Inference for one mean

Put your name here

Put the date here

Introduction

In this module, we'll learn about the Student t distribution and use it to perform a t-test for a single mean.

Instructions

Presumably, you have already created a new project and downloaded this file into it. From the `Run` menu above, select `Run All` to run all existing code chunks.

When prompted to complete an exercise or demonstrate skills, you will see the following lines in the document:

<div style="color:blue; text-align:center">— ANSWER —</div>

These lines demarcate the region of the R Markdown document in which you are to show your work.

Sometimes you will be asked to add your own R code. That will appear in this document as a code chunk with a request for you to add your own code, like so:

```
# Add code here
```

Be sure to remove the line `# Add code here` when you have added your own code. You should run each new code chunk you create by clicking on the dark green arrow in the upper-right corner of the code chunk.

Sometimes you will be asked to type up your thoughts. That will appear in the document with the words, "Please write up your answer here." Be sure to remove the line "Please write up your answer here" when you have written up your answer. In these areas of the assignment, please use contextually meaningful full sentences/paragraphs (unless otherwise indicated) and proper spelling, grammar, punctuation, etc. This is not R code, but rather a free response section where you talk about your analysis and conclusions. You may need to use inline R code in these sections.

When you are finished with the assignment, knit to PDF and proofread the PDF file **carefully**. Do not download the PDF file from the PDF viewer; rather, you should export the PDF file to your computer by selecting the check box next to the PDF file in the Files pane, clicking the `More` menu, and then clicking `Export`. Submit your assignment according to your professor's instructions.

Load Packages

We load the standard `mosaic` package as well as the `OIdata` package to get the `teacher` data and the `openintro` package for the `hsb2` data. The `broom` package will give us tidy output.

```
library(OIdata)
data(teacher)
library(openintro)
library(broom)
library(mosaic)
```

We'll set the seed for our simulations.

```
set.seed(5151977)
```

Simulating means

Systolic blood pressure (SBP) for women in the U.S. and Canada follows a normal distribution with a mean of 114 and a standard deviation of 14.

Suppose we gather a random sample of 4 women and measure their SBP. We can simulate doing that with the `rnorm` command:

```
SBP_sample <- rnorm(4, mean = 114, sd = 14)
SBP_sample
```

```
## [1]  99.75130 126.47739  99.53632 115.05247
```

We summarize our sample by taking the mean and standard deviation:

```
mean(SBP_sample)
```

```
## [1] 110.2044
```

```
sd(SBP_sample)
```

```
## [1] 13.05615
```

The sample mean $\bar{y} = 110.2043696$ is somewhat close to the true population mean $\mu = 114$ and the sample standard deviation $s = 13.0561519$ is somewhat close to the true population standard deviation $\sigma = 14$. (μ is the Greek letter "mu" and σ is the Greek letter "sigma".)

Let's simulate lots of samples of size 4. For each sample, we calculate the sample mean.

```
sims <- do(2000) * mean(rnorm(4, mean = 114, sd = 14))
head(sims)
```

```
##        mean
## 1 105.5526
## 2 110.9552
## 3 119.2016
## 4 111.0685
## 5 102.7575
## 6 109.9127
```

Again, we see that the sample means are close to 114, but there is some variability. Naturally, not every sample is going to have an average of exactly 114. So how much variability do we expect? Let's graph and find out. I'm going to set the x-axis manually so that we can do some comparisons later.

```
ggplot(sims, aes(x = mean)) +
    geom_histogram(binwidth = 1) +
    scale_x_continuous(limits = c(86, 142),
                       breaks = c(93, 100, 107, 114, 121, 128, 135))
```
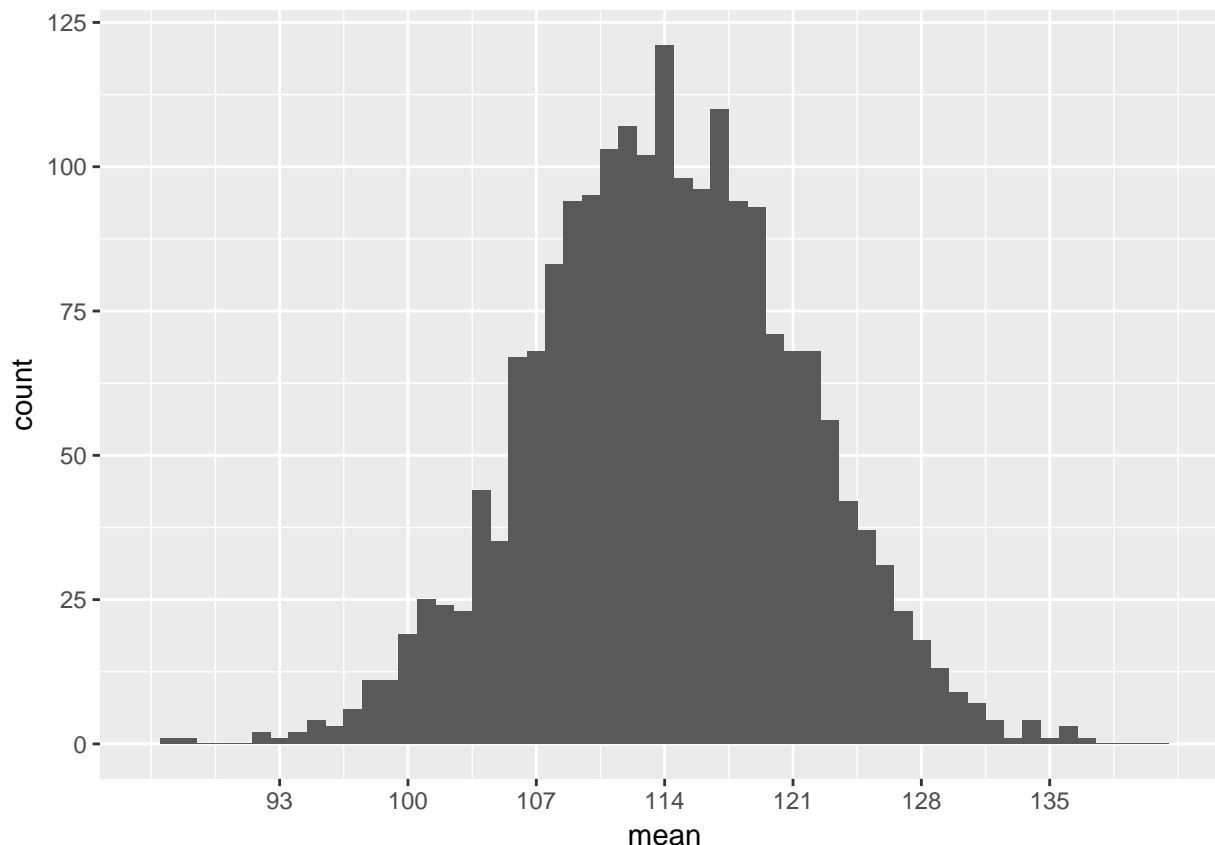
Most sample means are around 114, but there is a good range of possibilities from around 93 to 135. The population standard deviation σ is 14, but the standard deviation in this graph is clearly much smaller than that. (A large majority of the samples are within 14 of the mean!)

With some fancy mathematics, one can show that the standard deviation of this sampling distribution is not σ, but rather $\sigma/\sqrt{n}$. In other words, this sampling distribution of the mean has a standard error of

$$\frac{\sigma}{\sqrt{n}} = \frac{14}{\sqrt{4}} = 7.$$

This makes sense: as the sample size increases, we expect the sample mean to be more and more accurate, so the standard error should shrink with large sample sizes.

