431 Class 05

https://thomaselove.github.io/431-2024/

2024-09-10

## Today’s Agenda

* An introduction to the dm464 study
* Estimating the Difference between Two Population Means
  + using paired samples
  + using independent samples
* Comparisons using
  + ordinary least squares linear models (and their t-based equivalents),
  + the bootstrap, and
  + Bayesian linear models
* Most of the material in these slides is also discussed in Chapters 5-6 of [our course book](https://thomaselove.github.io/431-book/).

## Load packages and set theme

library(janitor)  
  
library(ggdist) ## new today  
library(knitr) ## new (sort of) today  
library(kableExtra) ## new today  
library(MKinfer) ## new today  
  
library(patchwork)  
library(rstanarm)  
library(easystats)  
library(tidyverse)  
  
theme\_set(theme\_bw())  
knitr::opts\_chunk$set(comment = NA)  
  
source("c05/data/Love-431.R") # for the lovedist() function

# Managing the Data

## Today’s data (dm464\_class05.csv)

The data describe a cohort of 464 adults with diabetes, measured at baseline, and then two years later.

dm5 <- read\_csv("c05/data/dm464\_class05.csv", show\_col\_types = FALSE)   
  
names(dm5)

[1] "ID.Code" "AGE" "sex" "Statin" "RESIDENCE" "A1c Base"   
[7] "A1c-End" "LDLBase"

dm5 <- janitor::clean\_names(dm5)  
  
names(dm5)

[1] "id\_code" "age" "sex" "statin" "residence" "a1c\_base"   
[7] "a1c\_end" "ldl\_base"

dim(dm5)

[1] 464 8

## Variable Definitions (1/2)

Cohort study where 464 adult subjects (ages 21-75) with a diabetes diagnosis were measured during a 12-month baseline period, and then again two years later.

| Variable | Definition |
| --- | --- |
| id\_code | Subject Code (unique for each subject/row) |
| age | Age at end of the baseline period, in years |
| sex | Female or Male |
| statin | Statin prescription in baseline period: 1 (yes) or 0 (no) |
| residence | place of residence in baseline: City or Suburbs |
| a1c\_base | baseline Hemoglobin A1c, % |
| a1c\_end | most recent Hemoglobin A1c at end of study, % |
| ldl\_base | baseline LDL cholesterol, mg/dl |

## Convert Characters to Factors

dm5 <- dm5 |>  
 mutate(across(where(is.character), as\_factor)) |>  
 mutate(id\_code = as.character(id\_code))  
  
dm5 |> tail() # final six observations

# A tibble: 6 × 8  
 id\_code age sex statin residence a1c\_base a1c\_end ldl\_base  
 <chr> <dbl> <fct> <dbl> <fct> <dbl> <dbl> <dbl>  
1 DM-9916 48 Male 1 City 6 6.2 140  
2 DM-9926 67 Female 1 City 6.3 12.7 90  
3 DM-9930 62 Male 0 Suburbs 6.1 6 88  
4 DM-9932 50 Male 0 Suburbs 7.6 7.8 96  
5 DM-9936 61 Female 0 City 6.5 8 144  
6 DM-9938 66 Male 1 City 7.1 7.4 81

* We might convert statin (1/0) to a binary factor, but I won’t, at least for now.

## Automated “Codebook”

* Why am I leaving out the id\_code results here?
* This is most useful for missingness and range checks.

data\_codebook(dm5 |> select(-id\_code))

select(dm5, -id\_code) (464 rows and 7 variables, 7 shown)  
  
ID | Name | Type | Missings | Values | N  
---+-----------+-------------+----------+-------------+------------  
1 | age | numeric | 0 (0.0%) | [21, 73] | 464  
---+-----------+-------------+----------+-------------+------------  
2 | sex | categorical | 0 (0.0%) | Female | 288 (62.1%)  
 | | | | Male | 176 (37.9%)  
---+-----------+-------------+----------+-------------+------------  
3 | statin | numeric | 0 (0.0%) | 0 | 97 (20.9%)  
 | | | | 1 | 367 (79.1%)  
---+-----------+-------------+----------+-------------+------------  
4 | residence | categorical | 0 (0.0%) | City | 357 (76.9%)  
 | | | | Suburbs | 107 (23.1%)  
---+-----------+-------------+----------+-------------+------------  
5 | a1c\_base | numeric | 0 (0.0%) | [4.2, 14.9] | 464  
---+-----------+-------------+----------+-------------+------------  
6 | a1c\_end | numeric | 0 (0.0%) | [4.4, 17] | 464  
---+-----------+-------------+----------+-------------+------------  
7 | ldl\_base | numeric | 0 (0.0%) | [40, 231] | 464  
-------------------------------------------------------------------

