

6.2: One-sample t-test — A new t-test

Now we are ready to talk about t-test. We will talk about three of them. We start with the one-sample t-test.

Commonly, the one-sample t-test is used to estimate the chances that your sample came from a particular population. Specifically, you might want to know whether the mean that you found from your sample, could have come from a particular population having a particular mean.

Straight away, the one-sample t-test becomes a little confusing (and I haven't even described it yet). Officially, it uses known parameters from the population, like the mean of the population and the standard deviation of the population. However, most times you don't know those parameters of the population! So, you have to estimate them from your sample. Remember from the chapters on descriptive statistics and sampling, our sample mean is an unbiased estimate of the population mean. And, our sample standard deviation (the one where we divide by $n-1$) is an unbiased estimate of the population standard deviation. When Gosset developed the t-test, he recognized that he could use these estimates from his samples, to make the t-test. Here is the formula for the one sample t-test, we first use words, and then become more specific:

Formulas for one-sample t-test

$$t = \frac{\text{measure of effect}}{\text{measure of error}}$$

$$t = \frac{\text{measure of effect}}{\text{measure of error}}$$

$$t = \frac{\text{Mean difference}}{\text{standard error}}$$

$$t = \frac{\bar{X} - \mu}{S_{\bar{X}}}$$

$$t = \frac{\text{Sample Mean} - \text{Population Mean}}{\text{Sample Standard Error}}$$

$$\text{Estimated Standard Error} = \text{Standard Error of Sample} = \frac{s}{\sqrt{N}}$$

Where, s is the sample standard deviation.

Some of you may have gone cross-eyed looking at all of this. Remember, we've seen it before when we divided our mean by the standard deviation in the first bit. The t-test is just a measure of a sample mean, divided by the standard error of the sample mean. That is it.

What does t represent?

t gives us a measure of confidence, just like our previous ratio for dividing the mean by a standard deviations. The only difference with t , is that we divide by the standard error of mean (remember, this is also a standard deviation, it is the standard deviation of the sampling distribution of the mean)

Note

What does the t in t-test stand for? Apparently nothing. Gosset originally labelled it z . And, Fisher later called it t , perhaps because t comes after s , which is often used for the sample standard deviation.

t is a property of the data that you collect. You compute it with a sample mean, and a sample standard error (there's one more thing in the one-sample formula, the population mean, which we get to in a moment). This is why we call t , a sample-statistic. It's a statistic we compute from the sample.

What kinds of numbers should we expect to find for these t 's? How could we figure that out?

Let's start small and work through some examples. Imagine your sample mean is 5. You want to know if it came from a population that also has a mean of 5. In this case, what would t be? It would be zero: we first subtract the sample mean from the population mean, $(5-5=0)$. Because the numerator is 0, t will be zero. So, $t = 0$, occurs, when there is no difference.

Let's say you take another sample, do you think the mean will be 5 every time, probably not. Let's say the mean is 6. So, what can t be here? It will be a positive number, because $(6-5 = +1)$. But, will t be $+1$? That depends on the standard error of the sample. If the standard error of the sample is 1, then t could be 1, because $(1/1 = 1)$.

If the sample standard error is smaller than 1, what happens to t ? It gets bigger right? For example, 1 divided by $(0.5 = 2)$. If the sample standard error was 0.5, t would be 2. And, what could we do with this information? Well, it be like a measure of confidence. As t gets bigger we could be more confident in the mean difference we are measuring.

Can t be smaller than 1? Sure, it can. If the sample standard error is big, say like 2, then t will be smaller than one (in our case), e.g., $(1/2 = .5)$. The direction of the difference between the sample mean and population mean, can also make the t become negative. What if our sample mean was 4. Well, then t will be negative, because the mean difference in the numerator will be negative, and the number in the bottom (denominator) will always be positive (remember why, it's the standard error, computed from the sample standard deviation, which is always positive because of the squaring that we did.).

So, that is some intuitions about what the kinds of values t can take. t can be positive or negative, and big or small.

Let's do one more thing to build our intuitions about what t can look like. How about we sample some numbers and then measure the sample mean and the standard error of the mean, and then plot those two things against each other. This will show us how a sample mean typically varies with respect to the standard error of the mean.

