IS5 in R: Testing Hypotheses (Chapter 15)

Nicholas Horton (nhorton@amherst.edu)

2025-01-20

Table of contents

Introduction and background

This document is intended to help describe how to undertake analyses introduced as examples in the Fifth Edition of *Intro Stats* (2018) by De Veaux, Velleman, and Bock. This file as well as the associated Quarto reproducible analysis source file used to create it can be found at http://nhorton.people.amherst.edu/is5.

This work leverages initiatives undertaken by Project MOSAIC (http://www.mosaic-web.org), an NSF-funded effort to improve the teaching of statistics, calculus, science and computing in the undergraduate curriculum. In particular, we utilize the `mosaic` package, which was written to simplify the use of R for introductory statistics courses. A short summary of the R needed to teach introductory statistics can be found in the mosaic package vignettes (https://cran.r-project.org/web/packages/mosaic). A paper describing the mosaic approach was published in the *R Journal*: https://journal.r-project.org/archive/2017/RJ-2017-024.

We begin by loading packages that will be required for our analyses.

```
library(mosaic)
library(tidyverse)
```

Chapter 15: Testing Hypotheses

Section 15.1: Hypotheses

Section 15.2: P-Values

Section 15.3: The Reasoning of Hypothesis Testing

Example 15.5: Finding a P-Value

It is straightforward to find p-values using summary statistics.

```
n <- 90
x <- 61
p <- .8
phat <- x / n
sdphat <- ((p * (1 - p)) / n)^.5
z <- (phat - p) / sdphat
pnorm(z)
```

```
[1] 0.00187324
```

```
# Or, without calculating the z-score:
pnorm(q = phat, mean = p, sd = sdphat)
```

```
[1] 0.00187324
```

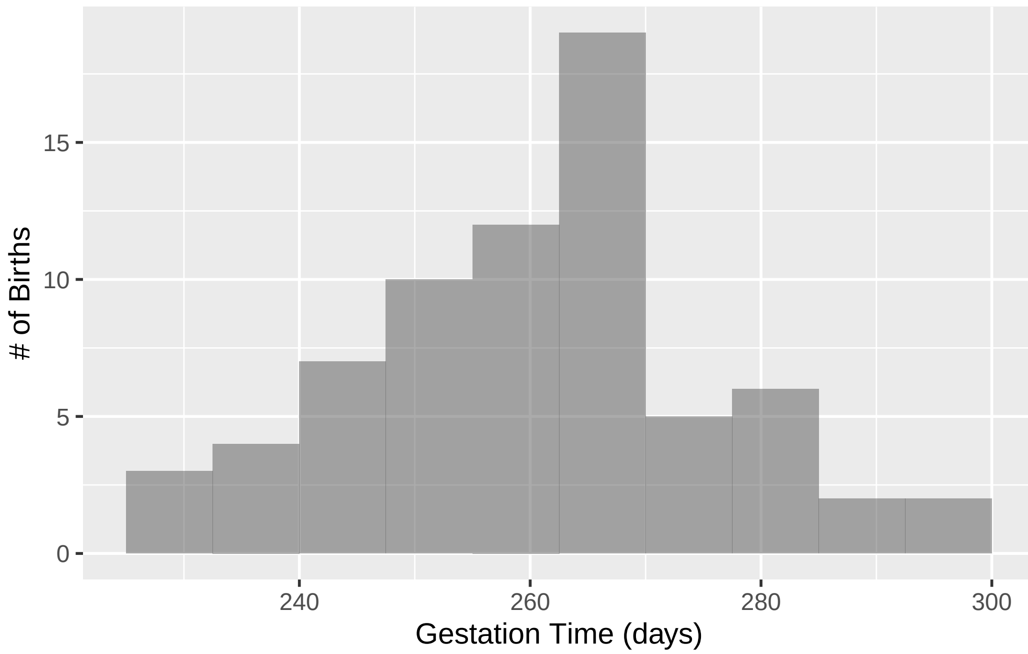
Section 15.4: A Hypothesis Test for the Mean

We begin by reading the data.

