431 Class 03

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

2024-09-03

## Today’s Agenda

* Work in R with a familiar data set (the 15 question “quick survey” from Class 02)
* Open RStudio, load in some data and a template to write Quarto code
  + We’ll do a little typing into the template today, but just a little.
    - We’ll then look at the completed Quarto document.
    - We’ll also inspect and knit the Quarto file after all of the code is included.
  + Then we’ll start over again with the slides.

These slides walk through everything in that Quarto document.

## Today’s Files

From our [431-data page](https://github.com/THOMASELOVE/431-data), or our [Class 03 README (data folder)](https://github.com/THOMASELOVE/431-classes-2024/tree/main/class03/data), you should find:

* 431-first-r-template.qmd
* quick\_survey\_2024.csv

and

* 431-class03-all-code.qmd

in addition to the usual slide materials.

## Today’s Plan

We’re using Quarto to gather together into a single document:

* the code we build,
* text commenting on and reacting to that code, and
* the output of the analyses we build.

Everything in these slides is also going into our Quarto file.

## Load packages and set theme

library(janitor)  
library(patchwork)  
library(easystats)  
library(tidyverse)  
  
theme\_set(theme\_bw())  
knitr::opts\_chunk$set(comment = NA)

Loading packages in R is like opening up apps on your phone. We need to tell R that, in addition to the base functions available in the software, we also have other functions we want to use.

* Why are we loading these packages, in particular?

## On the tidyverse meta-package

* We will use the series of packages called the tidyverse in every Quarto file we create.
  + The tidyverse was developed (in part) by Hadley Wickham, Chief Scientist at Posit (makers of RStudio).
  + dplyr for data wrangling, cleaning and transformation
  + ggplot2 is our main visualization package
  + other tidyverse packages help import data, work with factors and other common activities.

## More on today’s packages

* The janitor package has some tools for examining, cleaning and tabulating data (including tabyl() and clean\_names()) that we’ll use regularly.
* The patchwork package will help us show multiple ggplots together.
* The easystats meta-package contains several other packages that will help us (especially) with building models and presenting our results.
* It’s helpful to load the tidyverse package last.

## Today’s Data

Our data come from the Quick 15-item Survey we did in Class 02 ([pdf in Class 02 README](https://github.com/THOMASELOVE/431-classes-2024/blob/main/class02/431_surveyhandout_1perstudent_2024-08-29.pdf)), which we’ve done (in various forms) since 2014.

* A copy of these data (in .csv format) is on our [431-data page](https://github.com/THOMASELOVE/431-data), and also linked on our [Class 03 README](https://github.com/THOMASELOVE/431-classes-2024/tree/main/class03/data).

We’ll tackle several exploratory questions of interest…

## Read in data from .csv file

quicksur\_raw <-   
 read\_csv("c03/data/quick\_survey\_2024.csv", show\_col\_types = FALSE) |>  
 janitor::clean\_names()

* The <- assignment arrow creates quicksur\_raw
* We use read\_csv to read in data from the c03/data subfolder of my R project directory which contains the quick\_survey\_2024.csv file from our [431-data page](https://github.com/THOMASELOVE/431-data).
* We use show\_col\_types = FALSE to suppress some unnecessary output describing the column types
* We use clean\_names() from the janitor package
* Note the use of the pipe |> to direct the information flow

## What is the result?

dim(quicksur\_raw)

[1] 603 23

quicksur\_raw

# A tibble: 603 × 23  
 student glasses english statsofar love\_htcm smoke h\_left h\_right handedness  
 <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
 1 202401 n y 5 175 1 16 1 -0.88  
 2 202402 y y 3 180 1 3 10 0.54  
 3 202403 y y 2 183 1 0 10 1   
 4 202404 y n 7 188 1 8 10 0.11  
 5 202405 n y 5 183 1 1 15 0.88  
 6 202406 y y 3 185 1 1 14 0.87  
 7 202407 n y 4 183 1 2 15 0.76  
 8 202408 n y 7 188 1 6 14 0.4   
 9 202409 y n 7 188 1 8 10 0.11  
10 202410 n y 5 191 1 1 12 0.85  
# ℹ 593 more rows  
# ℹ 14 more variables: statfuture <dbl>, haircut <dbl>, lecture <dbl>,  
# alone <dbl>, height\_in <dbl>, hand\_span <dbl>, favcolor <chr>,  
# lastsleep <dbl>, pulse <dbl>, year <dbl>, lovetrueage <dbl>,  
# lovetrueht <dbl>, sex <chr>, ageguess <dbl>

## A more detailed look?

glimpse(quicksur\_raw)

