Solution

Jawaid Hakim

2022-09-30

Contents

1 Setup

1.1 Load packages

2 Load data

```
data("fastfood", package='openintro')
head(fastfood)
```

```
## # A tibble: 6 x 17
##   restaur~1 item  calor~2 cal_fat total~3 sat_fat trans~4 chole~5 sodium total~6
##   <chr>     <chr>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>  <dbl>   <dbl>
## 1 Mcdonalds Arti~     380      60       7       2     0          95   1110      44
## 2 Mcdonalds Sing~     840     410      45      17     1.5       130   1580      62
## 3 Mcdonalds Doub~    1130     600      67      27     3         220   1920      63
## 4 Mcdonalds Gril~     750     280      31      10     0.5       155   1940      62
## 5 Mcdonalds Cris~     920     410      45      12     0.5       120   1980      81
## 6 Mcdonalds Big ~     540     250      28      10     1          80    950      46
## # ... with 7 more variables: fiber <dbl>, sugar <dbl>, protein <dbl>,
## #   vit_a <dbl>, vit_c <dbl>, calcium <dbl>, salad <chr>, and abbreviated
## #   variable names 1: restaurant, 2: calories, 3: total_fat, 4: trans_fat,
## #   5: cholesterol, 6: total_carb
```

Create data frames for Mcdonalds and Dairy Queen restaurants.

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

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

Calculate mean and standard deviations.

```
mcmean <- mean(mcdonalds$cal_fat)
mcmean
```

```
## [1] 285.614
```

```
mcsd   <- sd(mcdonalds$cal_fat)
mcsd
```

```
## [1] 220.8993
```

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

```
## [1] 260.4762
```

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

```
## [1] 156.4851
```

3 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?

Let's plot density curve of fat calories from the two restaurants. Both plots appear to be Normal distributions, with Mcdonalds showing higher mean and also greater spread.

```
dqplot <- ggdensity(dairy_queen,
            'cal_fat',
            title = 'Dairy Queeen Cal from Fat',
            add = c('mean'),
            fill = 'Blue',
            palette = 'jco'
        )
```

```
## Warning: geom_vline(): Ignoring `mapping` because `xintercept` was provided.
```

```
## Warning: geom_vline(): Ignoring `data` because `xintercept` was provided.
```

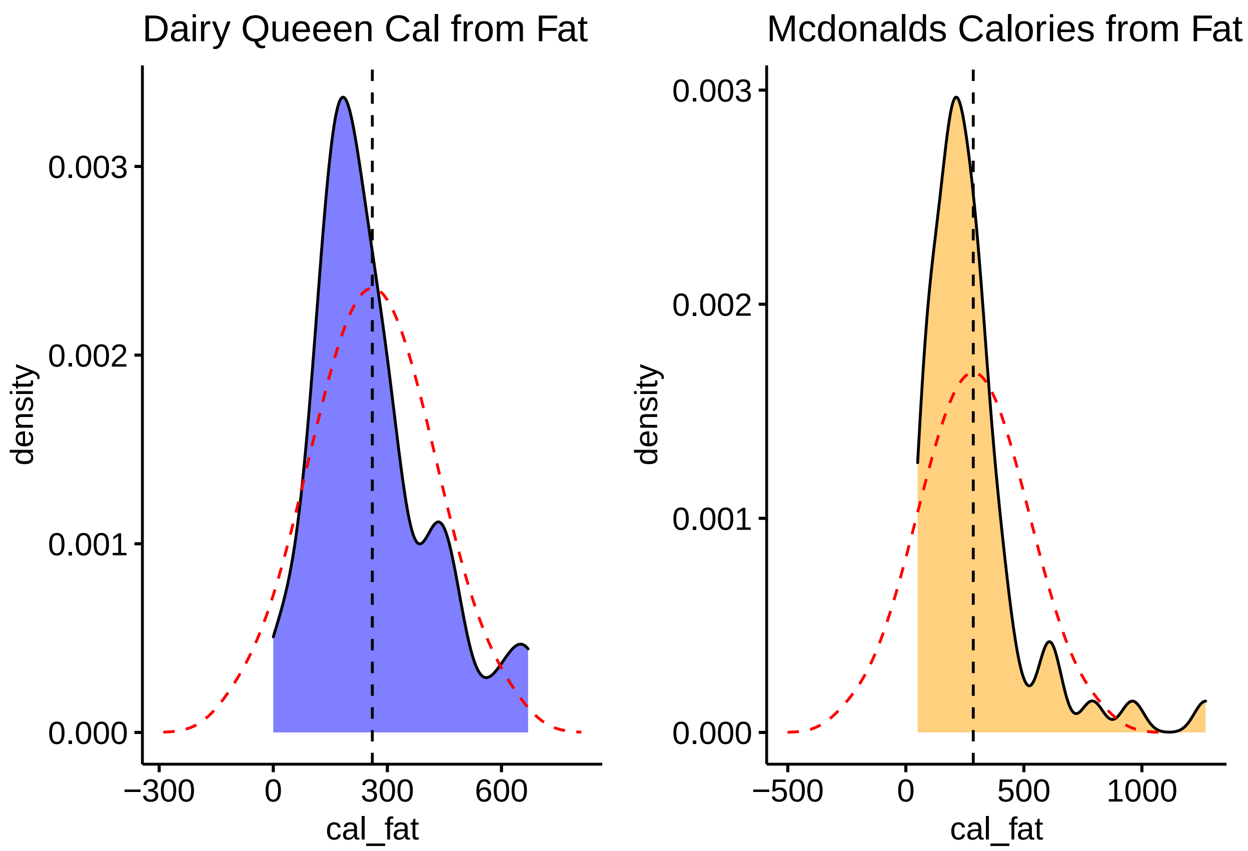
```
dqplot <- dqplot +
         stat_overlay_normal_density(color = "red", linetype = "dashed")

mdplot <- ggdensity(mcdonalds,
       'cal_fat',
       title = 'Mcdonalds Calories from Fat',
       add = c('mean'),
       fill = 'Orange',
       palette = 'jco'
)
```

```
## Warning: geom_vline(): Ignoring `mapping` because `xintercept` was provided.
## geom_vline(): Ignoring `data` because `xintercept` was provided.
```

```
mdplot <- mdplot +
         stat_overlay_normal_density(color = "red", linetype = "dashed")

ggarrange(dqplot, mdplot)
```



