

**Question 1.** (a) I pick `melt` function from `reshape` library. It's not hard to call. The usage is refer to [1].  
(b) To reshape the data with `melt` function, consider

---

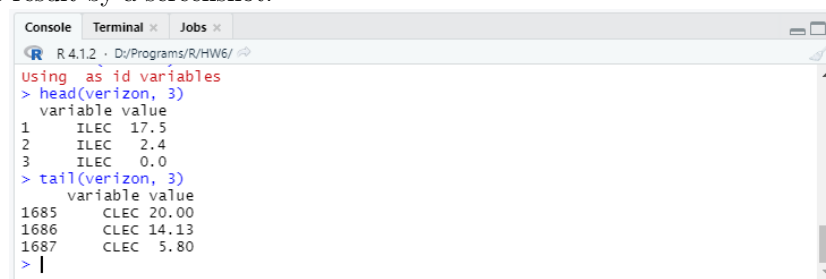
```
1 # Question 1 (b)
2 library("reshape")
3 page_loads <- read.csv(file="verizon_wide.csv") # read the file
4 verizon <- melt(page_loads,
5                 na.rm = TRUE,
6                 variable.name = "Type",
7                 value.name = "Time")
8 hosts <- split(x = verizon, f = verizon$variable)
```

---

(c) Use the command in console:

```
> head(verizon, 3)
> tail(verizon, 3)
```

I prefer to show the result by a screenshot.

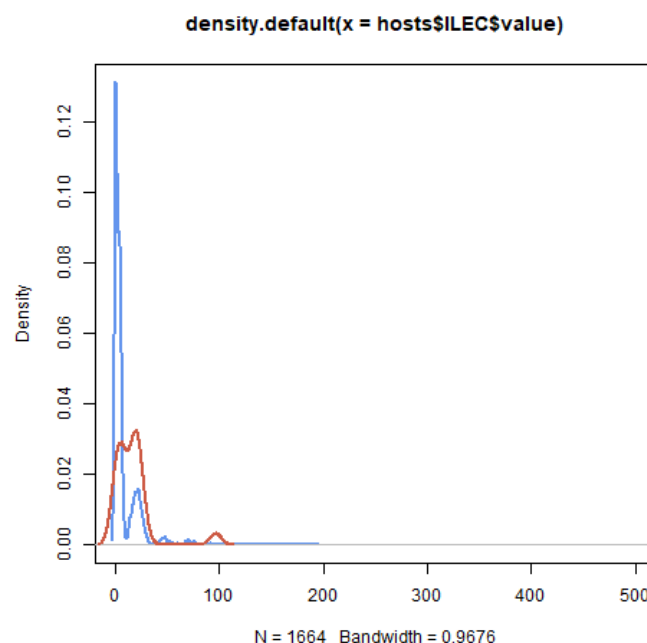


(d) Simply reuse the codes from the slides.

---

```
1 # Question 1 (d)
2 png(filename = "1d.png")
3 plot(density(hosts$ILEC$value), col="cornflowerblue", lwd=2, xlim=c(0, 500))
4 lines(density(hosts$CLEC$value), col="coral3", lwd=2)
5 legend(300, 0.5, lty=1, c("ILEC", "CLEC"), col=c("coral3", "cornflowerblue"))
6 dev.off()
```

---



**Question 2.** (a)  $H_0$ : the mean of response times for CLEC customers is equal to that of ILEC customers;  $H_1$ : The mean of response times for CLEC customers is greater than for ILEC customers.

(b) The code of  $t$ -test:

---

```
1 # Question 2 (b)
2 t.test(hosts$CLEC$value, hosts$ILEC$value, alt="greater",
3        var.equal=FALSE, conf.level=0.99) #(ii)
4 t.test(hosts$CLEC$value, hosts$ILEC$value, alt="greater",
5        var.equal=TRUE, conf.level=0.99) #(i)
```

---

(i) Assuming variances of the two populations are equal,  $t = 2.6125$ ,  $p\text{-value} = 0.004534$ . 99% CI is  $(0.8801387, \infty)$ . Since  $t$  is in this interval, we cannot reject  $H_0$ .

