Lab 4

Irene Jacob

2020-09-26

Lab 4

The data

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------------------------------

## v tibble  3.0.3     v purrr   0.3.4
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ---------------------------------------------------------------------------------------
## x tidyr::complete() masks RCurl::complete()
## x dplyr::filter()   masks stats::filter()
## x dplyr::lag()      masks stats::lag()
## x purrr::map()      masks maps::map()
```

```
library(openintro)
head(fastfood)
```

```
## # A tibble: 6 x 17
##   restaurant item  calories cal_fat total_fat sat_fat trans_fat cholesterol
##   <chr>      <chr>    <dbl>   <dbl>     <dbl>   <dbl>     <dbl>       <dbl>
## 1 Mcdonalds  Arti~      380      60         7       2       0            95
## 2 Mcdonalds  Sing~      840     410        45      17       1.5         130
## 3 Mcdonalds  Doub~     1130     600        67      27       3           220
## 4 Mcdonalds  Gril~      750     280        31      10       0.5         155
## 5 Mcdonalds  Cris~      920     410        45      12       0.5         120
## 6 Mcdonalds  Big ~      540     250        28      10       1            80
## # ... with 9 more variables: sodium <dbl>, total_carb <dbl>, fiber <dbl>,
## #   sugar <dbl>, protein <dbl>, vit_a <dbl>, vit_c <dbl>, calcium <dbl>,
## #   salad <chr>
```

Let's first focus on just products from McDonalds and Dairy Queen.

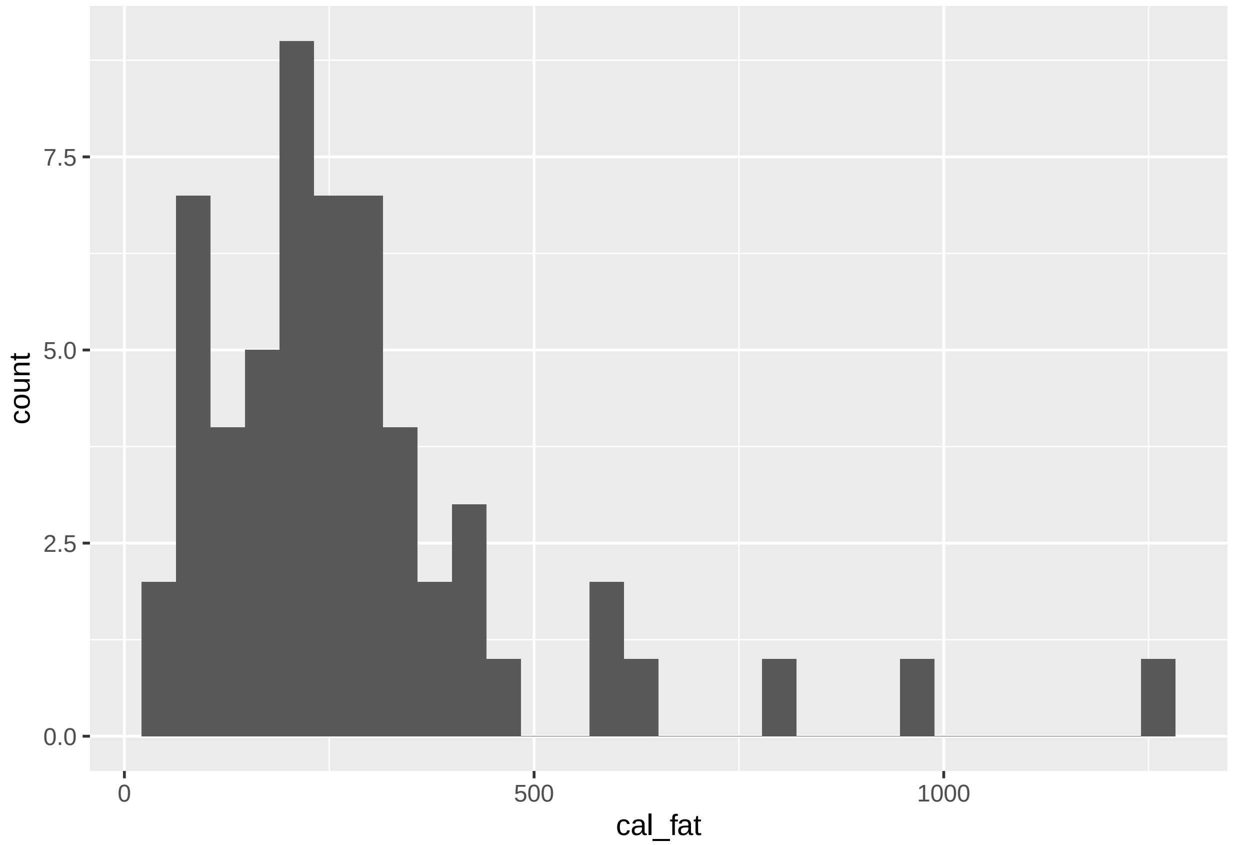
```
mcdonalds <- fastfood %>%
  filter(restaurant == "Mcdonalds")
dairy_queen <- fastfood %>%
  filter(restaurant == "Dairy Queen")
```

Exercise 1:

Make a plot (or plots) to visualize the distributions of the amount of calories from fat of the options from these two restaurants. How do their centers, shapes, and spreads compare?

Plot for Mcdonalds:

```
ggplot(data = mcdonalds, aes(cal_fat)) +
  geom_histogram()
```

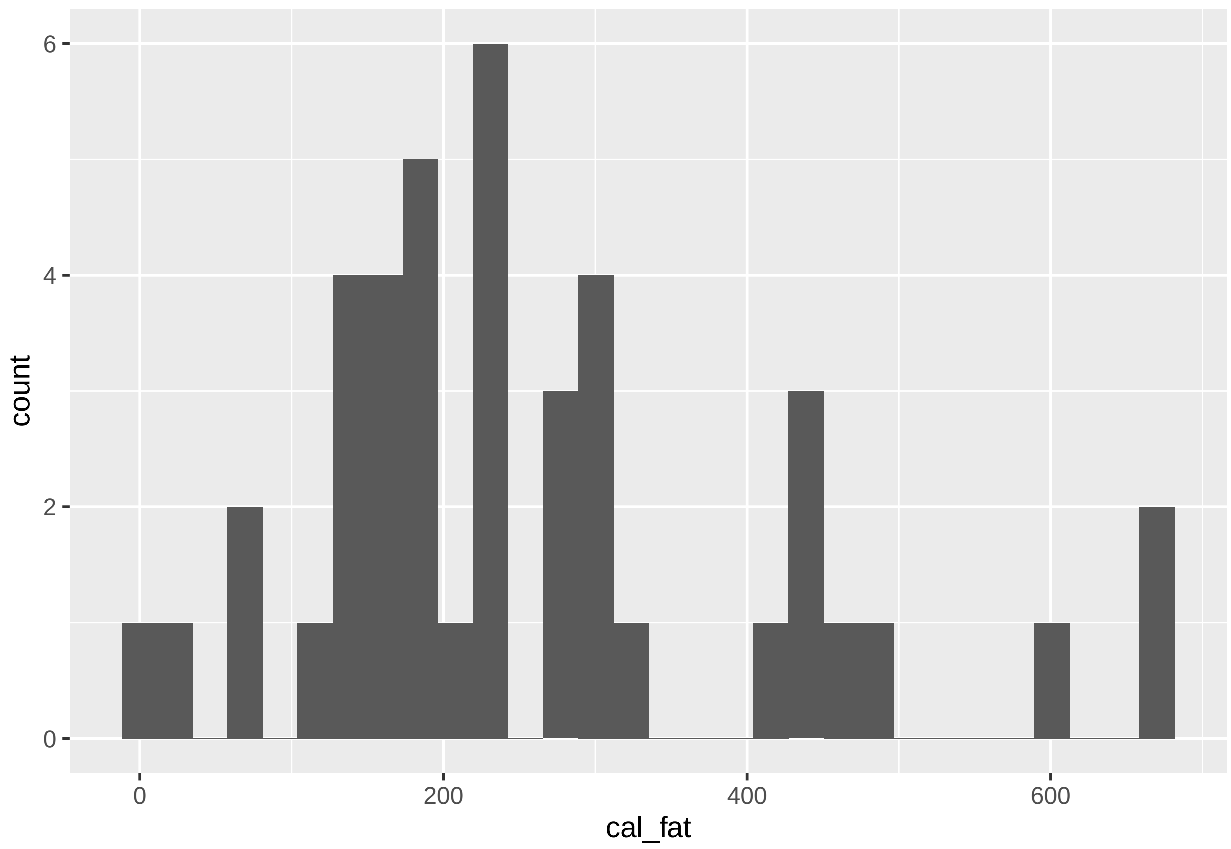