4 Exercise 2

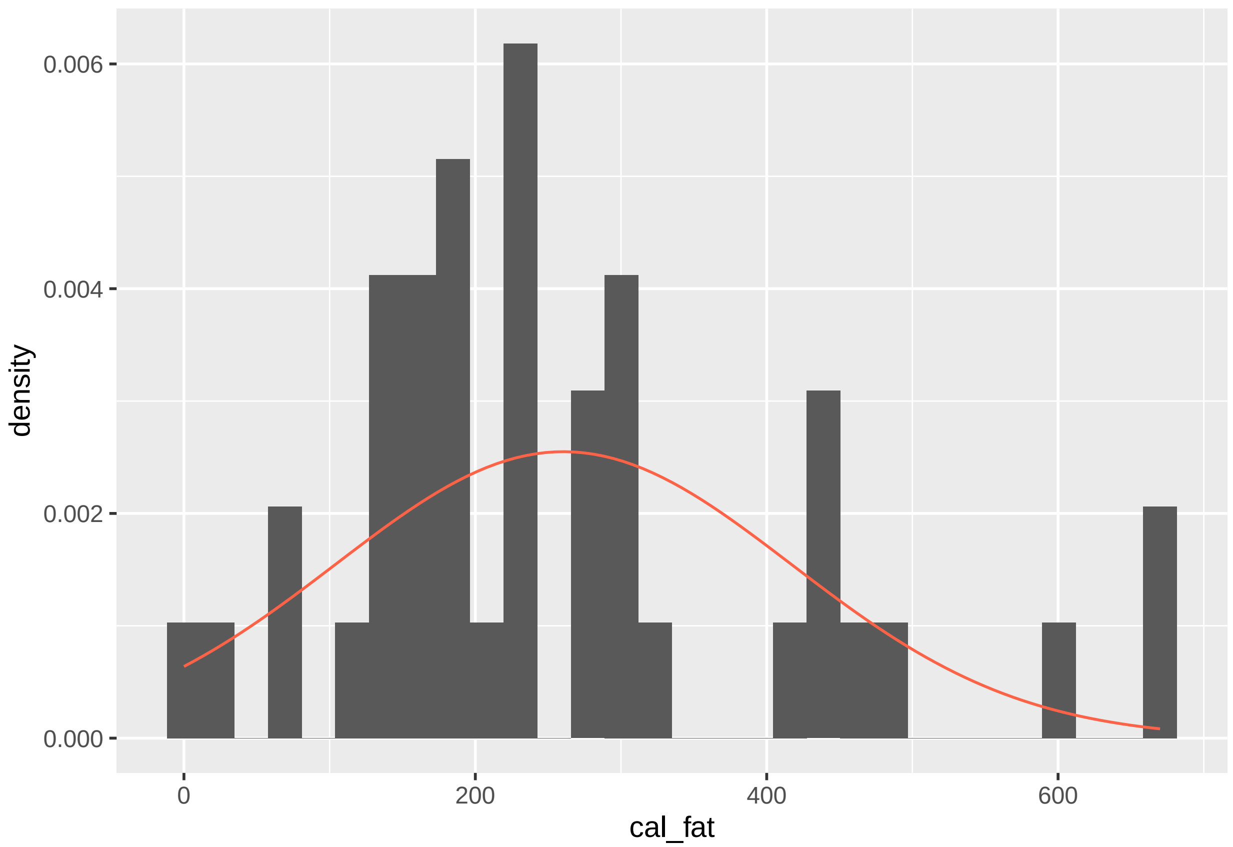
Based on the this plot, does it appear that the data follow a nearly normal distribution?

Let's plot the calorie fat observations for Dairy Queen along with the density curve.

There is a fit between the two plots so calories from fat for Dairy Queen seems to be a Normal Distribution.

```
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‘.
```



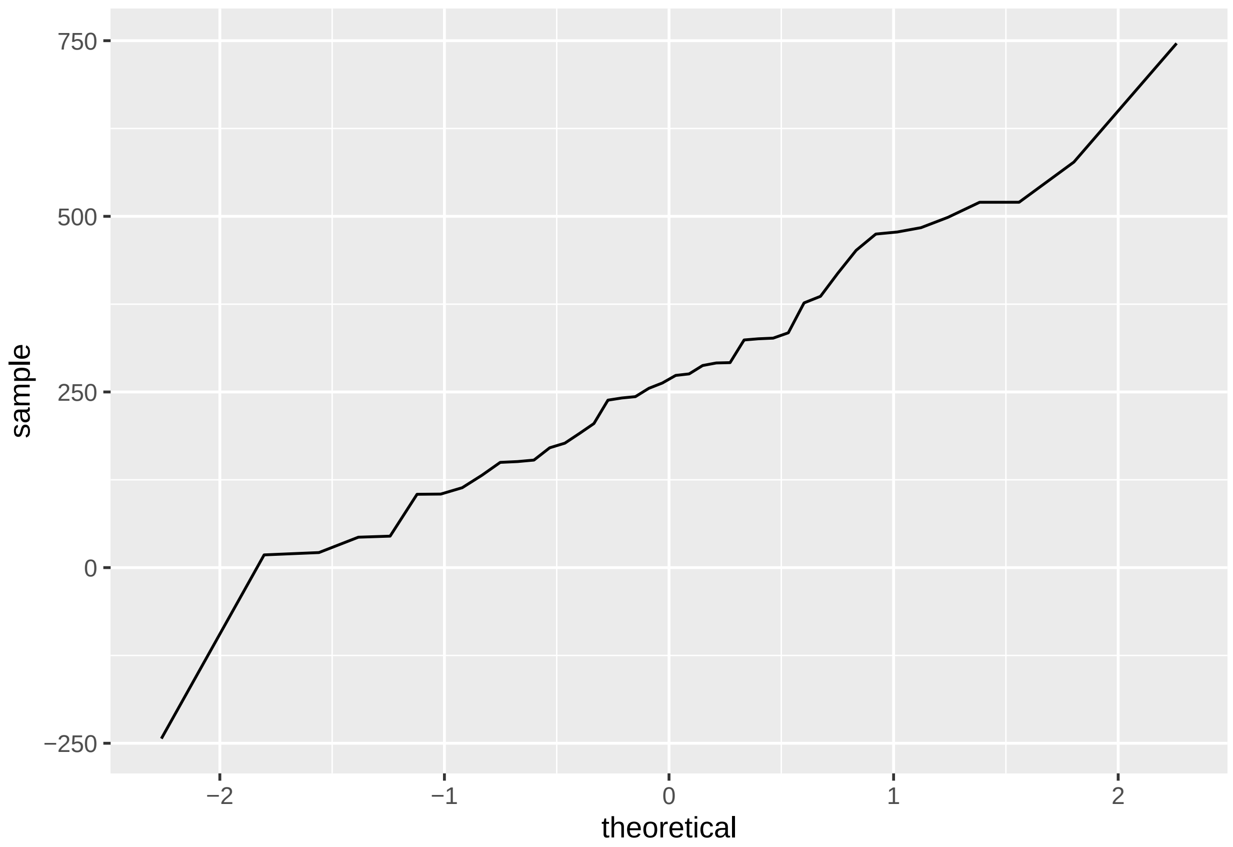
5 Exercise 3

> 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 data frame, it can be put directly into the sample argument and the data argument can be dropped.)