Let's re-scale the y-axis to use percentages instead of counts. Then we should be able to superimpose the normal model $N(114, 7)$ to check visually that it's the right fit.

```
ggplot(sims, aes(x = mean)) +
    geom_histogram(aes(y = ..density..), binwidth = 1) +
    scale_x_continuous(limits = c(86, 142),
                       breaks = c(93, 100, 107, 114, 121, 128, 135)) +
    stat_function(fun = dnorm, args = list(mean = 114, sd = 7),
                  color = "red")
```
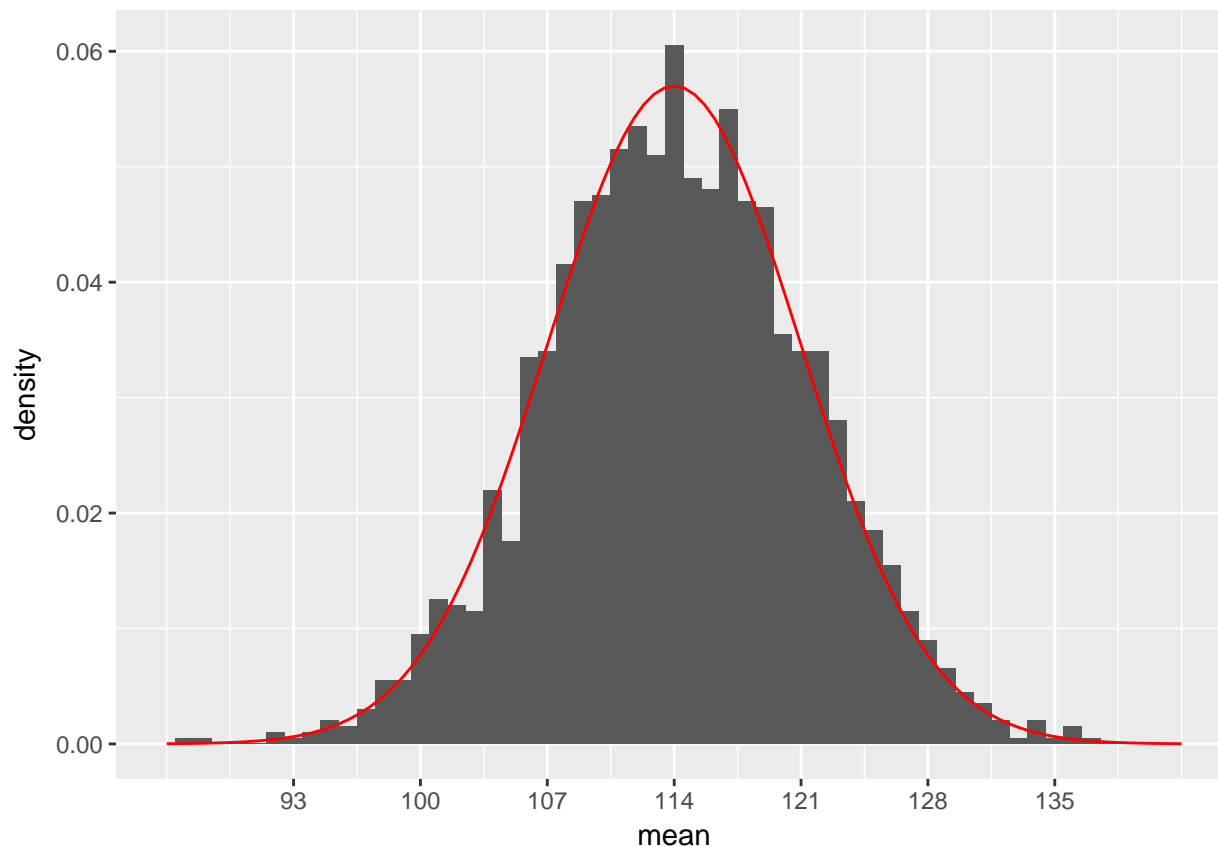
Looks pretty good!

All we do now is convert everything to z-scores. In other words, suppose we sample 4 individuals from a population distributed according to the normal model $N(0, 1)$. Now the standard error of the sampling distribution is

$$\frac{\sigma}{\sqrt{n}} = \frac{1}{\sqrt{4}} = 0.5.$$

The following code will accomplish all of this. (Don't worry about the messy syntax. All I'm doing here is making sure that this graph looks exactly the same as the previous graph, except now centered at $\mu = 0$ instead of $\mu = 114$.)

```r
sims_z <- data.frame(mean = scale(sims$mean, center = 114, scale = 14))
ggplot(sims_z, aes(x = mean)) +
    geom_histogram(aes(y = ..density..), binwidth = 1/14) +
    scale_x_continuous(limits = c(-2, 2),
                       breaks = c(-1.5, -1, -0.5, 0, 0.5, 1, 1.5)) +
    stat_function(fun = dnorm, args = list(mean = 0, sd = 0.5),
                  color = "red")
```
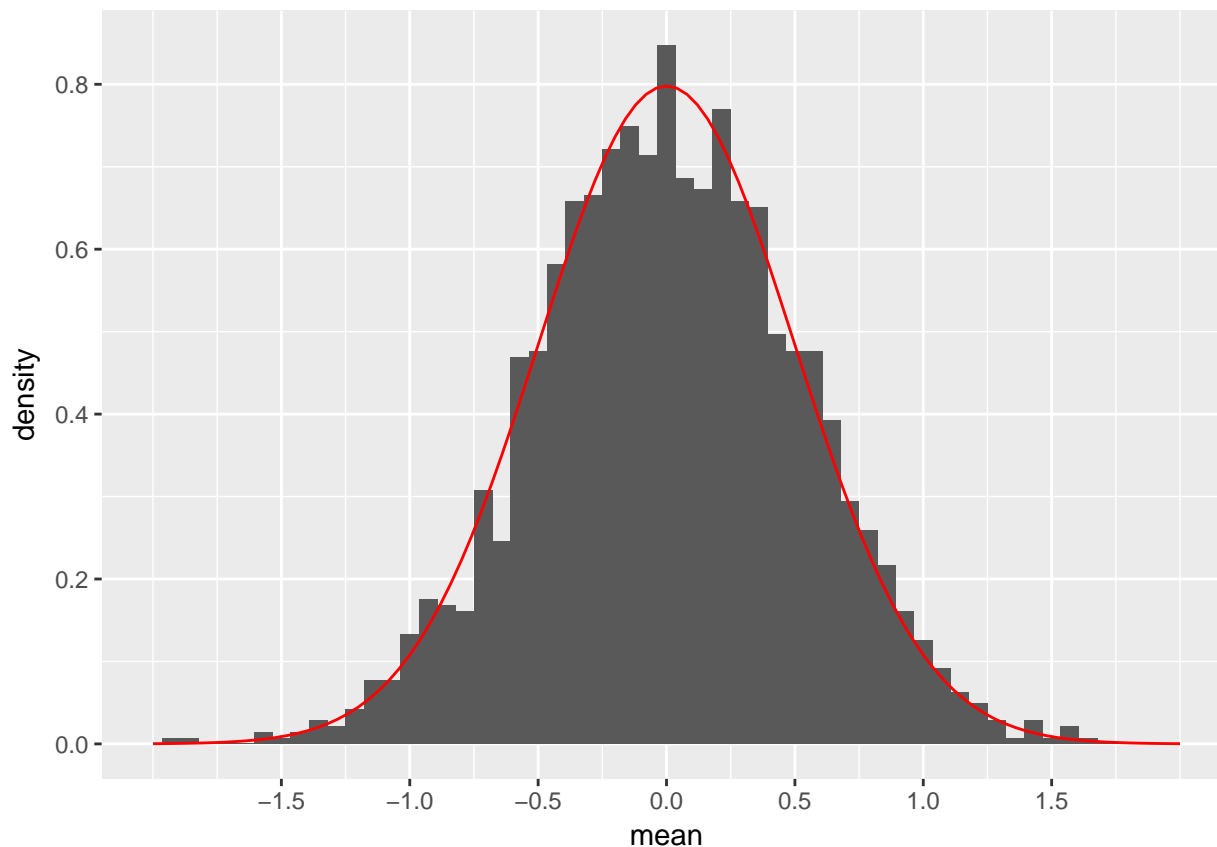
Unknown standard errors