## Today’s Analytic Questions

1. How large is the difference in the mean Hemoglobin A1c level at baseline as compared to the A1c level in the follow-up period?
2. How large is the difference in baseline LDL for patients who do have a statin prescription at baseline compared to those who don’t have a statin prescription?

# Compare the Hemoglobin A1c level at baseline to the A1c level in the follow-up period.

## Comparing a1c\_base to a1c\_end

We are comparing each subject’s a1c\_base to their a1c\_end, to learn something about the mean of those differences.

Does this planned analysis make use of paired samples or independent samples?

* Each subject provides an a1c\_base as well as an a1c\_end.
* Calculating the difference (a1c\_end - a1c\_base) makes sense for each individual subject.
* This will leave us with paired differences to study.

## Build the paired differences in A1c

dm5 <- dm5 |>  
 mutate(a1c\_diff = a1c\_end - a1c\_base)  
  
fivenum(dm5$a1c\_diff) ## min, q25, median, q75, max

[1] -5.9 -0.5 0.2 1.1 8.9

stem(dm5$a1c\_diff) ## stem-and-leaf display

The decimal point is at the |  
  
 -5 | 9  
 -4 | 620  
 -3 | 88753320000  
 -2 | 887765554333332222211000  
 -1 | 99988887776655555544444444333322222222211111110000000  
 -0 | 99998888777777666666555555555555544444443333333333333332222222222221  
 0 | 00000000001111111111111111111111111111222222222222222222222222223333+86  
 1 | 0000000001111111122222222223333333333344444555555555566777788899  
 2 | 000001112222333445555666677779  
 3 | 0002346667778999  
 4 | 112233699  
 5 | 019  
 6 | 04  
 7 | 88  
 8 | 9

## What does the distribution of A1c differences look like? (1/4)

* Plot p1 - Histogram with superimposed Normal curve

bw = 1 # specify width of bins in histogram  
  
p1 <- ggplot(dm5, aes(x = a1c\_diff)) +  
 geom\_histogram(binwidth = bw, fill = "black", col = "yellow") +  
 stat\_function(fun = function(x)   
 dnorm(x, mean = mean(dm5$a1c\_diff, na.rm = TRUE),   
 sd = sd(dm5$a1c\_diff, na.rm = TRUE)) \*   
 length(dm5$a1c\_diff) \* bw,  
 geom = "area", alpha = 0.5,   
 fill = "lightblue", col = "blue") +  
 labs(x = "Bill Length in mm", y = "Count",  
 title = "Histogram & Normal Curve")

## What does the distribution of A1c differences look like? (2/4)

* Plot p2 - Normal Q-Q plot

p2 <- ggplot(dm5, aes(sample = a1c\_diff)) +  
 geom\_qq() + geom\_qq\_line(col = "red") +  
 labs(y = "Bill Length in mm", x = "Standard Normal Distribution",  
 title = "Normal Q-Q plot")

## What does the distribution of A1c differences look like? (3/4)

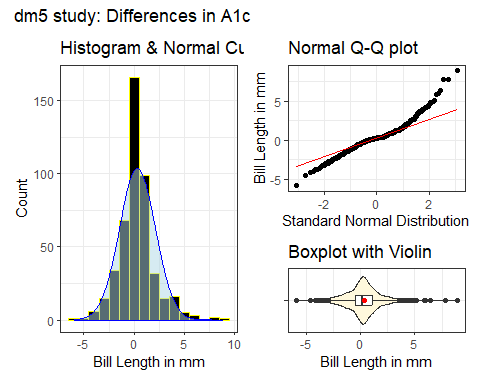
* Plot p3 - Boxplot with violin and mean

p3 <- ggplot(dm5, aes(x = a1c\_diff, y = "")) +  
 geom\_violin(fill = "cornsilk") +  
 geom\_boxplot(width = 0.2) +  
 stat\_summary(fun = mean, geom = "point", shape = 16, col = "red") +  
 labs(y = "", x = "Bill Length in mm", title = "Boxplot with Violin")

