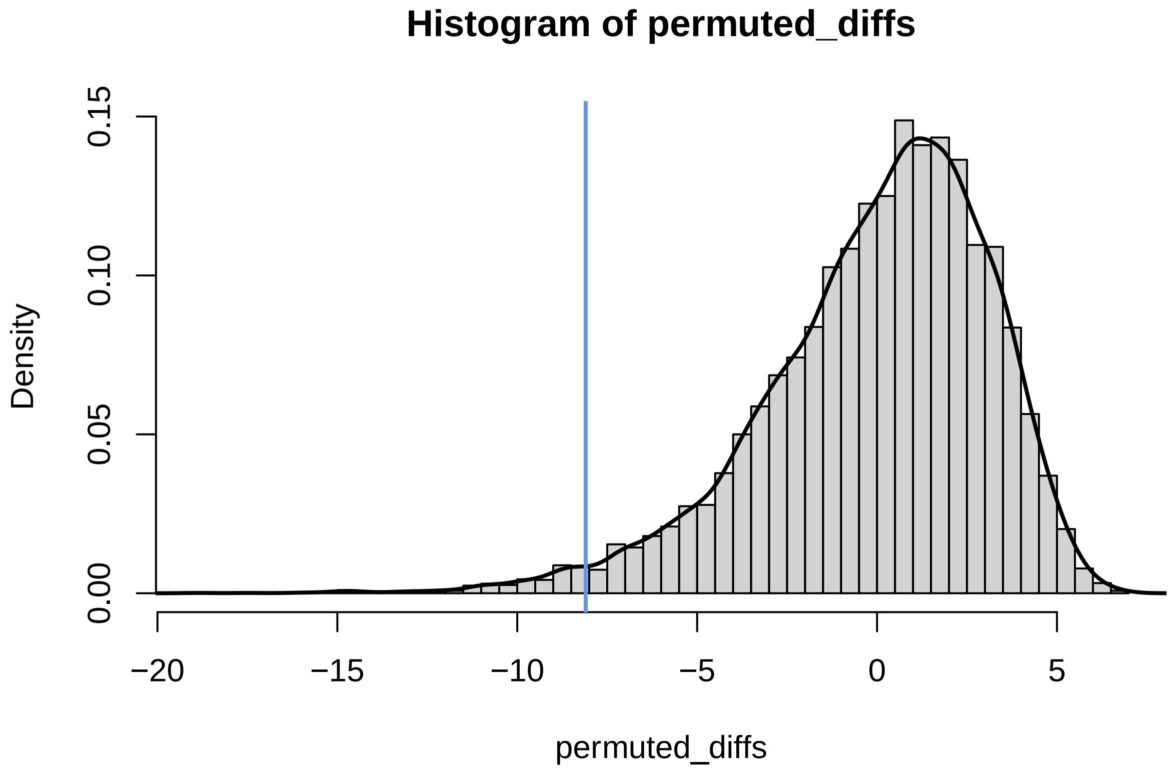
```
permute_diff <- function(values, groups) {
  permuted <- sample(values, replace = FALSE)
  grouped <- split(permuted, groups)
  permuted_diff <- mean(grouped$ILEC) - mean(grouped$CLEC)
}
nperms <- 10000
permuted_diffs <- replicate(nperms, permute_diff(loads_long$value,loads_long$variable))
```

(i) Visualize the distribution of permuted differences, and indicate the observed difference as well.

```
observed_diff <- mean(hosts$ILEC) - mean(hosts$CLEC)
observed_diff
```

```
## [1] -8.09752
```

```
hist(permuted_diffs, breaks = "fd", probability = TRUE)
lines(density(permuted_diffs),lwd=2)
abline(v=observed_diff, col="cornflowerblue", lwd = 2)
```



(ii) What are the one-tailed and two-tailed p-values of the permutation test?

```
p_1tailed <- sum(permuted_diffs>observed_diff) /nperms
p_1tailed
```

```
## [1] 0.9807
```

```
p_2tailed <- sum(abs(permuted_diffs)>observed_diff) /nperms
p_2tailed
```

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

(iii) Would you reject the null hypothesis at 1% significance in a one-tailed test?

The p value > (1-99%)/2, the claim is included in the 99% null distribution.

Question 3)

Let's use the Wilcoxon test to see if the response times for CLEC are different than ILEC.