If we want to run a hypothesis test, we will have a null hypothesis about the true value of the population mean μ. For example,

$$H_0 : \mu = 114$$

Now we gather a sample and compute the sample mean, say 110.2043696. We would like to be able to compare the sample mean $\bar{y}$ to the hypothesized value 114 using a z-score:

$$z = \frac{(\bar{y} - \mu)}{\sigma/\sqrt{n}} = \frac{(110.20 - 114)}{\sigma/\sqrt{4}}.$$

However, we have a problem: we don't know the true value of σ. (In our SBP example, we do happen to know it's 14, but we won't know this for a general research question.)

The best we can do with a sample is calculate this z-score replacing the unknown σ with the sample standard deviation s, 13.0561519. We'll call this a "t-score" instead of a "z-score":

$$t = \frac{(\bar{y} - \mu)}{s/\sqrt{n}} = \frac{(110.20 - 114)}{13.06/\sqrt{4}} = -0.58.$$

The problem is that s is not a perfect estimate of σ. We saw earlier that s is usually close to σ, but s has its own sampling variability. That means that our earlier simulation in which we assumed that σ was known and

equal to 14 was wrong for the type of situation that will arise when we run a hypothesis test. How wrong was it?

Simulating t-scores

Let's run the simulation again, but this time with the added uncertainty of using s to estimate σ.

The first step is to write a little function of our own to compute simulated t-scores. This function will take a sample of size n from the true population $N(\mu, \sigma)$, calculate the sample mean and sample standard deviation, then compute the t-score. Don't worry: you won't be required to do anything like this on your own.

```
sim_t <- function(n, mu, sigma) {
    sample_values <- rnorm(n, mean = mu, sd = sigma)
    y_bar <- mean(sample_values)
    s <- sd(sample_values)
    t <- (y_bar - mu)/(s / sqrt(n))
}
```

Now we can simulate doing this 2000 times.

```
sims_t <- do(2000) * sim_t(4, mu = 114, sigma = 14)
head(sims_t)
```

```
##         sim_t
## 1   3.1661223
## 2  -0.6996851
## 3   1.2384582
## 4  -0.1713226
## 5   0.3781026
## 6   0.1719113
```

Let's plot our simulated t-scores alongside a normal distribution.

```
ggplot(sims_t, aes(x = sim_t)) +
    geom_histogram(aes(y = ..density..), binwidth = 0.25) +
    scale_x_continuous(limits = c(-5, 5),
                        breaks = c(-4, -3, -2, -1, 0, 1, 2, 3, 4)) +
    stat_function(fun = dnorm,  args = list(mean = 0, sd = 1),
                    color = "red")
```

```
## Warning: Removed 29 rows containing non-finite values (stat_bin).
```
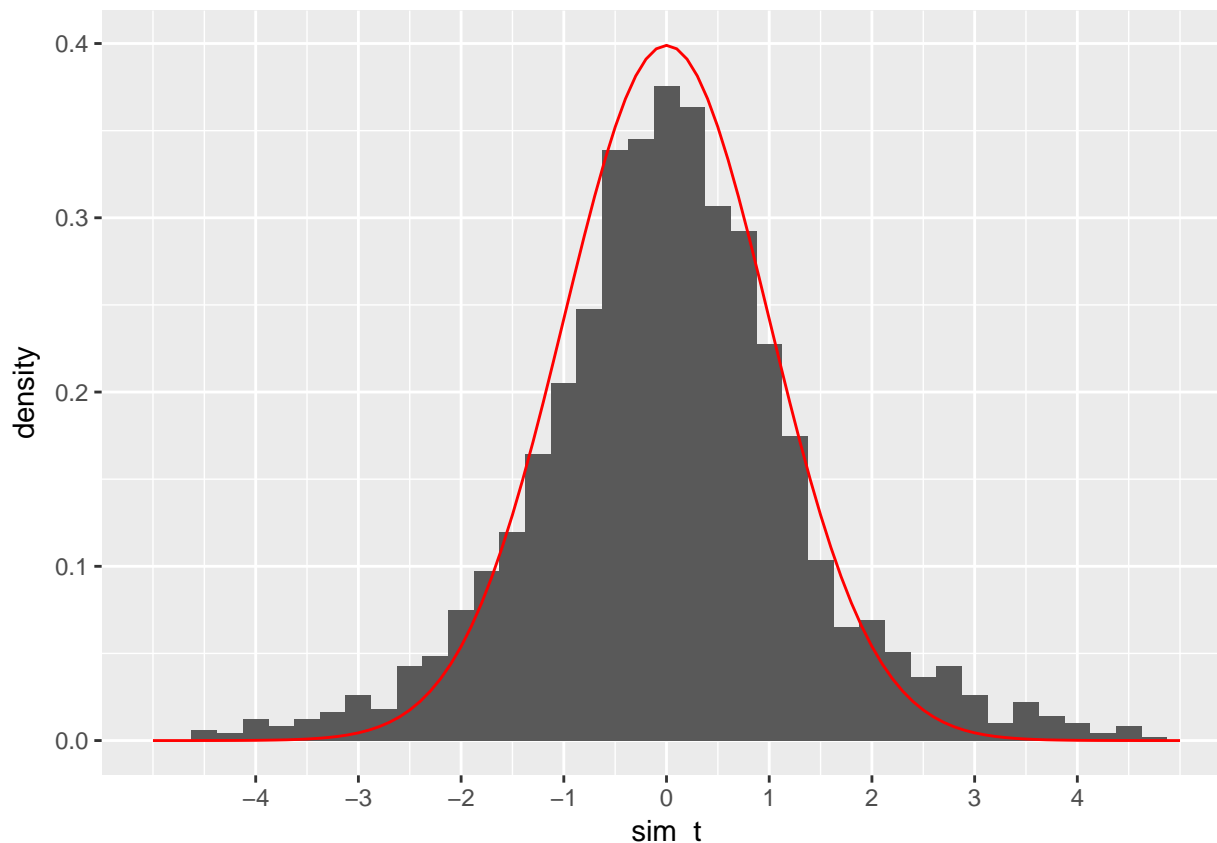
These t-scores are somewhat close to the normal model we had when we knew σ, but the fit doesn't look quite right. The peak of the simulated values isn't quite high enough, and the tails seem to spill out over the much thinner tails of the normal model.

William Gosset figured this all out in the early 20th century. He found a new function that is similar to a normal distribution, but is more spread out. This new function accounts for the extra variability one gets when using the sample standard deviation s as an estimate for the true population standard deviation σ.

Gosset published his findings under the pseudonym "Student" and this new function became known as the *Student t distribution*.

Gosset also realized that the spread of the t distribution depends on the sample size. This makes sense: the accuracy of s will be greater when we have a larger sample. In fact, for large enough samples, the t distribution is very close to a normal model.