## Shape of the distribution? (4/4)

* Use the patchwork package to combine plots p1, p2 and p3 into a single figure
* See [section 2.4.6 in our Course Book](https://thomaselove.github.io/431-book/02_viz.html#three-plots-at-once) for similar “Three Plots At Once” code.

p1 +   
 (p2 / p3 + plot\_layout(heights = c(2, 1))) +  
 plot\_annotation(title = "dm5 study: Differences in A1c")



## Summaries of A1c differences

OK, so the distribution of A1c differences looks symmetric, but outlier-prone relative to a Normal distribution.

* Can we get some numerical summaries?

dm5 |> reframe(lovedist(a1c\_diff))

# A tibble: 1 × 10  
 n miss mean sd med mad min q25 q75 max  
 <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
1 464 0 0.332 1.79 0.200 1.19 -5.9 -0.5 1.1 8.9

* What does the mean (0.33) mean in this context?

## Are our paired A1c values correlated?

* One way to see if pairing helps is to look for a positive correlation between the A1c values at the start (a1c\_base) and at the end (a1c\_end) of the study.

cor(dm5$a1c\_base, dm5$a1c\_end)

[1] 0.5965358

* Here, we see a meaningful and positive correlation between the paired A1c values, suggesting that the pairing helped to reduce nuisance variation.

## Estimating the Mean Difference

We’ll demonstrate three approaches today (see [Chapter 5](https://thomaselove.github.io/431-book/05_paired.html)) for these paired samples.

1. Ordinary least squares regression model (equivalent to a paired t procedure in this setting) fit with lm().
2. Bayesian regression model with a weakly informative prior.
3. Bootstrap confidence interval for the mean difference.

In each case, we’ll estimate the mean difference and form a **90%** uncertainty interval.

## Using lm() to fit an OLS model

fit1 <- lm(a1c\_diff ~ 1, data = dm5)  
  
model\_parameters(fit1, ci = 0.90)

Parameter | Coefficient | SE | 90% CI | t(463) | p  
-----------------------------------------------------------------  
(Intercept) | 0.33 | 0.08 | [0.20, 0.47] | 4.00 | < .001

Uncertainty intervals (equal-tailed) and p-values (two-tailed) computed  
 using a Wald t-distribution approximation.

* Sample mean difference is 0.33, with 90% uncertainty interval (0.20, 0.47).
* All three of these summaries are positive numbers. What does that indicate?

## Paired t approach = same as OLS

* Note that here we must include conf.level = 0.90 within the t.test() command and ci = 0.90 in the call to model\_parameters() in order to get the right (90%) uncertainty interval.

fit2 <- t.test(dm5$a1c\_diff, conf.level = 0.90)  
model\_parameters(fit2, ci = 0.90)

One Sample t-test  
  
Parameter | Mean | mu | Difference | 90% CI | t(463) | p  
------------------------------------------------------------------------  
dm5$a1c\_diff | 0.33 | 0.00 | 0.33 | [0.20, 0.47] | 4.00 | < .001  
  
Alternative hypothesis: true mean is not equal to 0

## Bayesian fit

Bayesian inference is an excellent choice for virtually every regression model, even when using **weakly informative default priors** (as we will do in 431), because it yields estimates which are stable, and because it helps us present the uncertainty associated with our estimates in useful ways.

set.seed(431123)  
  
fit3 <- stan\_glm(a1c\_diff ~ 1, data = dm5, refresh = 0)  
  
model\_parameters(fit3, ci = 0.90)

Parameter | Median | 90% CI | pd | Rhat | ESS | Prior  
------------------------------------------------------------------------------------  
(Intercept) | 0.33 | [0.19, 0.47] | 100% | 1.002 | 2431.00 | Normal (0.33 +- 4.46)

Uncertainty intervals (equal-tailed) and p-values (two-tailed) computed  
 using a MCMC distribution approximation.

