Introducing Hypothesis Testing

Matt Lawrence

October 27, 2021

Setting Up

Load the `gss_week7.csv` file on Canvas as a data frame called `gss_week7` and load the usual packages:

Comparing The Z and T Distributions

When the population standard deviation is unknown (almost always), the uncertainty of the standard error estimate is addressed by using a new distribution: the t distribution. This distribution also has a bell shape, but its tails are thicker than the normal model's.

When the sample size is large, z and t are the same. When in doubt, use t. It will always work since it adjusts for the sample size.

You can find t-values and associated probabilities using functions similar to the `norm()` functions. The difference is that with the `t()` functions you also give the degrees of freedom.

This is what we did before to get the z-value associated with the 95% confidence interval:

```
qnorm(.975)
```

```
## [1] 1.959964
```

The equivalent code with the t-distribution and a sample size of 1001 (so df = 1000):

```
qt(.975, df = 1000)
```

```
## [1] 1.962339
```

With smaller samples, using the t-distribution builds in extra room since our estimates are less reliable:

```
qt(.975, df = 100)
```

```
## [1] 1.983972
```

To find the probabilities associated with t-values, use `pt()` which also requires degrees of freedom:

```
pt(1.962339, df = 1000)
```

```
## [1] 0.975
```

And 1 - pt() also works the same way as 1 - pnorm():

```
1 - pt(1.983972, df = 100)
```

```
## [1] 0.02499997
```

Significance Testing

Let's test if the mean value of `childs` in 2018 differed from 2.2 (the 1984 mean, which we get from previous research).

What was the mean for `childs` in 2018?

REPLACE THIS LINE WITH YOUR CODE

```
mean(gss_week7$childs[gss_week7$year==2018])
```

```
## [1] 1.998255
```

The test statistic is calculated as: $t = \frac{y - y_{H0}}{SD/\sqrt{n}}$

Let's get the standard deviation and the square root of n

```
sd <- sd(gss_week7$childs[gss_week7$year==2018])
sqrt_n <- sqrt(length(gss_week7$childs[gss_week7$year==2018]))
```

For our test, the test statistic is:

```
(1.998255 - 2.2) / (sd/sqrt_n)
```

```
## [1] -3.389708
```

What is the t-value for the critical region for our sample size and an alpha level of .05?

REPLACE THIS LINE WITH YOUR CODE

```
qt(.025, df = 572)
```

```
## [1] -1.96412
```

Our test statistic is more extreme than that so we can reject the null hypothesis.

We could also use the p-value in the hypothesis test. If our p-value is less than the alpha level for our confidence range, we can reject the null hypothesis.

What is the p-value for our test statistic?

```
pt(-3.389708, df = 572)
```

```
## [1] 0.0003739406
```

That's for one side of the curve, so we need to double it for the test:

```
pt(-3.389708, df = 572) * 2
```

```
## [1] 0.0007478812
```

We can reject the null hypothesis because the p-value is less than .05 (the cutoff for a 95% test).

Shortcut!

R has a built-in function called `t.test()` that will calculate all of these test statistics. With one mean, we have to fill in the value of mu (μ) which is the null hypothesis value. If you leave it out, the default is zero.

```
t.test(gss_week7$childs[gss_week7$year==2018], mu = 2.2)
```

```
##
##  One Sample t-test
##
## data:  gss_week7$childs[gss_week7$year == 2018]
## t = -3.3897, df = 572, p-value = 0.0007479
## alternative hypothesis: true mean is not equal to 2.2
## 95 percent confidence interval:
##  1.881356 2.115153
## sample estimates:
## mean of x
##  1.998255
```

For another example, let's look at 2018's mean ideal number of children (variable = `chldidel`). Using `t.test()`, can we reject the null hypothesis that the 2018 mean is the same as 2.55 at the .05 alpha level?

REPLACE THIS LINE WITH YOUR CODE

```
t.test(gss_week7$chldidel[gss_week7$year==2018], mu = 2.55)
```

```
##
##  One Sample t-test
##
## data:  gss_week7$chldidel[gss_week7$year == 2018]
## t = 1.5856, df = 572, p-value = 0.1134
## alternative hypothesis: true mean is not equal to 2.55
## 95 percent confidence interval:
##  2.535483 2.686158
## sample estimates:
## mean of x
##   2.61082
```

We can change the default level of .05 to .01 using the conf.level option (which requires the confidence level, so .99 for the .01 alpha level). Can we reject the null hypothesis that the 2016 mean is the same as 2.5 at the .01 alpha level?

```
t.test(gss_week7$chldidel[gss_week7$year==2016], mu = 2.5,
       conf.level = .99)
```

```
##
##  One Sample t-test
##
## data:  gss_week7$chldidel[gss_week7$year == 2016]
## t = 1.5082, df = 720, p-value = 0.1319
## alternative hypothesis: true mean is not equal to 2.5
## 99 percent confidence interval:
##  2.466898 2.626029
## sample estimates:
## mean of x
##  2.546463
```

Inference For Single Proportions

We have previously seen that proportions and means have different standard errors. As a result, we use different hypothesis tests for them as well. We'll skip to the shortcut this time.

Example: Does the proportion of respondents whose number of children is equal to their ideal number of children differ from .33?

First let's create a binary variable identifying respondents whose number of children is equal to their ideal number of children with a 1 and everyone else with a 0.

```
gss_week7 <- gss_week7 |>
  mutate(has_ideal = ifelse(childs == chldidel, 1, 0))
```

For the test, we will need the frequency with a 1 and the total in the sample.

```
addmargins(table(gss_week7$has_ideal))
```

```
##
##    0    1  Sum
## 2034  889 2923
```

Enter those two values in `prop.test()` along with the null hypothesis value you want to test. The function calculates the proportion and compares it to the null hypothesis value.

```
prop.test(889, 2923, p = .33)
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  889 out of 2923, null probability 0.33
## X-squared = 8.7246, df = 1, p-value = 0.003139
```

```
## alternative hypothesis: true p is not equal to 0.33
## 95 percent confidence interval:
##  0.2875599 0.3212387
## sample estimates:
##         p
## 0.3041396
```

The prop.test function doesn't use the t-distribution so the test statistic is different (and cannot be compared to 1.96). But the p-value can still be compared to the alpha level and you can still compare the null hypothesis value to your confidence interval.

In the example above, can we reject the null hypothesis that the proportion differs from .33?

Another example: Does the proportion of respondents with less children than their ideal number differ from .5 at the 99% confidence level?

REPLACE THIS LINE WITH YOUR CODE

```
gss_week7 <- gss_week7 |>
  mutate(less_ideal = ifelse(childs < chldidel, 1, 0))
```

```
addmargins(table(gss_week7$less_ideal))
```

```
##
##    0    1  Sum
## 1437 1486 2923
```

```
prop.test(1486, 2923, p = .53, conf.level = .99)
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  1486 out of 2923, null probability 0.53
## X-squared = 5.3975, df = 1, p-value = 0.02017
## alternative hypothesis: true p is not equal to 0.53
## 99 percent confidence interval:
##  0.4844006 0.5323247
## sample estimates:
##         p
## 0.5083818
```