CUNY SPS DATA606 Lab4 - Normal distribution

Chinedu Onyeka

September 23rd, 2021

Getting Started

Let's load the packages.

```
library(tidyverse)
library(openintro)
```

```
library(tidyverse)
library(openintro)
data("fastfood", package='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  Artisan G~      380      60         7       2       0            95
## 2 Mcdonalds  Single Ba~      840     410        45      17       1.5         130
## 3 Mcdonalds  Double Ba~     1130     600        67      27       3           220
## 4 Mcdonalds  Grilled B~      750     280        31      10       0.5         155
## 5 Mcdonalds  Crispy Ba~      920     410        45      12       0.5         120
## 6 Mcdonalds  Big Mac          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>
```

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

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?

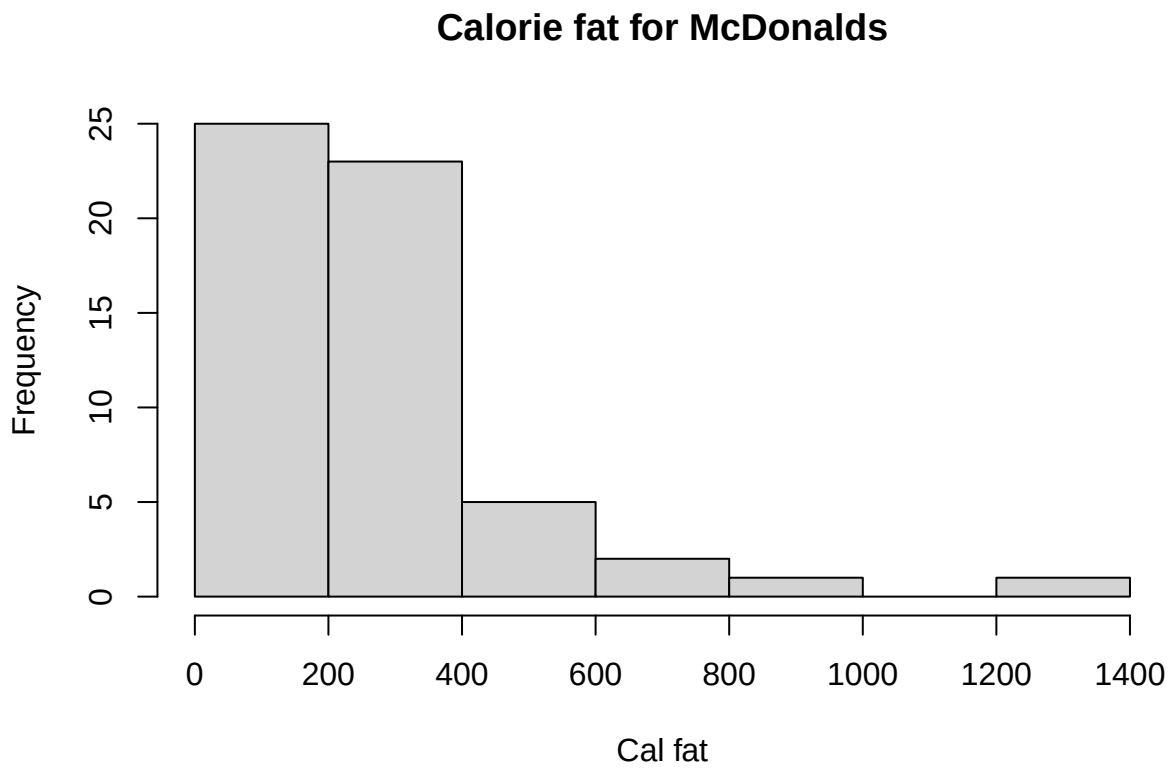
Solution 1:

Calories from fat for mcdonalds

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

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

```
hist(mcdonalds$cal_fat, main = "Calorie fat for McDonalds", xlab = "Cal fat")
```



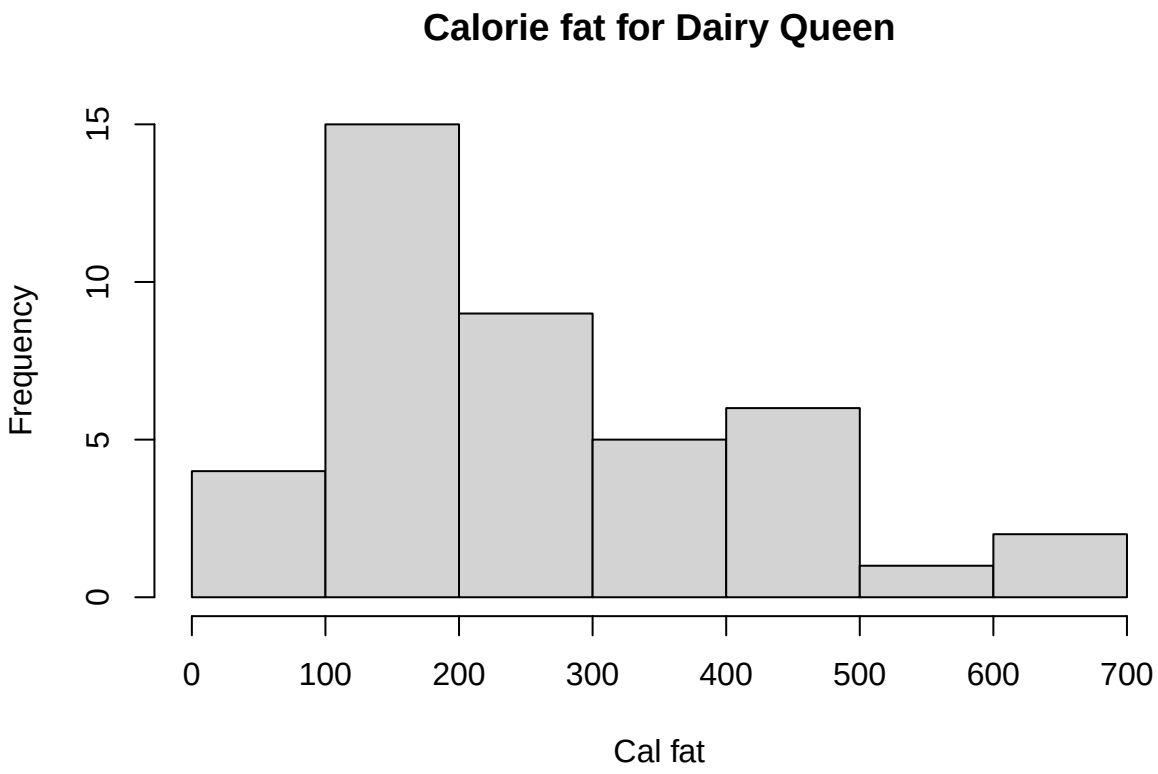
The mean is 285.6, median is 240. The mean > median > mode. Hence the distribution is right skewed and this clearly shows from the histogram.