## The bootstrap (with MKinfer)

set.seed(20240910)  
boot.t.test(dm5$a1c\_diff, conf.level = 0.90, R = 2000)

Bootstrap One Sample t-test  
  
data: dm5$a1c\_diff  
number of bootstrap samples: 2000  
bootstrap p-value < 5e-04   
bootstrap mean of x (SE) = 0.3315159 (0.08268618)   
90 percent bootstrap percentile confidence interval:  
 0.1952586 0.4692349  
  
Results without bootstrap:  
t = 4.0016, df = 463, p-value = 7.323e-05  
alternative hypothesis: true mean is not equal to 0  
90 percent confidence interval:  
 0.1950719 0.4682901  
sample estimates:  
mean of x   
 0.331681

## 90% Uncertainty Intervals

Let’s consider the approaches we have demonstrated to estimate the mean of the paired differences in Hemoglobin A1c levels from the start to the end of the study…

| Approach | Estimate & 90% Interval |
| --- | --- |
| Ordinary least squares / Paired t | 0.33 (0.20, 0.47) |
| Bayesian fit with stan\_glm() | 0.33 (0.19, 0.47) |
| Bootstrap (via boot.t.test) | 0.33 (0.20, 0.47) |

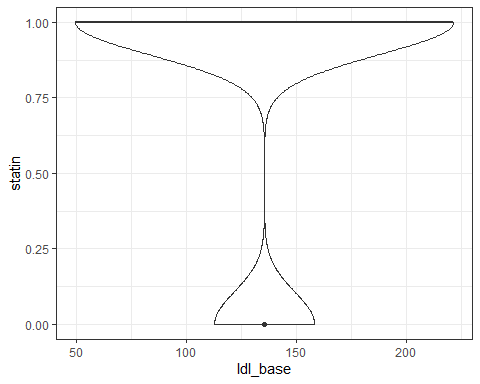
* Do our conclusions change here?

# Compare the baseline LDL for patients who do have a statin prescription at baseline to those who don’t.

## Comparing ldl\_base by statin level

ggplot(dm5, aes(x = ldl\_base, y = statin)) +  
 geom\_violin() +  
 geom\_boxplot(width = 0.3)

Warning: Continuous x aesthetic  
ℹ did you forget `aes(group = ...)`?



## Create a statin factor variable

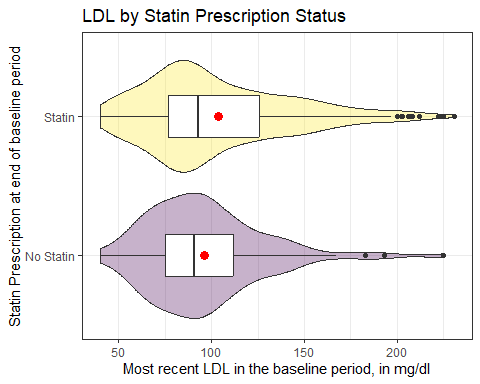
dm5 <- dm5 |>  
 mutate(statin\_f = as\_factor(statin),  
 statin\_f = fct\_recode(statin\_f,   
 "Statin" = "1", "No Statin" = "0"))  
  
dm5 |> tabyl(statin, statin\_f)

statin No Statin Statin  
 0 97 0  
 1 0 367

## Comparing ldl\_base by statin level

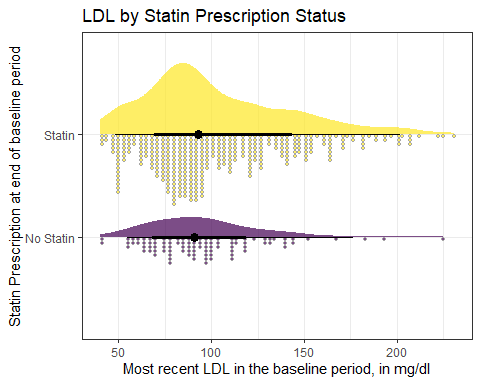
We’ll build a boxplot, including violins and means.