Most, not all, points fall along the center line, as one would expect from a small set of observations. This plot approximates the Normal distribution.

```
set.seed(1832)
sim_norm <- rnorm(n = nrow(dairy_queen), mean = dqmean, sd = dqsd)

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

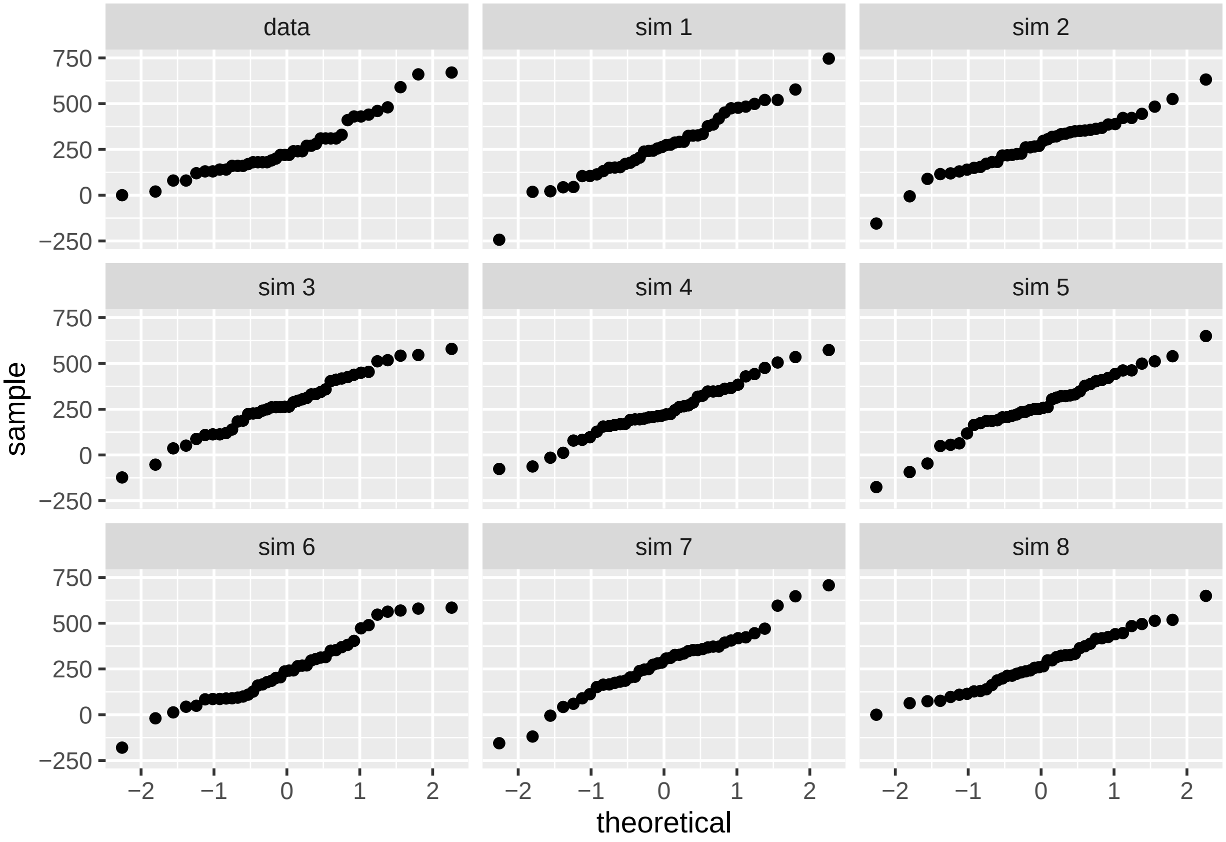


6 Exercise 4

> 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 calories are nearly normal?

The plots provide strong evidence that calories from fat for Dairy Queen are close to Normal.

```
set.seed(1832)
qqnormsim(sample = cal_fat, data = dairy_queen)
```

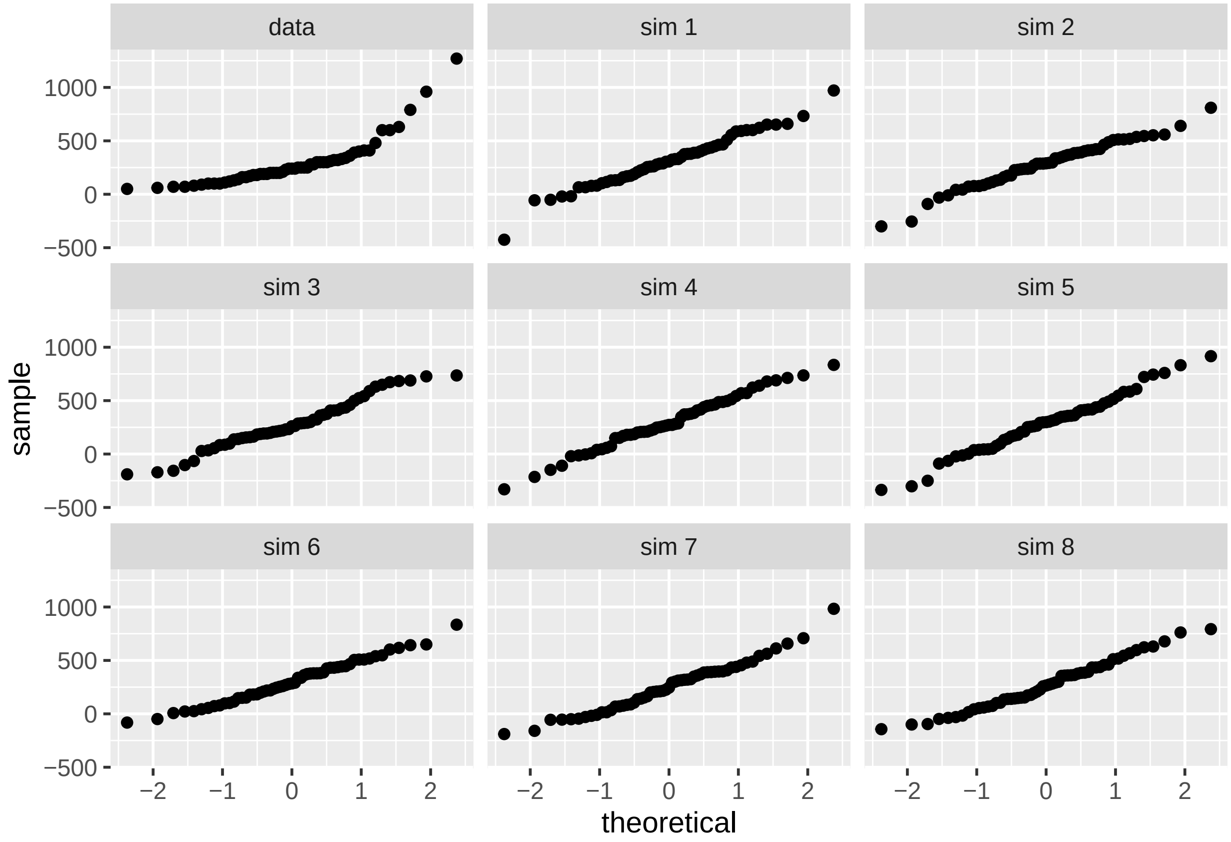


7 Exercise 5

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

Again, the plots provide strong evidence that calories from fat for Mcdonals are also close to Normal.

