Solution

Jawaid Hakim

2022-09-30

Contents

0.1 Load packages

0.2 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
```

0.3 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.
```

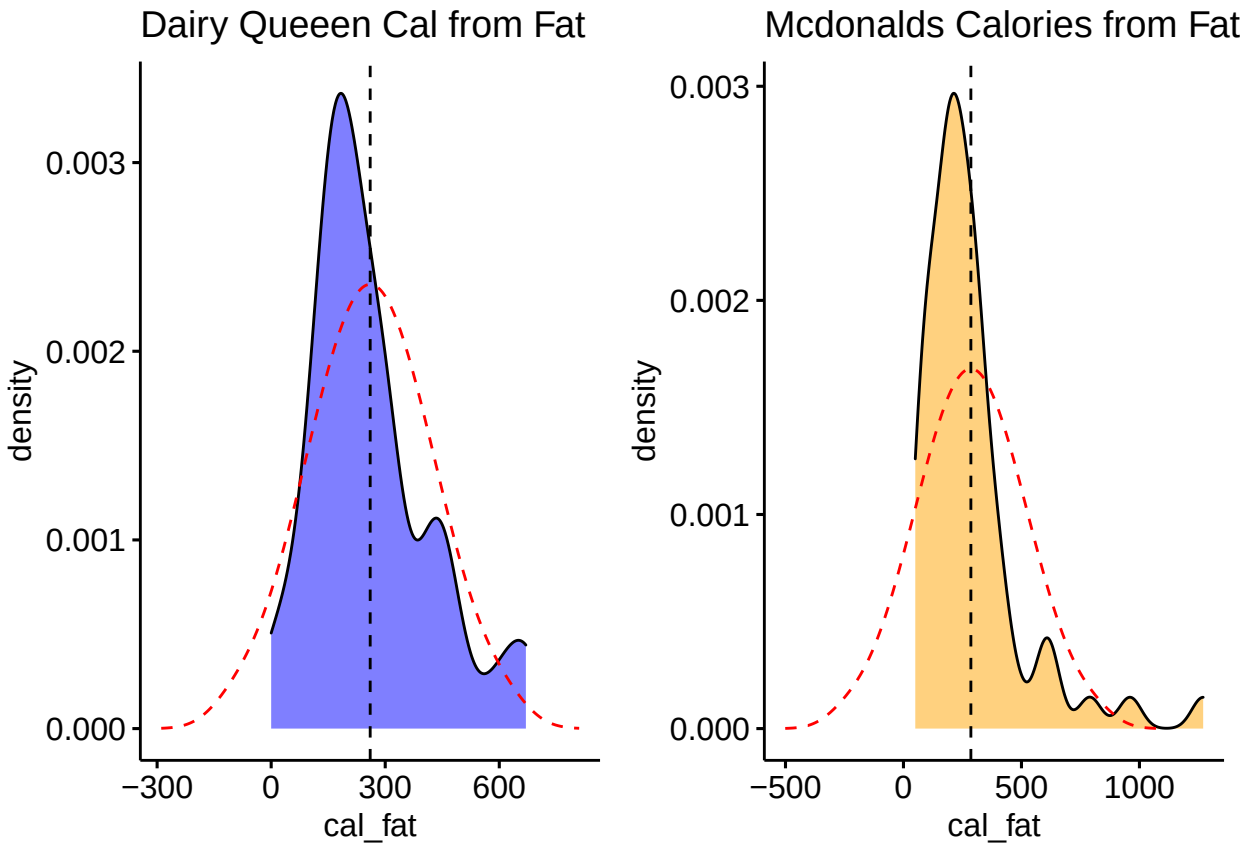
```
dqplot <- dqplot +
         stat_overlay_normal_density(color = "red", linetype = "dashed")

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)
```