(ii) Assuming variances of the two populations are not equal,  $t = 1.9834$ ,  $df = 22.346$ ,  $p\text{-value} = 0.02987$ . 99% CI is  $(-2.130858, \infty)$ . Since  $t$  is in this interval, we cannot reject  $H_0$ .

(c) I use the following code to run the permutation test:

---

```
1 # Question 2 (c)
2 observed_diff <- mean(hosts$CLEC$value) - mean(hosts$ILEC$value)
3
4 permute_diff <- function(values, groups)
5 {
6   permuted <- sample(values, replace = FALSE)
7   grouped <- split(permuted, groups)
8   permuted_diff <- mean(grouped[[1]]) - mean(grouped[[2]])
9 }
10 nperms <- 1000
11 permuted_diffs <- replicate(nperms,
12                             permute_diff(verizon$value, verizon$variable))
```

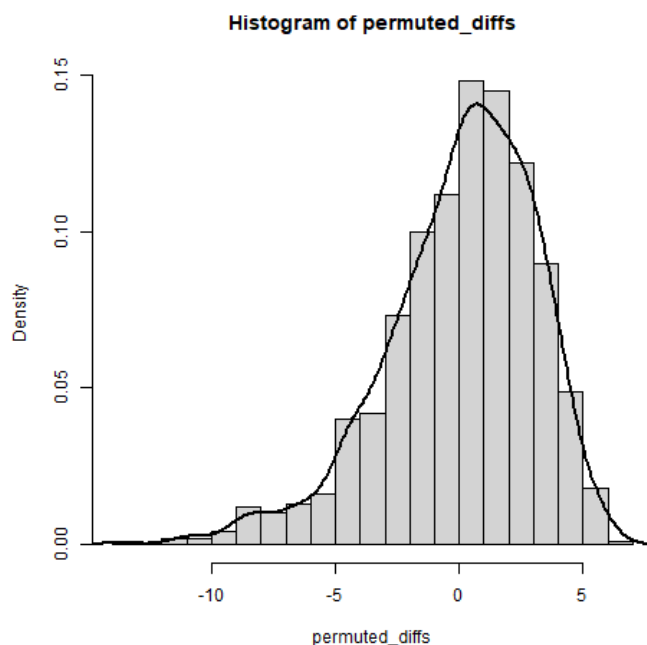
---

(i) The distribution of permuted differences can be drawn by

---

```
1 png(filename = "2c.png")
2 hist(permuted_diffs, breaks = "fd", probability = TRUE)
3 lines(density(permuted_diffs), lwd=2)
4 abline(v=observed_diff, lty="dashed")
5 dev.off()
```

---



The observed difference is roughly 8.09752, and it is larger than the boundary of this graph. Hence I think it's not suitable to plot on the diagram.

(ii) These  $p$ -values is computed by:

---

```
1 p_1tailed <- sum(permuted_diffs > observed_diff) / nperms
2 p_2tailed <- sum(abs(permuted_diffs) > observed_diff) / nperms
```

---

It gives one-tailed  $p$ -value = 0, two-tailed  $p$ -value = 0.02

(iii) Since the one-tailed  $p$ -value = 0, the permuted difference is same as observed difference in all 1000 permutation. Hence one can claim that we cannot reject  $H_0$  directly. ■

**Question 3.** (a) I prefer the vectorized form.

---

```
1 # Question 3 (a)
2 gt_eq <- function(a, b)
3 {
4   ifelse(a > b, 1, 0) + ifelse(a == b, 0.5, 0)
5 }
6 W <- sum(outer(hosts$CLEC$value, hosts$ILEC$value, FUN = gt_eq))
```

---

This gives  $W = 26820$ .