Rows: 603  
Columns: 23  
$ student <dbl> 202401, 202402, 202403, 202404, 202405, 202406, 202407, 20…  
$ glasses <chr> "n", "y", "y", "y", "n", "y", "n", "n", "y", "n", "y", "n"…  
$ english <chr> "y", "y", "y", "n", "y", "y", "y", "y", "n", "y", "y", "y"…  
$ statsofar <dbl> 5, 3, 2, 7, 5, 3, 4, 7, 7, 5, 6, 3, 5, 6, 3, 5, 5, 6, 4, 5…  
$ love\_htcm <dbl> 175, 180, 183, 188, 183, 185, 183, 188, 188, 191, 178, 180…  
$ smoke <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1…  
$ h\_left <dbl> 16, 3, 0, 8, 1, 1, 2, 6, 8, 1, 15, 16, 1, 1, 14, 19, 3, 1,…  
$ h\_right <dbl> 1, 10, 10, 10, 15, 14, 15, 14, 10, 12, 10, 10, 18, 15, 10,…  
$ handedness <dbl> -0.88, 0.54, 1.00, 0.11, 0.88, 0.87, 0.76, 0.40, 0.11, 0.8…  
$ statfuture <dbl> 7, 7, 7, 7, 7, 6, 5, 7, 7, 6, 7, 5, 7, 7, 6, 6, 6, 3, 7, 5…  
$ haircut <dbl> 50, 30, 80, 10, 45, 45, 130, 60, 40, 37, 12, 40, 176, 10, …  
$ lecture <dbl> 3, 5, 3, 2, 2, 3, 3, 3, 3, 2, 2, 3, 3, 3, 2, 5, 3, 2, 4, 4…  
$ alone <dbl> 3, 3, 2, 5, 3, 5, 5, 5, 5, 3, 2, 3, 4, 4, 2, 5, 4, 5, 3, 4…  
$ height\_in <dbl> 63.00, 69.00, 66.00, 68.50, 62.00, 64.00, 76.00, 72.00, 67…  
$ hand\_span <dbl> 15.0, 20.5, 20.5, 20.0, 14.5, 18.0, 22.0, 21.5, 22.0, 18.5…  
$ favcolor <chr> "purple", "blue", "green", "yellow", "yellow", "green", "g…  
$ lastsleep <dbl> 8.5, 10.0, 5.0, 5.0, 8.0, 8.0, 7.0, 6.0, 6.0, 5.5, 5.5, 5.…  
$ pulse <dbl> 75, 74, 90, 70, 80, 99, 70, 72, 64, 76, 80, 92, 88, 80, 88…  
$ year <dbl> 2024, 2024, 2024, 2024, 2024, 2024, 2024, 2024, 2024, 2024…  
$ lovetrueage <dbl> 57.5, 57.5, 57.5, 57.5, 57.5, 57.5, 57.5, 57.5, 57.5, 57.5…  
$ lovetrueht <dbl> 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188…  
$ sex <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA…  
$ ageguess <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA…

## Counting Categories

quicksur\_raw |> count(glasses)

# A tibble: 3 × 2  
 glasses n  
 <chr> <int>  
1 n 125  
2 y 222  
3 <NA> 256

quicksur\_raw |> count(glasses, english)

# A tibble: 9 × 3  
 glasses english n  
 <chr> <chr> <int>  
1 n n 29  
2 n y 95  
3 n <NA> 1  
4 y n 47  
5 y y 173  
6 y <NA> 2  
7 <NA> n 47  
8 <NA> y 205  
9 <NA> <NA> 4

## Favorite Color in 2024?

quicksur\_raw |>  
 filter(year == "2024") |>  
 tabyl(favcolor) |>  
 adorn\_pct\_formatting()

favcolor n percent valid\_percent  
 baby pink 2 3.6% 3.7%  
 black 1 1.8% 1.9%  
 blue 17 30.4% 31.5%  
 burgundy 1 1.8% 1.9%  
 green 17 30.4% 31.5%  
 hunter green 1 1.8% 1.9%  
 maroon 1 1.8% 1.9%  
 pink 1 1.8% 1.9%  
 pink-orange 1 1.8% 1.9%  
 purple 5 8.9% 9.3%  
 red 1 1.8% 1.9%  
 turquoise 1 1.8% 1.9%  
 yellow 5 8.9% 9.3%  
 <NA> 2 3.6% -