0.4 Exercise 2

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

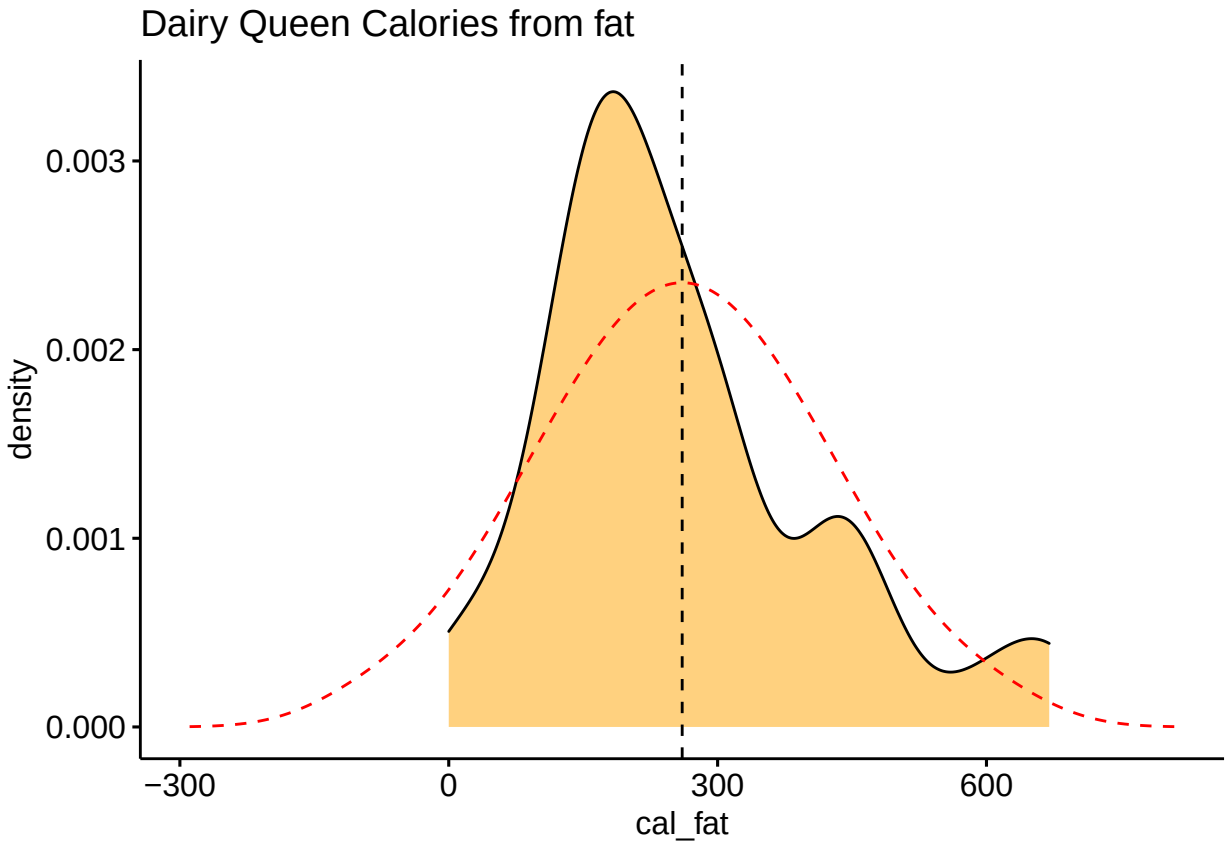
There is close fit between the two plots so calories from fat for Dairy Queen seems to be a Normal Distribution.

```
mdplot <- ggdensity(dairy_queen,
       'cal_fat',
       title = 'Dairy Queen Calories from fat',
       fill = 'Orange',
       add = c('mean'),
       palette = 'jco'
)
```