Gosset used the term *degrees of freedom* to describe how the sample size influences the spread of the t distribution. It's somewhat mathematical and technical, so suffice it to say here that the number of degrees of freedom is simply the sample size minus 1:

$$df = n - 1.$$

So is the t model correct for our simulated t-scores? Our sample size was 4, so we should use a t model with 3 degrees of freedom. Let's plot it in green on top of our previous graph and see:

```
ggplot(sims_t, aes(x = sim_t)) +
    geom_histogram(aes(y = ..density..), binwidth = 0.25) +
    scale_x_continuous(limits = c(-5, 5),
                       breaks = c(-4, -3, -2, -1, 0, 1, 2, 3, 4)) +
```

```
    stat_function(fun = dnorm,  args = list(mean = 0, sd = 1),
                  color = "red") +
    stat_function(fun = dt,  args = list(df = 3),
                  color = "green")
```

Warning: Removed 29 rows containing non-finite values (stat_bin).
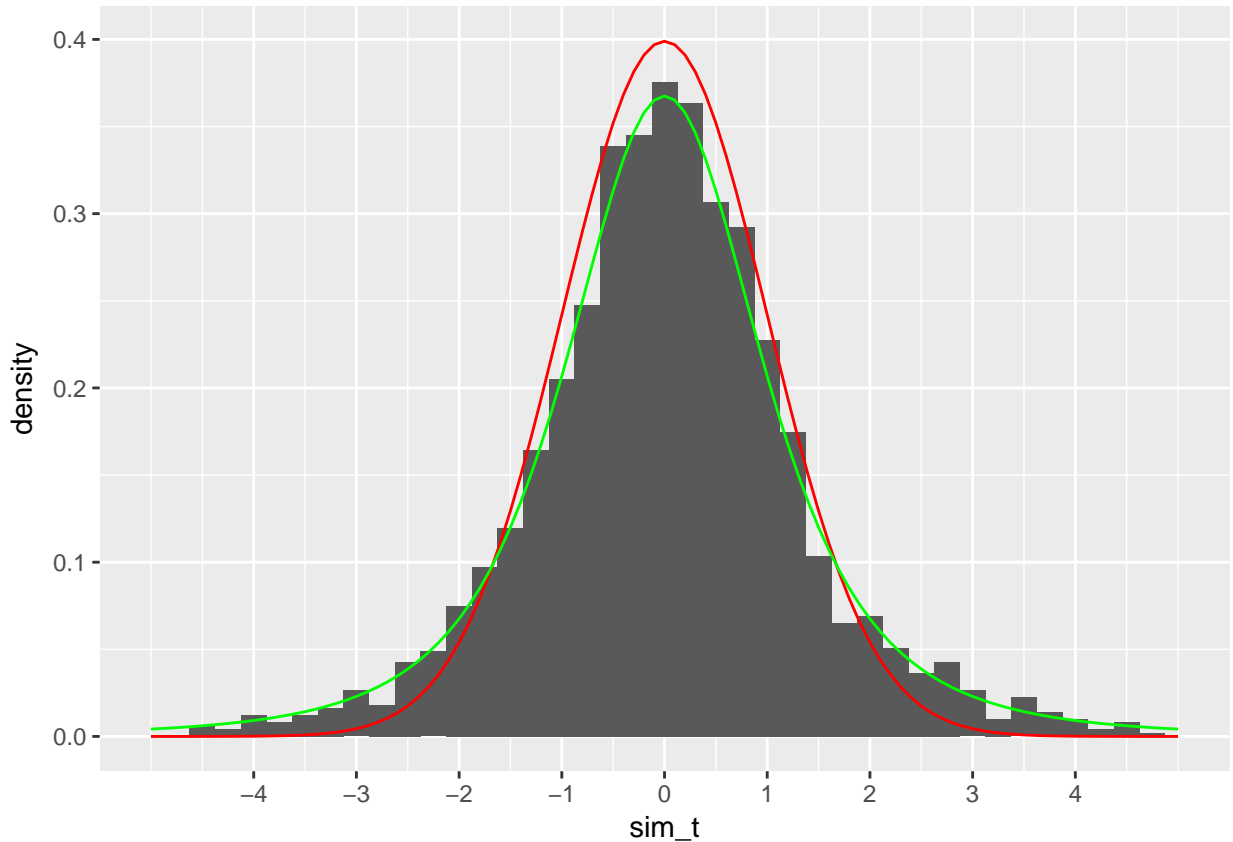
The green curve seems to fit the simulated values much better.

Inference for one mean

When we have a single numerical variable, we can ask if the sample mean is consistent or not with a null hypothesis. We will use a t model for our sampling distribution model as long as certain conditions are met.

One of the assumptions we made in the simulation above was that the true population was normally distributed. In general, we have no way of knowing if this is true. So instead we check the *nearly normal* condition: if a histogram or QQ plot of our data shows that the data is nearly normal, then there is a reasonable assumption that the whole population is shaped the same way.

If our sample size is large enough, the central limit theorem tells us that the sampling distribution gets closer and closer to a normal model. Therefore, we'll use a rule of thumb that says that if the sample size is greater than 30, we won't worry too much about any deviations from normality in the data.

The number 30 is somewhat arbitrary. If the sample size is 25 and a histogram shows only a little skewness, we're probably okay. But if the sample size is 10, we need for the data to be very normal to justify using the t model. The irony, of course, is that small sample sizes are the hardest to check for normality. We'll have to use our best judgment.

Outliers

We also need to be on the lookout for outliers. We've seen before that outliers can have a huge effect on means and standard deviations, especially when sample sizes are small. Whenever we find an outlier, we need to investigate.

Some outliers are mistakes. Perhaps someone entered data incorrectly into the computer. When it's clear that outliers are data entry errors, we are free to either correct them (if we know what error was made) or delete them from our data completely.

Some outliers are not necessarily mistakes, but should be excluded for other reasons. For example, if we are studying the weight of birds and we have sampled a bunch of hummingbirds and one emu, the emu's weight will appear as an outlier. It's not that its weight is "wrong", but it clearly doesn't belong in the analysis.

In general, though, outliers are "real" data that just happen to be unusual. It's not ethical simply to throw away such data points because they are inconvenient. (We only do so in very narrow and well-justified circumstances like the emu.) The best policy to follow when faced with such outliers is to run inference twice—once with the outlier included, and once with the outlier excluded. If when running a hypothesis test the conclusion is the same either way, then the outlier wasn't all that influential, so we leave it in. If when computing a confidence interval the endpoints don't change a lot either way, then we leave the outlier in. However, when conclusions or intervals are dramatically different depending on whether the outlier was in or out, then we have no choice but to state that honestly.

Research question

The `teacher` data from the `OIdata` package contains information on 71 teachers employed by the St. Louis Public School in Michigan. According to Google, the average teacher salary in Michigan was $63,024 in 2010. So does this data suggest that the teachers in this region are paid differently than teachers in other parts of Michigan?

Let's walk through the rubric.

Exploratory data analysis

Use data documentation (help files, code books, Google, etc.), the str command, and other summary functions to understand the data.