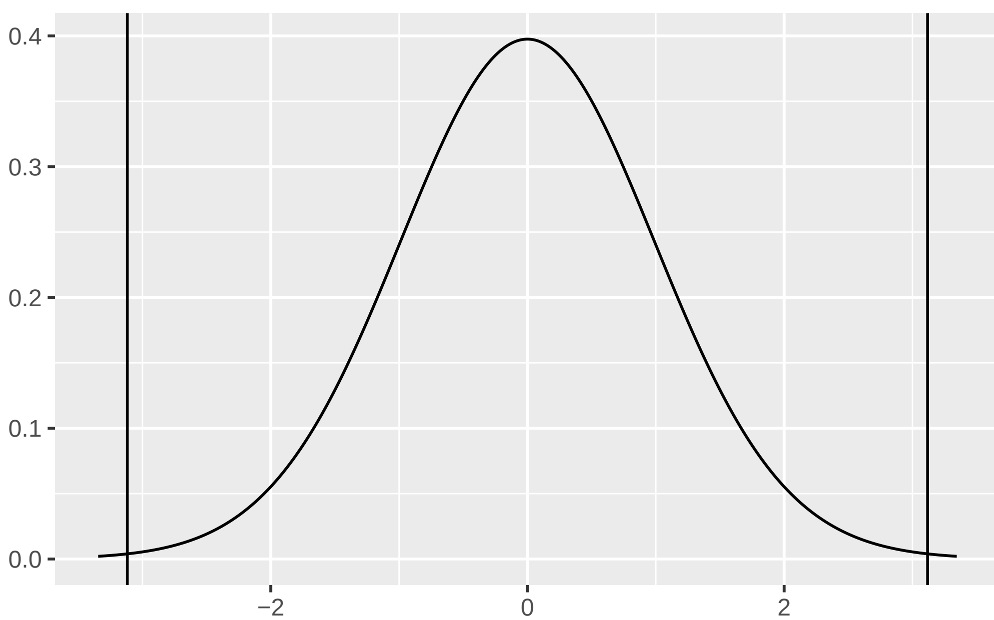
```
GestationTime <- read_csv("http://nhorton.people.amherst.edu/is5/data/Nashville.csv")
```

By default, `read_csv()` prints the variable names. These messages can be suppressed using the `message: false` code chunk option to save space and improve readability.

```
# 2. Model (page 482)
gf_histogram(~ Gestation, data = GestationTime, binwidth = 7.5, center = 3.75) |>
  gf_labs(x = "Gestation Time (days)", y = "# of Births")
```




```
# 3. Mechanics
gf_dist(dist = "t", df = 69) |>
  gf_vline(xintercept = -3.118) |>
  gf_vline(xintercept = 3.118) |>
  gf_labs(x = "", y = "") +
  xlim(-3.347, 3.347)
```