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

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

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



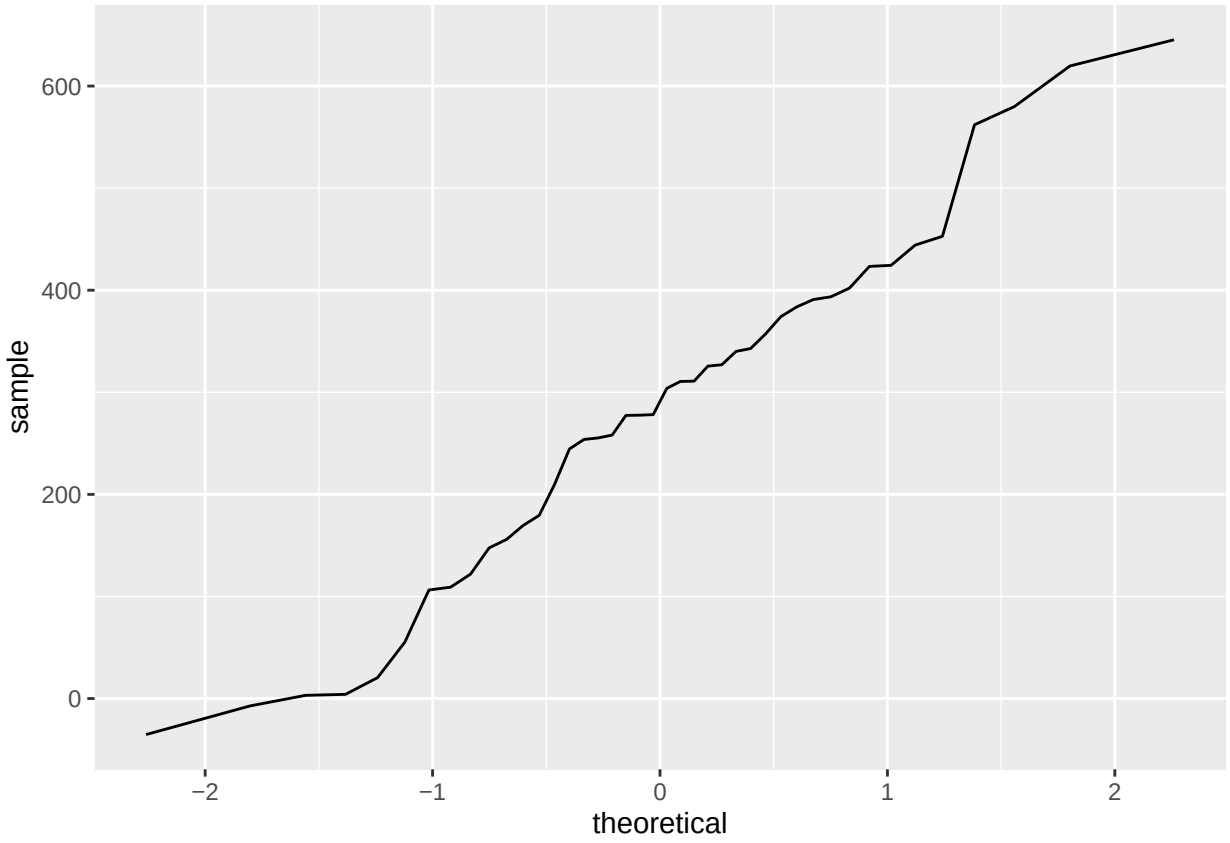
0.5 Exercise 3

> 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, with outliers ion the middle and right tail.