(a) Compute the W statistic comparing the values. You may use either the permutation approach (with either for-loops or the vectorized form) or the rank sum approach.

```
gt_eq <- function(a, b) {ifelse(a > b, 1, 0) +ifelse(a == b, 0.5, 0)}
W <- sum(outer(hosts$CLEC, hosts$ILEC, FUN =gt_eq))
W
```

```
## [1] 26820
```

(b) Compute the one-tailed p-value for W.

```
n1 <- length(hosts$CLEC)
n2 <- length(hosts$ILEC)
wilcox_p_1tail <- 1 - pwilcox(W, n1, n2)
wilcox_p_1tail
```

```
## [1] 0.0003688341
```

```
wilcox_p_2tail <- 2 * wilcox_p_1tail
wilcox_p_2tail
```

```
## [1] 0.0007376683
```

(c) Run the Wilcoxon Test again using the wilcox.test() function in R – make sure you get the same W as part [a]. Show the results.

```
wilcox.test(hosts$CLEC, hosts$ILEC, alternative = "two.sided")
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  hosts$CLEC and hosts$ILEC
## W = 26820, p-value = 0.000913
## alternative hypothesis: true location shift is not equal to 0
```

(d) At 1% significance, and one-tailed, would you reject the null hypothesis that the values of CLEC and ILEC are different from one another?

```
wilcox.test(hosts$CLEC, hosts$ILEC, conf.level = 0.99, alternative = "greater")
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  hosts$CLEC and hosts$ILEC
## W = 26820, p-value = 0.0004565
## alternative hypothesis: true location shift is greater than 0
```

The p value(0.0004565) < (1-99%)/2, reject H0.

Question 4)

One of the assumptions of some classical statistical tests is that our population data should be roughly normal. Let's explore one way of visualizing whether a sample of data is normally distributed.

(a) Follow the following steps to create a function to see how a distribution of values compares to a perfectly normal distribution. The ellipses (. . .) in the steps below indicate where you should write your own code.

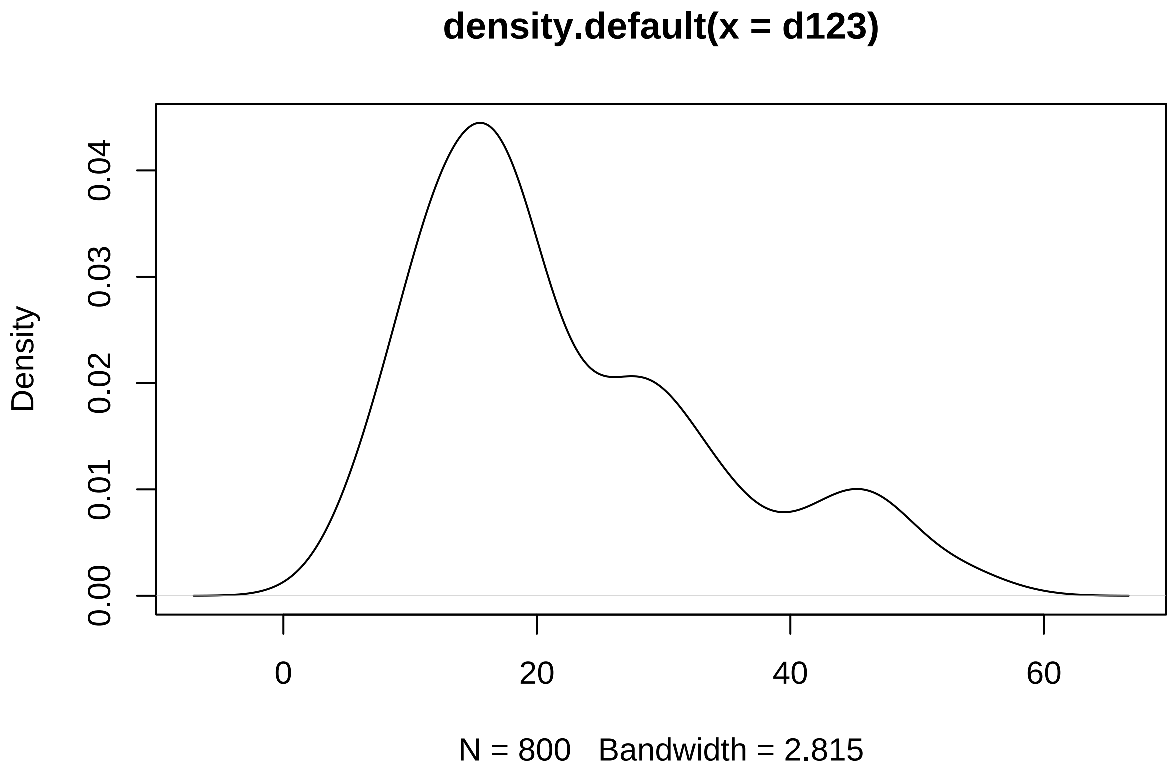
Make a function called norm_qq_plot() that takes a set of values):

```
norm_qq_plot <- function(values) {
  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)
}
```