```
summary(mcdonalds$cal_fat)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    50.0   160.0   240.0   285.6   320.0  1270.0
```

Plot for Diary Queen:

```
ggplot(data = dairy_queen, aes(cal_fat)) +
  geom_histogram()
```



```
summary(dairy_queen$cal_fat)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##     0.0   160.0   220.0   260.5   310.0   670.0
```

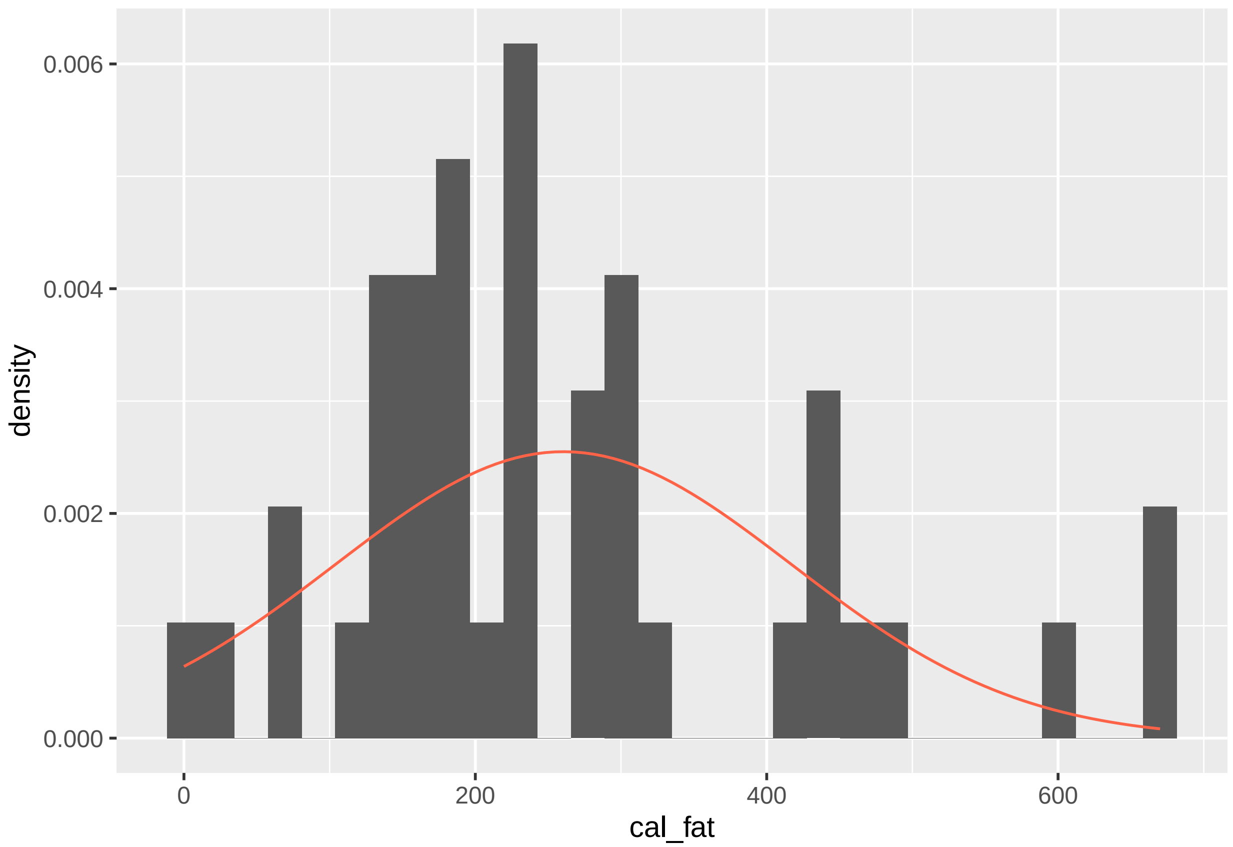
Mcdonalds and Diary_Queen are unimodal but Mcdonalds is right skewed. Mcdonalds has the larger value for max, min and mean when compared to Diary_Queen.

The normal distribution

```
dqmean <- mean(dairy_queen$cal_fat)
dqsd   <- sd(dairy_queen$cal_fat)
```

```
ggplot(data = dairy_queen, aes(x = cal_fat)) +
        geom_blank() +
        geom_histogram(aes(y = ..density..)) +
        stat_function(fun = dnorm, args = c(mean = dqmean, sd = dqsd), col = "tomato")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



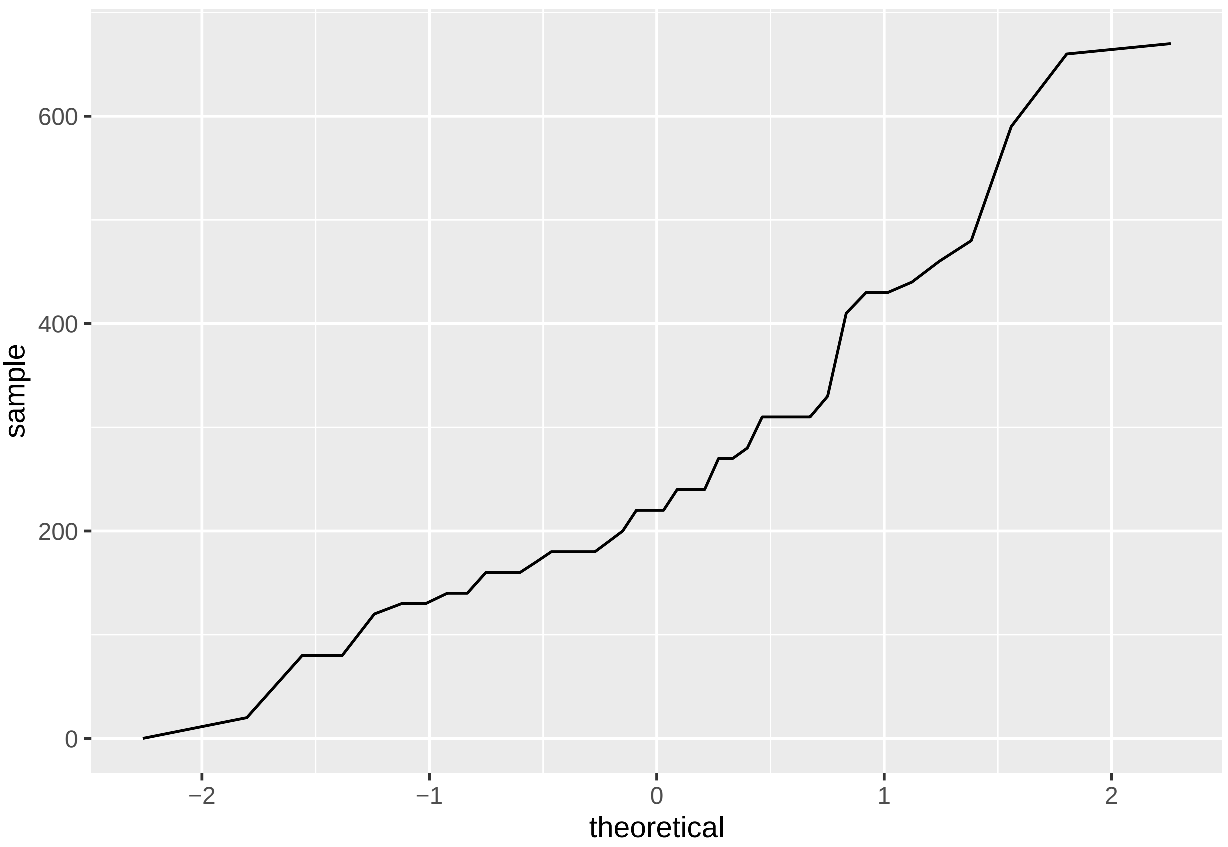
Exercise 2:

Based on the this plot, does it appear that the data follow a nearly normal distribution?

The values are spread out so it is an almost normal or nearly normal distribution.

Evaluating the normal distribution

```
ggplot(data = dairy_queen, aes(sample = cal_fat)) +
  geom_line(stat = "qq")
```

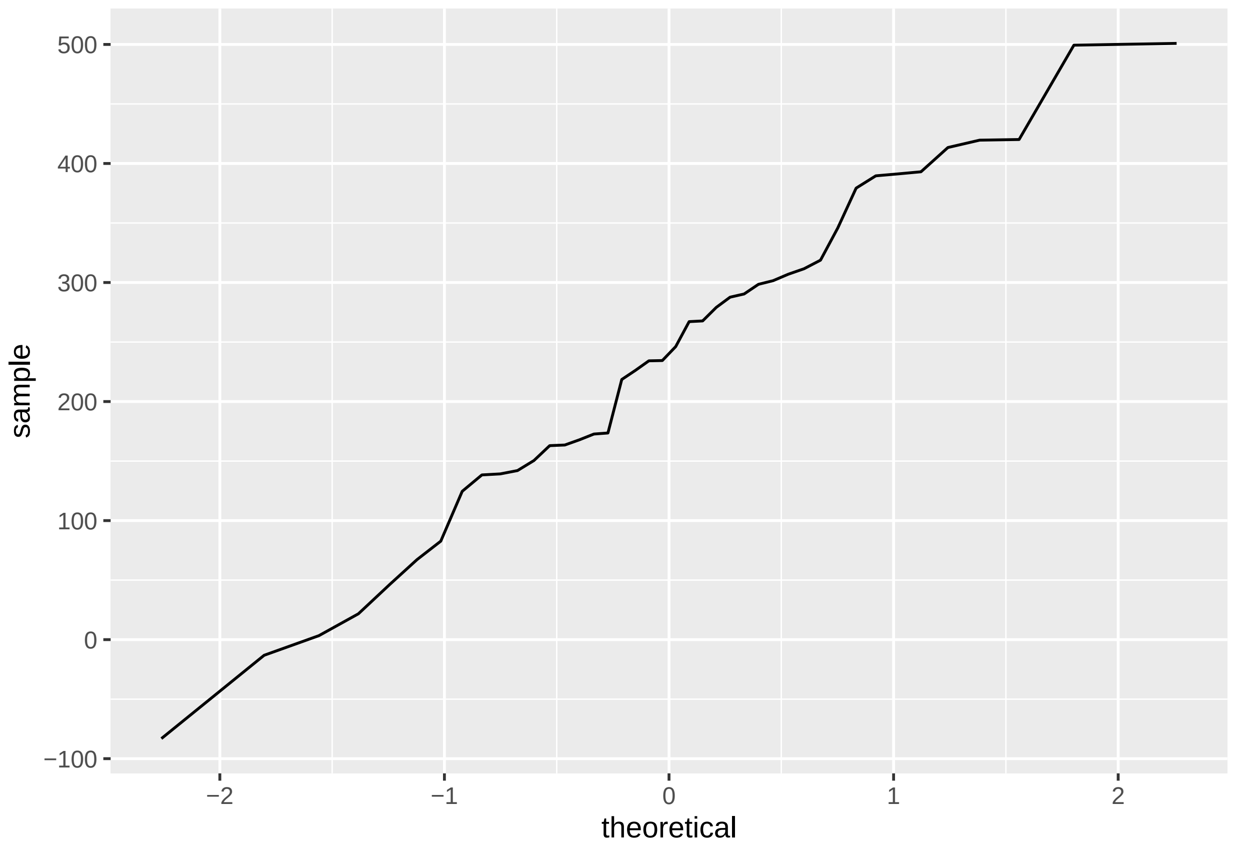


```
sim_norm <- rnorm(n = nrow(dairy_queen), mean = dqmean, sd = dqsd)
```

Exercise 3:

Make a normal probability plot of sim_norm. Do all of the points fall on the line? How does this plot compare to the probability plot for the real data? (Since sim_norm is not a dataframe, it can be put directly into the sample argument and the data argument can be dropped.)

```
ggplot(data = NULL, aes(sample = sim_norm)) +
  geom_line(stat = "qq")
```



All the points do not fall on the same line. This plot and the plot for the real data are not same but they look similar.

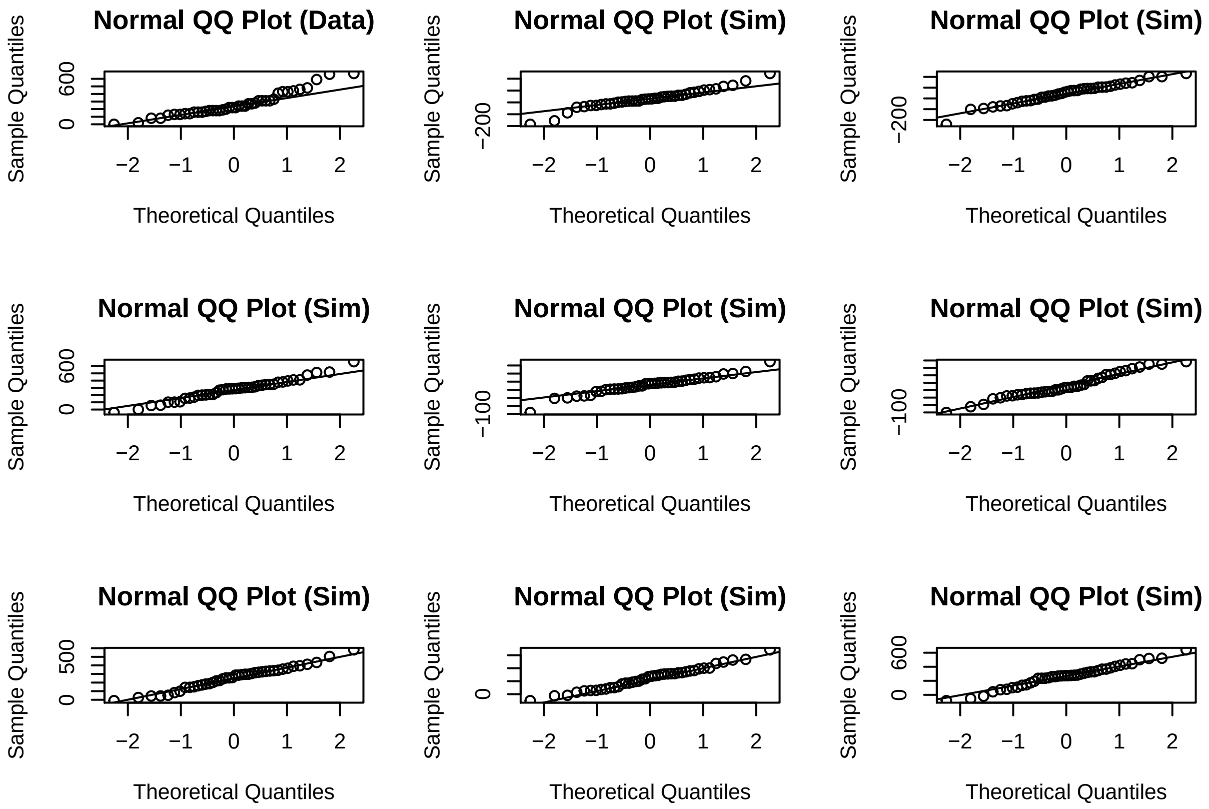
Even better than comparing the original plot to a single plot generated from a normal distribution is to compare it to many more plots using the following function. It shows the Q-Q plot corresponding to the original data in the top left corner, and the Q-Q plots of 8 different simulated normal data. It may be helpful to click the zoom button in the plot window.