ggplot(dm5, aes(x = ldl\_base, y = statin\_f,   
 fill = statin\_f)) +  
 geom\_violin() +  
 geom\_boxplot(width = 0.3, fill = "white") +  
 stat\_summary(fun = mean, geom = "point", shape = 16, size = 3,  
 col = "red") +  
 scale\_fill\_viridis\_d(alpha = 0.3) +  
 guides(fill = "none") +  
 labs(title = "LDL by Statin Prescription Status",  
 x = "Most recent LDL in the baseline period, in mg/dl",   
 y = "Statin Prescription at end of baseline period")



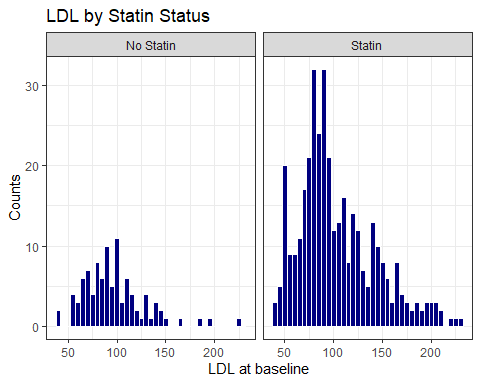
## Rain Cloud Plot

ggplot(dm5, aes(y = statin\_f, x = ldl\_base, fill = statin\_f)) +  
 stat\_slab(aes(thickness = after\_stat(pdf \* n)), scale = 0.7) +  
 stat\_dotsinterval(side = "bottom", scale = 0.7,   
 slab\_linewidth = NA) +  
 scale\_fill\_viridis\_d(alpha = 0.7) +  
 guides(fill = "none") +  
 labs(title = "LDL by Statin Prescription Status",  
 x = "Most recent LDL in the baseline period, in mg/dl",   
 y = "Statin Prescription at end of baseline period")



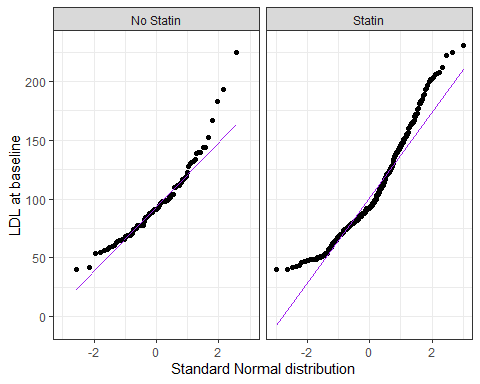
## Faceted Histograms

bw = 5  
ggplot(dm5, aes(x = ldl\_base)) +   
 geom\_histogram(binwidth = bw, fill = "navy", col = "white") +   
 facet\_wrap(~ statin\_f) +  
 labs(x = "LDL at baseline", y = "Counts",   
 title = "LDL by Statin Status")



## Faceted Normal Q-Q plots

ggplot(dm5, aes(sample = ldl\_base)) +   
 geom\_qq() + geom\_qq\_line(col = "purple") +   
 facet\_wrap(~ statin\_f) +  
 labs(x = "Standard Normal distribution", y = "LDL at baseline")



## Comparing ldl\_base by statin level

* LDL data in each statin group are skewed to the right, with the mean higher than the median.

dm5 |> group\_by(statin\_f) |> reframe(lovedist(ldl\_base))

# A tibble: 2 × 11  
 statin\_f n miss mean sd med mad min q25 q75 max  
 <fct> <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
1 No Statin 97 0 96.3 31.6 91 28.2 40 75 112 225  
2 Statin 367 0 103. 39.5 93 34.1 40 77 126 231

* Were these data collected using matched/paired samples or using independent samples?

## Estimating the Difference in Means

We’ll demonstrate four approaches today (see [Chapter 6](https://thomaselove.github.io/431-book/06_twogroups.html)) for these independent samples.

1. Ordinary least squares regression model (which is the same as a pooled t procedure in this setting) fit with lm().
2. Bayesian regression model with a weakly informative prior.
3. Welch t procedure without pooling the standard deviation across the two groups.
4. Bootstrap confidence interval for the difference in means.

Estimating difference in means with **90%** uncertainty interval.