(b) The one-tailed  $p$ -value:

---

```
1 # Question 3 (b)
2 n1 <- length(hosts$CLEC$value) # 23
3 n2 <- length(hosts$ILEC$value) # 1664
4 wilcox_p_1tail <- 1 - pwilcox(W, n1, n2)
```

---

The one-tailed  $p$ -value is 0.0003688341.

(c) By calling the built-in `wilcox.test()` function, say

---

```
1 # Question 3 (c)
2 wilcox.test(hosts$CLEC$value,
3             hosts$ILEC$value,
4             alternative = "greater",
5             conf.level=0.01)
```

---

The result shows  $W = 26820$  and the one-tailed  $p$ -value is 0.0004565.  $W$  is same with the result of hand-made approach in (a).

(d) Since both the  $p$ -values obtained in (c) and (d) is smaller than 0.01, we cannot reject the null hypothesis. ■

**Question 4.** (a) The complete implementation of the required function is:

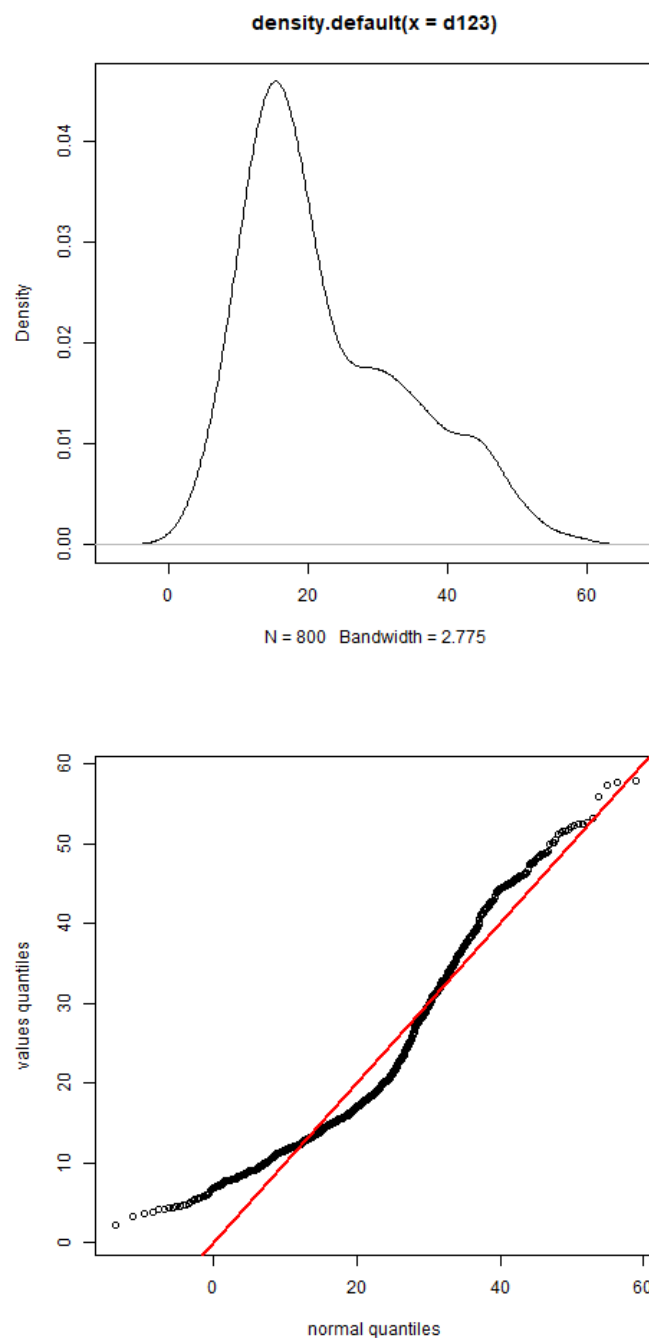
---

```
1 # Question 4 (a)
2 norm_qq_plot <- function(values)
3 {
4   probs1000 <- seq(0, 1, 0.001)
5   q_vals <- quantile(values, probs=probs1000)
6   q_norm <- qnorm(probs1000, mean=mean(values), sd=sd(values))
7   plot(q_norm, q_vals, xlab="normal quantiles", ylab="values quantiles")
8   abline(a=0, b=1, col="red", lwd=2)
9 }
```