Exercise 4:

Does the normal probability plot for the calories from fat look similar to the plots created for the simulated data? That is, do the plots provide evidence that the cal_fat are nearly normal?

The simulated data plot and the normal probability plot are similar which can be seen below:

```
qqnormsim(dairy_queen$cal_fat)
```

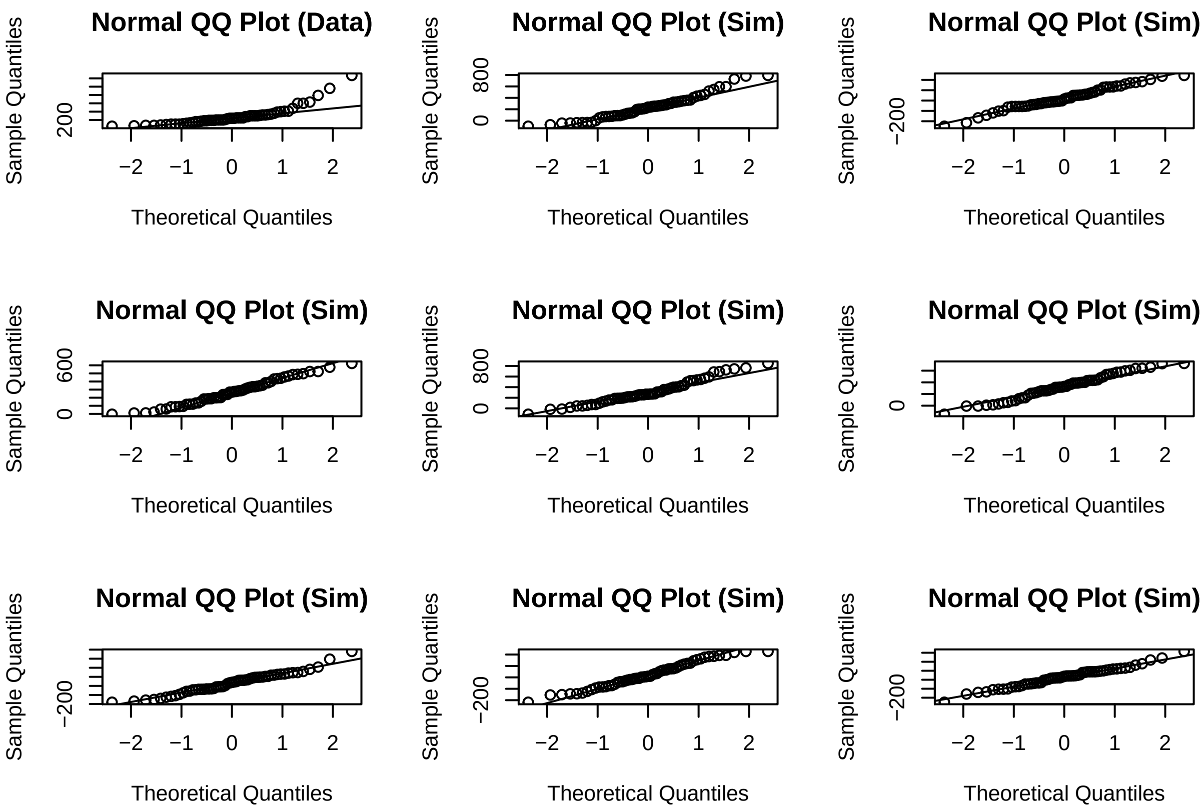


Exercise 5:

Using the same technique, determine whether or not the calories from McDonald's menu appear to come from a normal distribution.

The mcdonalds plot is almost or nearly normal as can be seen below.

```r
qqnormsim(mcdonalds$cal_fat)
```



Normal probabilities

```
1 - pnorm(q = 600, mean = dqmean, sd = dqsd)
```

```
## [1] 0.01501523
```

```
dairy_queen %>%
  filter(cal_fat > 600) %>%
  summarise(percent = n() / nrow(dairy_queen))
```

```
## # A tibble: 1 x 1
##   percent
##     <dbl>
## 1  0.0476
```

Exercise 6:

Write out two probability questions that you would like to answer about any of the restaurants in this dataset. Calculate those probabilities using both the theoretical normal distribution as well as the empirical distribution (four probabilities in all). Which one had a closer agreement between the two methods?

6.1. What is the probability that a randomly chosen food has cal_fat less than 100

```
m_ff <- mean(fastfood$cal_fat)
sd_ff <- sd(fastfood$cal_fat)

pnorm(q = 100, mean = m_ff, sd = sd_ff)
```

```
## [1] 0.2020902
```

```
fastfood %>%
  filter(cal_fat < 100) %>%
  summarise(percent = n() / nrow(fastfood))
```

```
## # A tibble: 1 x 1
##   percent
##     <dbl>
## 1   0.169
```

6.2. What is the probability that a randomly chosen food has calories more than 900

```
m_ff1 <- mean(fastfood$calories)
sd_ff1 <- sd(fastfood$calories)

1-pnorm(q = 900, mean = m_ff1, sd = sd_ff1)
```

```
## [1] 0.09564043
```

```
fastfood %>%
  filter(calories > 900) %>%
  summarise(percent = n() / nrow(fastfood))
```

