The Infer Package and Bayes Factors

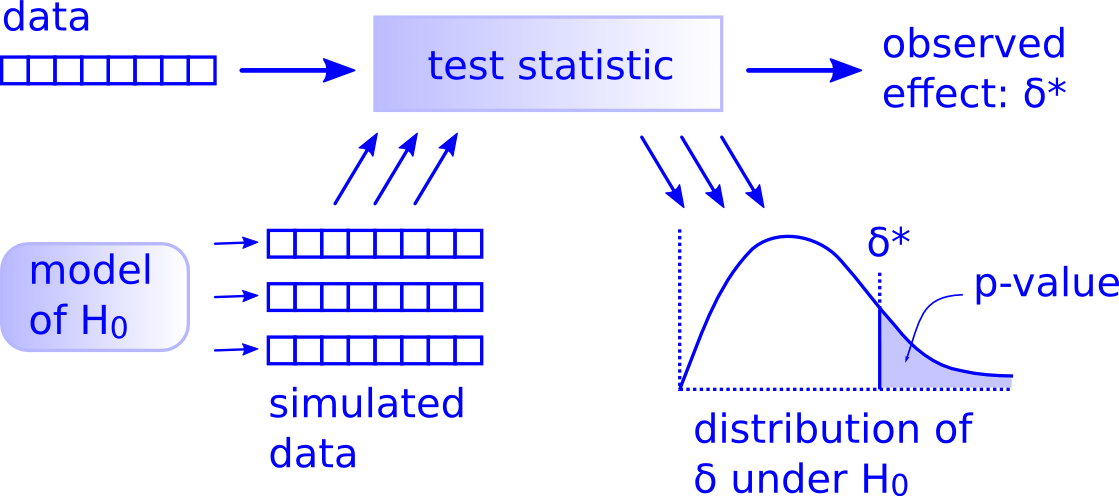
Jon Minton

# Introduction

This document will introduce a couple of concepts, and one R package, for performing statistical inference tests in R.

1. The There-is-only-one-test framework, and the R package infer
2. Bayes Factors, as distinct from Neyman-Pearson Hypothesis Testing frameoworks

# Proposed structure

* TIOOT/Infer
  + There-is-only-one-test : background
    - Allen Downey - a computer scientist’s take on statistical inference
      * 2011 - [There is only one test](http://allendowney.blogspot.com/2011/05/there-is-only-one-test.html)
      * 2016 - [There is still only one test](http://allendowney.blogspot.com/2016/06/there-is-still-only-one-test.html)
      * 
    - Null & Alternative Hypothesis
      * Null : The (boring) world in which the proposed relationship between predictor and response variables is *false*
      * Alternative : The (interesting) world in which the proposed relationship between predictor and response is *true*
    - P-values
      * P-values are a measure of conflict between data and a hypothesis, and are certainly not direct expressions of a probability of hypotheses.
        + [Communicating Uncertainty about facts, numbers, and science](https://royalsocietypublishing.org/doi/pdf/10.1098/rsos.181870)
      * P-values are usually *indirect* strategies for investigating support for , through quantifying *magnitude* and *direction* of conflict between data and
        + *Direction of conflict*: One-tailed cf two-tailed tests
        + *Magnitude of conflict*: Probability of observing values of effect as or more extreme under .
      * Neyman-Pearson Approach
        + Also prespecify : The magnitude of conflict between data and Null hypothesis, in expected direction, required to ‘reject the Null’
        + Typical values: 0.05, 0.01

CERN: ‘Five Sigma’ (almost zero)

* + - * + [The Sizeless Stare](https://www.amazon.co.uk/Cult-Statistical-Significance-Economics-Cognition/dp/0472050079)
      * Permutation approaches
        + A generic, computationally intensive, approach to producing a distribution of expected effect sizes under the Null
        + Key intuition:

says predicts/causes

Put another way: Values of are *informative* as to values of

Say is categorical, and can either be or

The data is a series of values of , with *labels* attached: the corresponding values of

Another way of expressing and :