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

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

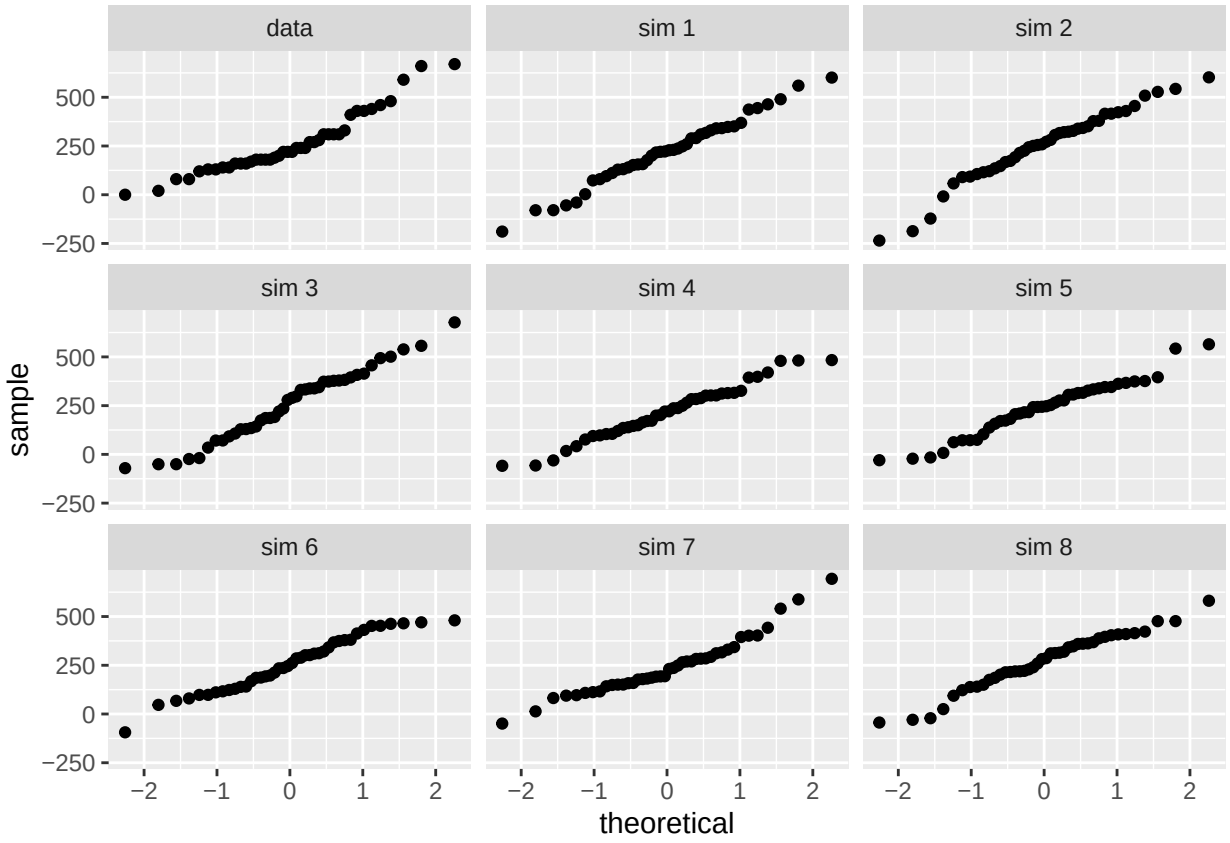


0.6 Exercise 4

> 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.