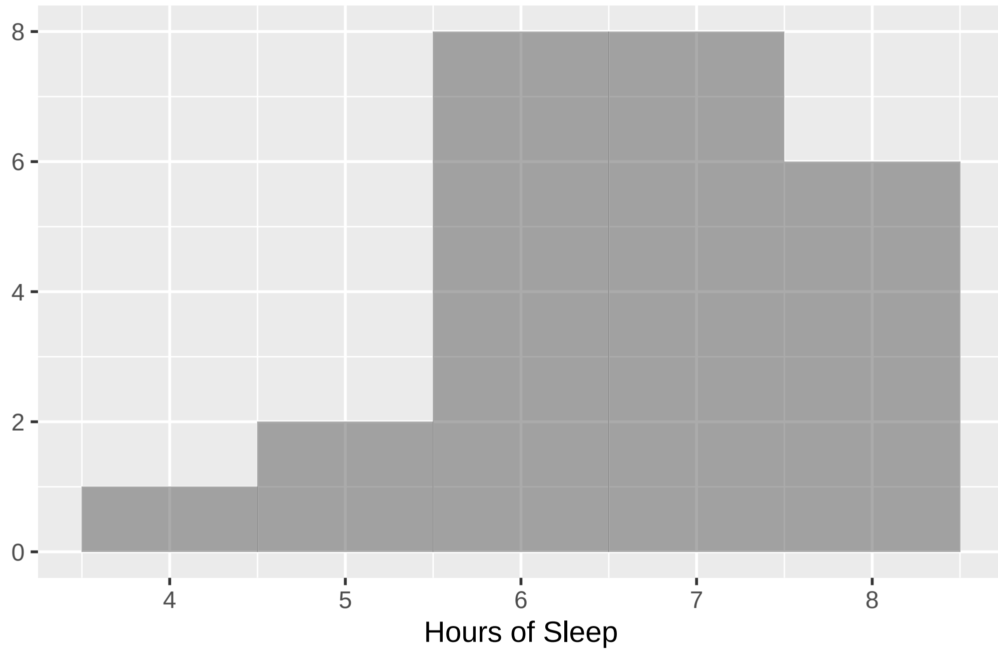
Step-By-Step Example: A One-Sample *t*-Test for the Mean

We begin by reading in the data.

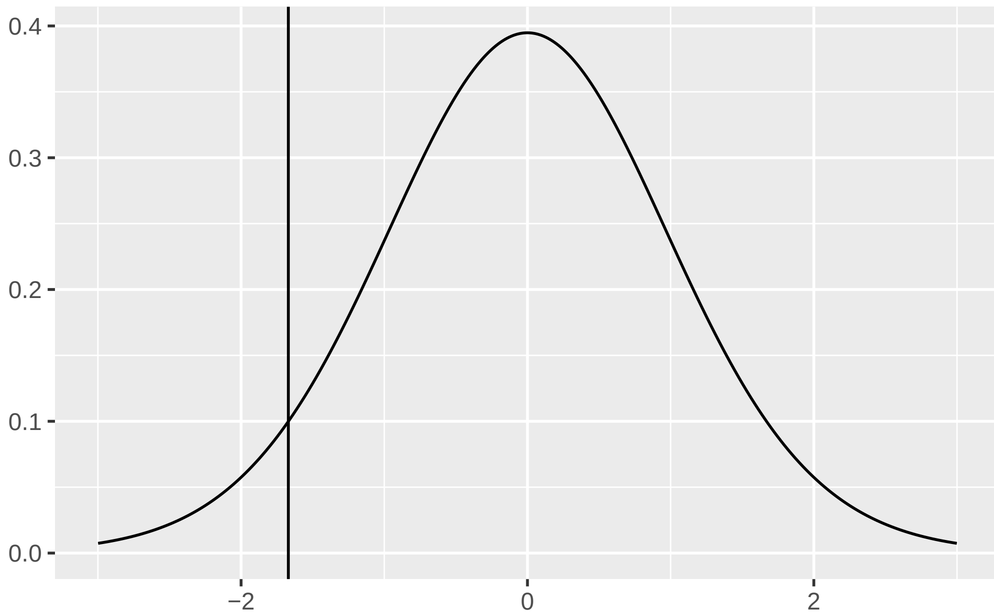
```
# page 485
Sleep <- read_csv("http://nhorton.people.amherst.edu/is5/data/Sleep.csv")
# Plan
df_stats(~ Sleep, data = Sleep)
```

```
  response min Q1 median Q3 max mean       sd  n missing
1    Sleep   4  6      7  7   8 6.64 1.075484 25       0
```

```
gf_histogram(~ Sleep, data = Sleep, binwidth = 1) |>
  gf_labs(x = "Hours of Sleep", y = "")
```




```
gf_dist(dist = "t", df = 24) |>
  gf_vline(xintercept = -1.67) |>
  gf_labs(x = "", y = "") +
  xlim(-3, 3)
```




```
# Mechanics
n <- 25
mean <- 7.0
df <- 24
y <- 6.64
s <- 1.075
sey <- s / (n^.5)
t <- (y - mean) / sey # t-statistic
pt(q = t, df = df) # p-value
```

```
[1] 0.05351625
```