[Type `library(OIdata)` then `?teacher` to read the help file.]

```
str(teacher)
```

```
## 'data.frame':    71 obs. of  8 variables:
##  $ id        : Factor w/ 71 levels "01","02","03",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ degree    : Factor w/ 2 levels "BA","MA": 1 2 2 1 1 1 1 2 1 1 ...
##  $ fte       : Factor w/ 2 levels "0.5","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ years     : num  5 15 16 10 26 28.5 12 32 25 12 ...
##  $ base      : int  45388 60649 60649 54466 65360 65360 58097 68230 65360 58097 ...
##  $ fica      : num  3472 4640 4640 4167 5000 ...
##  $ retirement: num  7689 10274 10274 9227 11072 ...
##  $ total     : num  56549 75563 75563 67859 81432 ...
```

Since `total` is a numerical variable, we can also summarize it using `favstats`:

```
favstats(teacher$total)
```

```
##        min       Q1    median       Q3       max      mean        sd  n missing
##   24793.41 63757.69 74646.85 81432.02 85007.76 70288.64 12439.34 71        0
```
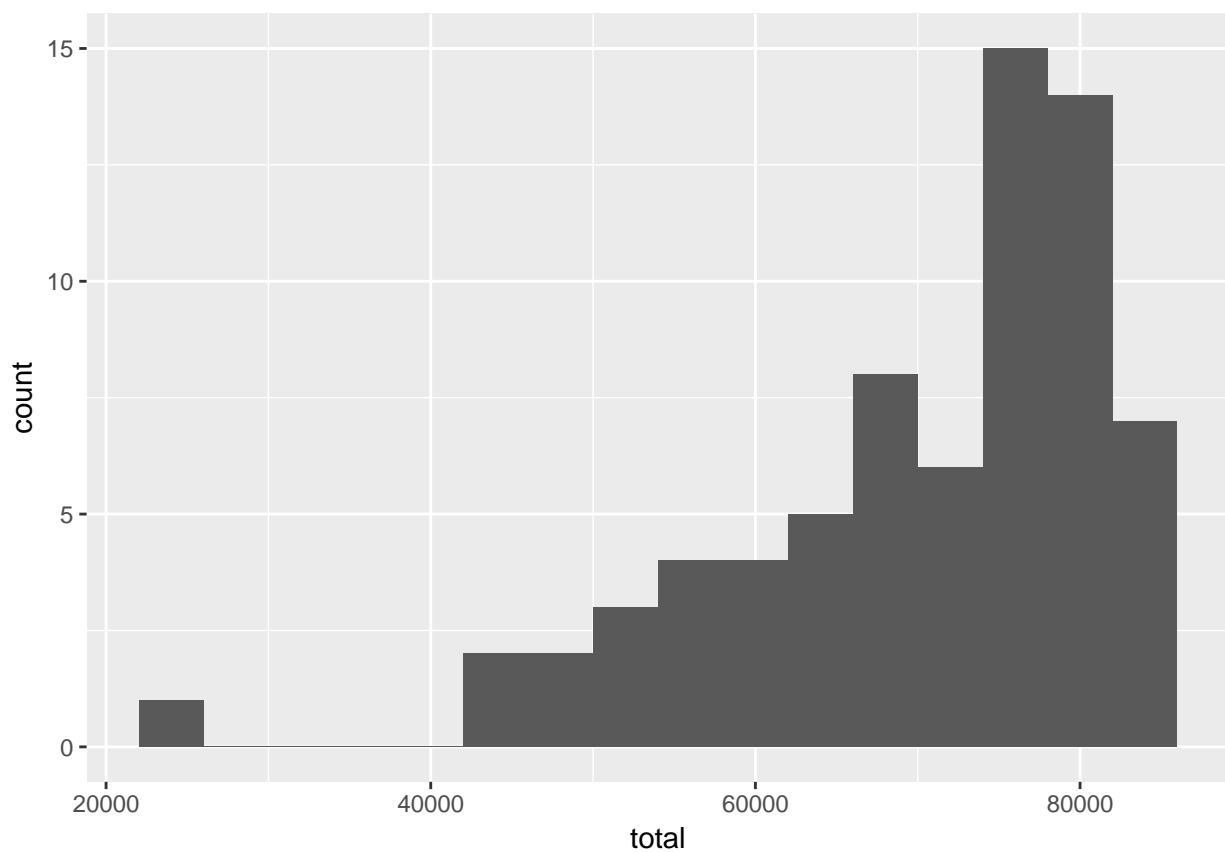
Prepare the data for analysis. [Not always necessary.]

Not necessary here, but see the next section to find out what we do when we discover an outlier.

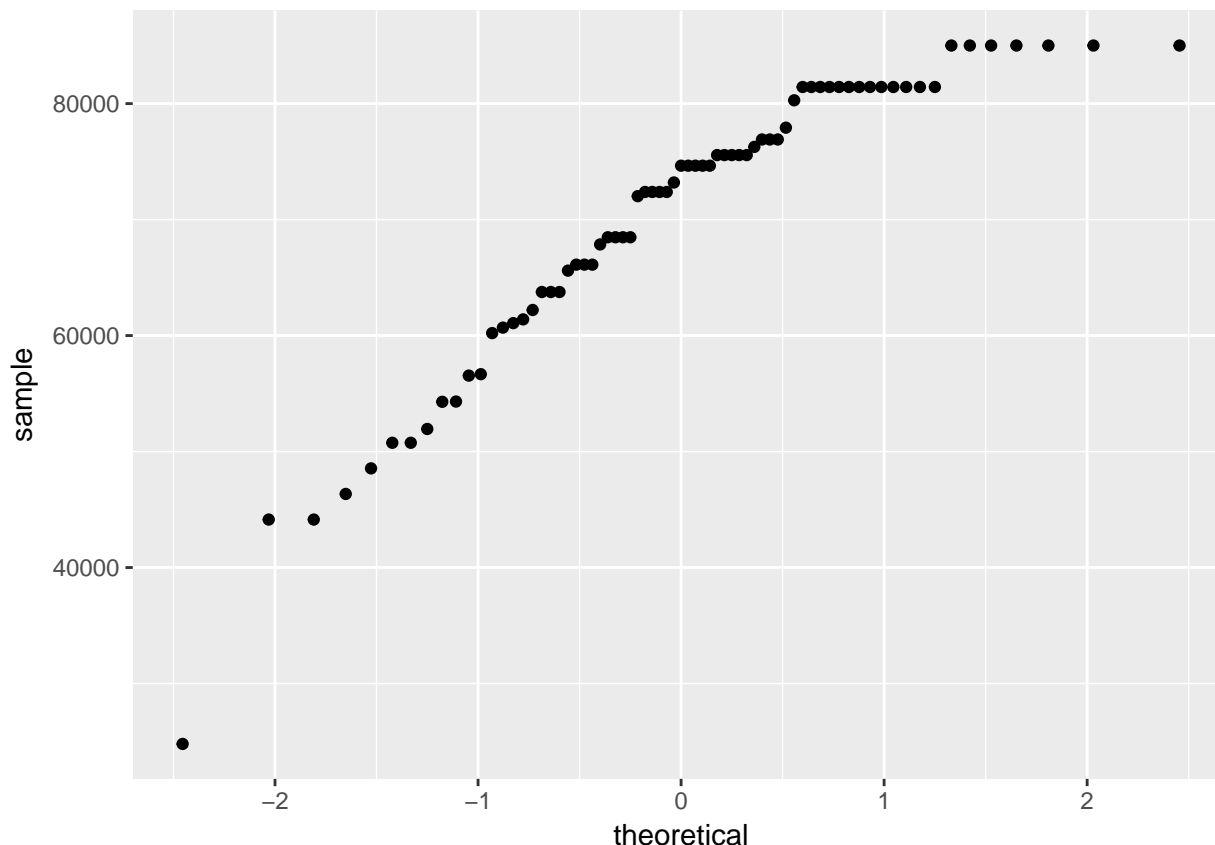**Make tables or plots to explore the data visually.**

Here is a histogram.

```
ggplot(teacher, aes(x = total)) +
    geom_histogram(binwidth = 4000)
```


And here is a QQ plot.

```
ggplot(teacher, aes(sample = total)) +
    geom_qq()
```

This distribution is quite skewed to the left. Of even more concern is the extreme outlier on the left.