---

(b) The plots show that my implementation is correct.

```
1 # Question 4 (b)
2 set.seed(48763)
3 d1 <- rnorm(n=500, mean=15, sd=5)
4 d2 <- rnorm(n=200, mean=30, sd=5)
5 d3 <- rnorm(n=100, mean=45, sd=5)
6 d123 <- c(d1, d2, d3)
7
8 png(filename = "4b-1.png")
9 plot(density(d123))
10 dev.off()
11 png(filename = "4b-2.png")
12 norm_qq_plot(d123)
13 dev.off()
```



[2] suggests that `d123` is not normally distributed. It looks more likely a “bimodal” distribution.

(c) Let’s plot the density and Q-Q plot first.

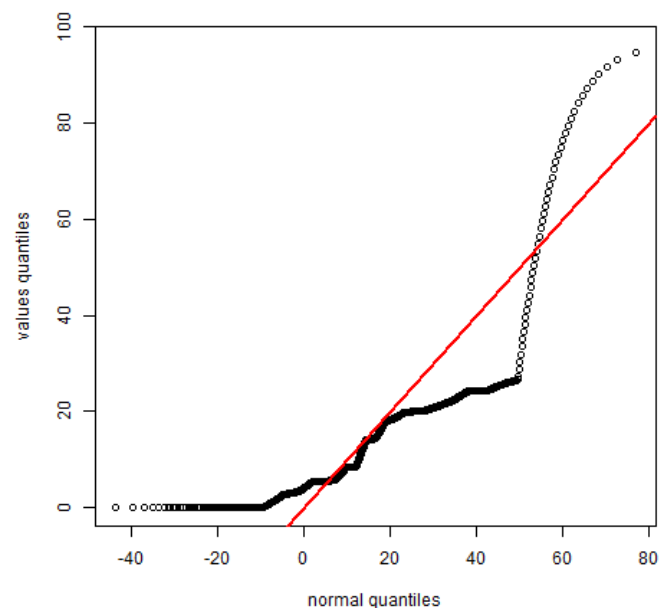
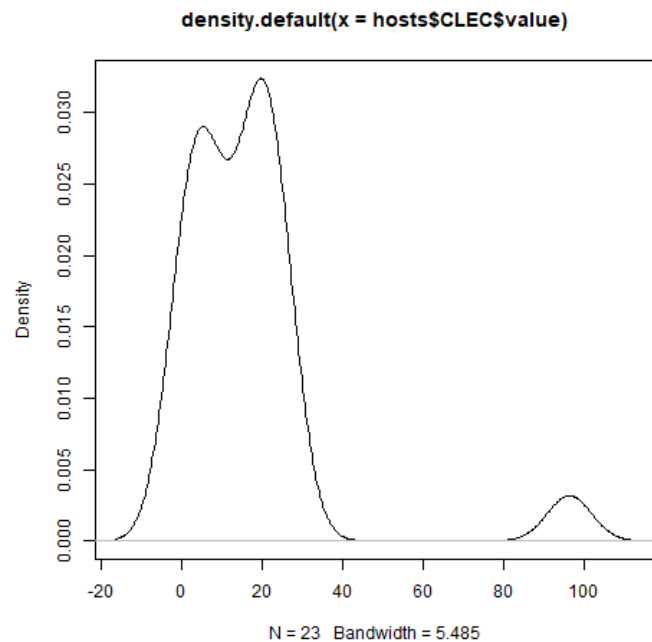
---