Calories from fat for dairy_queen

```
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
```

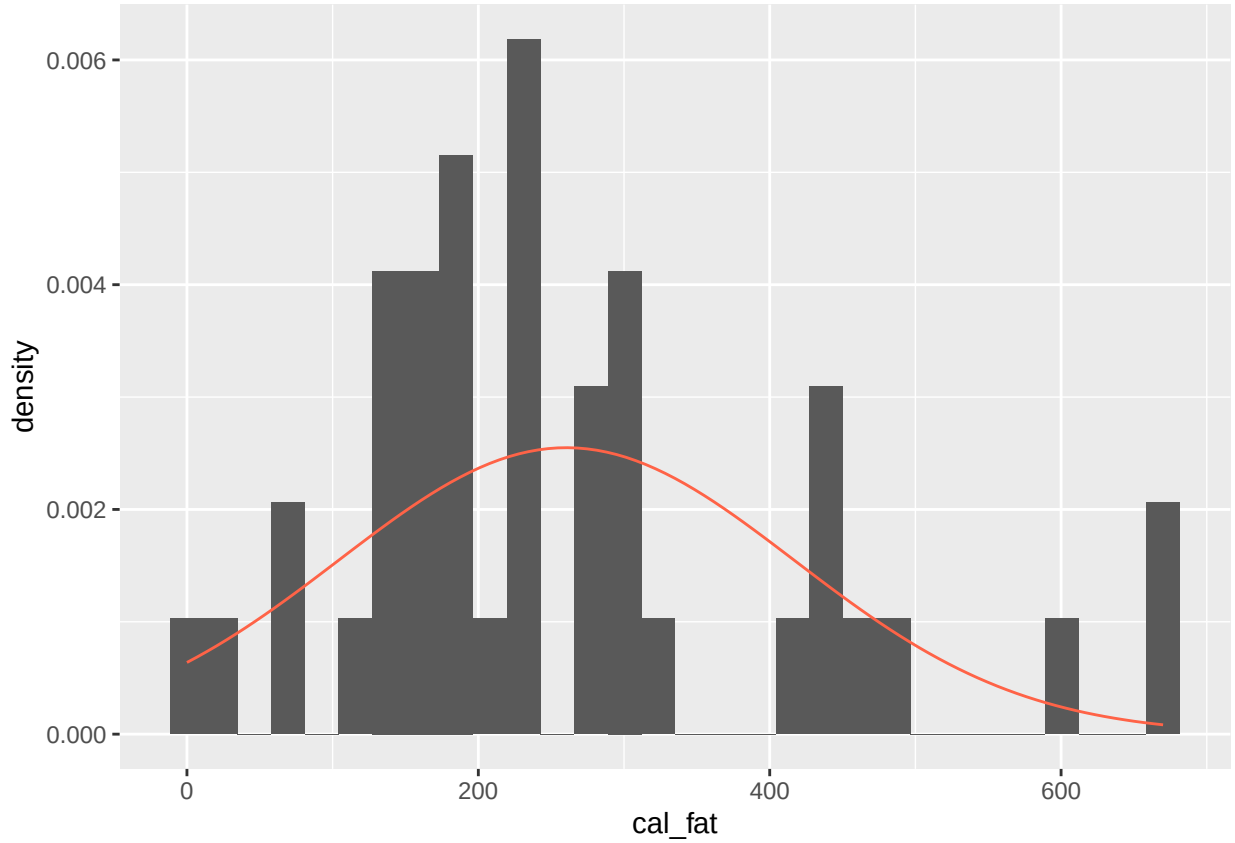
```
hist(dairy_queen$cal_fat, main = "Calorie fat for Dairy Queen", xlab = "Cal fat")
```



The mean is 260.5, median is 240. The mean > median > mode. Hence, this distribution is also slightly right skewed.

```
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")
```

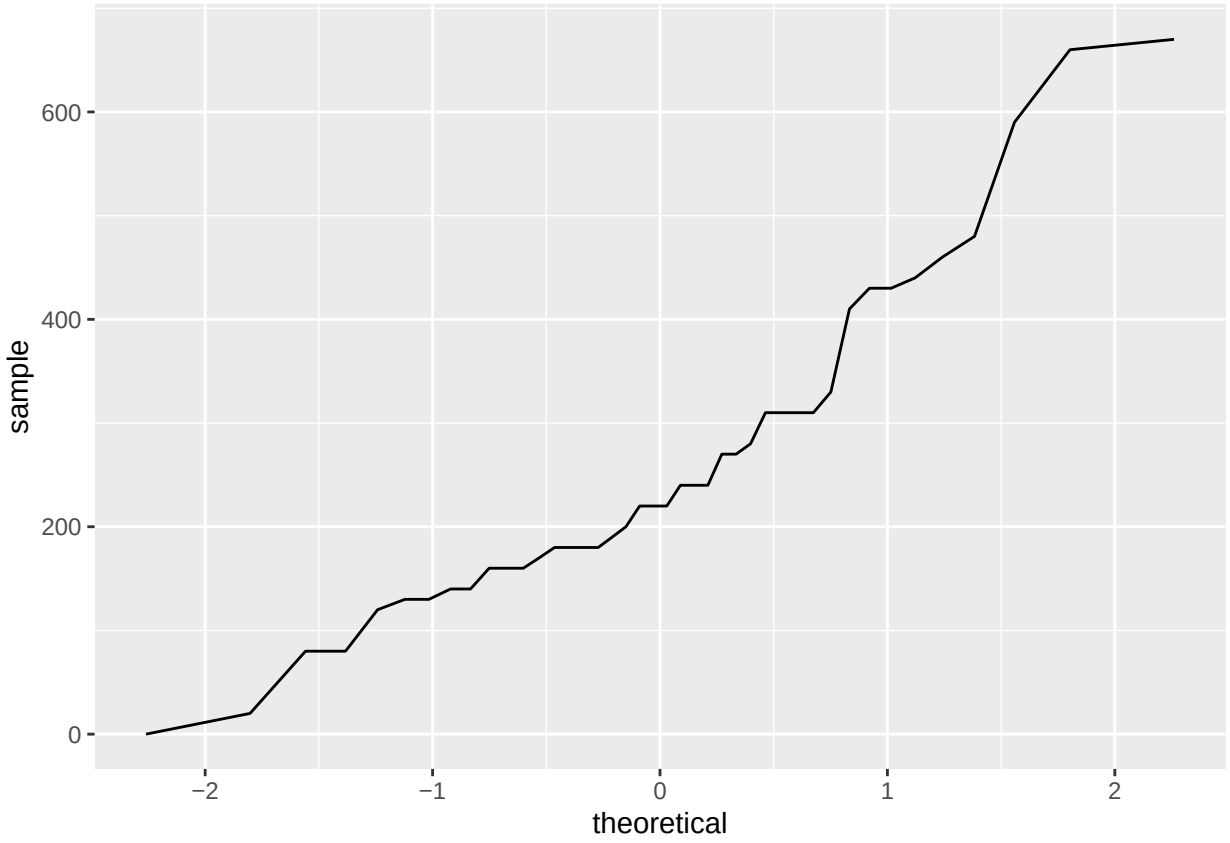


2. Based on the this plot, does it appear that the data follow a nearly normal distribution?

Solution 2:

Based on this plot, the distribution appears to follow a nearly normal distribution. Although there are some blank spaces in the histogram and the distribution is slightly right skewed.