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

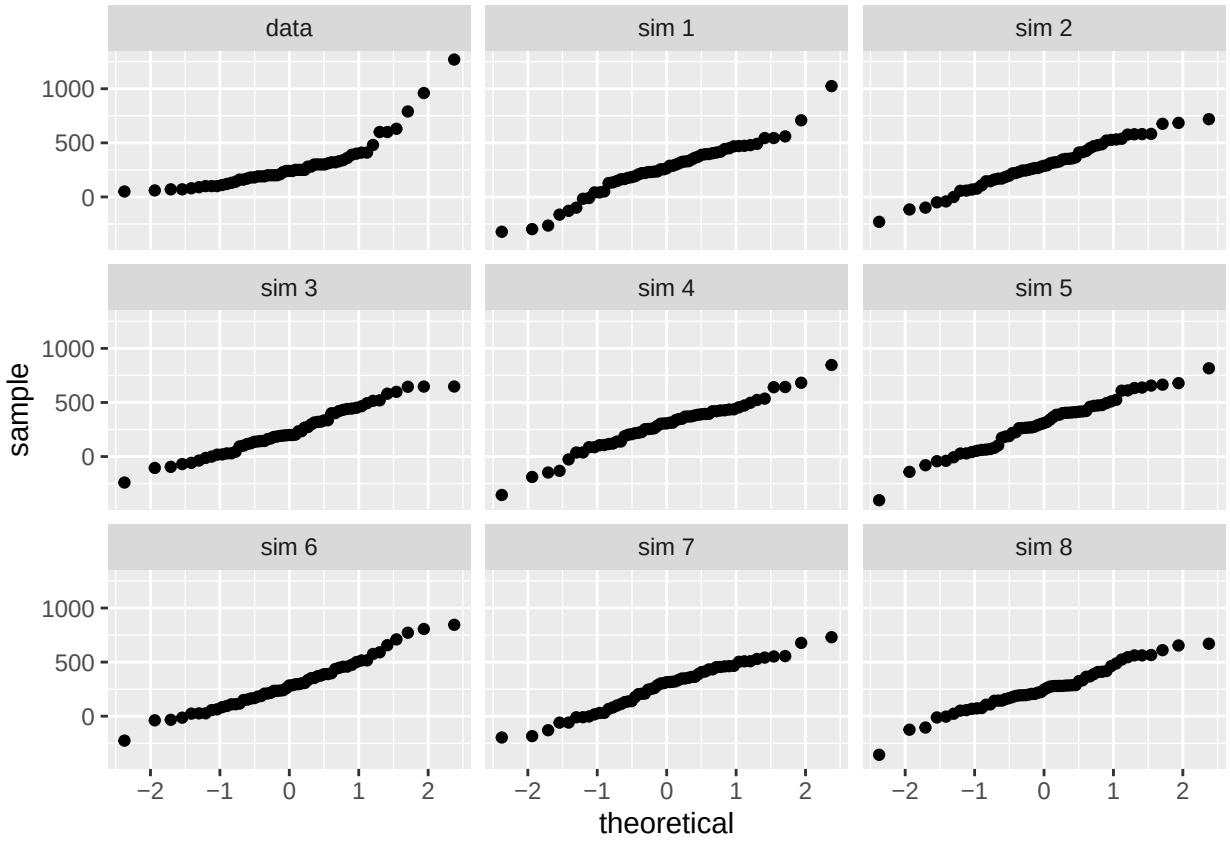


0.7 Exercise 5

> 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.

```
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.

0.8 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?

0.9 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 we compute the mean and standard deviation of sodium for both restaurants. Note, we explicitly set the seed for the RNG.

```
set.seed(738349)
```

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.

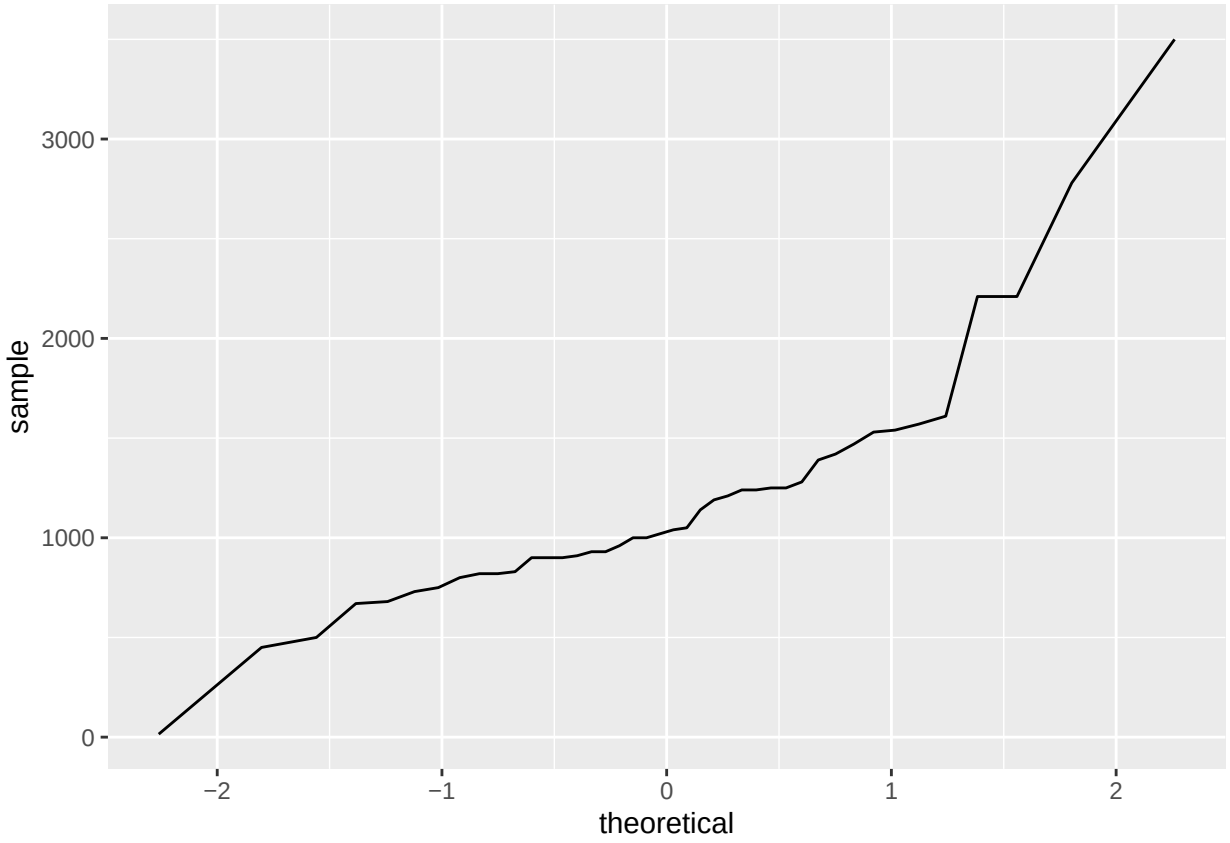
```
dqmean_sodium <- mean(dairy_queen$sodium) # mean for sodium
dqmean_sodium
```