```
set.seed(1832)
qqnormsim(sample = cal_fat, data = mcdonalds)
```



> 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
```

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
```

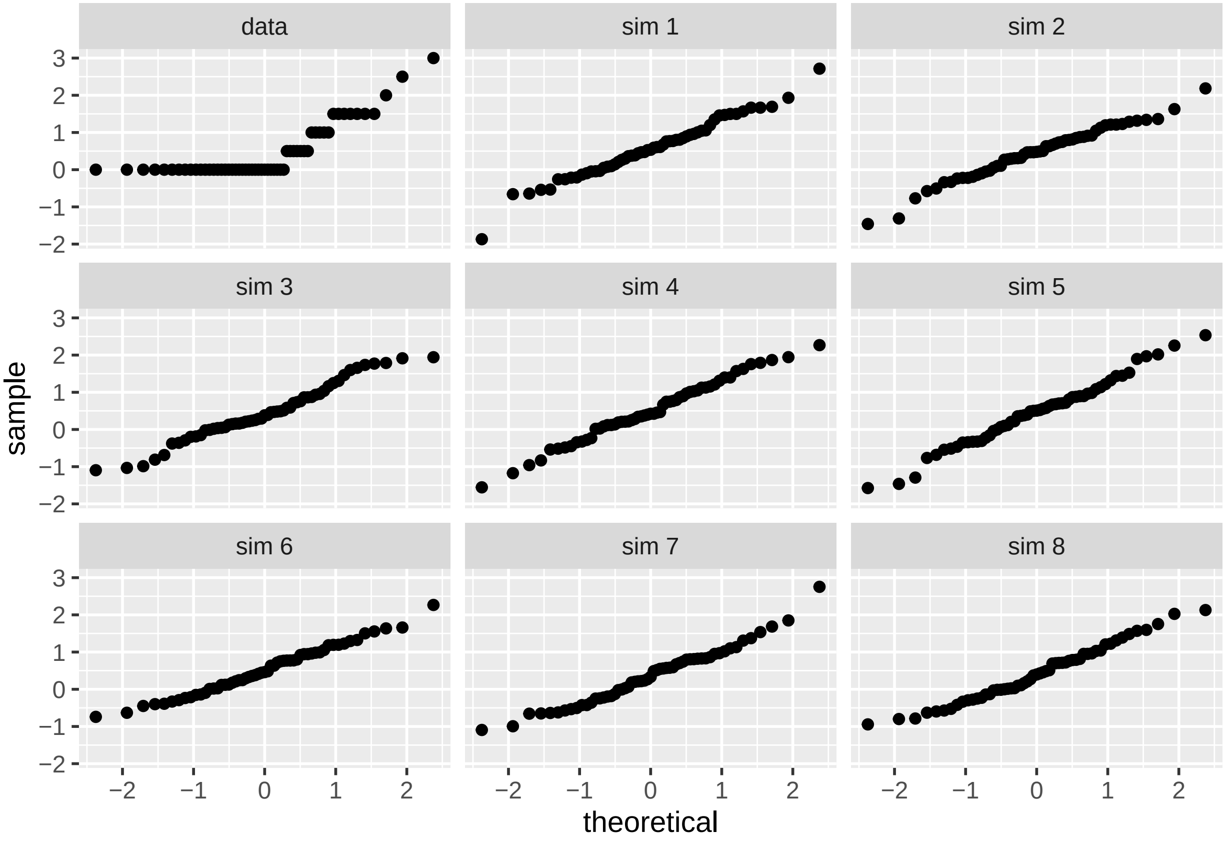
Although the probabilities are not exactly the same, they are reasonably close. The closer that your distribution is to being normal, the more accurate the theoretical probabilities will be.

8 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?

Let's first figure out if trans fat is a Normal distribution at Macdonalds. It has a definite stepwise shape but looks Normal.

```
set.seed(1832)
qqnormsim(sample = trans_fat, data = mcdonalds)
```



8.1 What is the probability that an item selected Mcdonalds will have less than 1.0 grams of trans fat?

First, we use pnorm to compute the probability of an item at Mcdonalds containing less than 1.0 grams of trans fat.