## Using lm() to fit an OLS model

fit4 <- lm(ldl\_base ~ statin\_f, data = dm5)  
  
model\_parameters(fit4, ci = 0.90)

Parameter | Coefficient | SE | 90% CI | t(462) | p  
--------------------------------------------------------------------------  
(Intercept) | 96.26 | 3.86 | [89.90, 102.62] | 24.94 | < .001  
statin f [Statin] | 7.17 | 4.34 | [ 0.02, 14.32] | 1.65 | 0.099

Uncertainty intervals (equal-tailed) and p-values (two-tailed) computed  
 using a Wald t-distribution approximation.

* Comparing those on a statin to those that aren’t, our model suggests that the mean difference in LDL levels is 7.17 mg/dl with 90% uncertainty interval (0.02, 14.32) mg/dl.

## Pooled t approach = same as OLS

fit5 <- t.test(ldl\_base ~ statin\_f, data = dm5,   
 var.equal = TRUE, conf.level = 0.90)  
  
model\_parameters(fit5, ci = 0.90)

Two Sample t-test  
  
Parameter | Group | statin\_f = No Statin | statin\_f = Statin | Difference | 90% CI | t(462) | p  
---------------------------------------------------------------------------------------------------------------  
ldl\_base | statin\_f | 96.26 | 103.43 | -7.17 | [-14.32, -0.02] | -1.65 | 0.099  
  
Alternative hypothesis: true difference in means between group No Statin and group Statin is not equal to 0

## Bayesian regression model

set.seed(20240910)  
  
fit6 <- stan\_glm(ldl\_base ~ statin\_f, data = dm5, refresh = 0)  
  
model\_parameters(fit6, ci = 0.90)

Parameter | Median | 90% CI | pd | Rhat | ESS | Prior  
-----------------------------------------------------------------------------------------------  
(Intercept) | 96.14 | [89.75, 102.50] | 100% | 1.000 | 3994.00 | Normal (101.93 +- 95.20)  
statin\_fStatin | 7.22 | [-0.06, 14.56] | 94.85% | 0.999 | 3765.00 | Normal (0.00 +- 233.87)

Uncertainty intervals (equal-tailed) and p-values (two-tailed) computed  
 using a MCMC distribution approximation.

## Welch t approach (no pooling)

fit7 <- t.test(ldl\_base ~ statin\_f, data = dm5,   
 var.equal = FALSE, conf.level = 0.90)  
  
model\_parameters(fit7, ci = 0.90)

Welch Two Sample t-test  
  
Parameter | Group | statin\_f = No Statin | statin\_f = Statin | Difference | 90% CI | t(183.36) | p  
------------------------------------------------------------------------------------------------------------------  
ldl\_base | statin\_f | 96.26 | 103.43 | -7.17 | [-13.48, -0.86] | -1.88 | 0.062  
  
Alternative hypothesis: true difference in means between group No Statin and group Statin is not equal to 0

## Bootstrap, with MKinfer

set.seed(431022)  
boot.t.test(ldl\_base ~ statin\_f, var.equal = TRUE, R = 2000,  
 data = dm5, conf.level = 0.90)

Bootstrap Two Sample t-test  
  
data: ldl\_base by statin\_f  
number of bootstrap samples: 2000  
bootstrap p-value = 0.094   
bootstrap difference of means (SE) = -7.208243 (4.343016)   
90 percent bootstrap percentile confidence interval:  
 -14.286936 -0.341577  
  
Results without bootstrap:  
t = -1.6523, df = 462, p-value = 0.09915  
alternative hypothesis: true difference in means is not equal to 0  
90 percent confidence interval:  
 -14.32216921 -0.01795271  
sample estimates:  
mean in group No Statin mean in group Statin   
 96.25773 103.42779

## Bootstrap without pooling sd

set.seed(431023)  
boot.t.test(ldl\_base ~ statin\_f, var.equal = FALSE, R = 2000,  
 data = dm5, conf.level = 0.90)

Bootstrap Welch Two Sample t-test  
  
data: ldl\_base by statin\_f  
number of bootstrap samples: 2000  
bootstrap p-value = 0.096   
bootstrap difference of means (SE) = -7.230676 (3.798356)   
90 percent bootstrap percentile confidence interval:  
 -13.6630903 -0.8817706  
  
Results without bootstrap:  
t = -1.8786, df = 183.36, p-value = 0.06188  
alternative hypothesis: true difference in means is not equal to 0  
90 percent confidence interval:  
 -13.4797701 -0.8603518  
sample estimates:  
mean in group No Statin mean in group Statin   
 96.25773 103.42779