```
## [1] 1181.786
```

```
dqsd_sodium   <- sd(dairy_queen$sodium)   # sd
dqsd_sodium
```

```
## [1] 609.9398
```

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



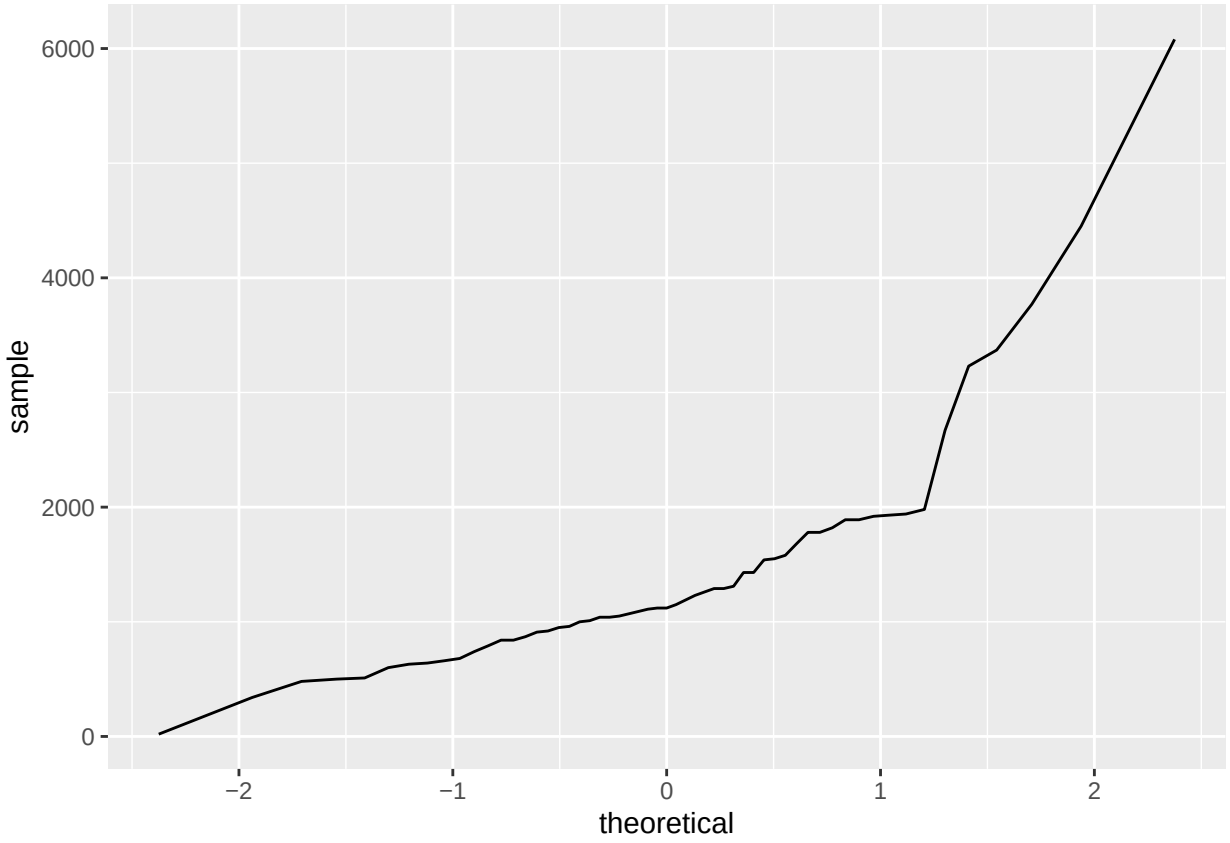