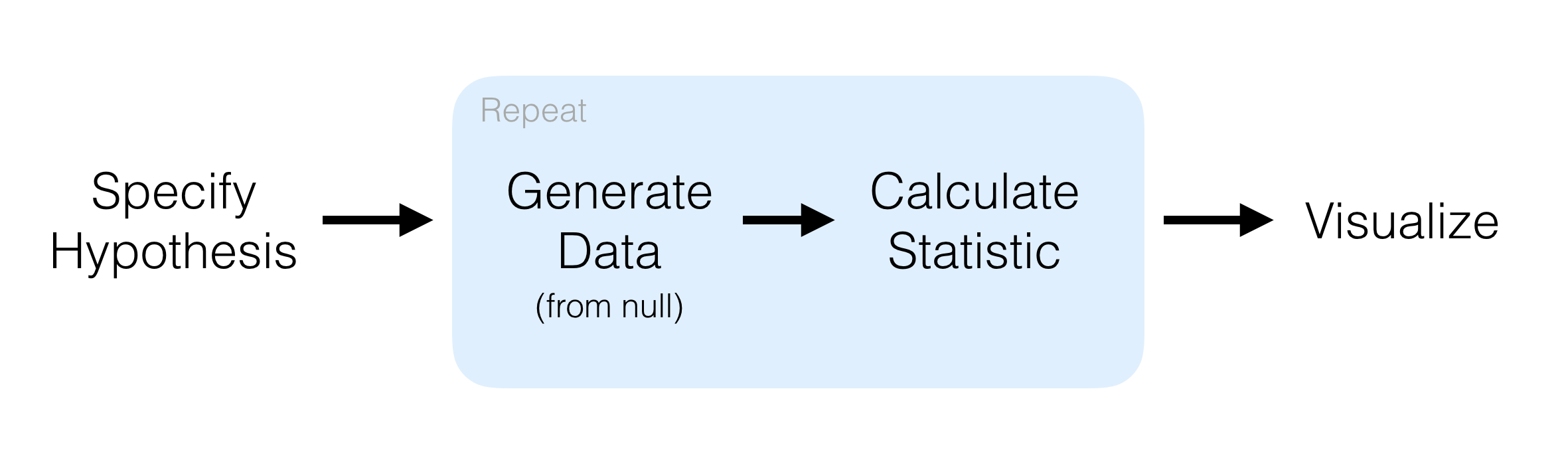
: These labels *matter* (Are informative of )

: These labels *don’t matter* (Are not informative of )

A corollary of : If the labels don’t matter, there’s no harm (information lost) by reallocating them to values at random

Random reallocation: Permutation

Repeat many times to estimate the Null distribution

* + [Infer package](https://github.com/tidymodels/infer)
    - 
    - Verbs
      * specify
      * hypothesize
      * generate
      * calculate
  + DataCamp courses [(Learner beware)](https://www.buzzfeednews.com/article/daveyalba/datacamp-sexual-harassment-metoo-tech-startup)
    - [Foundations of Inference](https://www.datacamp.com/courses/foundations-of-inference)
    - [Inference for Linear Regression](https://www.datacamp.com/courses/inference-for-linear-regression)
    - [Inference for Numeric Data](https://www.datacamp.com/courses/inference-for-numerical-data)
    - [Inference for Categorical Data](https://www.datacamp.com/courses/inference-for-categorical-data)
  + [Avoiding Datacamp](https://bookdown.org/cteplovs/ismaykim/ismaykim.pdf)

# Examples of infer

pacman::p\_load(tidyverse, infer)

# Initial example: mtcars

* [mtcars dataset](https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/mtcars.html)

## Variables of interest

* **vs**: Engine (0 - V-shaped; 1 = straight)
* **am**: Transmission (0 = automatic, 1 = manual)
* **mpg**: Miles per gallon

## Prep the data

mtcars <- as.data.frame(mtcars) %>%  
 mutate(cyl = factor(cyl),  
 vs = factor(vs),  
 am = factor(am),  
 gear = factor(gear),  
 carb = factor(carb))

## First research question

* Does engine type influence(!) Transmission Type
* Null: It does not
* Alternative: It does