## Using summary() on Quantities

quicksur\_raw |>   
 select(love\_htcm, haircut, height\_in, lastsleep) |>  
 summary()

love\_htcm haircut height\_in lastsleep   
 Min. :152.0 Min. : 0.00 Min. :57.0 Min. : 2.000   
 1st Qu.:178.0 1st Qu.: 13.50 1st Qu.:64.0 1st Qu.: 6.000   
 Median :183.0 Median : 25.00 Median :67.0 Median : 7.000   
 Mean :181.9 Mean : 31.47 Mean :67.2 Mean : 6.909   
 3rd Qu.:185.0 3rd Qu.: 40.00 3rd Qu.:70.0 3rd Qu.: 8.000   
 Max. :196.0 Max. :250.00 Max. :77.5 Max. :12.000   
 NA's :385 NA's :12 NA's :9 NA's :8

* Numerical summaries (five quantiles, plus the mean) for:
  + your guess of my height (in cm), last haircut price ($), your height (in inches), and last night’s hours of sleep
* How many observations are available for these measures?

# Manage the data into an analytic tibble called qsdat

## Variables we’ll look at closely today

We’ll place these seven variables into our analytic data frame (tibble.)

* student: student identification (numerical code)
* year: indicates year when survey was taken (August)
* english: y = prefers to speak English, else n
* smoke: 1 = never smoker, 2 = quit, 3 = current
* pulse: pulse rate (beats per minute)
* height\_in: student’s height (in inches)
* haircut: price of student’s last haircut (in $)

## Select our variables

qsdat <- quicksur\_raw |>  
 select(student, year, english, smoke,   
 pulse, height\_in, haircut)

* The select() function chooses the variables (columns) we want to keep in our new tibble called qsdat.
* What should the result of this code look like?

## What do we have now?

dim(qsdat)

[1] 603 7

qsdat

# A tibble: 603 × 7  
 student year english smoke pulse height\_in haircut  
 <dbl> <dbl> <chr> <dbl> <dbl> <dbl> <dbl>  
 1 202401 2024 y 1 75 63 50  
 2 202402 2024 y 1 74 69 30  
 3 202403 2024 y 1 90 66 80  
 4 202404 2024 n 1 70 68.5 10  
 5 202405 2024 y 1 80 62 45  
 6 202406 2024 y 1 99 64 45  
 7 202407 2024 y 1 70 76 130  
 8 202408 2024 y 1 72 72 60  
 9 202409 2024 n 1 64 67 40  
10 202410 2024 y 1 76 60 37  
# ℹ 593 more rows

## Initial Numeric Summaries

* Is everything the “type” of variable it should be?
* Are we getting the summaries we want?

summary(qsdat)

student year english smoke   
 Min. :201401 Min. :2014 Length:603 Min. :1.000   
 1st Qu.:201661 1st Qu.:2016 Class :character 1st Qu.:1.000   
 Median :201948 Median :2019 Mode :character Median :1.000   
 Mean :201944 Mean :2019 Mean :1.088   
 3rd Qu.:202213 3rd Qu.:2022 3rd Qu.:1.000   
 Max. :202456 Max. :2024 Max. :3.000   
 NA's :2   
 pulse height\_in haircut   
 Min. : 30.00 Min. :57.0 Min. : 0.00   
 1st Qu.: 65.00 1st Qu.:64.0 1st Qu.: 13.50   
 Median : 72.00 Median :67.0 Median : 25.00   
 Mean : 73.93 Mean :67.2 Mean : 31.47   
 3rd Qu.: 80.00 3rd Qu.:70.0 3rd Qu.: 40.00   
 Max. :112.00 Max. :77.5 Max. :250.00   
 NA's :79 NA's :9 NA's :12

## What should we be seeing?

* Categorical variables should list the categories, with associated counts.
  + To accomplish this, the variable needs to be represented in R with a factor, rather than as a character or numeric variable.
* Quantitative variables should show the minimum, median, mean, maximum, etc.

names(qsdat)

[1] "student" "year" "english" "smoke" "pulse" "height\_in"  
[7] "haircut"

## Categorical variables as factors

We want the year and smoke information treated as categorical, rather than as quantitative, and the english information as a factor, too. Also, do we want to summarize the student ID codes?

* We use the mutate() function to help with this.

qsdat <- qsdat |>  
 mutate(year = as\_factor(year),  
 smoke = as\_factor(smoke),  
 english = as\_factor(english),  
 student = as.character(student))

* Note that it’s as\_factor() but as.character(). Sigh.

## Next step: Recheck the summaries and do range checks

* Do these summaries make sense?
* Are the minimum and maximum values appropriate?
* How much missingness are we to deal with?

