More on t - Part 3

Chriss Jordan Oboa

2022-04-22

Packages and data

We load our usual packages: `tidyverse` as a general tool, and `infer` to help with random sampling.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(infer)
```

We also load the same population from our earlier activities. This population is close to what Gosset used in his work.

```
population <- read_csv("data/population.csv")
```

```
## Rows: 3000 Columns: 1
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (1): height
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Select a single random sample of size 4 from the population:

```
samp1 <- population %>%
  slice_sample(n = 4)
```

Click on samp1 in the environment pane and see your sample. (Is it the same as your neighbor's sample?)

Summarize the sample statistics: the mean height for the people in your sample, the standard deviation for your sample, and the estimated standard error for mean heights.

```
samp1 %>%
  summarize(mean_samp1 = mean(height),  sd_samp1 = sd (height), est_se= sd(height)/sqrt(4))
```

```
## # A tibble: 1 x 3
##   mean_samp1 sd_samp1 est_se
```

```
##        <dbl>    <dbl>  <dbl>
## 1       64.8     3.22   1.61
```

Remember, we compute a t-confidence interval using the formula: statistic +- t*(SE)

where we approximate SE using s/sqrt(n) and t* is based on the sample size and the desired confidence level. Remind yourself of the appropriate value of t* to use by using `qt(p = 0.975, df = 3)`

```
qt(p = 0.975, df = 3)
```

```
## [1] 3.182446
```

Compute, by hand, the 95% confidence interval for *your* sample.

Below is the command for how to have R automatically compute the 95% t-confidence interval for us. Run the command and compare the confidence interval you obtain with the one you found above. They should be roughly the same.

```
t.test(height~1, data = samp1, conf.level = 0.95)
```

```
##
##  One Sample t-test
##
## data:  height
## t = 40.309, df = 3, p-value = 3.36e-05
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  59.70085 69.93595
## sample estimates:
## mean of x
##   64.8184
```