In the following figure, I pulled 1,000 samples of $N=10$ from a normal distribution (mean = 0, sd = 1). Each time I measured the mean and standard error of the sample. That gave two descriptive statistics for each sample, letting us plot each sample as dot in a scatterplot

```
library(ggplot2)
sample_mean<-length(1000)
sample_se<-length(1000)
for(i in 1:1000){
  s<-rnorm(10,0,1)
  sample_mean[i]<-mean(s)
  sample_se[i]<-sd(s)/sqrt(length(s))
}
plot(sample_mean,sample_se)
```

run

restart

restart & run all

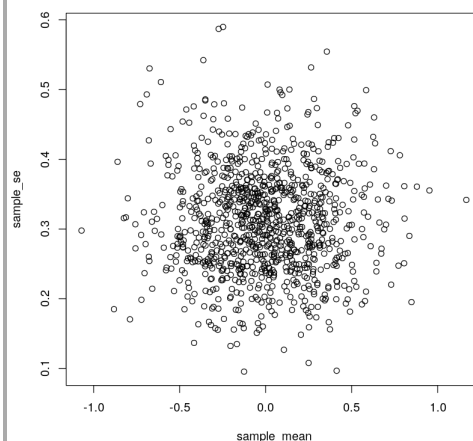


Figure 1: A scatterplot with sample mean on the x-axis, and standard error of the mean on the y-axis.

What we get is a cloud of dots. You might notice the cloud has a circular quality. There's more dots in the middle, and fewer dots as they radiate out from the middle. The dot cloud shows us the general range of the sample mean, for example most of the dots are in between -1 and 1. Similarly, the range for the sample standard error is roughly between .2 and .5. Remember, each dot represents one sample.

We can look at the same data a different way. For example, rather than using a scatterplot, we can divide the mean for each dot, by the standard error for each dot. Below is a histogram showing what this looks like:

```
sample_mean<-length(1000)
sample_se<-length(1000)
for(i in 1:1000){
  s<-rnorm(10,0,1)
  sample_mean[i]<-mean(s)
  sample_se[i]<-sd(s)/sqrt(length(s))
}
hist(sample_mean/sample_se, breaks=30)
```

run restart restart & run all

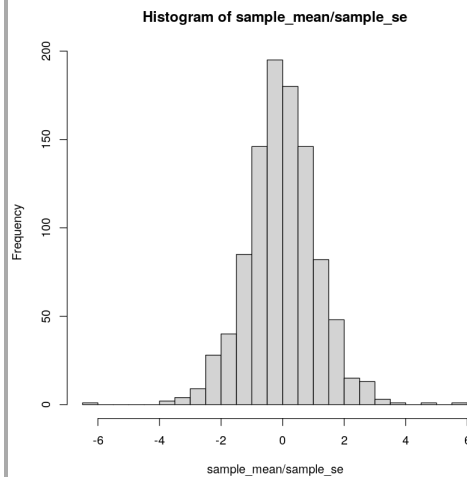


Figure \(\PageIndex{2}\): A histogram of the sample means divided by the sample standard errors, this is a t-distribution.

Interesting, we can see the histogram is shaped like a normal curve. It is centered on 0, which is the most common value. As values become more extreme, they become less common. If you remember, our formula for t , was the mean divided by the standard error of the mean. That's what we did here. This histogram is showing you a t -distribution.

Calculating t from data

Let's briefly calculate a t-value from a small sample. Let's say we had 10 students do a true/false quiz with 5 questions on it. There's a 50% chance of getting each answer correct.