With any outlier, we need to investigate. Go to the Console and type

`View(teacher)`

(Be sure to use a capital "V" in `View`).

Exercise

In the spreadsheet view, you can use the arrows in the column headers to sort the data. Sort by `total` (ascending) and see if you can figure out how the person with the lowest total salary is different from all the other teachers. (Hint: you may need to Google "fte".)

–––––––––––––––––––––––– ANSWER ––––––––––––––––––––––––

Please write up your answer here.

Based on your answer to the above exercise, hopefully it's clear that this is an outlier for which we can easily justify exclusion. We can use the `filter` command to get only the rows we want. There are lots of ways to do this, but it's easy enough to grab only salaries above $40,000. (There's only one salary below $40,000, so that outlier will be excluded.)

CAUTION: If you are copying and pasting from this example to use for another research question, the following code chuck is specific to this research question and not applicable in

other contexts.

```
teacher2 <- filter(teacher, total > 40000)
```

Check to make sure this had the desired effect:
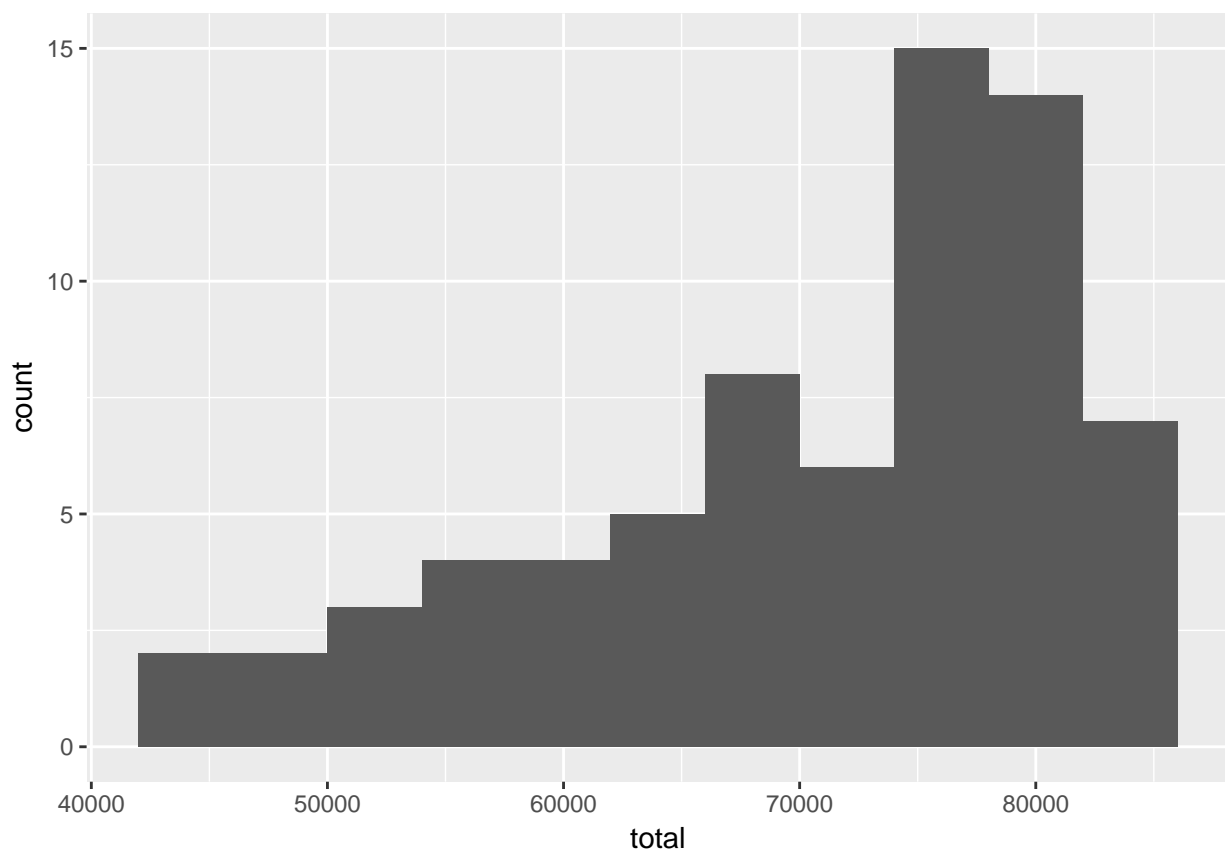
```
favstats(teacher2$total)
```

```
##      min      Q1   median      Q3      max     mean       sd  n missing
##  44138.5 63757.69 74646.85 81432.02 85007.76 70938.57 11249.61 70       0
```
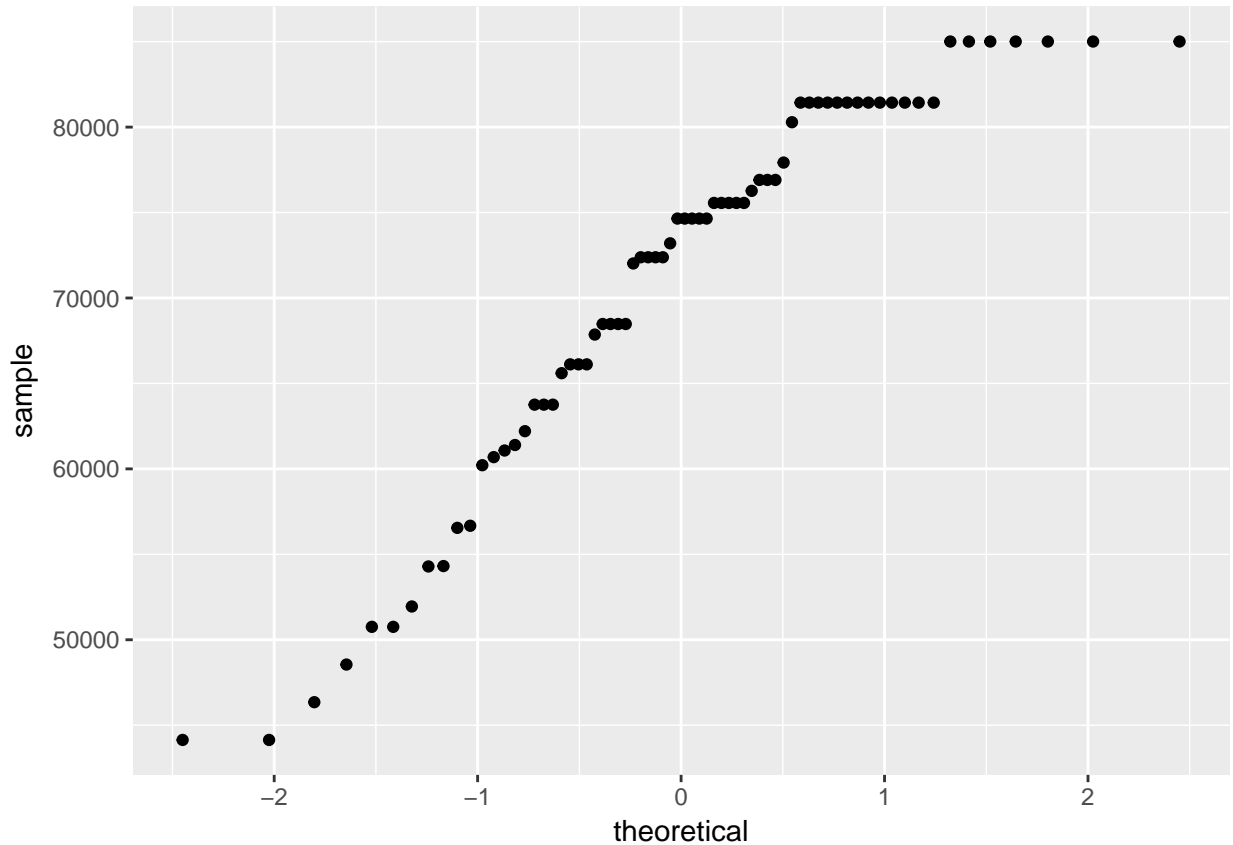
Notice how the min is no longer $24,793.41 and the sample size is 70 instead of 71.