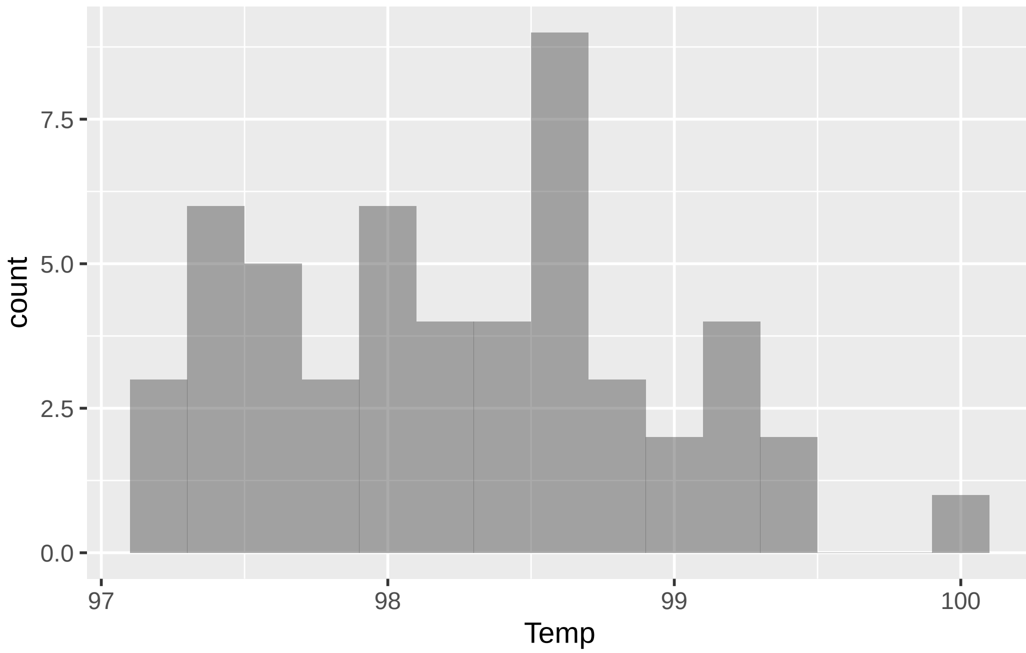
Section 15.5: Intervals and Tests

It is straightforward to calculate confidence intervals and carry out hypothesis tests.

```
# page 487
Temperatures <-
  read_csv("http://nhorton.people.amherst.edu/is5/data/Normal_temperature.csv")
df_stats(~ Temp, data = Temperatures)
```

```
  response  min     Q1 median   Q3 max     mean        sd  n missing
1     Temp 97.2 97.675   98.2 98.7 100 98.28462 0.6823789 52       0
```

```r
gf_histogram(~ Temp, data = Temperatures, binwidth = .2)
```




```r
# Confidence interval
y <- mean(~ Temp, data = Temperatures)
y
```

```
[1] 98.28462
```

```r
s <- sd(~ Temp, data = Temperatures)
s
```

```
[1] 0.6823789
```

```r
n <- nrow(Temperatures)
n
```

```
[1] 52
```

```r
tstats <- qt(df = n - 1, p = c(.005, .995))
tstats
```

```
[1] -2.675722  2.675722
```

```r
y + (tstats * (s / (n^.5)))
```

```
[1] 98.03141 98.53782
```

```r
# Hypothesis test
mu <- 98.6
t <- (y - mu) / (s / (n^.5))
t
```

```
[1] -3.332856
```

```r
2 * pt(q = t, df = n - 1) # two sided test
```

```
[1] 0.001605849
```