Every student completes the 5 questions, we grade them, and then we find their performance (mean percent correct). What we want to know is whether the students were guessing. If they were all guessing, then the sample mean should be about 50%, it shouldn't be different from chance, which is 50%. Let's look at the table:

```
suppressPackageStartupMessages(library(dplyr))
students <- 1:10
scores <- c(50,70,60,40,80,30,90,60,70,60)
mean_scores <- mean(scores)
Difference_from_Mean <- scores-mean_scores
Squared_Deviations <- Difference_from_Mean^2
the_df<-data.frame(students,
                   scores,
                   mean=rep(mean_scores,10),
                   Difference_from_Mean,
                   Squared_Deviations)
the_df <- the_df %>%
  rbind(c("Sums",colSums(the_df[1:10,2:5]))) %>%
```

```

rbind(c("Means", colMeans(the_df[1:10,2:5]))) %>%
rbind(c(" ", " ", " ", " ", "sd ", round(sd(the_df[1:10,2]), digits=2))) %>%
rbind(c(" ", " ", " ", " ", "SEM ", round(sd(the_df[1:10,2])/sqrt(10), digits=2))) %>%
rbind(c(" ", " ", " ", " ", "t", (61-50)/round(sd(the_df[1:10,2])/sqrt(10), digits=2)))
knitr::kable(the_df)

```

run

restart

restart & run all

students	scores	mean	Difference_from_Mean	Squared_Deviations
1	50	61	-11	121
2	70	61	9	81
3	60	61	-1	1
4	40	61	-21	441
5	80	61	19	361
6	30	61	-31	961
7	90	61	29	841
8	60	61	-1	1
9	70	61	9	81
10	60	61	-1	1
Sums	610	610	0	2890
Means	61	61	0	289
			sd	17.92
			SEM	5.67
			t	1.94003527336861

You can see the `scores` column has all of the test scores for each of the 10 students. We did the things we need to do to compute the standard deviation.

Remember the sample standard deviation is the square root of the sample variance, or:

$$\sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{N-1}}$$

$$\sqrt{\frac{2890}{10-1}} = 17.92$$

The standard error of the mean, is the standard deviation divided by the square root of N

$$\sqrt{\frac{s}{N}} = \frac{17.92}{10} = 5.67$$

t is the difference between our sample mean (61), and our population mean (50, assuming chance), divided by the standard error of the mean.

$$t = \frac{\bar{X} - \mu}{S_{\bar{X}}} = \frac{\bar{X} - \mu}{SEM} = \frac{61-50}{5.67} = 1.94$$

And, that is you how calculate t , by hand. It's a pain. I was annoyed doing it this way. In the lab, you learn how to calculate t using software, so it will just spit out t . For example in R, all you have to do is this:

```

scores <- c(50,70,60,40,80,30,90,60,70,60)
t.test(scores, mu=50)

```

run

restart

restart & run all

One Sample t-test

data: scores

t = 1.9412, df = 9, p-value = 0.08415

alternative hypothesis: true mean is not equal to 50

95 percent confidence interval:

```
48.18111 73.81889
sample estimates:
mean of x
61
```

How does t behave?

If t is just a number that we can compute from our sample (it is), what can we do with it? How can we use t for statistical inference?

Remember back to the chapter on sampling and distributions, that's where we discussed the sampling distribution of the sample mean. Remember, we made a lot of samples, then computed the mean for each sample, then we plotted a histogram of the sample means. Later, in that same section, we mentioned that we could generate sampling distributions for any statistic. For each sample, we could compute the mean, the standard deviation, the standard error, and now even t , if we wanted to. We could generate 10,000 samples, and draw four histograms, one for each sampling distribution for each statistic.

This is exactly what I did, and the results are shown in the four figures below. I used a sample size of 20, and drew random observations for each sample from a normal distribution, with mean = 0, and standard deviation = 1. Let's look at the sampling distributions for each of the statistics. t was computed assuming with the population mean assumed to be 0.