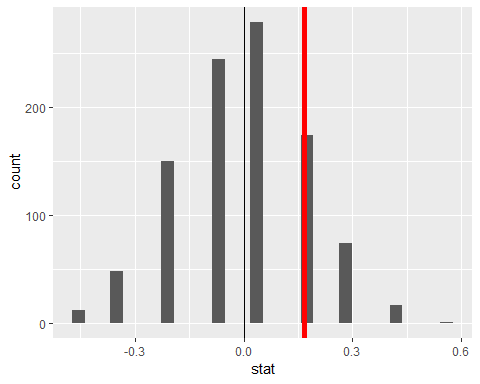
# Observed difference in proportions   
obs\_diff\_rq1 <-   
 mtcars %>%   
 select(am, vs) %>%   
 group\_by(vs) %>%   
 summarise(prop = mean(am == 1)) %>%   
 summarise(diff\_in\_props = diff(prop)) %>%   
 pull(diff\_in\_props)  
  
obs\_diff\_rq1

## [1] 0.1666667

# Null distribution:   
null\_rq1 <-   
 mtcars %>%  
 specify(am ~ vs, success = "1") %>%  
 hypothesize(null = "independence") %>%  
 generate(reps = 1000, type = "permute") %>%  
 calculate(stat = "diff in props", order = c("1", "0"))

# Visualise the Null  
  
null\_rq1 %>%   
 ggplot(aes(x = stat)) +   
 geom\_histogram() +   
 geom\_vline(xintercept = 0) +   
 geom\_vline(xintercept = obs\_diff\_rq1, colour = "red", size = 2)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



# P-value (two sided)  
  
null\_rq1 %>%   
 summarise(p\_val = mean(stat > obs\_diff\_rq1))

## # A tibble: 1 x 1  
## p\_val  
## <dbl>  
## 1 0.092

To answer the RQ in Neyman-Pearson terms, with alpha of 0.05, two-tailed: > Is the above value either greater than 0.975, or less than 0.025?

## Second research question

Do manual transmission vehicles (from the 1970s) have higher MPG?

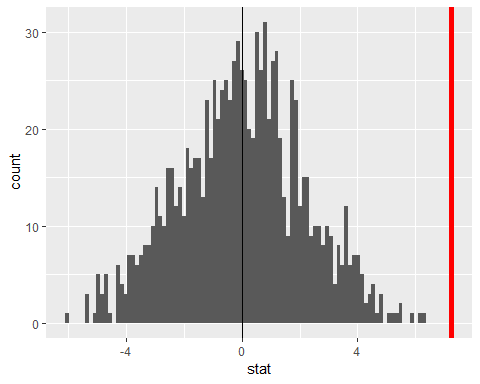
* Alternative: They do
* Null: They don’t

# Observed difference in means  
obs\_diff\_rq2 <-   
 mtcars %>%   
 group\_by(am) %>%   
 summarise(mean\_mpg = mean(mpg)) %>%   
 summarise(diff\_in\_means = diff(mean\_mpg)) %>%   
 pull(diff\_in\_means)  
  
obs\_diff\_rq2

## [1] 7.244939

null\_rq2 <-   
 mtcars %>%  
 specify(response = mpg, explanatory = am) %>%  
 hypothesize(null = "independence") %>%   
 generate(reps = 1000, type = "permute") %>%  
 calculate(stat = "diff in means", order = c("1", "0"))

# Visualise  
  
null\_rq2 %>%   
 ggplot(aes(x = stat)) +   
 geom\_histogram(bins = 100) +  
 geom\_vline(xintercept = 0) +   
 geom\_vline(xintercept = obs\_diff\_rq2, colour = "red", size = 2)



# P-value (one-sided)  
  
null\_rq2 %>%   
 summarise(p\_val = mean(stat > obs\_diff\_rq2))