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

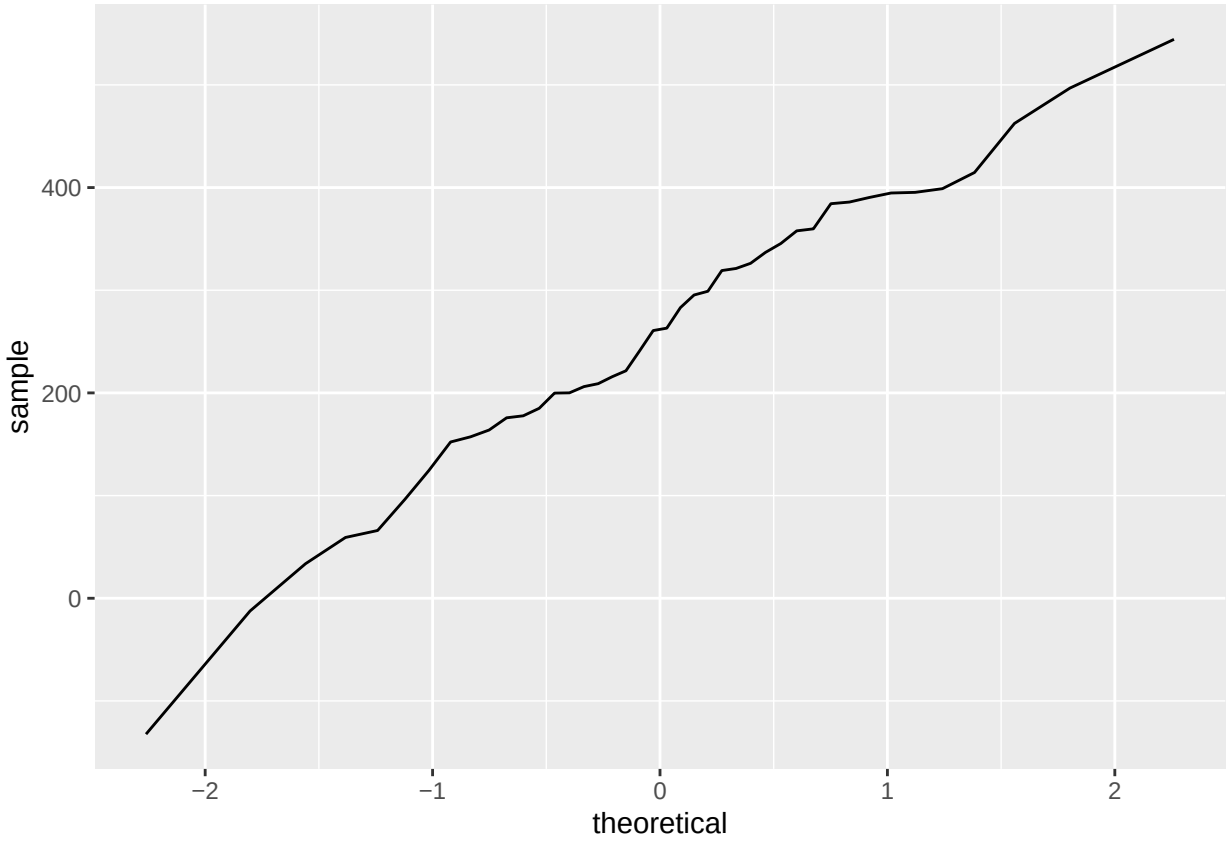



```
sim_norm <- rnorm(n = nrow(dairy_queen), mean = dqmean, sd = dqsd)
```

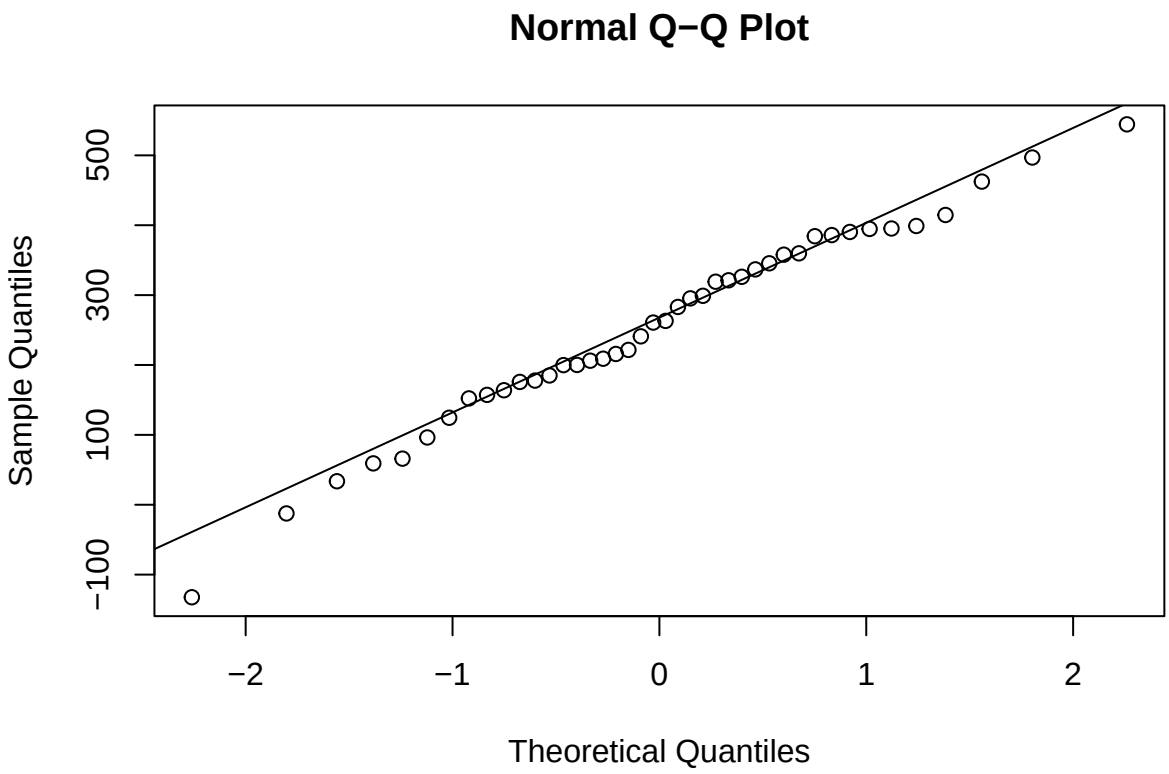
3. Make a normal probability plot of `sim_norm`. Do all of the points fall on the line? How does this plot compare to the probability plot for the real data? (Since `sim_norm` is not a data frame, it can be put directly into the `sample` argument and the `data` argument can be dropped.)

Solution 3:

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

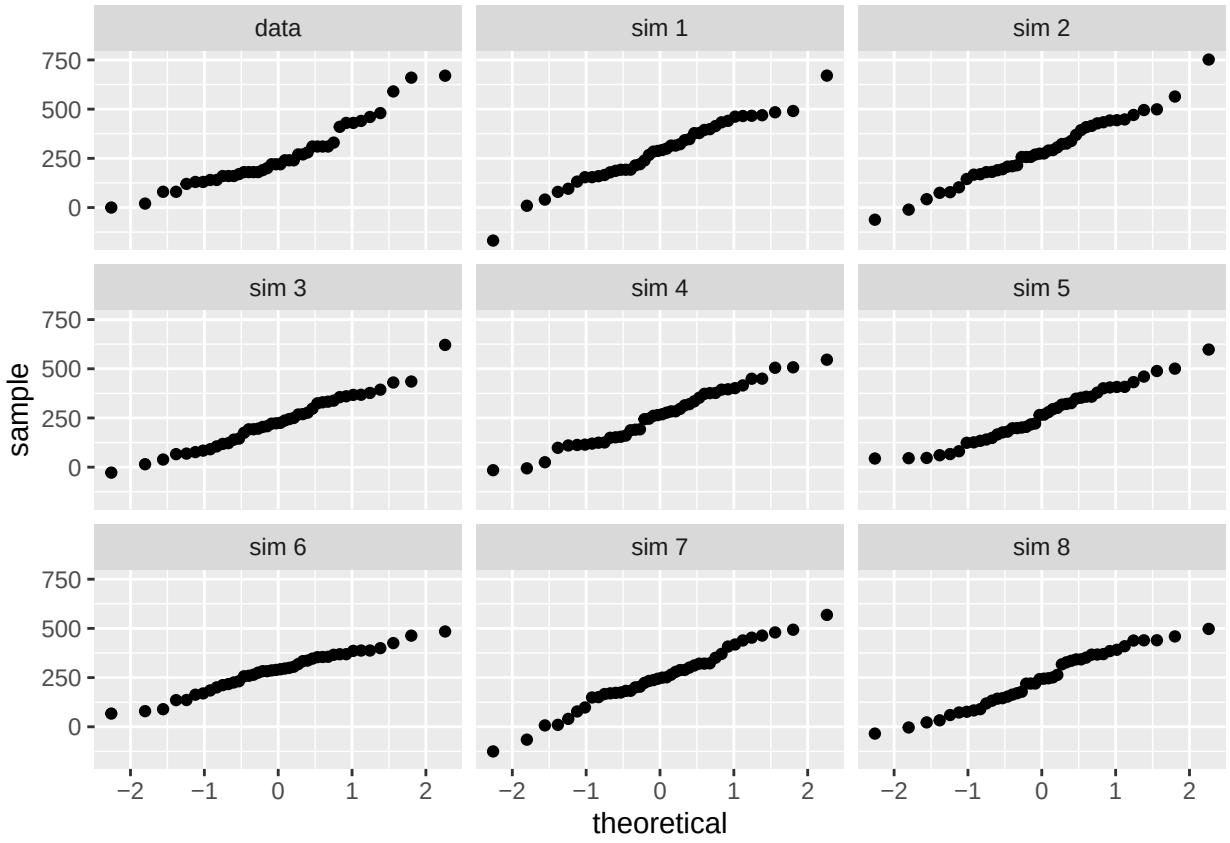



```
qqnorm(sim_norm)
qqline(sim_norm)
```



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.

```
qqnormsim(sample = cal_fat, data = dairy_queen)
```



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 calories are nearly normal?

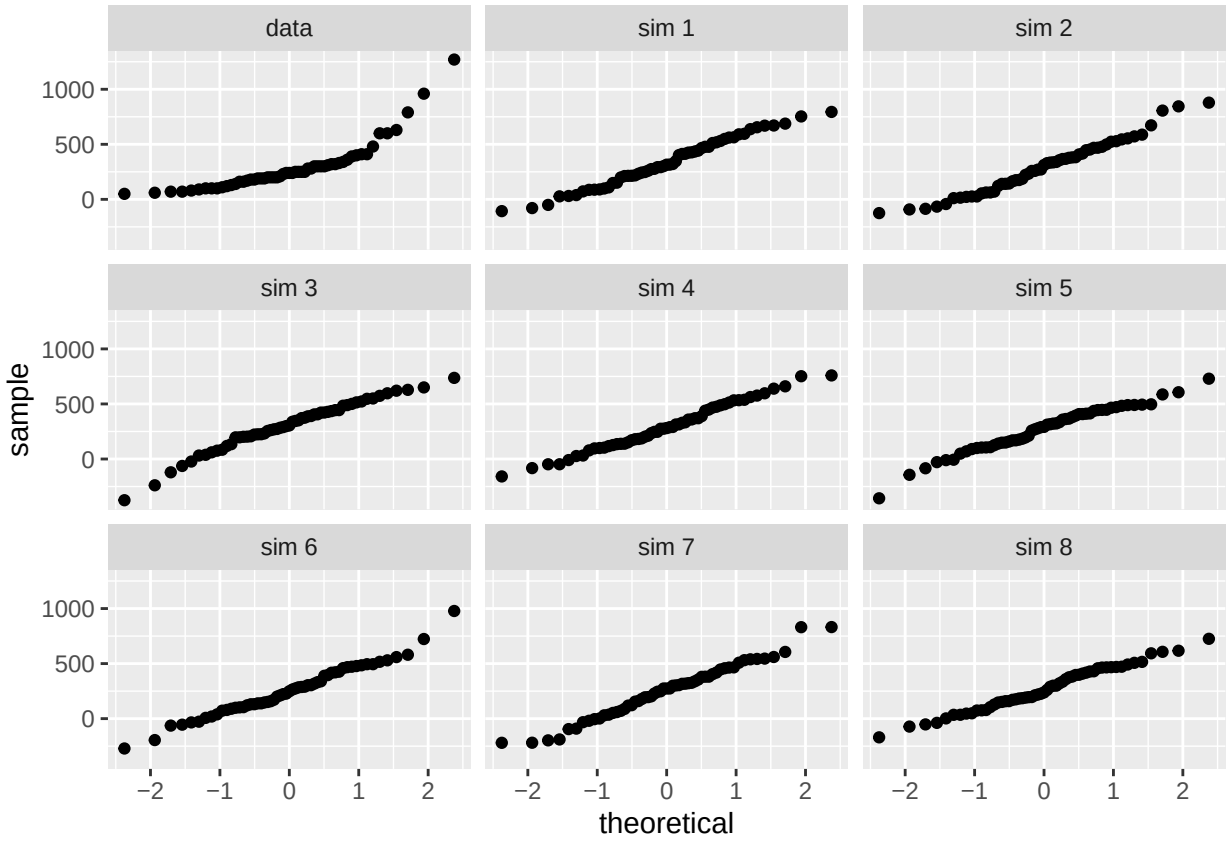
Solution 4:

Yes the normal probability plot for the calories from fat look similar to the plots created for the simulated data and it provides evidence that the calories are nearly normal.

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

Solution 5:

```
qqnormsim(sample = cal_fat, data = mcdonalds)
```



From the qq plot, we can say that the calories from McDonald's menu appear to come from a normal distribution.

Normal probabilities

Okay, so now you have a slew of tools to judge whether or not a variable is normally distributed. Why should you care?

It turns out that statisticians know a lot about the normal distribution. Once you decide that a random variable is approximately normal, you can answer all sorts of questions about that variable related to probability. Take, for example, the question of, "What is the probability that a randomly chosen Dairy Queen product has more than 600 calories from fat?"

If we assume that the calories from fat from Dairy Queen's menu are normally distributed (a very close approximation is also okay), we can find this probability by calculating a Z score and consulting a Z table (also called a normal probability table). In R, this is done in one step with the function `pnorm()`.

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

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

Note that the function `pnorm()` gives the area under the normal curve below a given value, `q`, with a given mean and standard deviation. Since we're interested in the probability that a Dairy Queen item has more than 600 calories from fat, we have to take one minus that probability.

Assuming a normal distribution has allowed us to calculate a theoretical probability. If we want to calculate the probability empirically, we simply need to determine how many observations fall above 600 then divide this number by the total sample size.

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

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

Although the probabilities are not exactly the same, they are reasonably close. The closer that your distribution is to being normal, the more accurate the theoretical probabilities will be.

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?