```
mcmean_sodium <- mean(mcdonalds$sodium)
mcmean_sodium
```

```
## [1] 1437.895
```

```
mcsd_sodium   <- sd(mcdonalds$sodium)
mcsd_sodium
```

```
## [1] 1036.172
```

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



0.10 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. One possible explanation for might be if reported sodium content is being rounded up/down.

0.11 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.

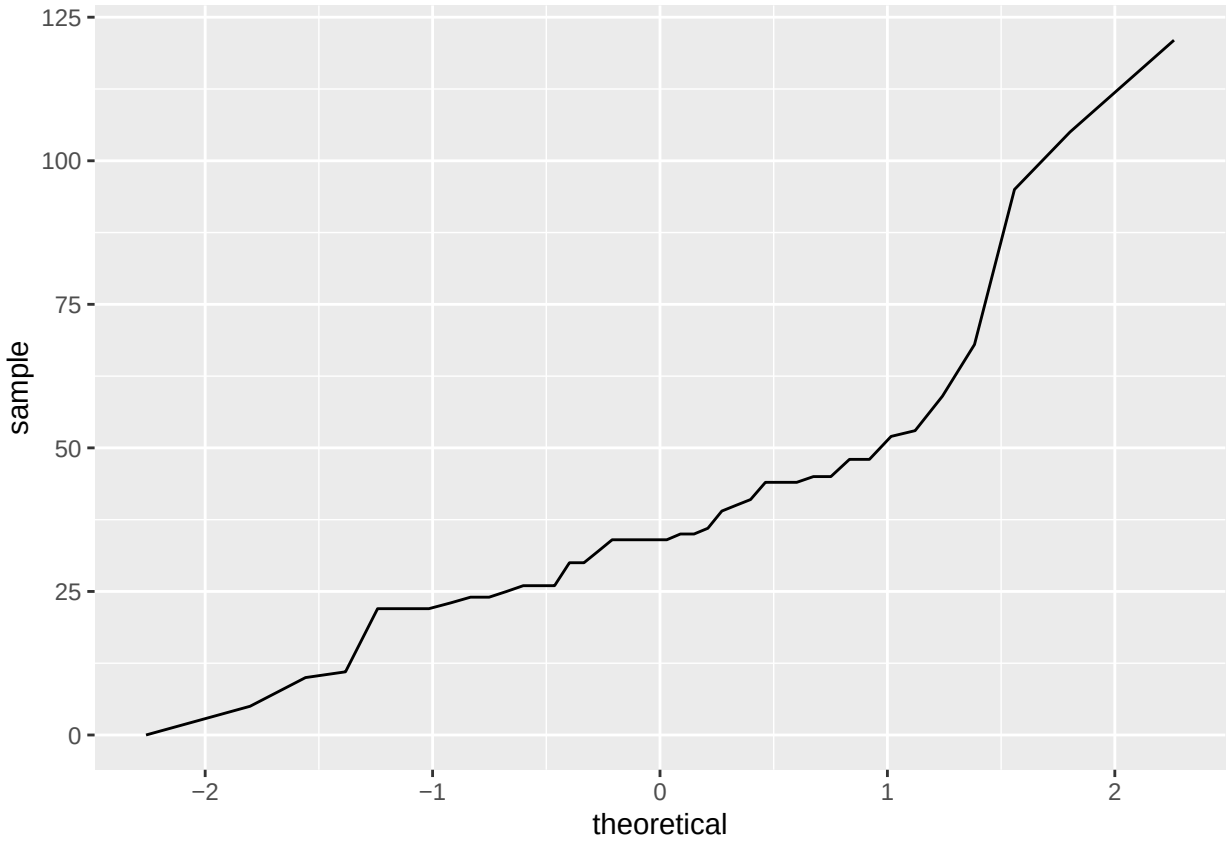
```
dqmean_carb <- mean(dairy_queen$total_carb) # mean for sodium
dqmean_carb
```