## 90% Uncertainty Intervals

Estimating the mean difference in LDL levels at baseline for the “Statin” group minus the “No Statin” group…

| Approach | Estimate & 90% Interval |
| --- | --- |
| Ordinary least squares / Pooled t | 7.17 (0.02, 14.32) |
| Bayesian fit with stan\_glm() | 7.22 (-0.06, 14.56) |
| Welch t without pooling sd | 7.17 (0.86, 13.48) |
| Bootstrap with pooled sd | 7.21 (0.34, 14.29) |
| Bootstrap with unpooled sd | 7.23 (0.88, 13.66) |

* Do our conclusions change here?

## Repeating Today’s Agenda

* An introduction to the dm464 study
* Estimating the Difference between Two Population Means
  + using paired samples
  + using independent samples
* Comparisons using
  + ordinary least squares linear models,
  + the bootstrap, and
  + Bayesian linear models
* For more examples, see Chapters 5-6 of [our course book](https://thomaselove.github.io/431-book/).
* At this point, you should be able to do [Lab 2](https://github.com/THOMASELOVE/431-labs-2024/tree/main/lab2).

## Session Information

xfun::session\_info()

R version 4.4.1 (2024-06-14 ucrt)  
Platform: x86\_64-w64-mingw32/x64  
Running under: Windows 11 x64 (build 22631)  
  
Locale:  
 LC\_COLLATE=English\_United States.utf8   
 LC\_CTYPE=English\_United States.utf8   
 LC\_MONETARY=English\_United States.utf8  
 LC\_NUMERIC=C   
 LC\_TIME=English\_United States.utf8   
  