## # A tibble: 1 x 1  
## p\_val  
## <dbl>  
## 1 0

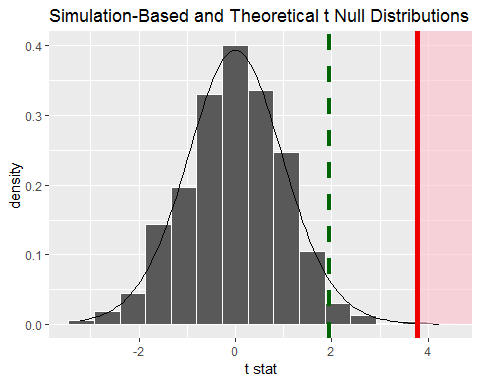
To answer the RQ in Neyman-Pearson terms, with alpha of 0.05, two-tailed:

Is the above value either greater than 0.975, or less than 0.025?

# RQ2 using the t-distribution   
  
obs\_diff\_rq2\_t <-   
 mtcars %>%   
 t\_test(mpg ~ am, order = c("1", "0"), alternative = "less")  
  
mtcars %>%  
 specify(response = mpg, explanatory = am) %>%  
 hypothesize(null = "independence") %>%   
 generate(reps = 1000, type = "permute") %>%  
 calculate(stat = "t", order = c("1", "0")) %>%   
 visualize(method = "both") +   
 shade\_p\_value(obs\_stat = obs\_diff\_rq2\_t, direction = "greater") +  
 geom\_vline(xintercept = 1.96, colour = "darkgreen", size = 1.5, linetype = "dashed")

## Warning: Check to make sure the conditions have been met for the  
## theoretical method. {infer} currently does not check these for you.

## Warning: The first row and first column value of the given `obs\_stat` will  
## be used.



# Second Example (If we get time)

RQ: Are males taller than females (on average)

* : No!
* : Yes!

## Data Source - Kaggle

* From [this page](https://www.kaggle.com/majidarif17/weight-and-heightcsv).
* Provenance unknown…
* Unit appears to be inches

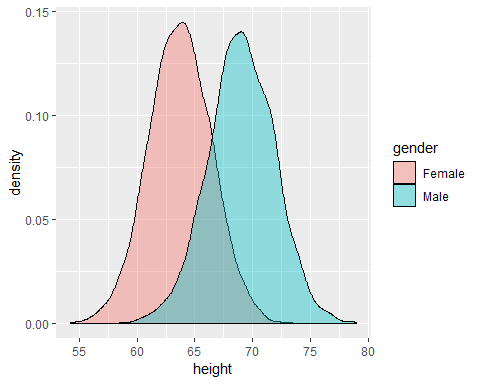
The dataset has many observations (though they may be synthetic), and so I’m going to create a variant with far fewer observations too

## Load and standardise data

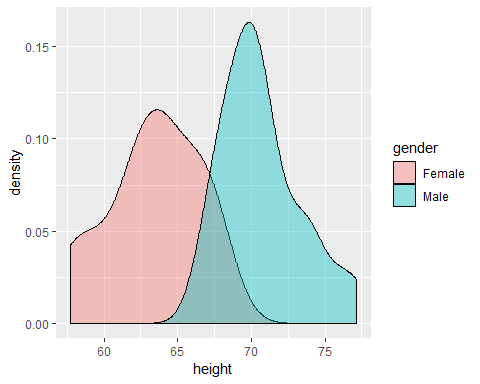
# pacman::p\_load(readxl)  
# height\_data <- read\_excel("data/height\_data.xlsx")  
height\_data <- read\_csv("data/kaggle\_weight-height.csv")

## Parsed with column specification:  
## cols(  
## Gender = col\_character(),  
## Height = col\_double(),  
## Weight = col\_double()  
## )

tidy\_height <-   
 height\_data %>%   
 select(gender = Gender, height = Height)   
  
set.seed(20)   
tiny\_tidy\_height <-   
 tidy\_height %>%   
 sample\_n(size = 40, replace = FALSE)  
  
tidy\_height %>%   
 ggplot(aes(x = height, group = gender, fill = gender)) +   
 geom\_density(alpha = 0.4)



tiny\_tidy\_height %>%   
 ggplot(aes(x = height, group = gender, fill = gender)) +   
 geom\_density(alpha = 0.4)