Solution 6:

Question 1: What is the probability that a randomly chosen mcdonald's product has more than 800 calories from fat?

```
mc_mean <- mean(mcdonalds$cal_fat)
mc_sd <- sd(mcdonalds$cal_fat)

#Empirical probability
prob_more_mc_800cal_emp <- 1 - pnorm(q = 800, mean = mc_mean, sd = mc_sd)
paste0("The probability that a randomly chosen mcdonalds' product has more than 800 calories from fat is
```

```
## [1] "The probability that a randomly chosen mcdonalds' product has more than 800 calories from fat is
```

```
#Theoretical probability
prob_more_mc_800cal_theoretical <-  mcdonalds %>%
  filter(cal_fat > 800) %>%
  summarise(percent = n() / nrow(mcdonalds))

paste0("The theoretical probability that a randomly chosen mcdonalds' product has more than 800 calories
```

```
## [1] "The theoretical probability that a randomly chosen mcdonalds' product has more than 800 calories
```

Question 2: What is the probability that a randomly chosen Dairy Dueen product has less than 500 calories from fat?

```
dq_mean <- mean(dairy_queen$cal_fat)
dq_sd <- sd(dairy_queen$cal_fat)

#Empirical probability
prob_more_dq_800cal_emp <- pnorm(q = 500, mean = dq_mean, sd = dq_sd)
paste0("The probability that a randomly chosen Dairy Queen product has less than 500 calories from fat i
```

```
## [1] "The probability that a randomly chosen Dairy Queen product has less than 500 calories from fat
```

```
#Theoretical probability
prob_more_dq_800cal_theoretical <-  dairy_queen %>%
  filter(cal_fat < 500) %>%
  summarise(percent = n() / nrow(dairy_queen))

paste0("The theoretical probability that a randomly chosen mcdonalds' product has less than 500 calorie
```

```
## [1] "The theoretical probability that a randomly chosen mcdonalds' product has less than 500 calorie
```

The second question (Question 2) has a closer agreement between the two methods.

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?

Solution 7:

We first check the dataframe to find the restaurants in the fastfood dataframe:

```
restaurants_list <- fastfood %>% distinct(restaurant)
restaurants_list
```

```
## # A tibble: 8 x 1
##   restaurant
##   <chr>
## 1 Mcdonalds
## 2 Chick Fil-A
## 3 Sonic
## 4 Arbys
## 5 Burger King
## 6 Dairy Queen
## 7 Subway
## 8 Taco Bell
```

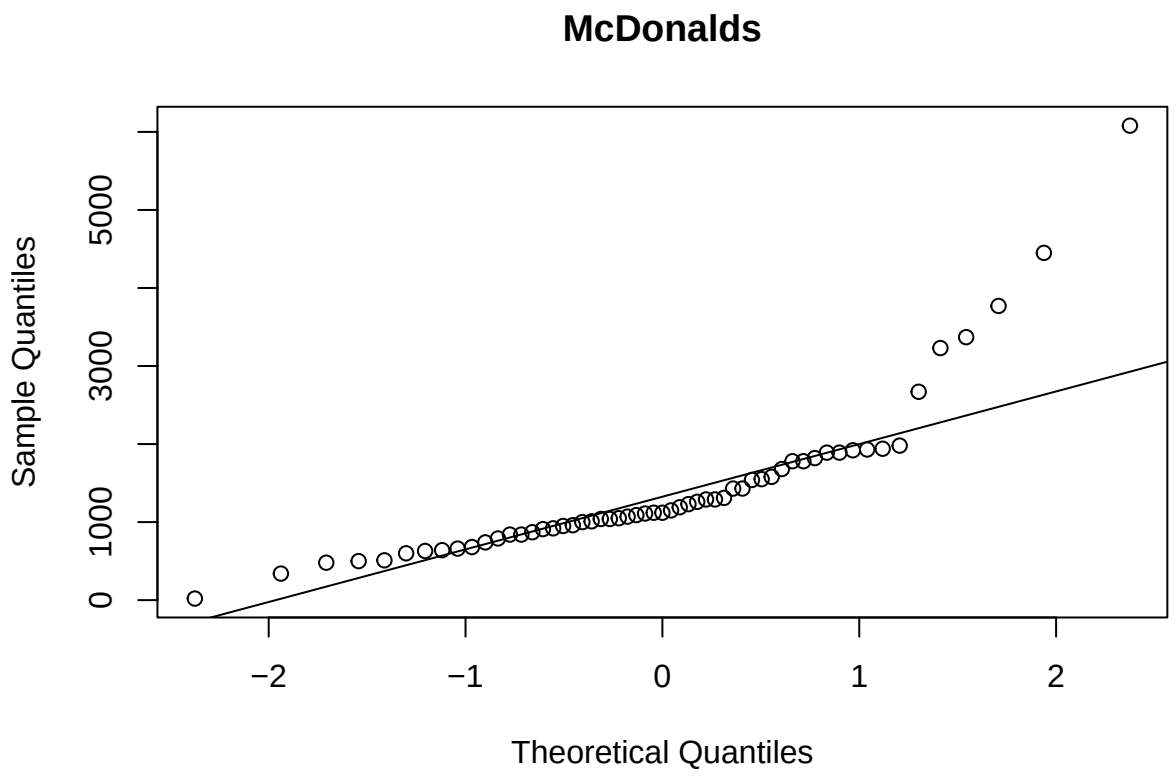
There are eight(8) distinct restaurants in the fastfood dataframe. Hence, we will draw a qq plot for all 8 restaurants and find the one with closest to normal distribution for sodium.

```
#Subset Each of the restaurants:
mcdonalds <- fastfood %>%
  filter(restaurant == "Mcdonalds")

chickfilA <- fastfood %>%
  filter(restaurant == "Chick Fil-A")

sonic <- fastfood %>%
  filter(restaurant == "Sonic")

arbys <- fastfood %>%
  filter(restaurant == "Arbys")
```

```
burgerking <- fastfood %>%
  filter(restaurant == "Burger King")

dairyqueen <- fastfood %>%
  filter(restaurant == "Dairy Queen")

subway <- fastfood %>%
  filter(restaurant == "Subway")

tacobell <- fastfood %>%
  filter(restaurant == "Taco Bell")
```

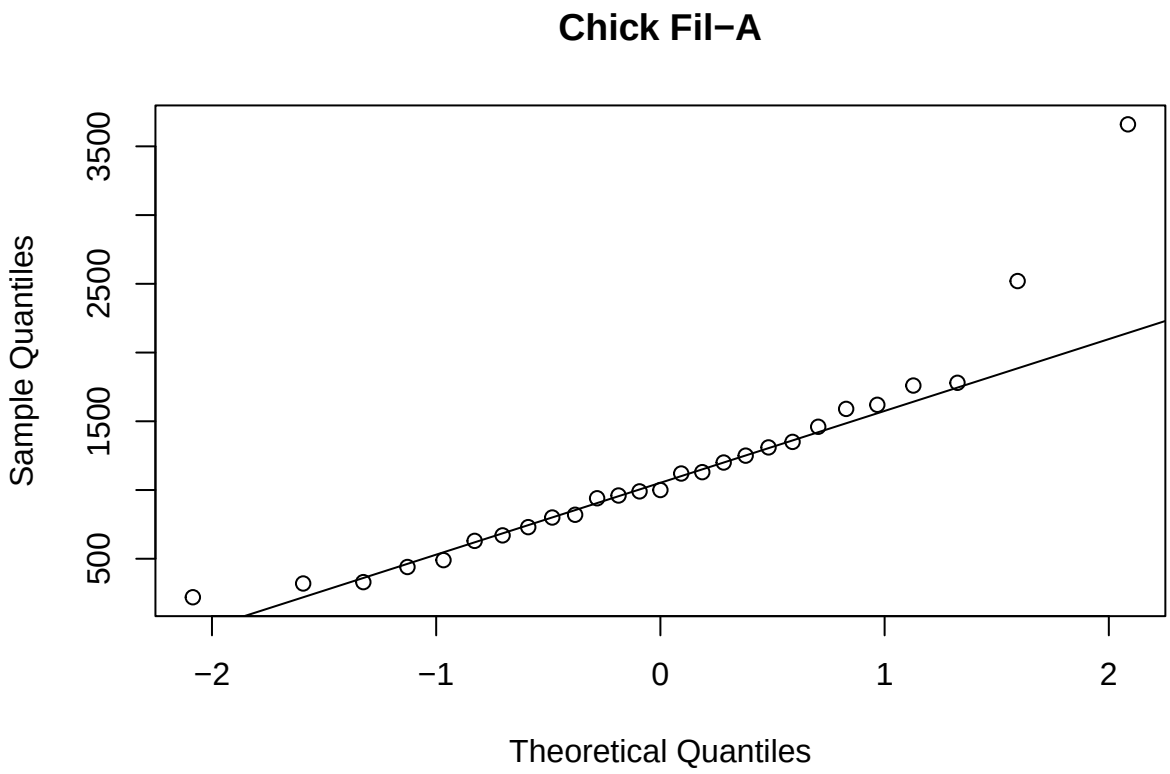
Normal Plot for McDonalds

```
qqnorm(mcdonalds$sodium, main = "McDonalds")
qqline(mcdonalds$sodium)
```