```
mcmean_trans_fat <- mean(mcdonalds$trans_fat)
mcmean_trans_fat
```

```
## [1] 0.4649123
```

```
mcsd_trans_fat   <- sd(mcdonalds$trans_fat)
mcsd_trans_fat
```

```
## [1] 0.7249363
```

```
pnorm(q = 1.0, mean = mcmean_trans_fat, sd = mcsd_trans_fat)
```

```
## [1] 0.7697783
```

Next, we compute the probability explicitly from the data. There is close agreement between the two numbers.

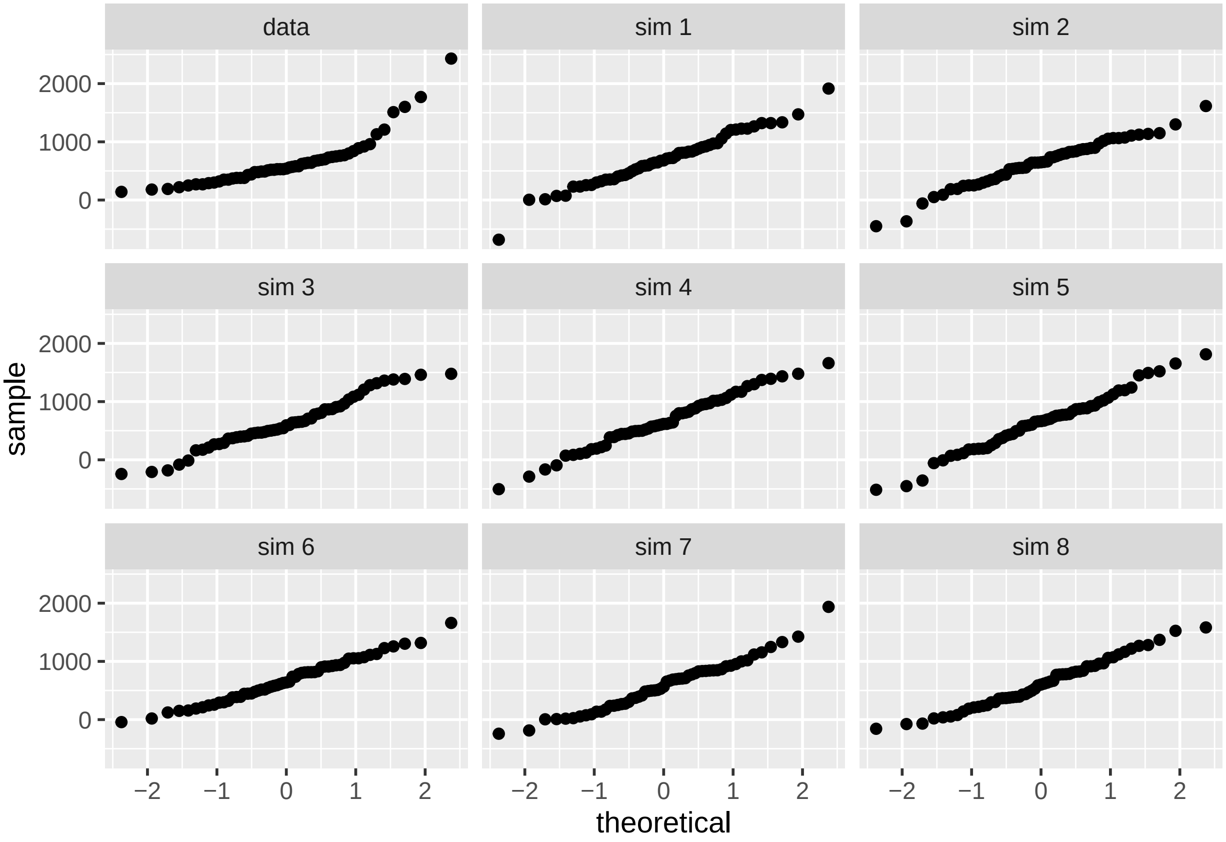
```
mcdonalds %>%
  filter(trans_fat < 1.0) %>%
  summarise(percent = n() / nrow(mcdonalds))
```

```
## # A tibble: 1 x 1
##   percent
##     <dbl>
## 1   0.737
```

8.2 What is the probability that an item selected Mcdonalds will have greater than 500 calories?

Let's first figure out if calories is a Normal distribution at Macdonalds. Looks Normal.

```
set.seed(1832)
qqnormsim(sample = calories, data = mcdonalds)
```



Compute probability using pnorm.

```
mcmean_calories <- mean(mcdonalds$calories)
mcmean_calories
```

```
## [1] 640.3509
```

```
mcsd_calories   <- sd(mcdonalds$calories)
mcsd_calories
```

```
## [1] 410.6961
```

```
1 - pnorm(q = 500, mean = mcmean_calories, sd = mcsd_calories)
```

```
## [1] 0.6337263
```

Next, we compute the probability explicitly from the data. There is close agreement between the two numbers.

```
mcdonalds %>%
  filter(calories > 500) %>%
  summarise(percent = n() / nrow(mcdonalds))
```