Package version:  
 abind\_1.4-5 arrangements\_1.1.9 askpass\_1.2.0   
 backports\_1.5.0 base64enc\_0.1-3 bayesplot\_1.11.1   
 bayestestR\_0.14.0 BH\_1.84.0.0 bit\_4.0.5   
 bit64\_4.0.5 blob\_1.2.4 boot\_1.3-31   
 broom\_1.0.6 bslib\_0.8.0 cachem\_1.1.0   
 callr\_3.7.6 cellranger\_1.1.0 checkmate\_2.3.2   
 cli\_3.6.3 clipr\_0.8.0 coda\_0.19-4.1   
 codetools\_0.2-20 colorspace\_2.1-1 colourpicker\_1.3.0   
 commonmark\_1.9.1 compiler\_4.4.1 conflicted\_1.2.0   
 correlation\_0.8.5 cpp11\_0.5.0 crayon\_1.5.3   
 crosstalk\_1.2.1 curl\_5.2.2 data.table\_1.16.0   
 datasets\_4.4.1 datawizard\_0.12.3 DBI\_1.2.3   
 dbplyr\_2.5.0 desc\_1.4.3 digest\_0.6.37   
 distributional\_0.4.0 dplyr\_1.1.4 DT\_0.33   
 dtplyr\_1.3.1 dygraphs\_1.1.1.6 easystats\_0.7.3   
 effectsize\_0.8.9 emmeans\_1.10.4 estimability\_1.5.1   
 evaluate\_0.24.0 exactRankTests\_0.8-35 fansi\_1.0.6   
 farver\_2.1.2 fastmap\_1.2.0 fontawesome\_0.5.2   
 forcats\_1.0.0 foreach\_1.5.2 fs\_1.6.4   
 gargle\_1.5.2 generics\_0.1.3 ggdist\_3.3.2   
 ggplot2\_3.5.1 ggridges\_0.5.6 glmnet\_4.1-8   
 glue\_1.7.0 gmp\_0.7-5 googledrive\_2.1.1   
 googlesheets4\_1.1.1 graphics\_4.4.1 grDevices\_4.4.1   
 grid\_4.4.1 gridExtra\_2.3 gtable\_0.3.5   
 gtools\_3.9.5 haven\_2.5.4 highr\_0.11   
 hms\_1.1.3 htmltools\_0.5.8.1 htmlwidgets\_1.6.4   
 httpuv\_1.6.15 httr\_1.4.7 ids\_1.0.1   
 igraph\_2.0.3 inline\_0.3.19 insight\_0.20.4   
 isoband\_0.2.7 iterators\_1.0.14 janitor\_2.2.0   
 jomo\_2.7-6 jquerylib\_0.1.4 jsonlite\_1.8.8   
 kableExtra\_1.4.0 knitr\_1.48 labeling\_0.4.3   
 later\_1.3.2 lattice\_0.22-6 lazyeval\_0.2.2   
 lifecycle\_1.0.4 lme4\_1.1-35.5 loo\_2.8.0   
 lubridate\_1.9.3 magrittr\_2.0.3 markdown\_1.13   
 MASS\_7.3-61 Matrix\_1.7-0 matrixStats\_1.4.0   
 memoise\_2.0.1 methods\_4.4.1 mgcv\_1.9.1   
 mice\_3.16.0 miceadds\_3.17-44 mime\_0.12   
 miniUI\_0.1.1.1 minqa\_1.2.8 mitml\_0.4-5   
 mitools\_2.4 MKdescr\_0.8 MKinfer\_1.2   
 modelbased\_0.8.8 modelr\_0.1.11 multcomp\_1.4-26   
 munsell\_0.5.1 mvtnorm\_1.3-1 nlme\_3.1-164   
 nloptr\_2.1.1 nnet\_7.3-19 numDeriv\_2016.8.1.1   
 openssl\_2.2.1 ordinal\_2023.12.4.1 pan\_1.9   
 parallel\_4.4.1 parameters\_0.22.2 patchwork\_1.2.0   
 performance\_0.12.3 pillar\_1.9.0 pkgbuild\_1.4.4   
 pkgconfig\_2.0.3 plyr\_1.8.9 posterior\_1.6.0   
 prettyunits\_1.2.0 processx\_3.8.4 progress\_1.2.3   
 promises\_1.3.0 ps\_1.7.7 purrr\_1.0.2   
 quadprog\_1.5-8 QuickJSR\_1.3.1 R6\_2.5.1   
 ragg\_1.3.2 rappdirs\_0.3.3 RColorBrewer\_1.1.3   
 Rcpp\_1.0.13 RcppArmadillo\_14.0.0.1 RcppEigen\_0.3.4.0.2   
 RcppParallel\_5.1.9 readr\_2.1.5 readxl\_1.4.3   
 rematch\_2.0.0 rematch2\_2.1.2 report\_0.5.9   
 reprex\_2.1.1 reshape2\_1.4.4 rlang\_1.1.4   
 rmarkdown\_2.28 rpart\_4.1.23 rstan\_2.32.6   
 rstanarm\_2.32.1 rstantools\_2.4.0 rstudioapi\_0.16.0   
 rvest\_1.0.4 sandwich\_3.1-0 sass\_0.4.9   
 scales\_1.3.0 see\_0.9.0 selectr\_0.4.2   
 shape\_1.4.6.1 shiny\_1.9.1 shinyjs\_2.1.0   
 shinystan\_2.6.0 shinythemes\_1.2.0 snakecase\_0.11.1   
 sourcetools\_0.1.7.1 splines\_4.4.1 StanHeaders\_2.32.10   
 stats\_4.4.1 stats4\_4.4.1 stringi\_1.8.4   
 stringr\_1.5.1 survival\_3.7-0 svglite\_2.1.3   
 sys\_3.4.2 systemfonts\_1.1.0 tensorA\_0.36.2.1   
 textshaping\_0.4.0 TH.data\_1.1-2 threejs\_0.3.3   
 tibble\_3.2.1 tidyr\_1.3.1 tidyselect\_1.2.1   
 tidyverse\_2.0.0 timechange\_0.3.0 tinytex\_0.52   
 tools\_4.4.1 tzdb\_0.4.0 ucminf\_1.2.2   
 utf8\_1.2.4 utils\_4.4.1 uuid\_1.2.1   
 V8\_5.0.0 vctrs\_0.6.5 viridisLite\_0.4.2   
 vroom\_1.6.5 withr\_3.0.1 xfun\_0.47   
 xml2\_1.3.6 xtable\_1.8-4 xts\_0.14.0   
 yaml\_2.3.10 zoo\_1.8-12