```
## [1] 38.69048
```

```
dqmean_carb   <- sd(dairy_queen$total_carb)   # sd
dqmean_carb
```

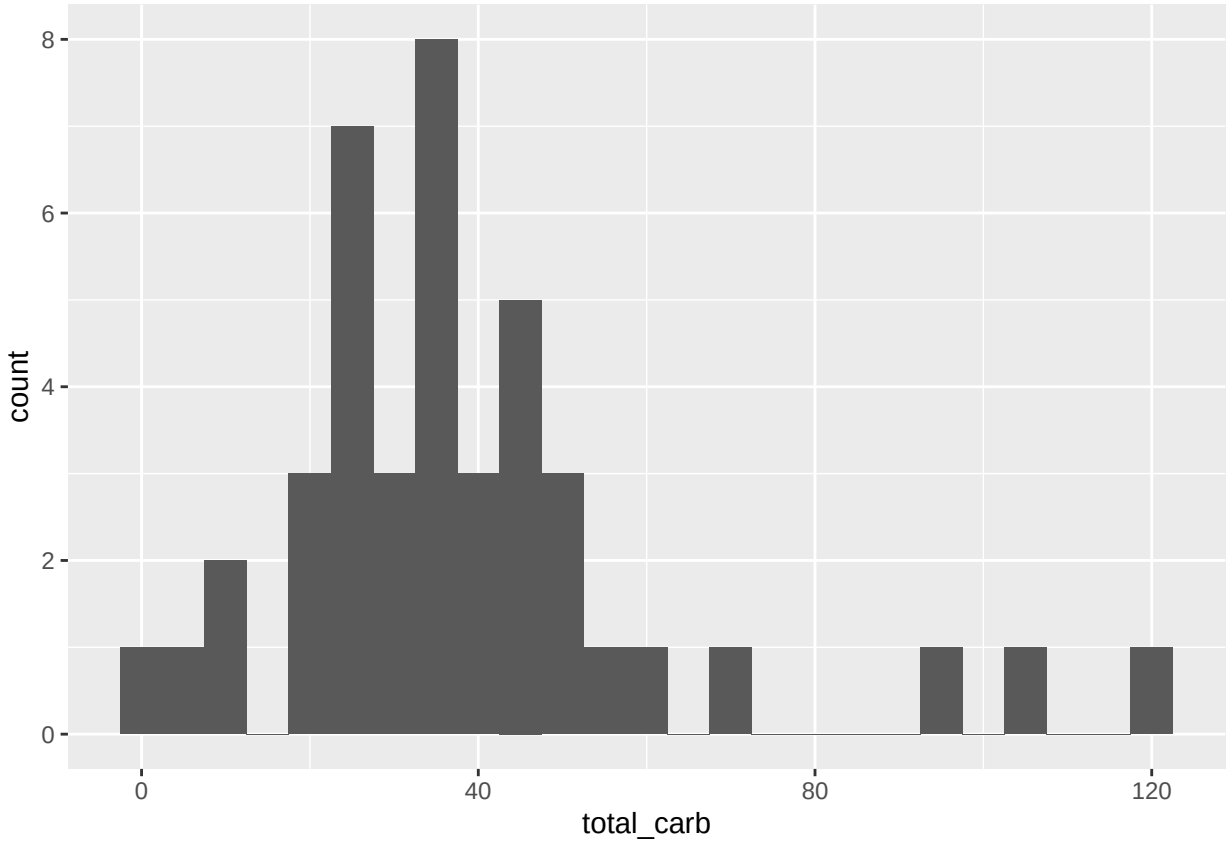
```
## [1] 23.72966
```

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



Let's plat a histogram to verify right skew. Histogram confirms a long right tail.

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


8
6
4
2
0
count
0
40
80
120
total_carb