```
## # A tibble: 1 x 1
##   percent
##     <dbl>
## 1   0.614
```

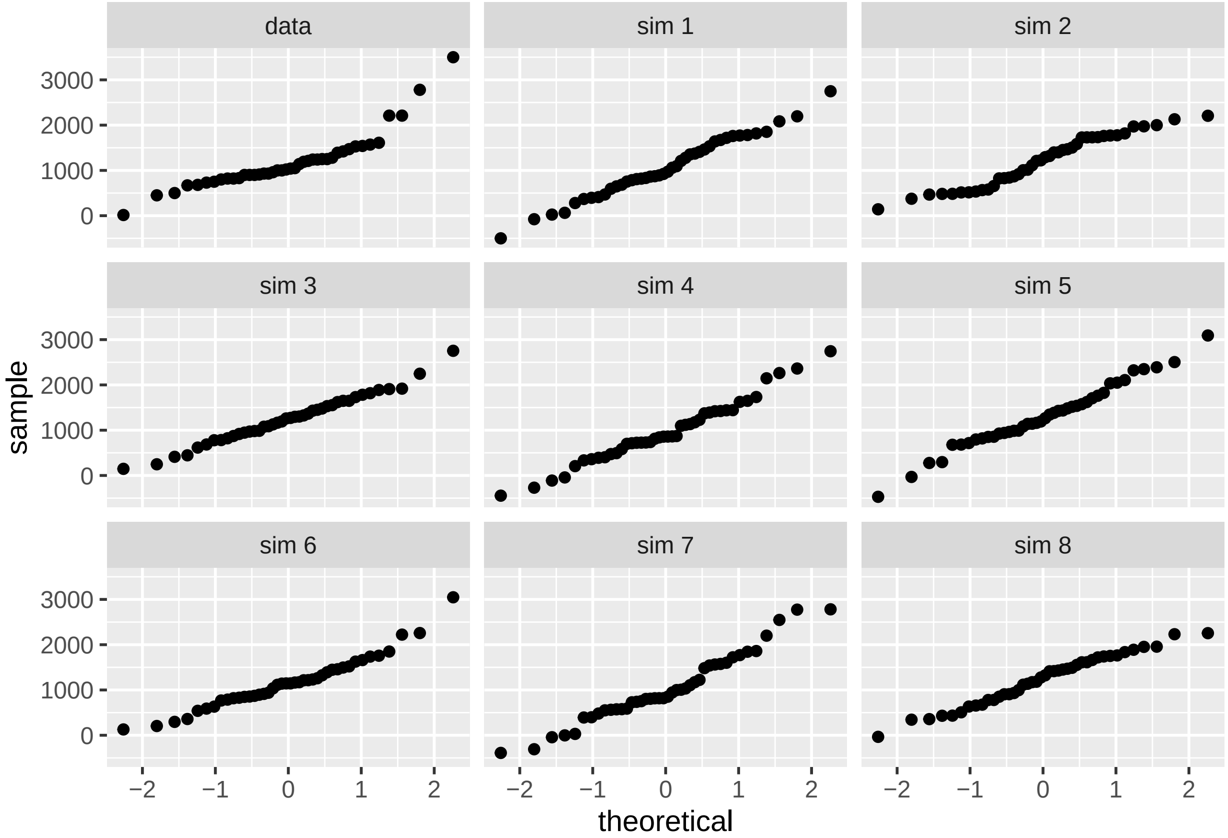
9 Exercise 7

> 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?

Let's do qq-plot for sodium content of items from both restaurants.

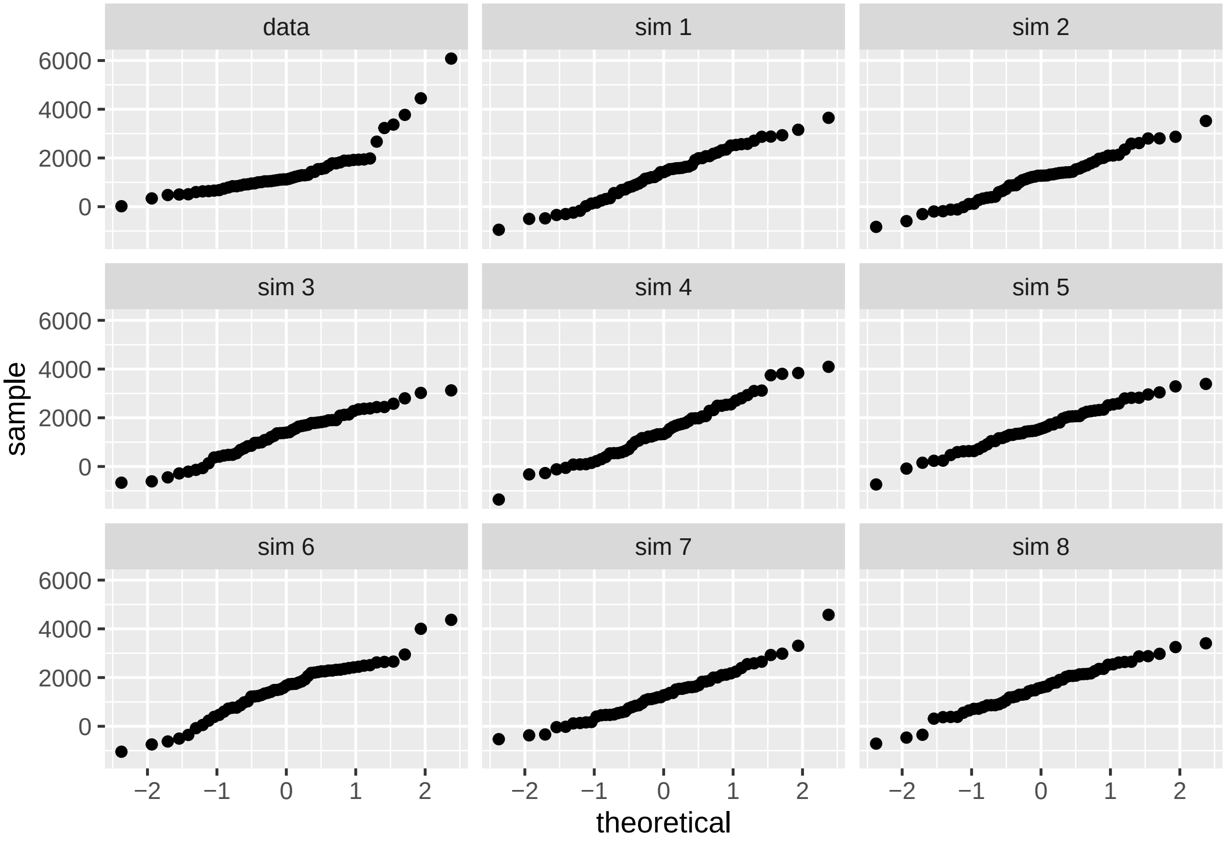
First, for Dairy Queen.

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