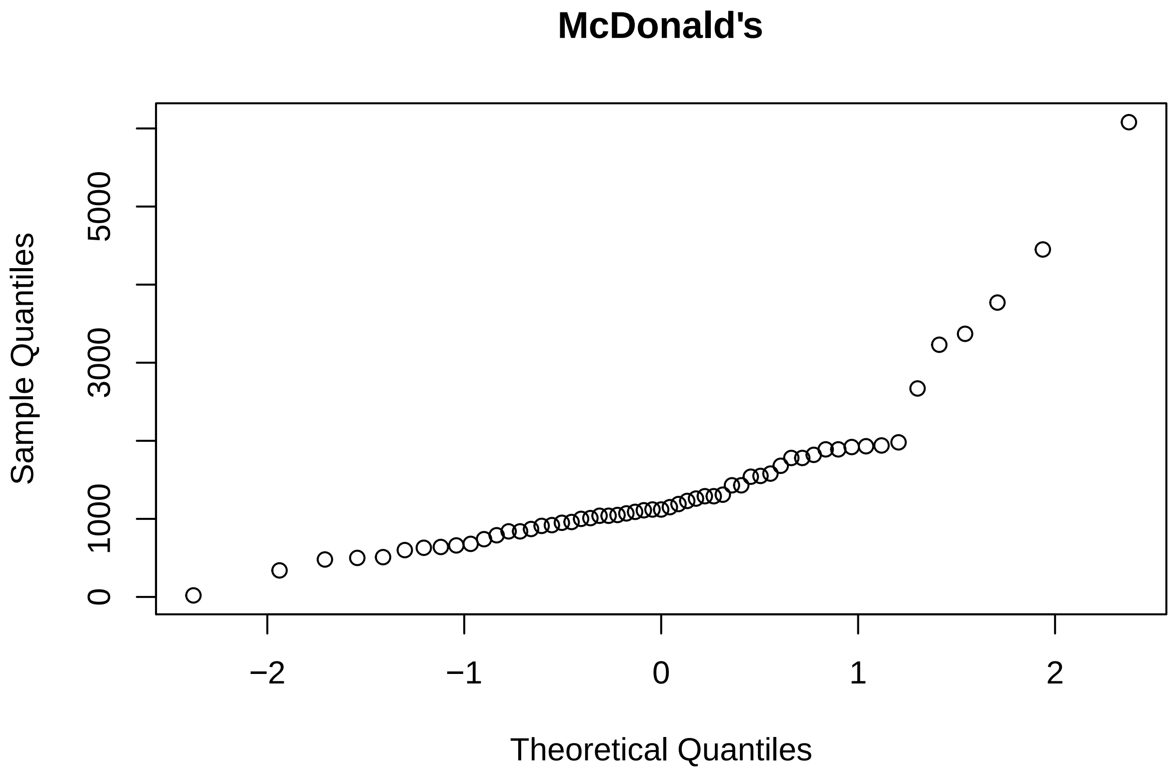
```
## # A tibble: 1 x 1
##   percent
##     <dbl>
## 1  0.0951
```

More Practice

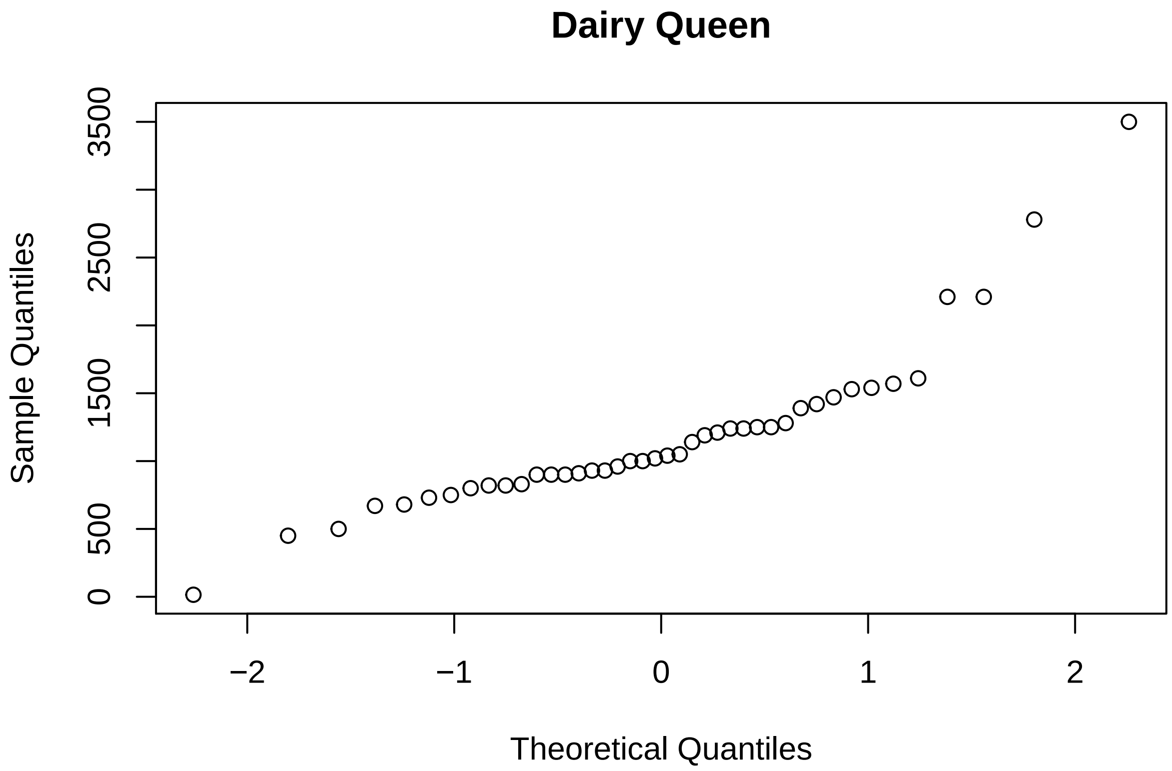
Exercise 7:

Now let's consider some of the other variables in the dataset. Out of all the different restaurants, which ones' distribution is the closest to normal for sodium?

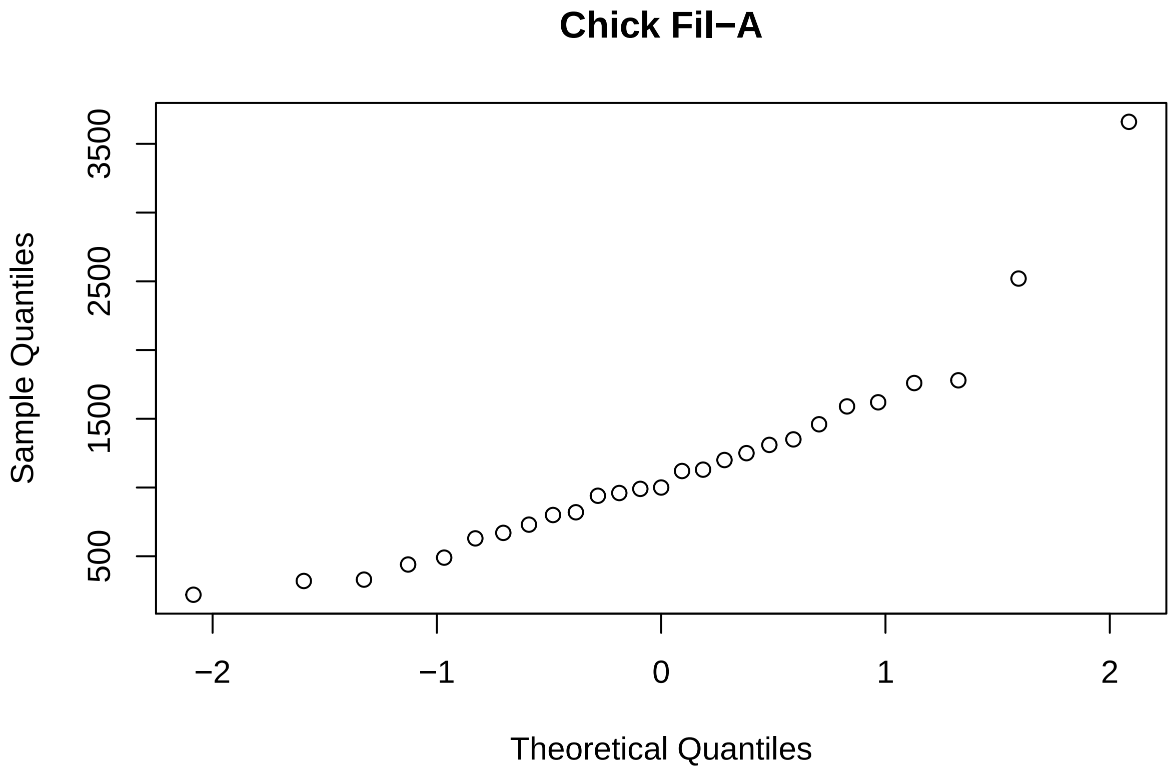
```
chick_fil_a <- fastfood %>%
  filter(restaurant == "Chick Fil-A")
sonic <- fastfood %>%
  filter(restaurant == "Sonic")
arbys <- fastfood %>%
  filter(restaurant == "Arbys")
burgerking <- fastfood %>%
  filter(restaurant == "Burger King")
subway <- fastfood %>%
  filter(restaurant == "Subway")
taco_bell <- fastfood %>%
  filter(restaurant == "Taco Bell")


qqnorm(mcdonalds$sodium, main = "McDonald's")
```