## Hypothesis test

obs\_mean\_diff <-   
 tidy\_height %>%   
 group\_by(gender) %>%   
 summarise(mean\_height = mean(height)) %>%   
 summarise(mean\_diff = mean\_height[gender == "Male"] - mean\_height[gender == "Female"]) %>%   
 pull(mean\_diff)  
  
obs\_mean\_diff  
  
tiny\_obs\_mean\_diff <-   
 tiny\_tidy\_height %>%   
 group\_by(gender) %>%   
 summarise(mean\_height = mean(height)) %>%   
 summarise(mean\_diff = mean\_height[gender == "Male"] - mean\_height[gender == "Female"]) %>%   
 pull(mean\_diff)  
  
tiny\_obs\_mean\_diff  
  
  
tidy\_height %>%   
 specify(height ~ gender) %>%  
 hypothesize(null = "independence") %>%   
 generate(reps = 1000, type = "permute") %>%  
 calculate(stat = "diff in means", order = c("Male", "Female")) %>%   
 visualize() # %>%  
 # shade\_p\_value(  
 # obs\_stat = obs\_mean\_diff,  
 # direction = "greater"  
 # )  
  
tiny\_tidy\_height %>%   
 specify(height ~ gender) %>%  
 hypothesize(null = "independence") %>%   
 generate(reps = 1000, type = "permute") %>%  
 calculate(stat = "diff in means", order = c("Male", "Female")) %>%   
 visualize()

The range of Null differences is far outside the observed difference, and so the probability there’s no true difference given this dataset is effectively zero.

# Bayes Factors

* Analogy: Courts and Burden of Proof
  + Criminal Courts: ‘Beyond Reasonable Doubt’ (Neyman-Pearson/Classical approach)
  + Civil Courts: ‘Balance of Probabilities’ (The Bayes Factor: Likelihood Ratio)
* Likelihood and Probability
  + The likelihood of the model given the data is proportional to the probability of the data given the model
  + Likelihood is always relative rather than absolute
    - A model is never *likely*, just *more/less likely* than another model
* Bayes Factors
  + B > 1 : More support for Alternative Hypothesis
  + B < 1 : More support for Null Hypothesis
  + How much more support?
    - < 3: ‘anecdotal’
    - < 10: ‘moderate’
    - < 30: ‘strong’
    - etc

## Example using mtcars dataset

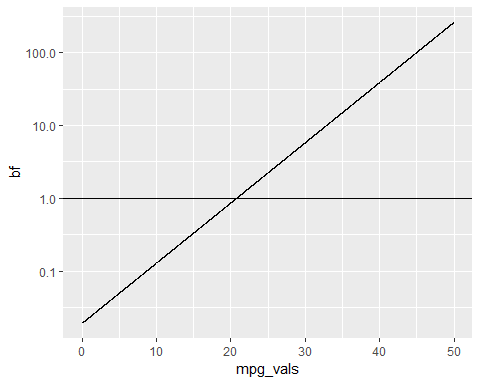
Given some value of MPG, what’s the relative likelihood the car is manual (Alt) or automatic (Null) transmission?

summaries <-   
 mtcars %>%   
 group\_by(am) %>%   
 summarise(  
 mean\_mpg = mean(mpg),   
 sd\_mpg = sd(mpg)  
 )  
  
mu\_manual <- summaries %>% filter(am == 1) %>% pull(mean\_mpg)  
sd\_manual <- summaries %>% filter(am == 1) %>% pull(sd\_mpg )  
mu\_automatic <- summaries %>% filter(am == 0) %>% pull(mean\_mpg)  
sd\_automatic <- summaries %>% filter(am == 0) %>% pull(sd\_mpg )