## Now, how’s our summary?

summary(qsdat)

student year english smoke pulse   
 Length:603 2020 : 67 y :473 1 :558 Min. : 30.00   
 Class :character 2016 : 64 n :123 2 : 33 1st Qu.: 65.00   
 Mode :character 2019 : 61 NA's: 7 3 : 10 Median : 72.00   
 2021 : 58 NA's: 2 Mean : 73.93   
 2024 : 56 3rd Qu.: 80.00   
 2022 : 54 Max. :112.00   
 (Other):243 NA's :79   
 height\_in haircut   
 Min. :57.0 Min. : 0.00   
 1st Qu.:64.0 1st Qu.: 13.50   
 Median :67.0 Median : 25.00   
 Mean :67.2 Mean : 31.47   
 3rd Qu.:70.0 3rd Qu.: 40.00   
 Max. :77.5 Max. :250.00   
 NA's :9 NA's :12

* Some things to look for appear on the next slide.

## What to look for…

* Are we getting counts for all variables that are categorical?
  + Do the category levels make sense?
* Are we getting means and medians for all variables that are quantities?
  + Do the minimum and maximum values make sense for each of these quantities?
* Which variables have missing data, as indicated by NA's?

## The summary for year is an issue

* Just to fill in the gap left by the summary() result, how many students responded each year?

qsdat |> tabyl(year) |> adorn\_totals() |> adorn\_pct\_formatting()

year n percent  
 2014 42 7.0%  
 2015 49 8.1%  
 2016 64 10.6%  
 2017 48 8.0%  
 2018 51 8.5%  
 2019 61 10.1%  
 2020 67 11.1%  
 2021 58 9.6%  
 2022 54 9.0%  
 2023 53 8.8%  
 2024 56 9.3%  
 Total 603 100.0%

## Five Questions of (some) Interest

1. What is the distribution of pulse rates among students in 431 since 2014?
2. Does the distribution of student heights change materially over time?
3. Is the Normal distribution a good model for student heights? How about student haircut prices?
4. Do taller people appear to have paid less for their most recent haircut?
5. Do students have a more substantial tobacco history if they prefer to speak English or a language other than English?

## Question 1

What is the distribution of pulse rates among students in 431 since 2014?

qsdat |> tabyl(pulse)

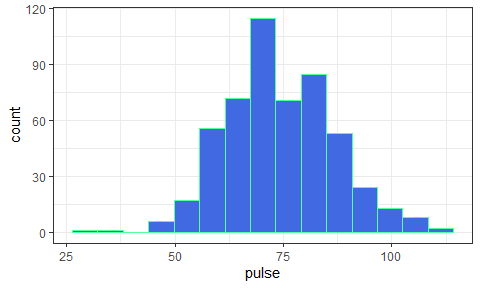
pulse n percent valid\_percent  
 30 1 0.001658375 0.001908397  
 33 1 0.001658375 0.001908397  
 44 1 0.001658375 0.001908397  
 46 1 0.001658375 0.001908397  
 48 4 0.006633499 0.007633588  
 50 4 0.006633499 0.007633588  
 52 5 0.008291874 0.009541985  
 54 6 0.009950249 0.011450382  
 55 2 0.003316750 0.003816794  
 56 12 0.019900498 0.022900763  
 57 1 0.001658375 0.001908397  
 58 5 0.008291874 0.009541985  
 59 1 0.001658375 0.001908397  
 60 37 0.061359867 0.070610687  
 62 11 0.018242123 0.020992366  
 63 2 0.003316750 0.003816794  
 64 32 0.053067993 0.061068702  
 65 7 0.011608624 0.013358779  
 66 20 0.033167496 0.038167939  
 68 42 0.069651741 0.080152672  
 69 2 0.003316750 0.003816794  
 70 32 0.053067993 0.061068702  
 71 1 0.001658375 0.001908397  
 72 38 0.063018242 0.072519084  
 74 21 0.034825871 0.040076336  
 75 8 0.013266998 0.015267176  
 76 22 0.036484245 0.041984733  
 77 2 0.003316750 0.003816794  
 78 16 0.026533997 0.030534351  
 79 2 0.003316750 0.003816794  
 80 56 0.092868988 0.106870229  
 81 1 0.001658375 0.001908397  
 82 7 0.011608624 0.013358779  
 84 21 0.034825871 0.040076336  
 85 4 0.006633499 0.007633588  
 86 12 0.019900498 0.022900763  
 88 23 0.038142620 0.043893130  
 89 1 0.001658375 0.001908397  
 90 13 0.021558872 0.024809160  
 91 1 0.001658375 0.001908397  
 92 12 0.019900498 0.022900763  
 94 5 0.008291874 0.009541985  
 95 1 0.001658375 0.001908397  
 96 5 0.008291874 0.009541985  
 98 4 0.006633499 0.007633588  
 99 1 0.001658375 0.001908397  
 100 8 0.013266998 0.015267176  
 104 6 0.009950249 0.011450382  
 106 1 0.001658375 0.001908397  
 108 1 0.001658375 0.001908397  
 110 1 0.001658375 0.001908397  
 112 1 0.001658375 0.001908397  
 NA 79 0.131011609 NA