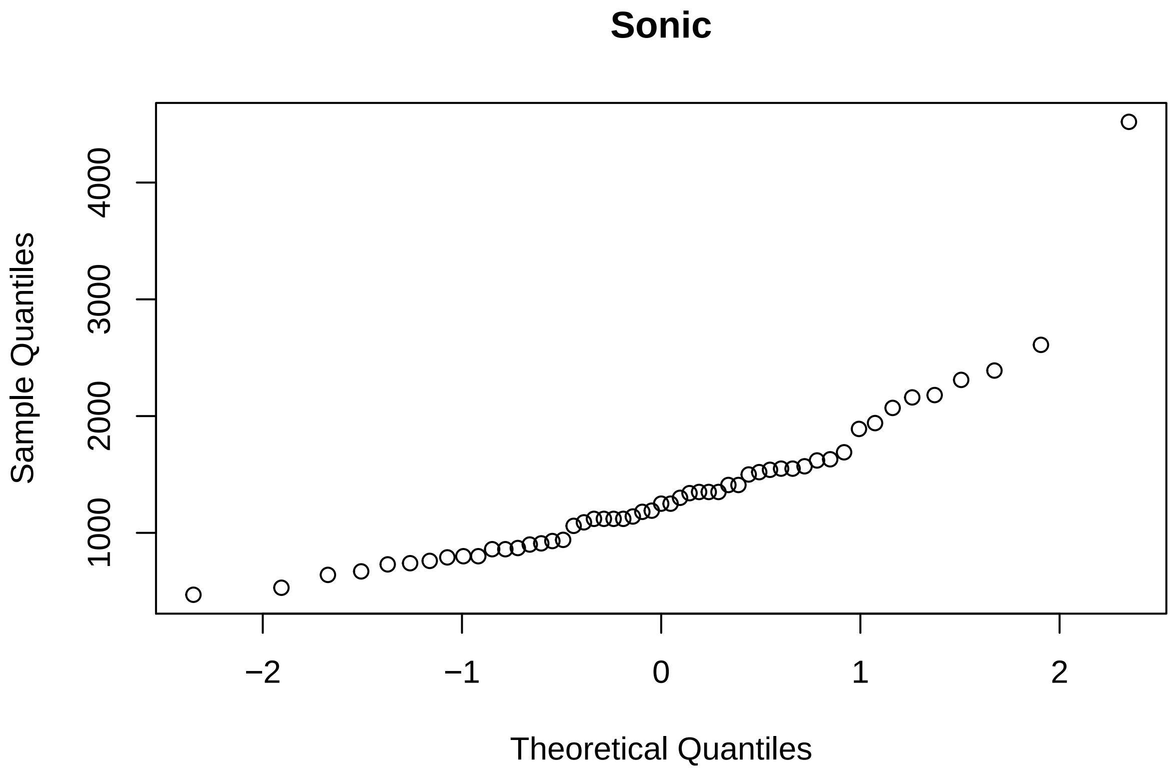
```
qqnorm(dairy_queen$sodium, main = "Dairy Queen")
```



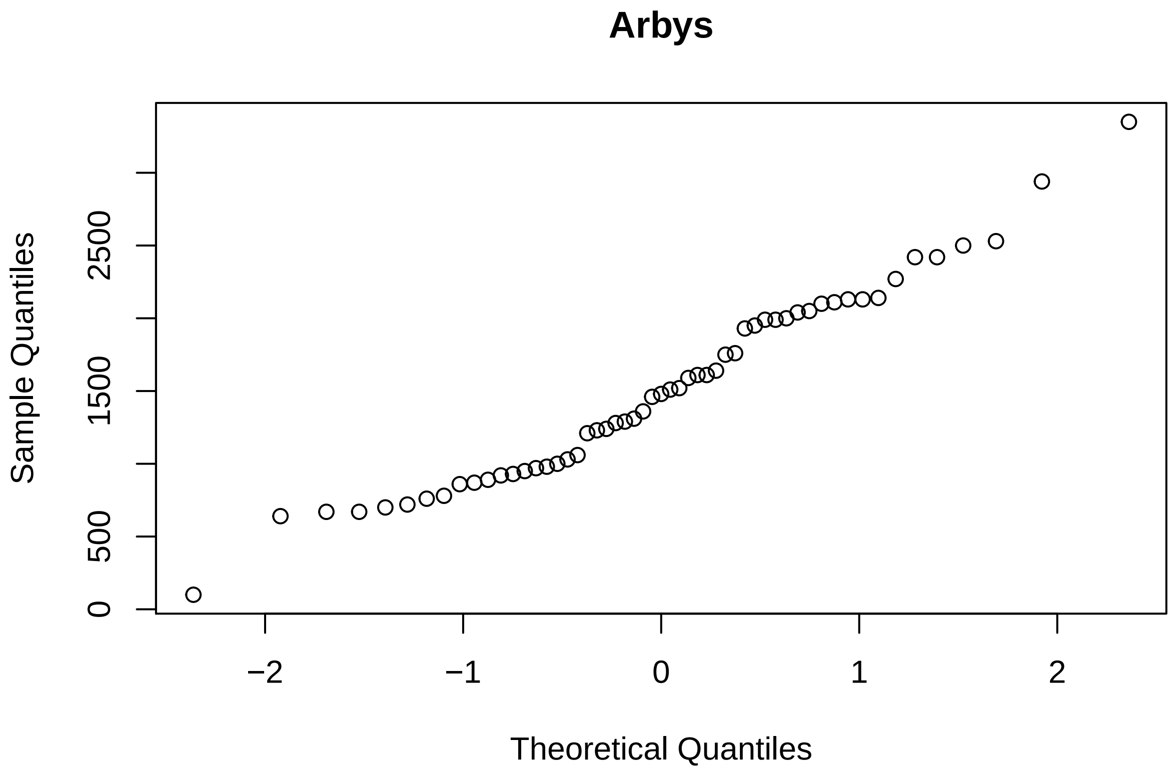
```
qqnorm(chick_fil_a$sodium, main = "Chick Fil-A")
```