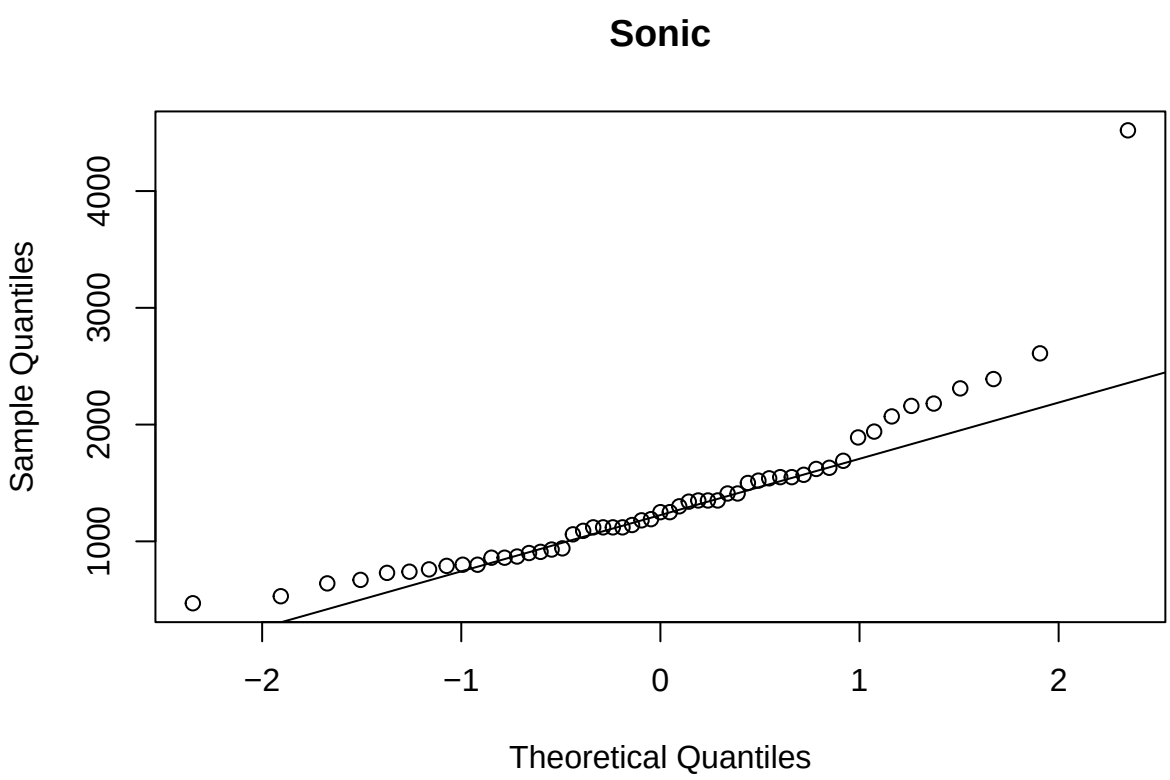
Normal Plot for Chick Fil-A

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



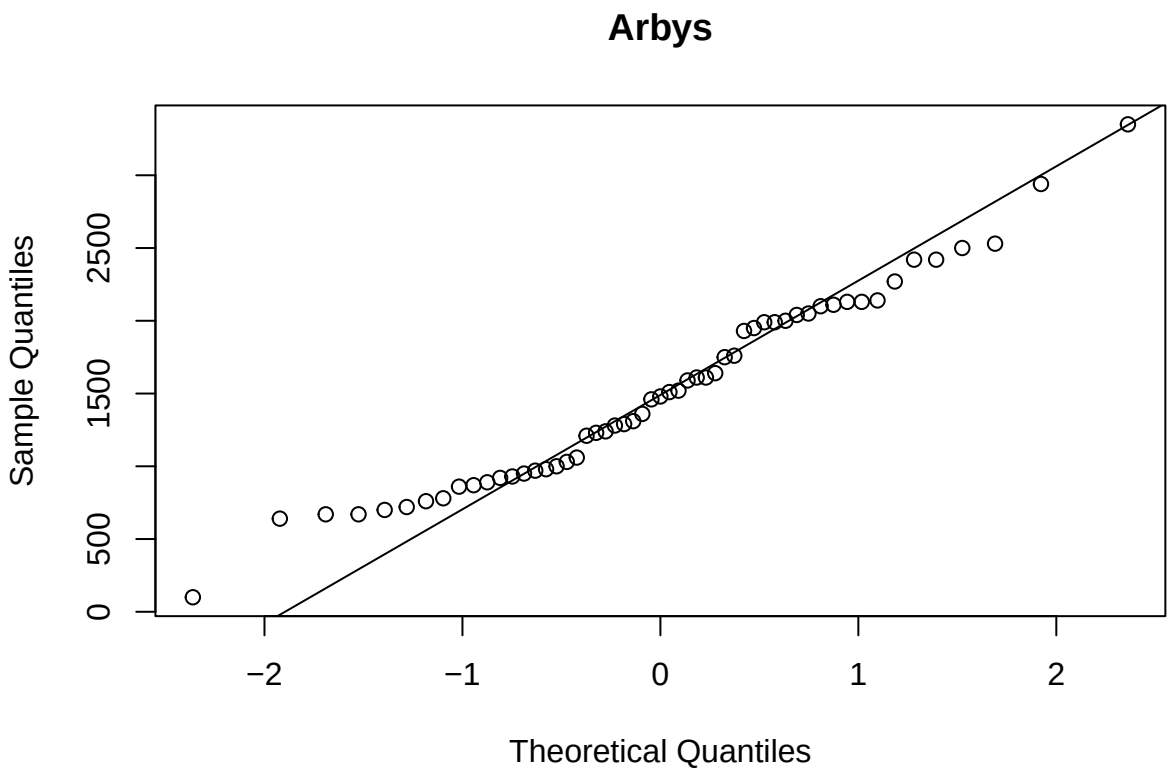
Normal Plot for Sonic

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



Normal Plot for Arbys

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

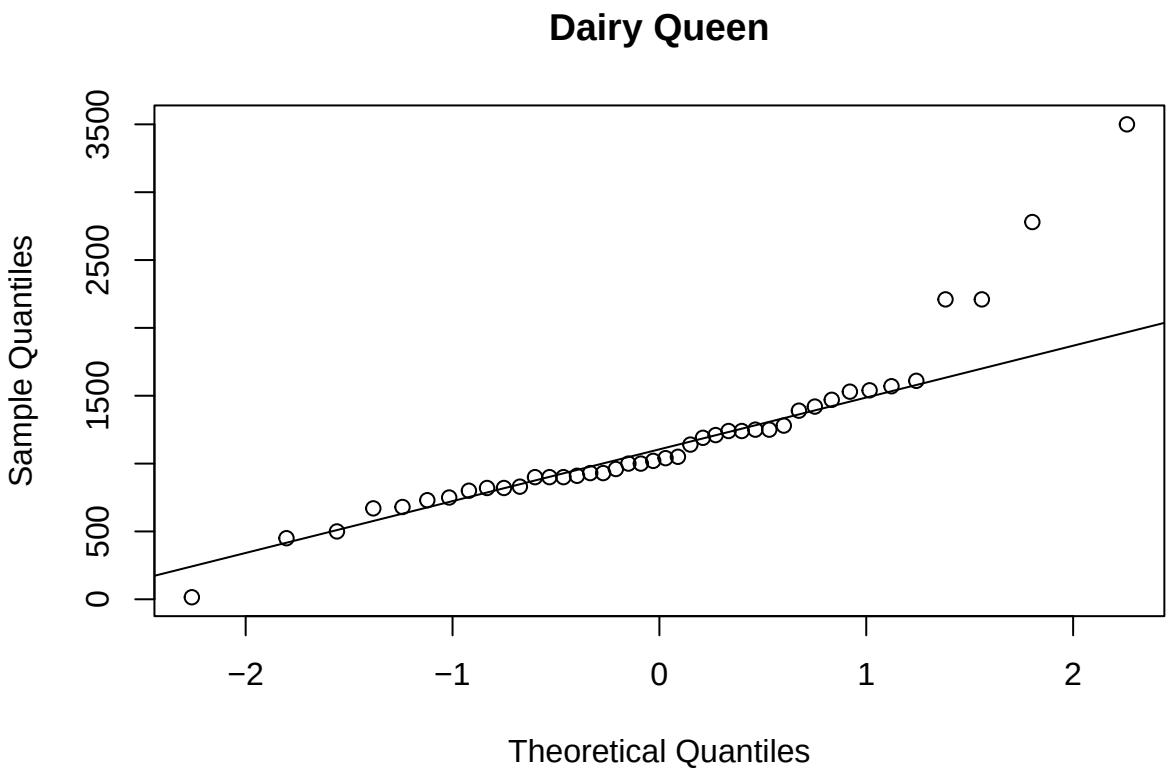


Normal Plot for Burger King

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