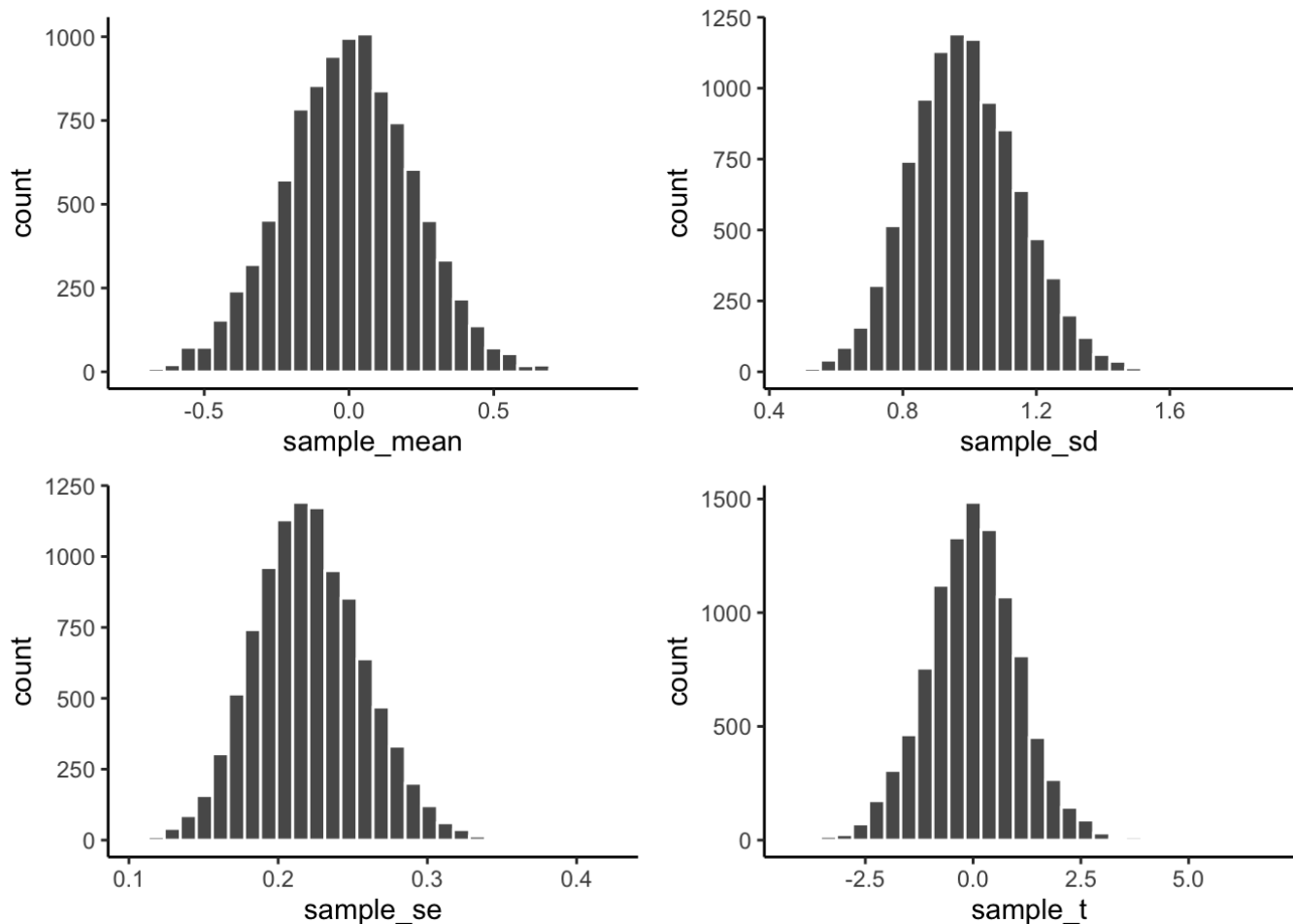


Figure 6.2.5: Sampling distributions for the mean, standard deviation, standard error of the mean, and t .

We see four sampling distributions. This is how statistical summaries of these summaries behave. We have used the word chance windows before. These are four chance windows, measuring different aspects of the sample. In this case, all of the samples came from the same normal distribution. Because of sampling error, each sample is not identical. The means are not identical, the standard deviations are not identical, sample standard error of the means are not identical, and the t 's of the samples are not identical. They all have some variation, as shown by the histograms. This is how samples of size 20 behave.

We can see straight away, that in this case, we are unlikely to get a sample mean of 2. That's way outside the window. The range for the sampling distribution of the mean is around $-.5$ to $+.5$, and is centered on 0 (the population mean, would you believe!). We are unlikely to get sample standard deviations of between $.6$ and 1.5 , that is a different range, specific to the sample standard deviation.

Same thing with the sample standard error of the mean, the range here is even smaller, mostly between $.1$, and $.3$. You would rarely find a sample with a standard error of the mean greater than $.3$. Virtually never would you find one of say 1 (for this situation).