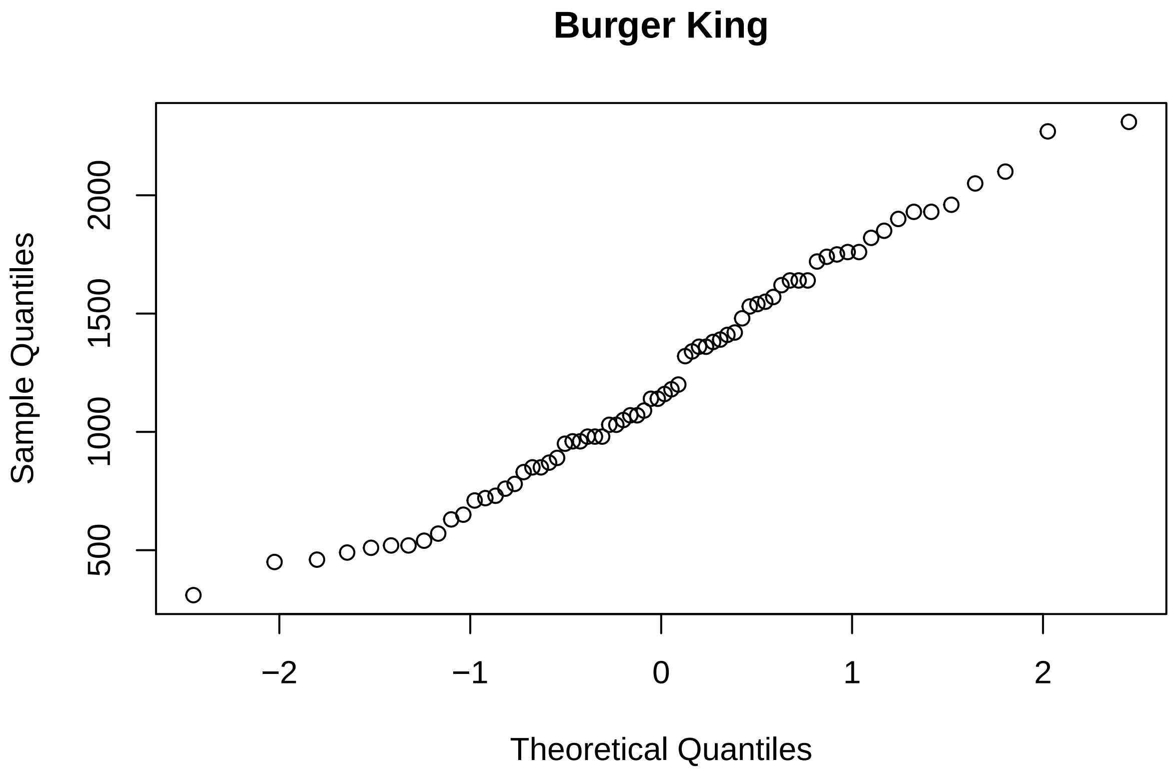
```
qqnorm(sonic$sodium, main = "Sonic")
```



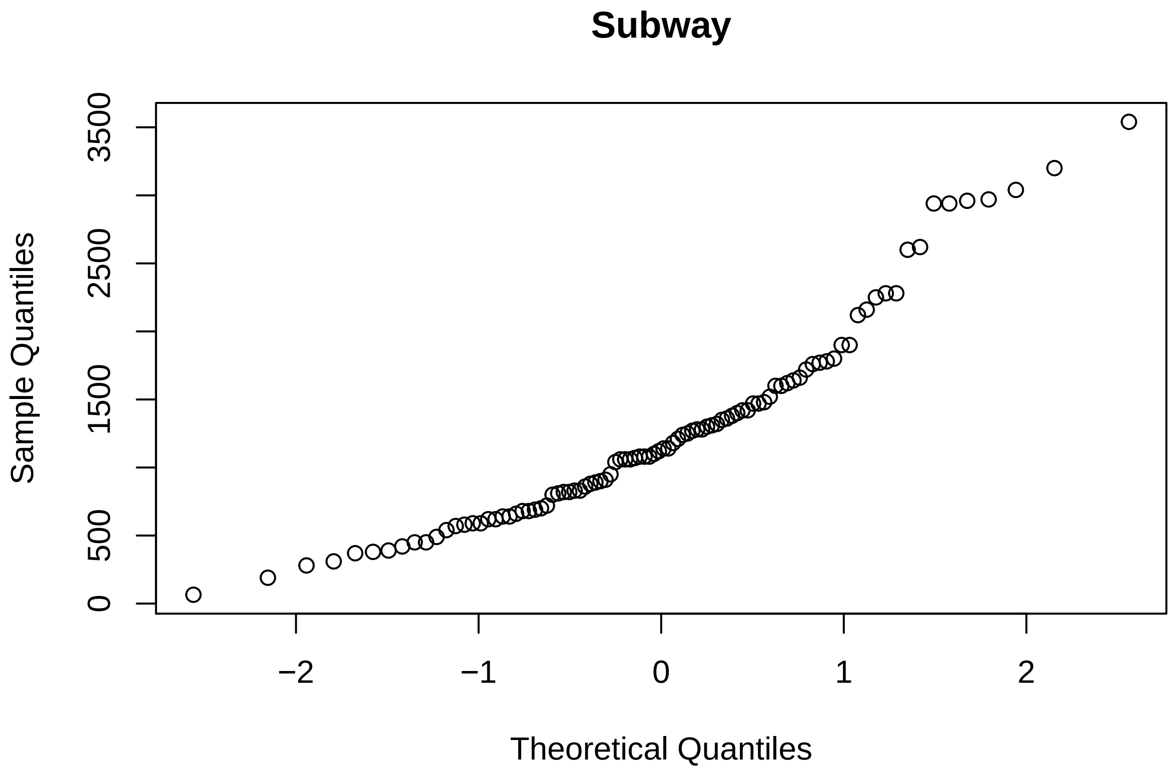
```
qqnorm(arbys$sodium, main = "Arbys")
```



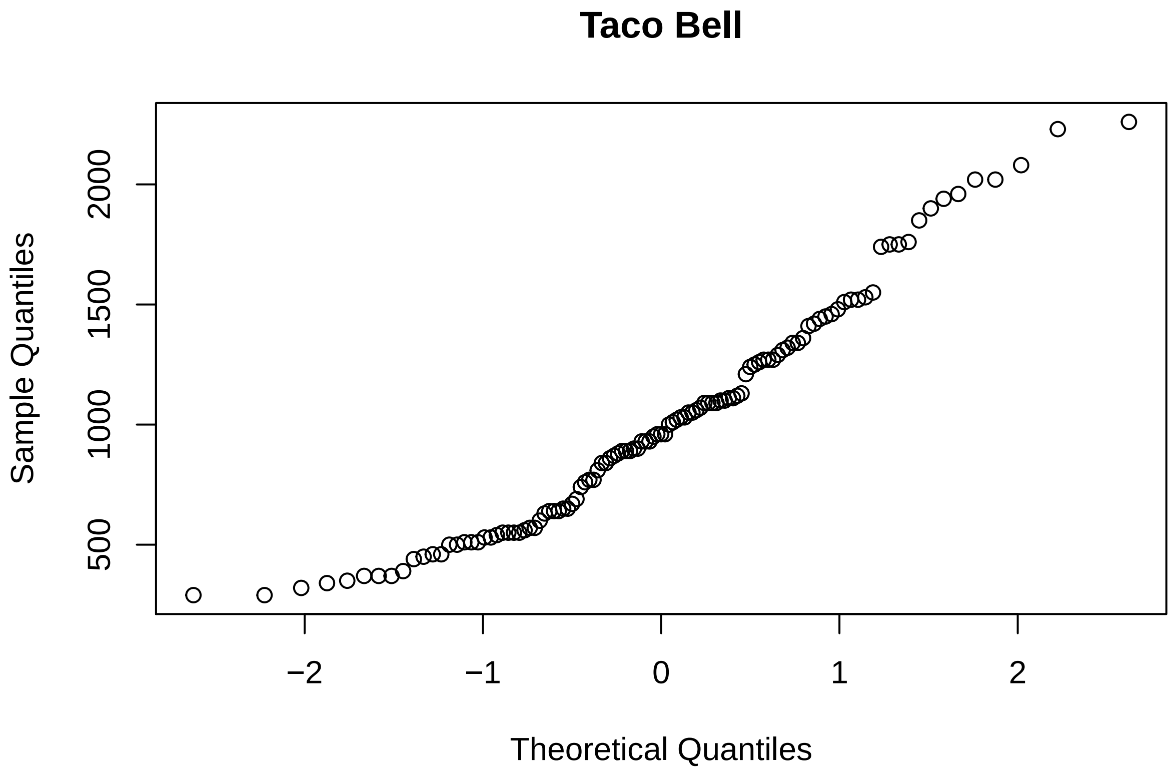
```
qqnorm(burgerking$sodium, main = "Burger King")
```



```r
qqnorm(subway$sodium, main = "Subway")
```



```r
qqnorm(taco_bell$sodium, main = "Taco Bell")
```