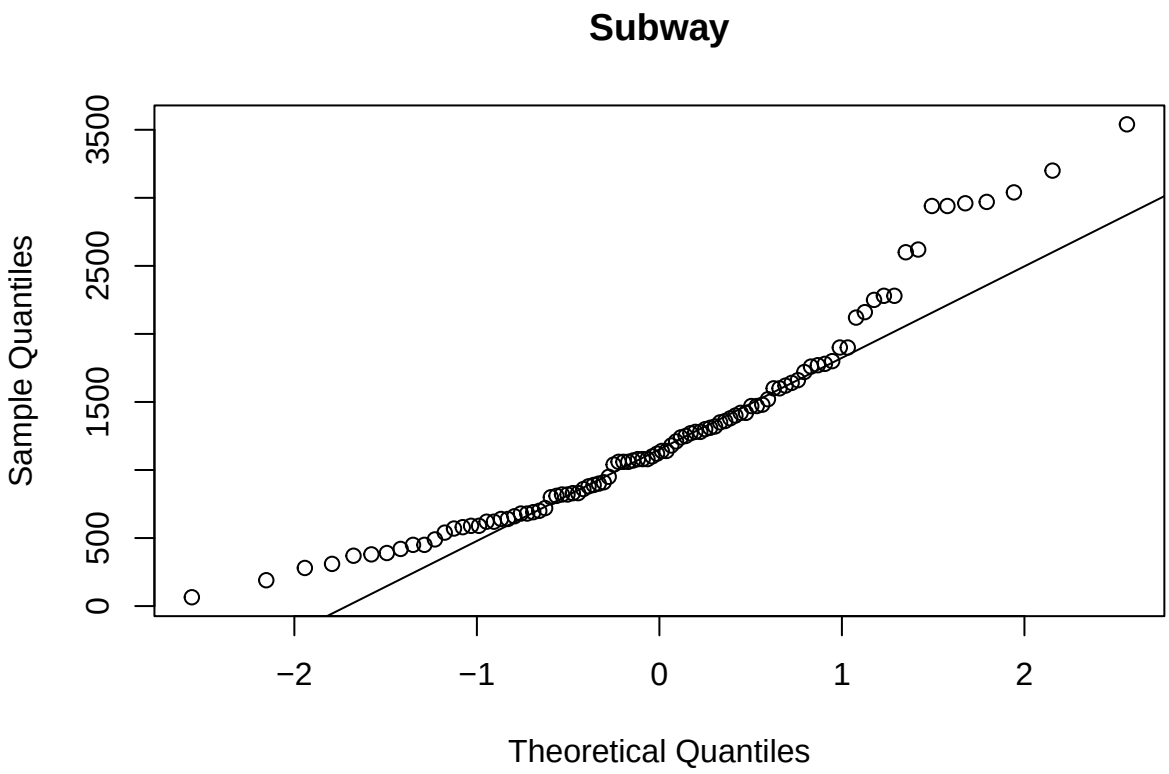
Normal Plot for Dairy Queen

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



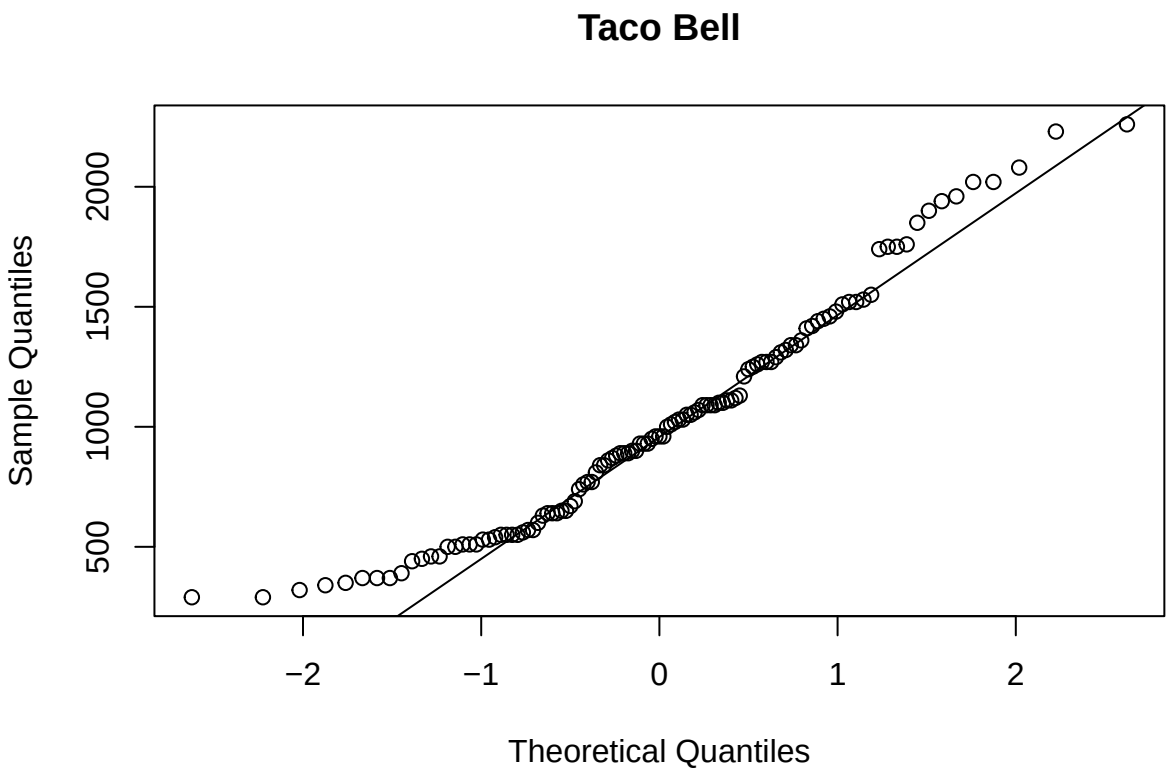
Normal Plot for Subway

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



Normal Plot for Taco Bell

```
qqnorm(tacobell$sodium, main = "Taco Bell")
qqline(tacobell$sodium)
```



The plot for Burger King and Arbys appears to be closest to a normal distribution for sodium.