Random Matters: Bootstrap Hypothesis Tests and Intervals

The boostrap is a flexible alternative approach to inference.

```r
numsamp <- 10000

# What does do() do?
mean(~ Temp, data = resample(Temperatures)) # Mean of one random resample
```

```
[1] 98.14231
```

```r
mean(~ Temp, data = resample(Temperatures)) # Mean of another random resample
```

```
[1] 98.26923
```

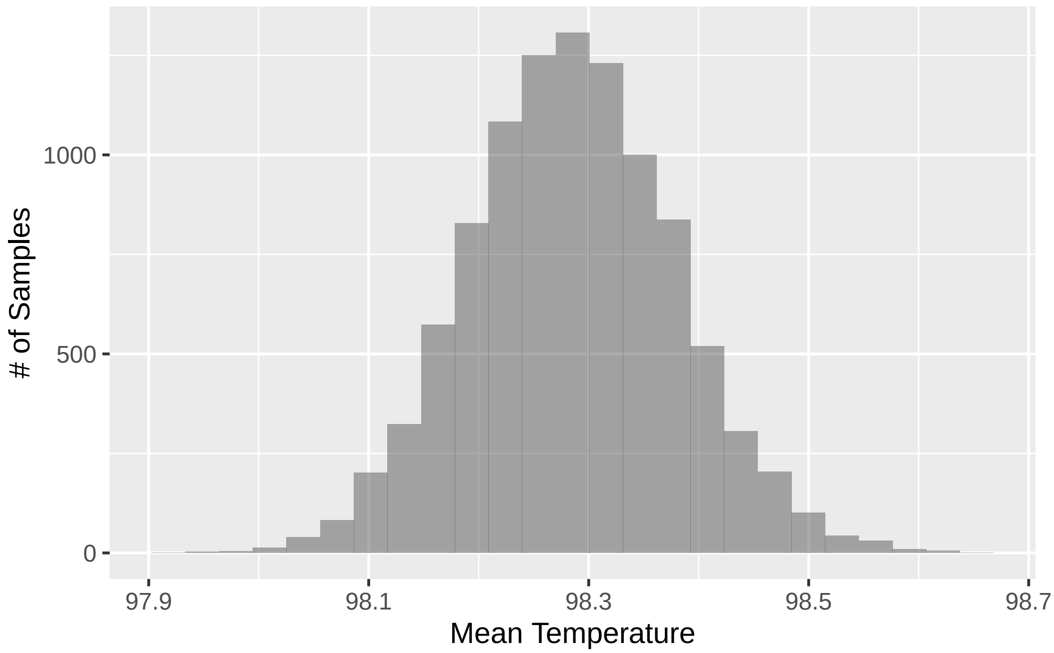
```r
do(2) * mean(~ Temp, data = resample(Temperatures)) # Calculates means of two resamples
```

```
      mean
1 98.10769
2 98.24038
```

```
# We will use do() a numsamp number of times
resampletemps <- do(numsamp) * mean(~ Temp, data = resample(Temperatures))
```

For more information about `resample()`, refer to the resampling vignette in mosaic: https://cran.r-project.org/web/packages/mosaic/vignettes/Resampling.html