Here are the new plots:

```
ggplot(teacher2, aes(x = total)) +
    geom_histogram(binwidth = 4000)
```



```
ggplot(teacher2, aes(sample = total)) +
    geom_qq()
```

The left skew is still present, but we have removed the outlier.

Hypotheses

Identify the sample (or samples) and a reasonable population (or populations) of interest.

The sample consists of 70 teachers employed by the St. Louis Public School in Michigan. We are using these 70 teachers as a hopefully representative sample of all teachers in that region of Michigan.

Express the null and alternative hypotheses as contextually meaningful full sentences.

H_0 : Teachers in the St. Louis region earn \$63,024 on average. (In other words, these teachers are the same as the teachers anywhere else in Michigan.)

H_A : Teachers in the St. Louis region do not earn \$63,024 on average. (In other words, these teachers are *not* the same as the teachers anywhere else in Michigan.)

Express the null and alternative hypotheses in symbols (when possible).

$H_0 : \mu = 63024$

$H_A : \mu \neq 63024$

Model

Identify the sampling distribution model.

We will use a t model with 69 degrees of freedom.

Check the relevant conditions to ensure that model assumptions are met.

- Random
  - We know this isn't a random sample. We're not sure if this school is representative of other schools in the region, so we'll proceed with caution.
- 10%
  - This is also suspect, as it's not clear that there are 700 teachers in the region. One way to look at it is this: if there are 10 or more schools in the region, and all the school are about the size of the St. Louis Public School under consideration, then we should be okay.
- Nearly Normal
  - For this, we note that the sample size is much larger than 30, so we should be okay, even with the skewness in the data.

Mechanics

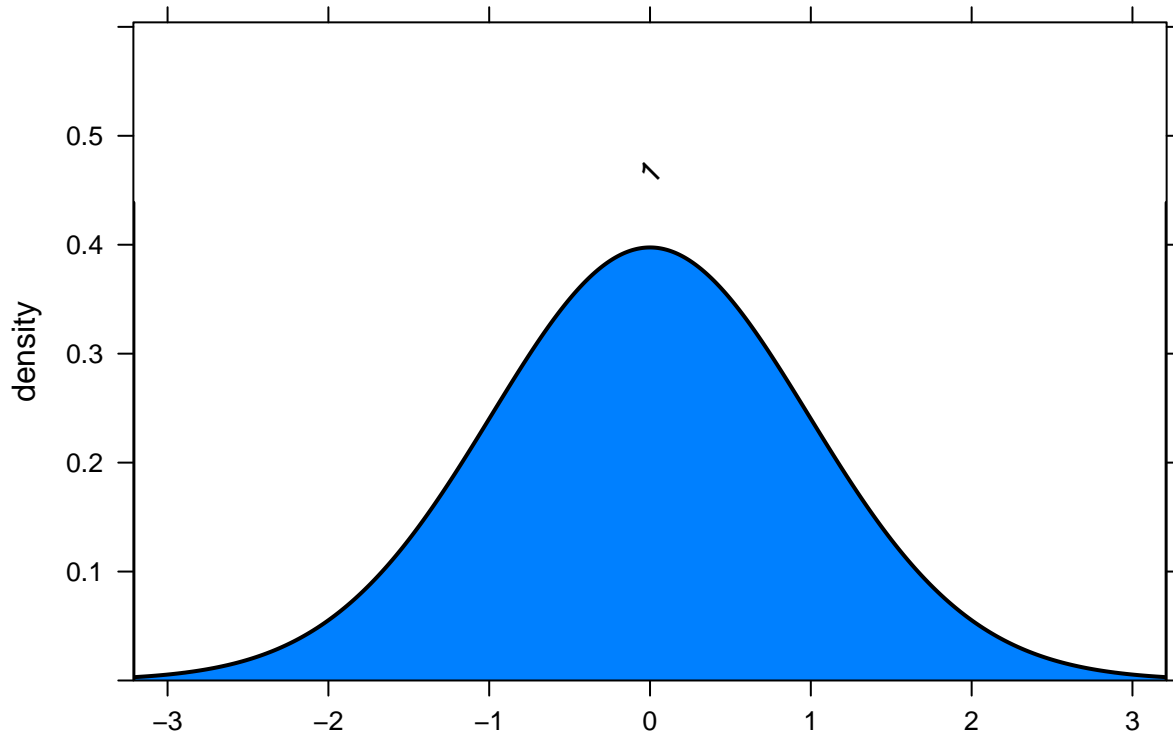**Compute and report the test statistic.**

```
total_test <- tidy(t.test(teacher2$total, mu = 63024))
t <- total_test$statistic
t
```

```
## [1] 5.886253
```

The t-score is 5.886253.

Plot the null distribution.

```
pdist("t", df = total_test$parameter, q = c(-t, t), invisible = TRUE)
```

Commentary: The `pdist` command is the same as always except now we use a `"t"` model instead of `"norm"`. We know there are 69 degrees of freedom, but we might as well use the value stored in the `t.test` output to make our code reusable. Finally, note that `q = c(-t, t)` will plot in both tails of the distribution since we're running a two-sided test. Of course, since the t-score is crazy huge, it doesn't actually appear in the resulting graph.

Calculate and report the P-value.

```
P <- total_test$p.value
P
```

```
## [1] 0.0000001290869
```

$P < 0.001$

Commentary: When the P-value is this small, it is traditional to report simply $P < 0.001$.

Conclusion

State the statistical conclusion.

We reject the null hypothesis.

State (but do not overstate) a contextually meaningful conclusion.

There is sufficient evidence that teachers in the St. Louis region do not earn $63,024 on average.

Identify the possibility of either a Type I or Type II error and state what making such an error means in the context of the hypotheses.

If we've made a Type I error, then the truth is that teachers in this region do make around $63,024 on average, but our sample was way off.

Confidence interval

Check the relevant conditions to ensure that model assumptions are met.

All the conditions have been checked already.

Calculate the confidence interval.

```
total_test$conf.low
```

```
## [1] 68256.2
```

```
total_test$conf.high
```

```
## [1] 73620.95
```

State (but do not overstate) a contextually meaningful interpretation.

We are 95% confident that the true mean salary for teachers in the St. Louis region is captured in the interval ($68256.2, $73620.95).

Commentary: As these are dollar amounts, it makes sense to round them to two decimal places. Even then, R is finicky and sometimes it will not respect your wishes.)

Inference using summary statistics

In the previous example, we had access to the actual data frame. In some situations, you are not given the data; rather, all you have are summary statistics about the data. This certainly happens with homework problems from a textbook, but it can happen in "real life" too. If you're reading a research article, you will rarely have access to the original data used in the analysis. All you can see is what the researchers report in their paper.

For a t-test, often you have nothing but a sample size n, the sample mean $\bar{y}$, and the sample standard deviation s.