Exercise 8:

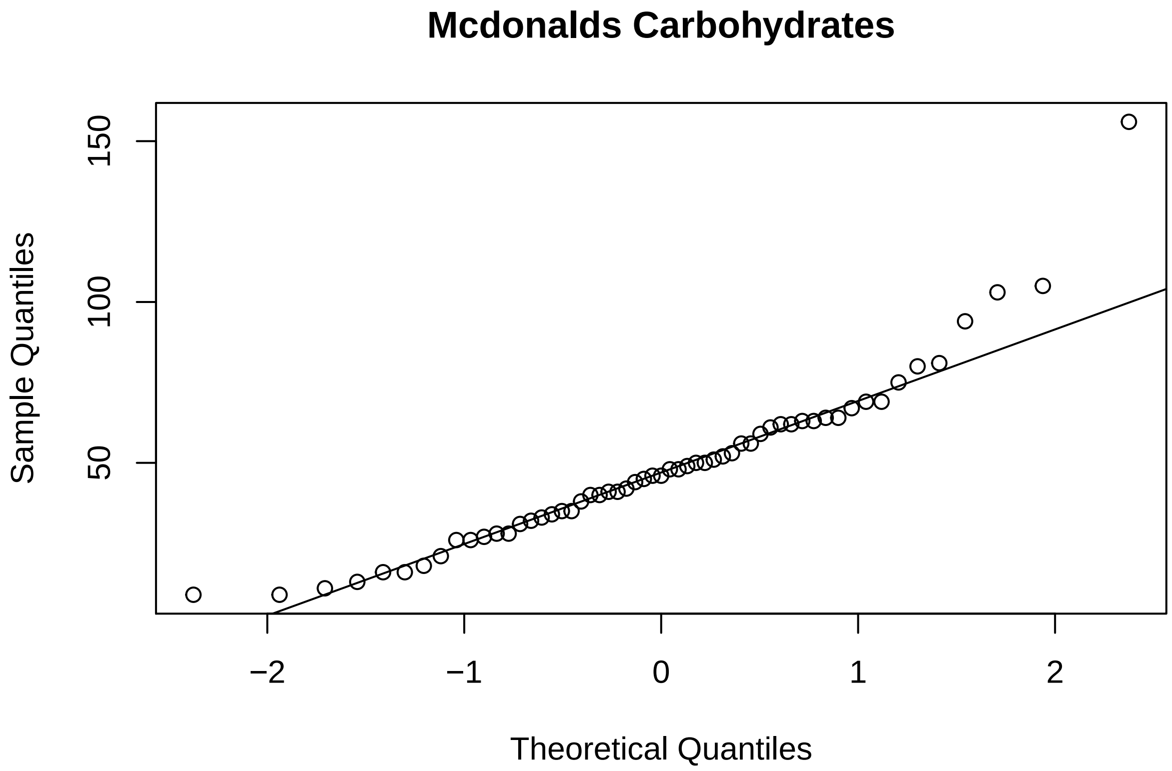
Note that some of the normal probability plots for sodium distributions seem to have a stepwise pattern. why do you think this might be the case?

Each fastfood restaurant provides a number of items which vary in their sodium level. This causes the stepwise pattern in the normal distribution plots.

Exercise 9:

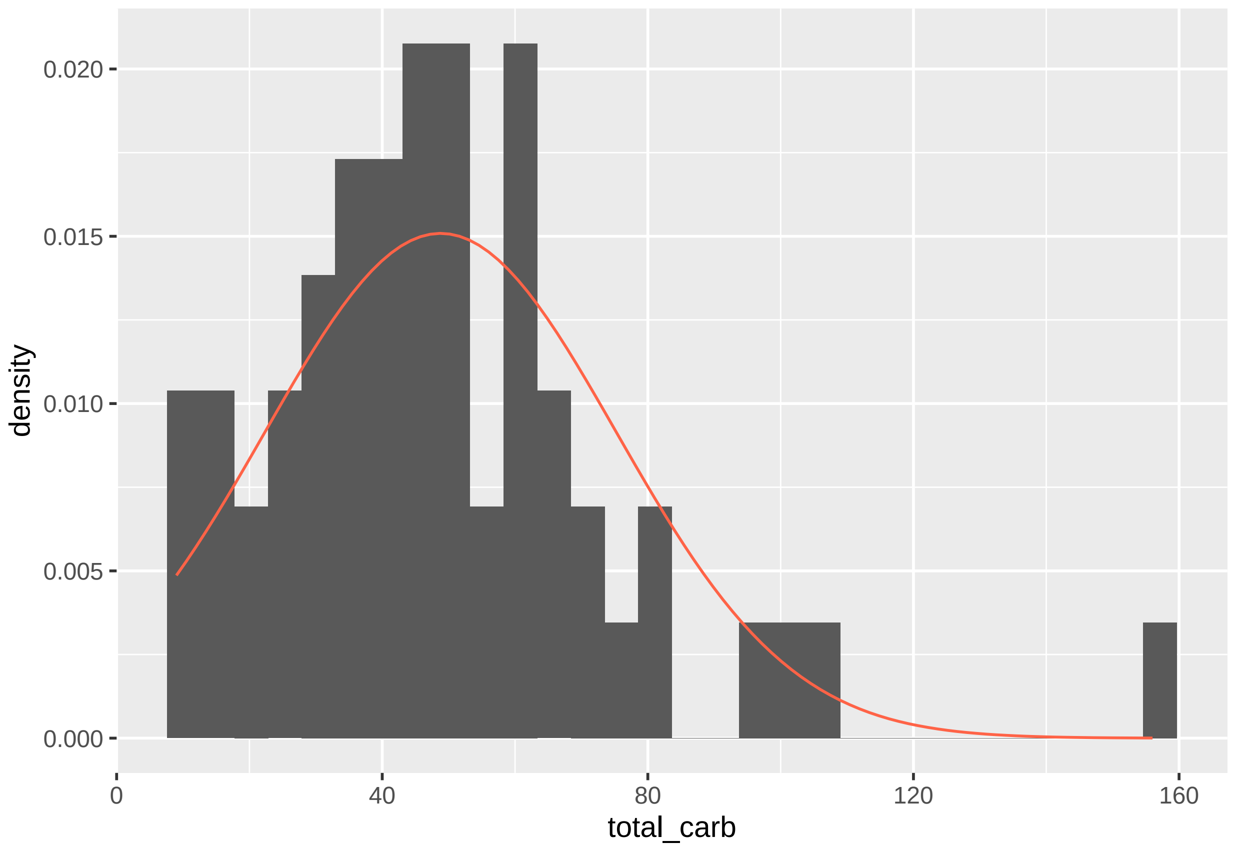
As you can see, normal probability plots can be used both to assess normality and visualize skewness. Make a normal probability plot for the total carbohydrates from a restaurant of your choice. Based on this normal probability plot, is this variable left skewed, symmetric, or right skewed? Use a histogram to confirm your findings.

```
new_m <- mean(mcdonalds$total_carb)
new_sd <- sd(mcdonalds$total_carb)

qqnorm(mcdonalds$total_carb, main = "Mcdonalds Carbohydrates")
qqline(mcdonalds$total_carb)
```



```
ggplot(data = mcdonalds, aes(total_carb)) +
  geom_blank() +
  geom_histogram(aes(y = ..density..)) +
  stat_function(fun = dnorm, args = c(mean = new_m, sd = new_sd), col = "tomato")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



It is clear that the distribution is right skewed.