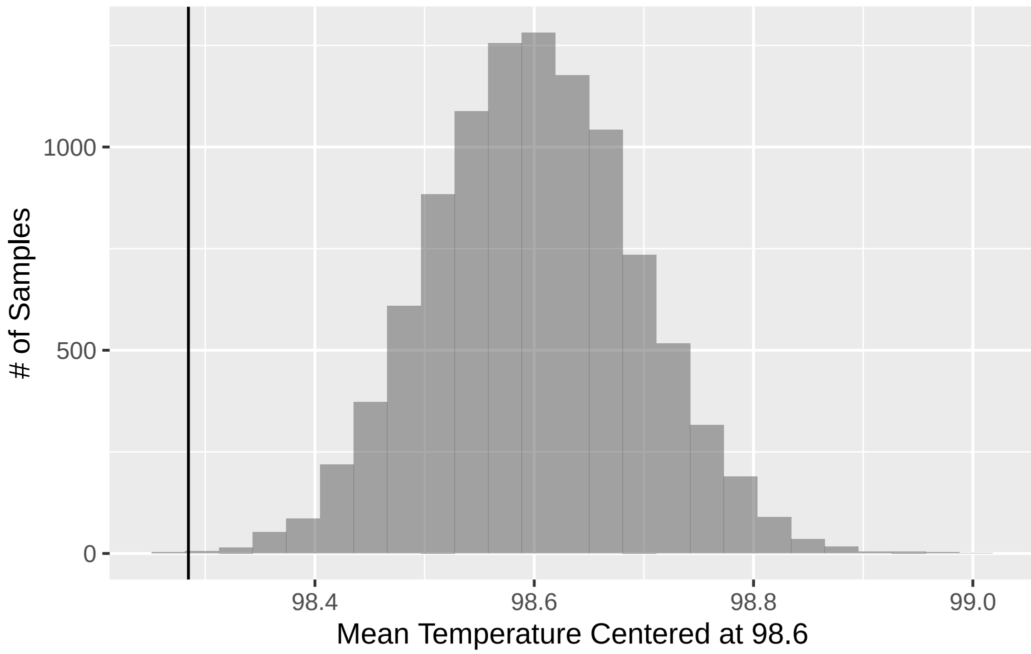
```
gf_histogram(~ mean, data = resampletemps) |>
  gf_labs(x = "Mean Temperature", y = "# of Samples")
```




```
qdata(~ mean, p = c(.005, .995), data = resampletemps) # reject null hypothesis
```

```
    0.5%    99.5%
98.05192 98.53846
```

```
# Making a model-centric distribution
Temperatures2 <- Temperatures |>
  mutate(Temp = Temp + .315)
resampletemps2 <- do(numsamp) * mean(~ Temp, data = resample(Temperatures2))
gf_histogram(~ mean, data = resampletemps2) |>
  gf_vline(xintercept = mean(~ Temp, data = Temperatures)) |>
  gf_labs(x = "Mean Temperature Centered at 98.6", y = "# of Samples")
```



Step-By-Step Example: Tests and Intervals

We begin by creating the dataset.

```
# Creating the data set
Baseball <- bind_rows(
  do(1308) * (winner <- "HOME"),
  do(2431 - 1308) * (winner <- "AWAY")
) |>
  rename(winner = result)
# Mechanics (page 490)
n <- nrow(Baseball)
p <- .5
phat <- Baseball |>
  filter(winner == "HOME") |>
  nrow() / n
phat
```

```
[1] 0.5380502
```

```
sdphat <- ((p * (1 - p)) / n)^.5
sdphat
```

```
[1] 0.01014092
```

```r
z <- (phat - p) / sdphat # z-value
z
```

```
[1] 3.752142
```

```r
1 - pnorm(z) # p-value
```

```
[1] 8.76651e-05
```

```r
# Or, without calculating the z-score:
1 - pnorm(q = phat, mean = p, sd = sdphat)
```

```
[1] 8.76651e-05
```

```r
# Mechanics (page 491)
sep <- ((phat * (1 - phat)) / n)^.5
sep
```

```
[1] 0.01011152
```

```r
me <- 1.96 * sep
phat - me # lower bound of 95% confidence
```

```
[1] 0.5182316
```

```r
phat + me # upper bound of 95% confidence
```

```
[1] 0.5578688
```

Section 15.6: P-Values and Decisions: What to Tell About a Hypothesis Test