Bayes factor (Manual cf automatic) given MPG

calc\_bayes\_factor <- function(value, mu\_null, sd\_null, mu\_alt, sd\_alt){  
 dnorm(value, mean = mu\_alt, sd = sd\_alt) / dnorm(value, mean = mu\_null, sd = sd\_alt)  
}

bf\_schedule <-   
 tibble(  
 mpg\_vals = seq(0, 50, by = 0.01)  
 ) %>%   
 mutate(bf = map\_dbl(mpg\_vals, calc\_bayes\_factor, mu\_null = mu\_automatic, sd\_null = sd\_automatic, mu\_alt = mu\_manual, sd\_alt = sd\_manual )  
 )   
  
bf\_schedule %>%   
 ggplot(aes(x = mpg\_vals, y = bf)) +   
 geom\_line() +   
 scale\_y\_log10() +  
 geom\_hline(yintercept = 1)



## Example using our heights data

summaries <-   
 tidy\_height %>%   
 group\_by(gender) %>%   
 summarise(  
 mean\_height = mean(height),  
 sd\_height = sd(height)  
 )   
  
mu\_male <- summaries %>% filter(gender == "Male") %>% pull(mean\_height)  
sd\_male <- summaries %>% filter(gender == "Male") %>% pull(sd\_height )  
mu\_female <- summaries %>% filter(gender == "Female") %>% pull(mean\_height)  
sd\_female <- summaries %>% filter(gender == "Female") %>% pull(sd\_height )  
  
tiny\_summaries <-   
 tiny\_tidy\_height %>%   
 group\_by(gender) %>%   
 summarise(  
 mean\_height = mean(height),  
 sd\_height = sd(height)  
 )   
  
mu\_male\_tiny <- tiny\_summaries %>% filter(gender == "Male") %>% pull(mean\_height)  
sd\_male\_tiny <- tiny\_summaries %>% filter(gender == "Male") %>% pull(sd\_height )  
mu\_female\_tiny <- tiny\_summaries %>% filter(gender == "Female") %>% pull(mean\_height)  
sd\_female\_tiny <- tiny\_summaries %>% filter(gender == "Female") %>% pull(sd\_height )

bf\_schedule <-   
 tibble(  
 height\_vals = seq(50, 90, by = 1)  
 ) %>%   
 mutate(bf = map\_dbl(height\_vals, calc\_bayes\_factor, mu\_null = mu\_female, sd\_null = sd\_female, mu\_alt = mu\_male, sd\_alt = sd\_male)  
 )   
  
tiny\_bf\_schedule <-   
 tibble(  
 height\_vals = seq(50, 90, by = 1)  
 ) %>%   
 mutate(bf = map\_dbl(height\_vals, calc\_bayes\_factor, mu\_null = mu\_female\_tiny, sd\_null = sd\_female\_tiny, mu\_alt = mu\_male\_tiny, sd\_alt = sd\_male\_tiny)  
 )   
  
  
  
bf\_schedule %>%   
 ggplot(aes(x = height\_vals, y = bf)) +   
 geom\_line() +   
 scale\_y\_log10() +  
 geom\_hline(yintercept = 1)



tiny\_bf\_schedule %>%   
 ggplot(aes(x = height\_vals, y = bf)) +   
 geom\_line() +   
 scale\_y\_log10() +  
 geom\_hline(yintercept = 1)



Finally, what’s the relative likelihood that I’m male?

my\_height <- 1.7145 \* 39.3701 # to get to height in inches  
  
calc\_bayes\_factor(value = my\_height, mu\_null = mu\_female, sd\_null = sd\_female, mu\_alt = mu\_male, sd\_alt = sd\_male)

## [1] 2.084433

calc\_bayes\_factor(value = my\_height, mu\_null = mu\_female\_tiny, sd\_null = sd\_female\_tiny, mu\_alt = mu\_male\_tiny, sd\_alt = sd\_male\_tiny)

## [1] 1.616706

So, from height alone, it’s twice as likely I’m male than female, if using the full dataset, and about 60% more likely I’m make if using the small sample.

# Conclusion/Discussion

* There-is-only-one-test makes it easy to think through hypothesis testing
* The infer package makes it straightforward to apply this approach
* Computationally intensive and analytic approaches do the same thing in different ways
* Classical (Neyman-Pearson style) hypothesis tests may not answer the question you want to ask
* Bayes Factors are an alterantive and often more pragmatic way of weighing the evidence