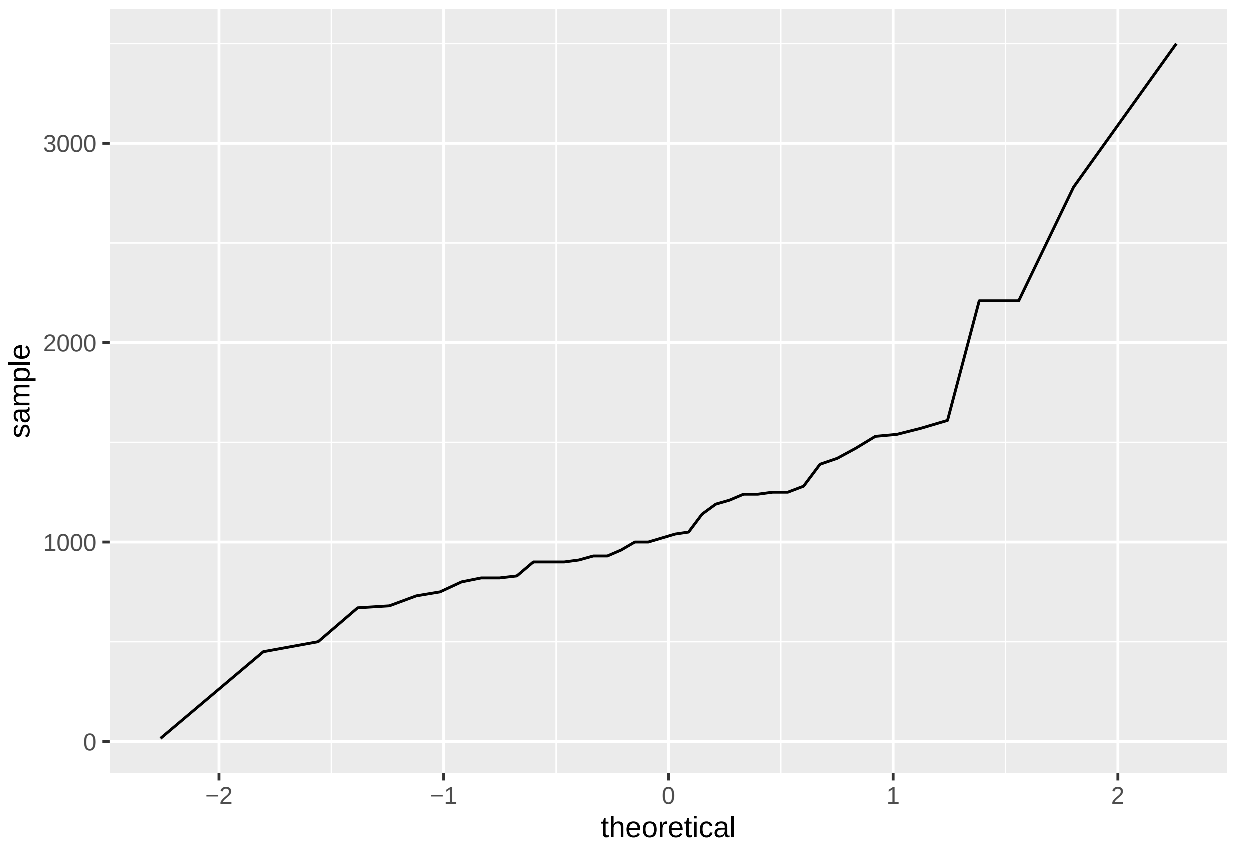
Next, for Mcdonalds.

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

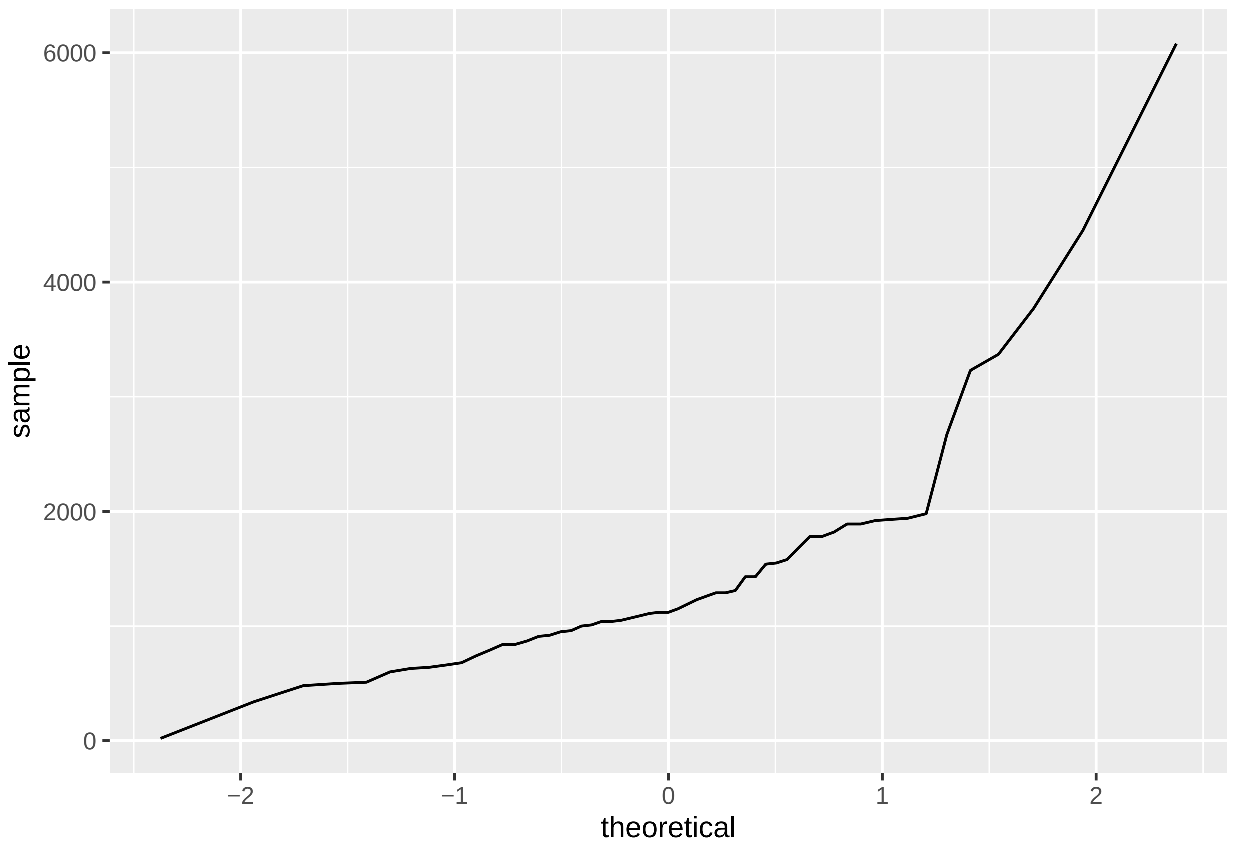


Let's verify that the sodium distributions are Normal using the qq-plots. The two distributions appear to be normal but contains *steps*, i.e. large gradations.

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