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?

Solution 8:

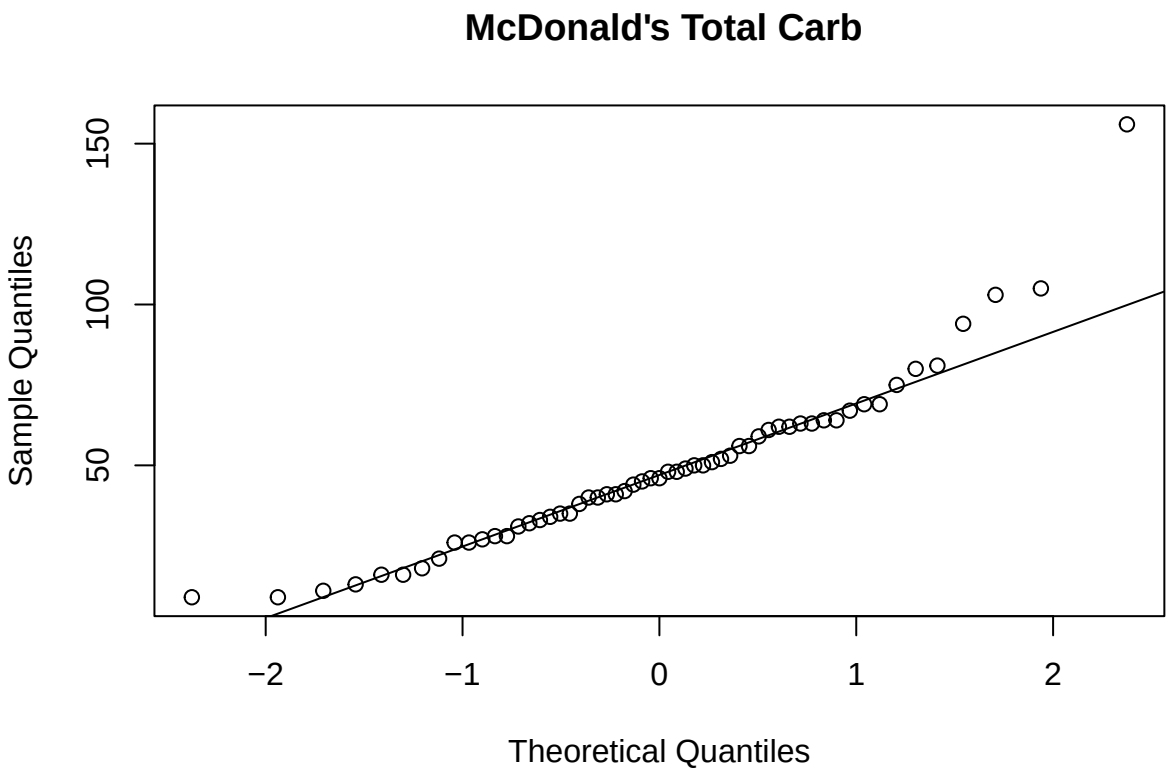
This stepwise pattern may arise as a result of variabilities in sodium content of different products by a particular restaurant.

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.

Solution 9:

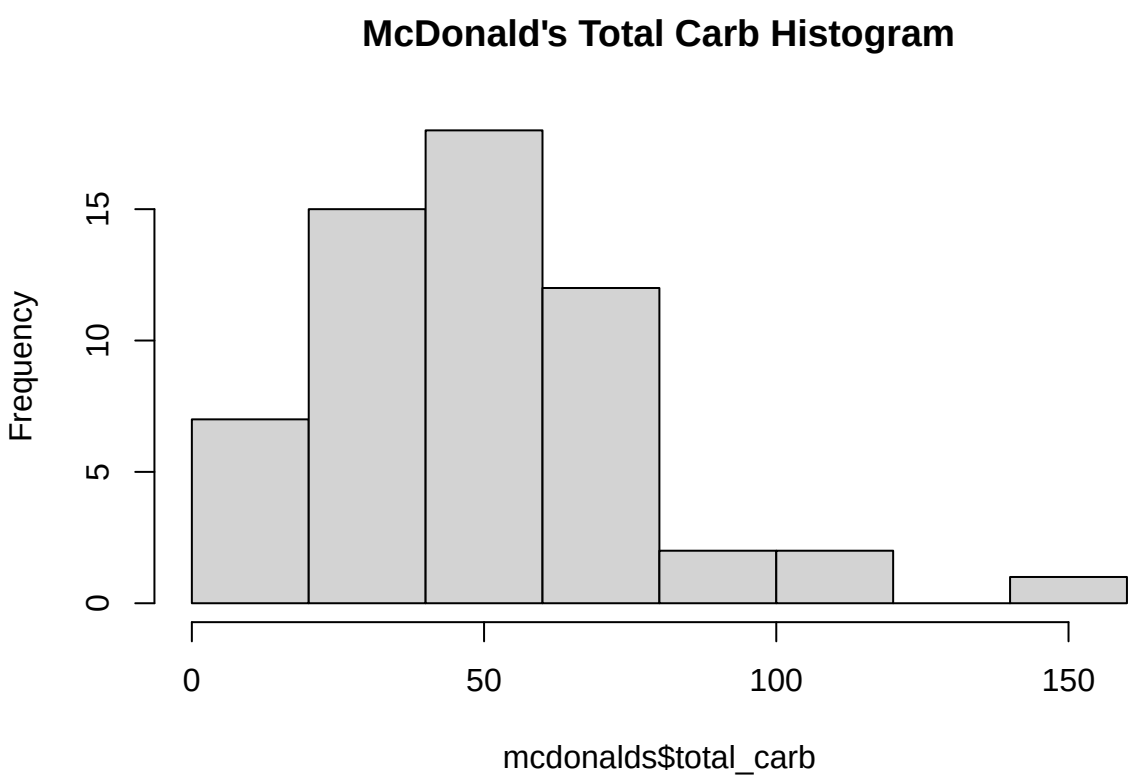
Normal Plot for the total carbohydrates for McDonalds

```
qqnorm(mcdonalds$total_carb, main = " McDonald's Total Carb")
qqline(mcdonalds$total_carb)
```



Histogram Plot for total carbohydrates for McDonalds

```
hist(mcdonalds$total_carb, main = "McDonald's Total Carb Histogram")
```



From the qq plot, we can see deviations on the upper right side (right tail). Also, from the histogram, we can confirm that the distribution for the total carbohydrates for McDonalds is right skewed.