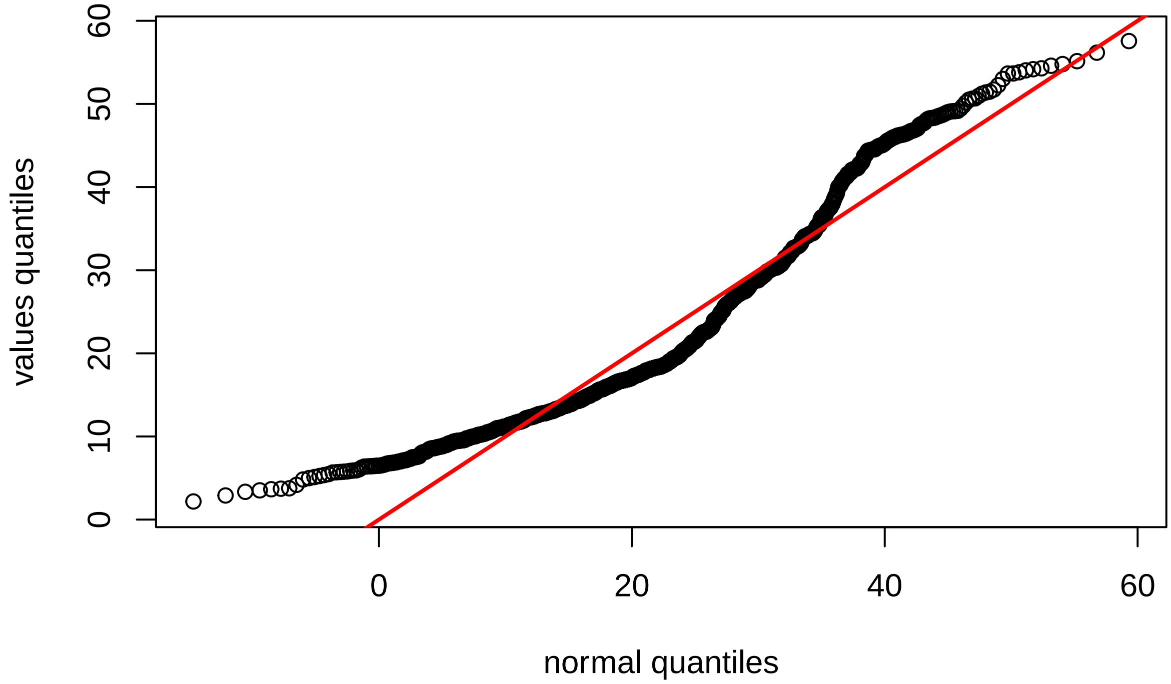
(b) Confirm that your function works by running it against the values of our d123 distribution from week 3 and checking that it looks like the plot on the right:

```
set.seed(978234)
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)

plot(density(d123))
```