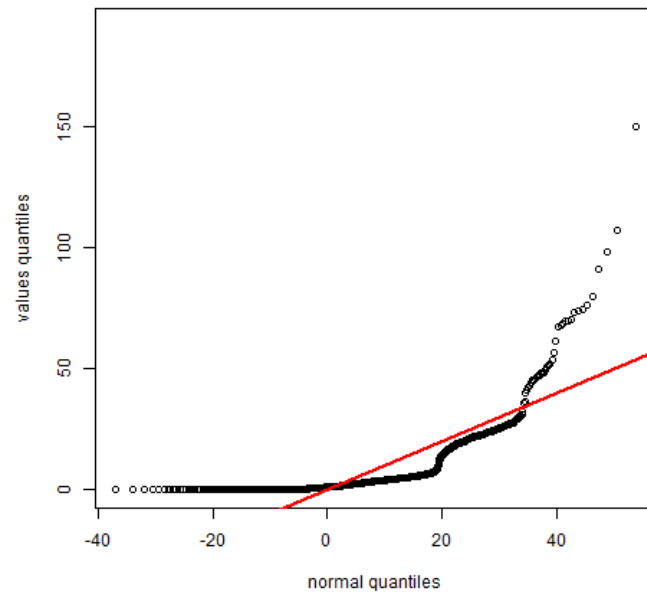
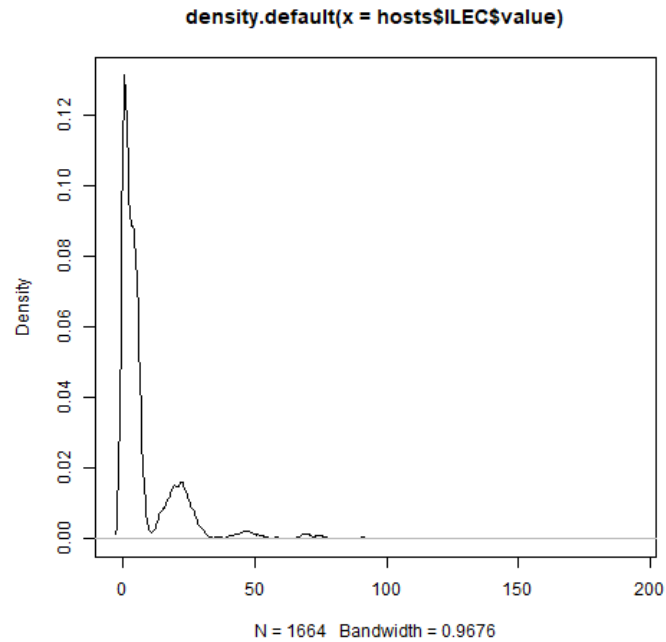
```

1 # Question 4 (c)
2 png(filename = "4c-CLEC-density.png")
3 plot(density(hosts$CLEC$value))
4 dev.off()
5
6 png(filename = "4c-CLEC-qq.png")
7 norm_qq_plot(hosts$CLEC$value)
8 dev.off()
9
10 png(filename = "4c-ILEC-density.png")
11 plot(density(hosts$ILEC$value))
12 dev.off()
13
14 png(filename = "4c-ILEC-qq.png")
15 norm_qq_plot(hosts$ILEC$value)
16 dev.off()

```

---





I really don't think any of CLEC or ILEC follows a normal distribution by any means, neither by the density plot nor the Q-Q plot. Perhaps ILEC follows a bimodal distribution. I guess CLEC may be approximate by a Gaussian-mixture model.

■

## References

- [1] reshape2 (version 1.4.4), RDocumentation,  
<https://www.rdocumentation.org/packages/reshape2/versions/1.4.4/topics/melt>
- [2] A Q-Q Plot Dissection Kit, <https://seankross.com/2016/02/29/A-Q-Q-Plot-Dissection-Kit.html>