Now, look at t . Its range is basically between -3 and $+3$ here. 3s barely happen at all. You pretty much never see a 5 or -5 in this situation.

All of these sampling windows are chance windows, and they can all be used in the same way as we have used similar sampling distributions before (e.g., Crump Test, and Randomization Test) for statistical inference. For all of them we would follow the same process:

1. Generate these distributions
2. Look at your sample statistics for the data you have (mean, SD, SEM, and t)
3. Find the likelihood of obtaining that value or greater
4. Obtain that probability
5. See if you think your sample statistics were probable or improbable.

We'll formalize this in a second. I just want you to know that what you will be doing is something that you have already done before. For example, in the Crump test and the Randomization test we focused on the distribution of mean differences. We could do that again here, but instead, we will focus on the distribution of t values. We then apply the same kinds of decision rules to the t distribution, as we did for the other distributions. Below you will see a graph you have already seen, except this time it is a distribution of t s, not mean differences:

Remember, if we obtained a single t from one sample we collected, we could consult this chance window below to find out the t we obtained from the sample was likely or unlikely to occur by chance.

```
library(ggplot2)
options(warn=-1)
all_df<-data.frame()
for(i in 1:10000){
  sample<-rnorm(20,0,1)
  sample_mean<-mean(sample)
  sample_sd<-sd(sample)
  sample_se<-sd(sample)/sqrt(length(sample))
  sample_t<-as.numeric(t.test(sample, mu=0)$statistic)
  t_df<-data.frame(i,sample_mean,sample_sd,sample_se,sample_t)
  all_df<-rbind(all_df,t_df)
}
sample_t<-all_df$sample_t
ggplot(all_df,aes(x=sample_t))+
  annotate("rect", xmin=min(sample_t), xmax=max(sample_t), ymin=0,
    ymax=Inf, alpha=0.5, fill="red") +
  annotate("rect", xmin=min(sample_t), xmax=-1.94, ymin=0,
    ymax=Inf, alpha=0.7, fill="light grey") +
  annotate("rect", xmin=1.94, xmax=max(sample_t), ymin=0,
    ymax=Inf, alpha=0.7, fill="light grey") +
  geom_rect(aes(xmin=-Inf, xmax=min(sample_t), ymin=0,
    ymax=Inf), alpha=.5, fill="lightgreen")+
  geom_rect(aes(xmin=max(sample_t), xmax=Inf, ymin=0,
    ymax=Inf), alpha=.5, fill="lightgreen")+
  geom_histogram(bins=50, color="white")+
  theme_classic()+
  geom_vline(xintercept = min(sample_t))+
  geom_vline(xintercept = max(sample_t))+
  geom_vline(xintercept = -1.94)+
  geom_vline(xintercept = 1.94)+
  ggtitle("Histogram of mean sample_ts between two samples (n=20) \n
    both drawn from the same normal distribution (u=0, sd=1)")+
  xlim(-8,8)+
  geom_label(data = data.frame(x = 0, y = 250, label = "CHANCE"),
    aes(x = x, y = y, label = label))+
  geom_label(data = data.frame(x = -7, y = 250, label = "NOT \n CHANCE"),
    aes(x = x, y = y, label = label))+
  geom_label(data = data.frame(x = 7, y = 250, label = "NOT \n CHANCE"),
    aes(x = x, y = y, label = label))+
  # geom_label(data = data.frame(x = min(sample_t), y = 600,
  #   label = paste0("min \n",round(min(sample_t)))),
  #   aes(x = x, y = y, label = label))+
  # geom_label(data = data.frame(x = max(sample_t), y = 600,
  #   label = paste0("max \n",round(max(sample_t)))),
  #   aes(x = x, y = y, label = label))+
  geom_label(data = data.frame(x = -4, y = 250,
    label = "?"),
    aes(x = x, y = y, label = label))+
  geom_label(data = data.frame(x = 4, y = 250,
    label = "?"),
    aes(x = x, y = y, label = label))+
  xlab("mean sample_t")
```