Unlike the `binom.test` or `prop.test` commands that allow you to use either the raw data or summary statistics, the `t.test` command does not allow this. Instead, you have to calculate the t-score directly and use `pdist` to get the P-value.

For example, suppose you are told only that $n = 61$, $\bar{y} = 4.35$, and $s = 1.8$. Also suppose our hypotheses are

$H_0 : \mu = 5$

$H_A : \mu \neq 5$

The t-score is then

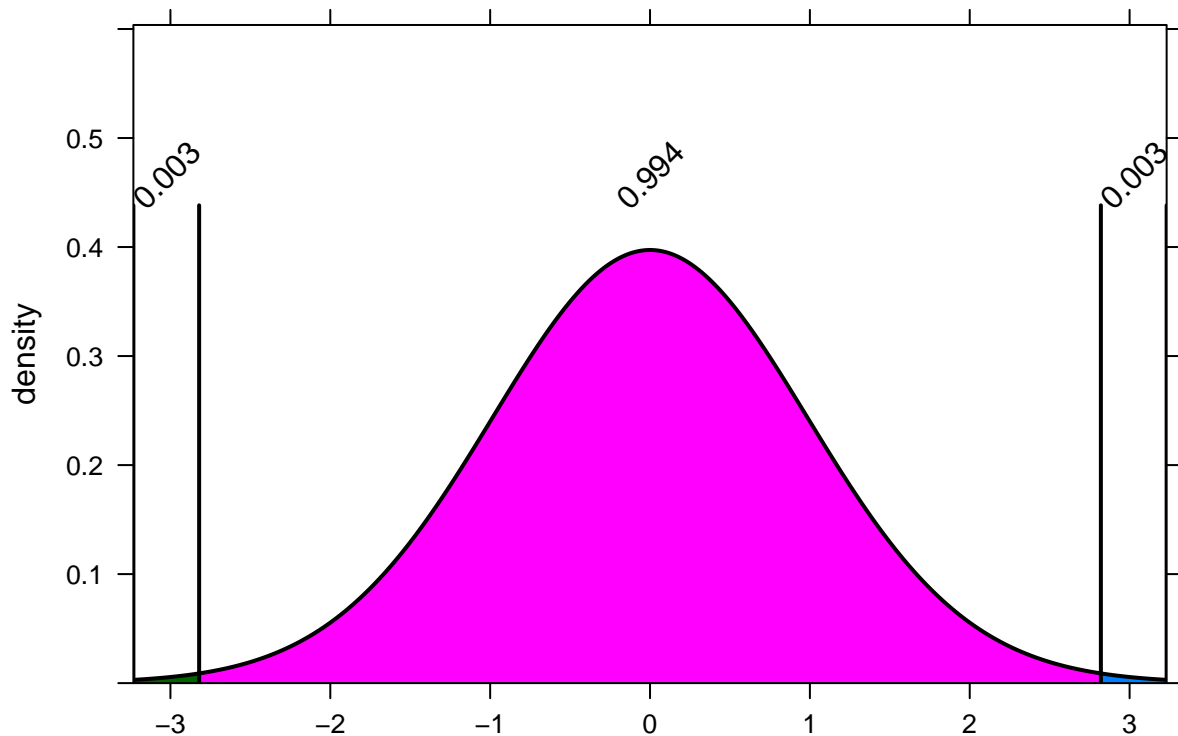$$t = \frac{(4.35 - 5)}{(1.8/\sqrt{61})}.$$

In R:

```
t <- (4.35 - 5)/(1.8/sqrt(61))
t
```

```
## [1] -2.820368
```

The t-score is -2.8203679.

The model is a t model with 60 degrees of freedom:

```
pdist("t", df = 60, q = c(-t, t), invisible = TRUE)
```


The P-value is also found using `pdist`. Since t is negative, we can use the area to its left and multiply by 2:

```
2 * pdist("t", df = 60, q = t, plot = FALSE)
```

```
## [1] 0.006492485
```

Since $P < 0.05$, we reject the null.

Be careful: with only summary statistics, we can't do any exploratory data analysis, so it may be impossible to check conditions. The only condition we have to check with the raw data is the nearly normal condition. In this example, though, since $n = 61$ we have a large enough sample size that we're not too worried.

Your turn

In the High School and Beyond survey (the `hsb2` data set from the `openintro` package), among the many scores that are recorded are standardized math scores. Suppose that these scores are normalized so that a score of 50 represents some kind of international average. (This is not really true. I had to make something up here to give you a baseline number with which to work.) The question is, then, are American students different from this international baseline?

The rubric outline is reproduced below. You may refer to the worked example above and modify it accordingly. Remember to strip out all the commentary. That is just exposition for your benefit in understanding the steps, but is not meant to form part of the formal inference process.

Another word of warning: the copy/paste process is not a substitute for your brain. You will often need to modify more than just the names of the data frames and variables to adapt the worked examples to your own work. Do not blindly copy and paste code without understanding what it does. And you should **never** copy and paste text. All the sentences and paragraphs you write are expressions of your own analysis. They must reflect your own understanding of the inferential process.

Exploratory data analysis

Use data documentation (help files, code books, Google, etc.), the str command, and other summary functions to understand the data.

<div align="center">ANSWER</div>

```r
# Add code here to understand the data.
```

Prepare the data for analysis. [Not always necessary.]

<div align="center">ANSWER</div>

```r
# Add code here to prepare the data for analysis.
```

Make tables or plots to explore the data visually.

<div align="center">ANSWER</div>

```r
# Add code here to make tables or plots.
```

Hypotheses

Identify the sample (or samples) and a reasonable population (or populations) of interest.

—— ANSWER ——

Please write up your answer here.

Express the null and alternative hypotheses as contextually meaningful full sentences.

—— ANSWER ——

H_0 : Null hypothesis goes here.

H_A : Alternative hypothesis goes here.

Express the null and alternative hypotheses in symbols (when possible).

—— ANSWER ——

$H_0 : math$

$H_A : math$

Model

Identify the sampling distribution model.

—— ANSWER ——

Please write up your answer here.

Check the relevant conditions to ensure that model assumptions are met.

—— ANSWER ——

Please write up your answer here. (Some conditions may require R code as well.)

Mechanics

Compute and report the test statistic.

—— ANSWER ——

```
# Add code here to compute the test statistic.
```

Please write up your answer here.

Plot the null distribution.

```
# Add code here to plot the null distribution.
```

Calculate and report the P-value.

```
# Add code here to calculate the P-value.
```

Please write up your answer here.

Conclusion

State the statistical conclusion.

Please write up your answer here.

State (but do not overstate) a contextually meaningful conclusion.

Please write up your answer here.

Identify the possibility of either a Type I or Type II error and state what making such an error means in the context of the hypotheses.

Please write up your answer here.

Confidence interval

Check the relevant conditions to ensure that model assumptions are met.

ANSWER

Please write up your answer here. (Some conditions may require R code as well.)

Calculate the confidence interval.

ANSWER

```
# Add code here to calculate the confidence interval.
```


State (but do not overstate) a contextually meaningful interpretation.

ANSWER

Please write up your answer here.

Exercises

After running inference above, answer the following questions:

1. Even though the result was *statistically* significant, do you think the result is *practically* significant? By this, I mean, are scores for American students so vastly different than 50? Do we have a lot of reason to brag about American scores based on your analysis?

ANSWER

Please write up your answer here.

2. What makes it possible for a small effect like this to be statistically significant even if it's not practically very different from 50? In other words, what has to be true of data to detect small but statistically significant effects?

ANSWER

Please write up your answer here.