```
ggplot(data = mcdonalds, aes(sample = sodium)) + geom_line(stat = "qq")
```



10 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?

Step wise pattern might be due to large **jumps** in sodium content of items. Let's sort the sodium data to see if it is the case.

We see large jumps from 1980 to 2607, from 26to to 3230, from 3370 to 3770, from 3770 to 4450, and from 4450 to 6080.

```
sort(mcdonalds$sodium)
```

```
##  [1]   20  340  480  500  510  600  630  640  660  680  740  790  840  840  870
## [16]  910  920  950  960 1000 1010 1040 1040 1050 1070 1090 1110 1120 1120 1150
## [31] 1190 1230 1260 1290 1290 1310 1430 1430 1540 1550 1580 1680 1780 1780 1820
## [46] 1890 1890 1920 1930 1940 1980 2670 3230 3370 3770 4450 6080
```

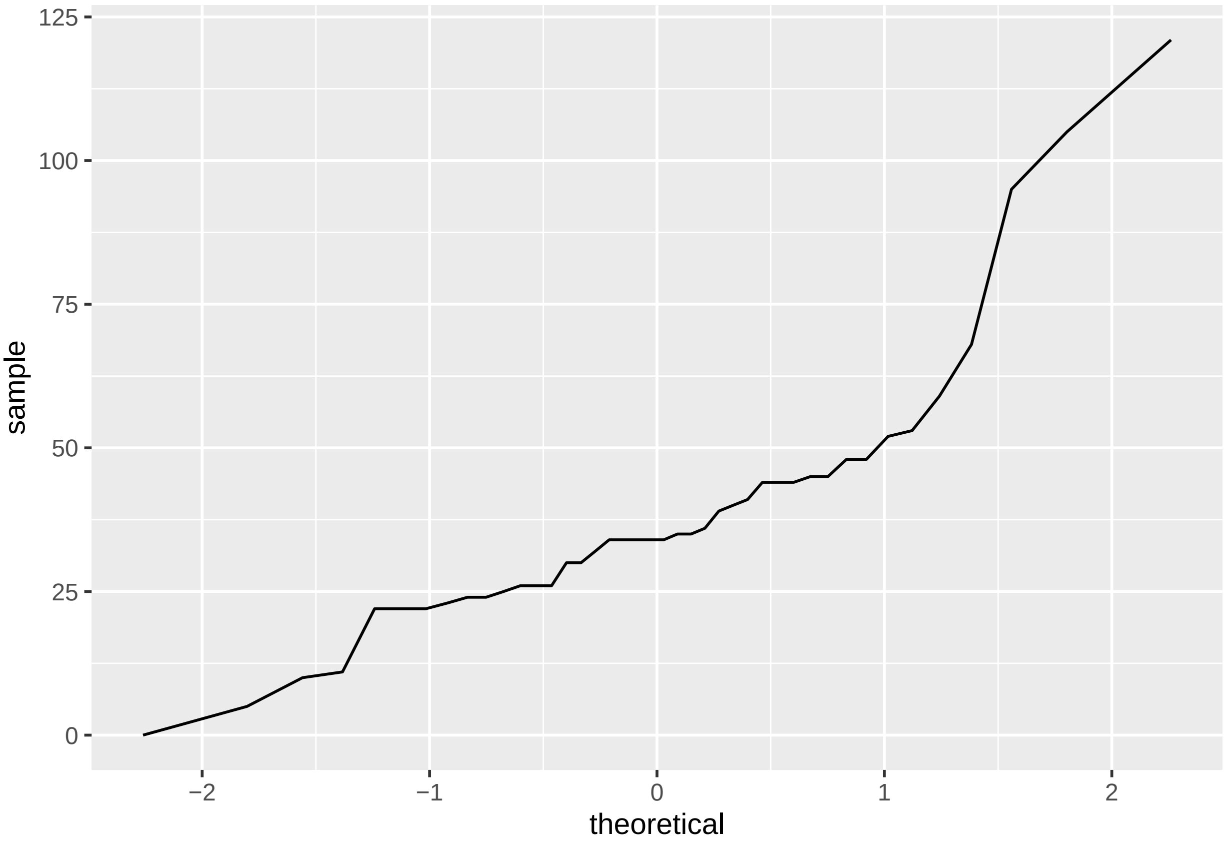
11 Exercise 9

> 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.

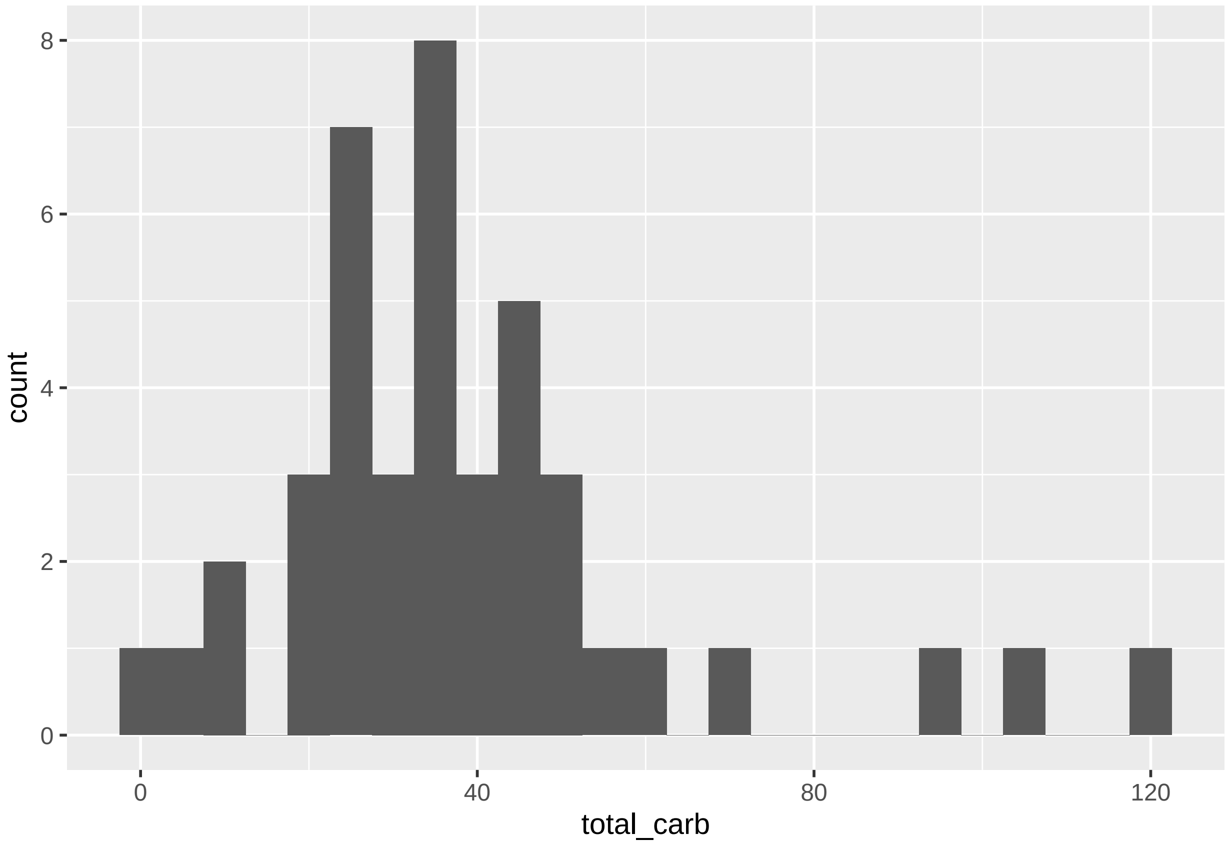
Let's look at total carbs at Dairy Queen. First we make sure that carbs distribution is Normal. It does look Normal and appears to be skewed on the right.

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



The histogram plotconfirms right skewness.

```
ggplot(dairy_queen, aes(x=total_carb)) + geom_histogram(binwidth = 5)
```


count
0
2
4
6
8
0
40
80
120
total_carb