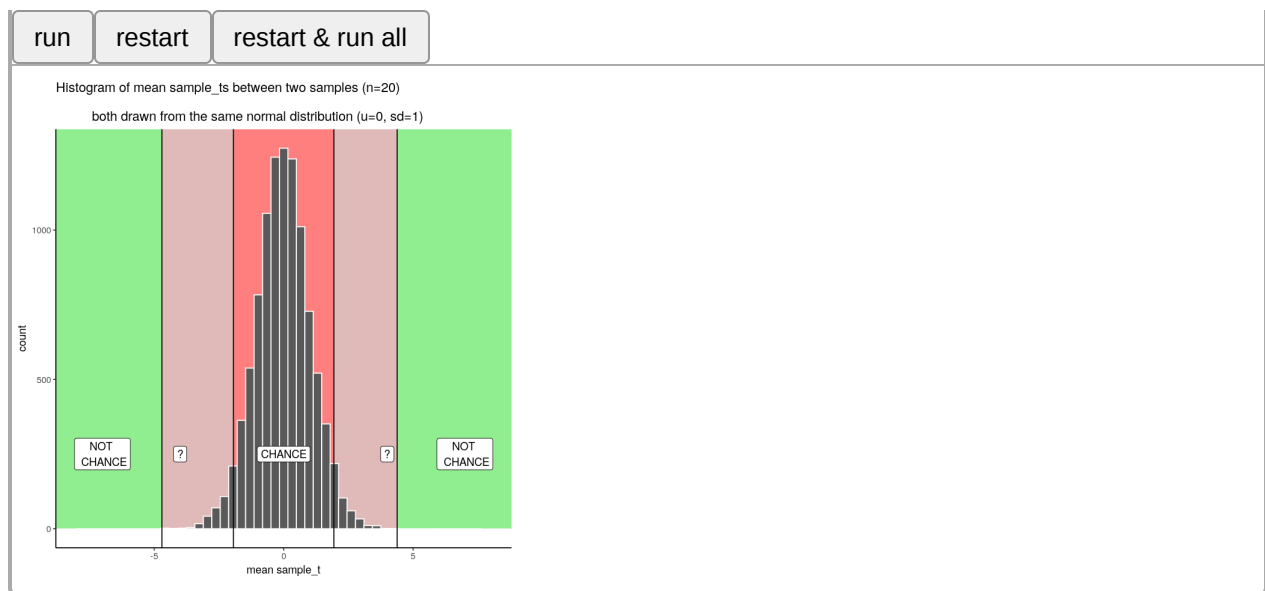


Figure 6.2.8: Applying decision criteria to the t-distribution.

Making a decision

From our early example involving the TRUE/FALSE quizzes, we are now ready to make some kind of decision about what happened there. We found a mean difference of

```
scores <- c(50,70,60,40,80,30,90,60,70,60)
mean(scores) - 50
```

run
restart
restart & run all

11

. We found a $t =$

```
scores <- c(50,70,60,40,80,30,90,60,70,60)
t.test(scores, mu=50)$statistic
```

run
restart
restart & run all

t: 1.94117647058824

. The probability of this t or larger occurring is $p =$

```
scores <- c(50,70,60,40,80,30,90,60,70,60)
t.test(scores, mu=50)$p.value
```

run
restart
restart & run all

0.0841503080536893

. We were testing the idea that our sample mean of

```
scores <- c(50,70,60,40,80,30,90,60,70,60)
mean(scores)
```

run
restart
restart & run all

could have come from a normal distribution with mean = 50. The t test tells us that the t for our sample, or a larger one, would happen with $p = 0.0841503$. In other words, chance can do it a kind of small amount of time, but not often. In English, this means that all of the students could have been guessing, but it wasn't that likely that were just guessing.

We're guessing that you are still a little bit confused about t values, and what we are doing here. We are going to skip ahead to the next t -test, called a paired samples t -test. We will also fill in some more things about t -tests that are more obvious when discussing paired samples t -test. In fact, spoiler alert, we will find out that a paired samples t -test is actually a one-sample t -test in disguise (WHAT!), yes it is. If the one-sample t -test didn't make sense to you, read the next section.

This page titled 6.2: One-sample t -test — A new t -test is shared under a CC BY-SA 4.0 license and was authored, remixed, and/or curated by Matthew J. C. Crump via source content that was edited to the style and standards of the LibreTexts platform.