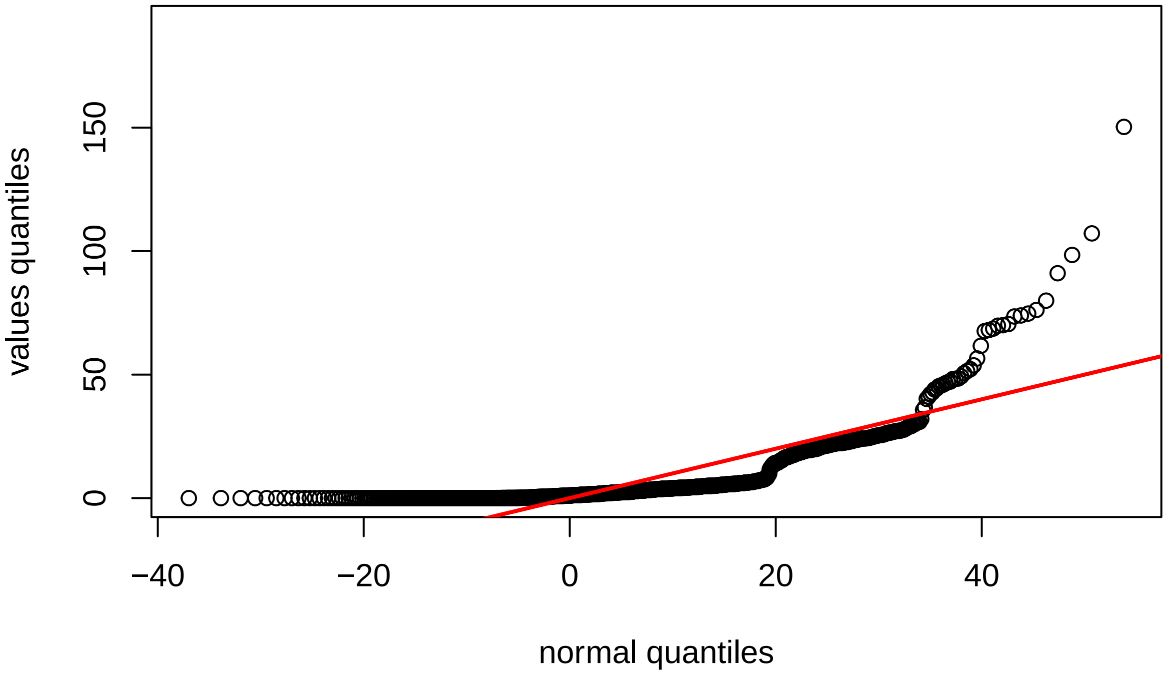


```
norm_qq_plot(d123)
```

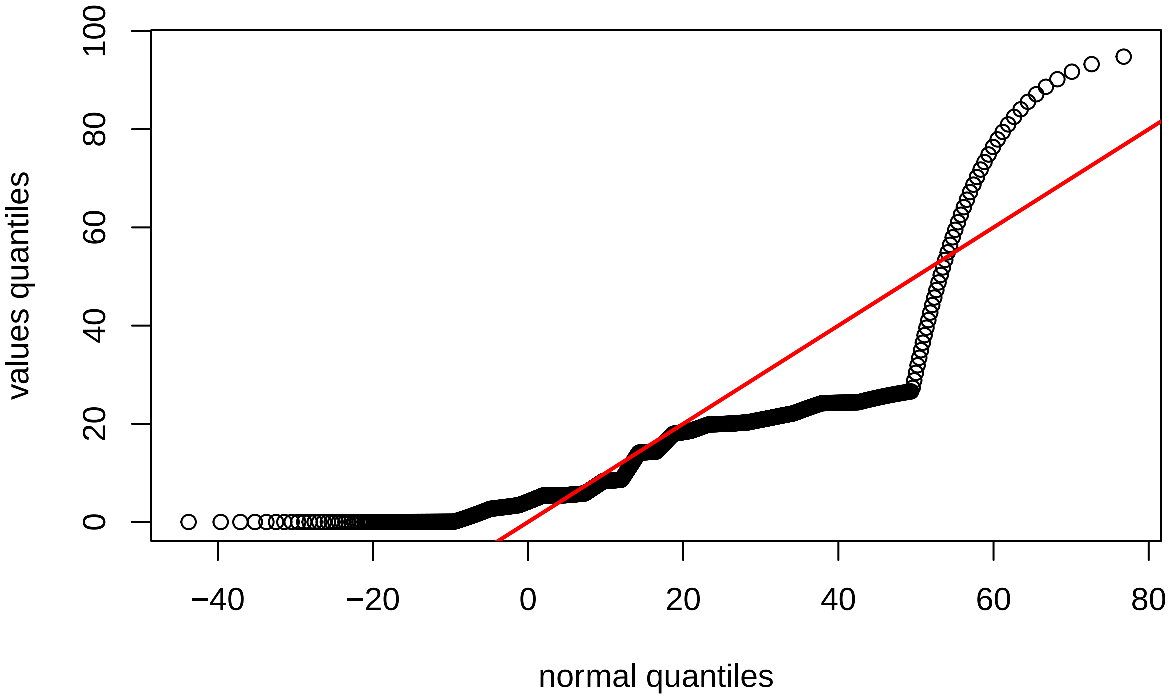


(c) Use your normal Q-Q plot function to check if the values from each of the CLEC and ILEC samples we compared in question 2 could be normally distributed. What's your conclusion?

```
norm_qq_plot(hosts$ILEC)
```



```
norm_qq_plot(hosts$CLEC)
```



- ILEC's plot is "Skwed Right" histogram, due to the upper tail is a little bit far away from red line.
- CLEC's plot is normally distributed, because of its symmetry.