## Histogram, first try

* What is the distribution of student pulse rates?

ggplot(data = qsdat, aes(x = pulse)) +  
 geom\_histogram(bins = 15, fill = "royalblue", col = "seagreen1")

Warning: Removed 79 rows containing non-finite outside the scale range  
(`stat\_bin()`).



## Describing the Pulse Rates

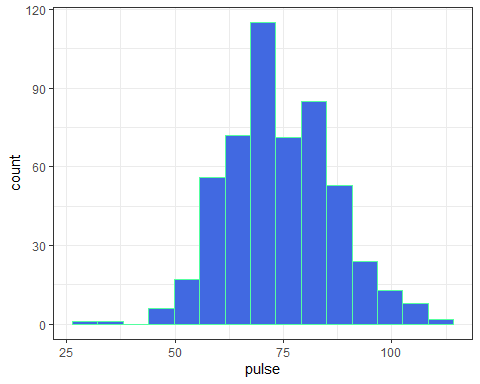
How might we describe this distribution?

* What is the center?
* How much of a range around that center do we see? How spread out are the data?
* What is the shape of this distribution?
  + Is it symmetric, or is it skewed to the left or to the right?

(Histogram is replotted on the next slide)

## Histogram (with warning suppressed)

ggplot(data = qsdat, aes(x = pulse)) +  
 geom\_histogram(bins = 15, fill = "royalblue", col = "seagreen1")



## Some Key Numerical Summaries

qsdat |> select(pulse) |> summary()

pulse   
 Min. : 30.00   
 1st Qu.: 65.00   
 Median : 72.00   
 Mean : 73.93   
 3rd Qu.: 80.00   
 Max. :112.00   
 NA's :79

length(qsdat$pulse)

[1] 603

sd(qsdat$pulse, na.rm = TRUE)

[1] 12.43657

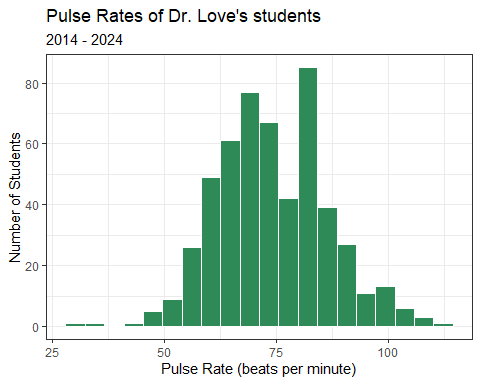
mad(qsdat$pulse, na.rm = TRUE)

[1] 11.8608

* Do these summaries help us describe the data?

## Histogram, version 2

dat1 <- qsdat |>  
 filter(complete.cases(pulse))  
  
ggplot(data = dat1, aes(x = pulse)) +  
 geom\_histogram(fill = "seagreen", col = "white", bins = 20) +  
 labs(title = "Pulse Rates of Dr. Love's students",  
 subtitle = "2014 - 2024",  
 y = "Number of Students",  
 x = "Pulse Rate (beats per minute)")



* How did we deal with missing data?
* How did we add axis labels and titles to the plot?
* What is the distinction between fill and col?
* How many bins should we use?

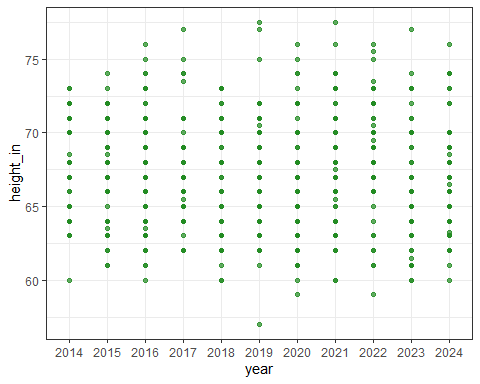
## Question 2

Does the distribution of student heights change over time?

(Plot shown on next slide)

ggplot(qsdat, aes(x = year, y = height\_in)) +  
 geom\_point(alpha = 0.7, color = "forestgreen")

Warning: Removed 9 rows containing missing values or values outside the scale range  
(`geom\_point()`).



## Yearly Five-Number Summaries

qsdat |>  
 filter(complete.cases(height\_in)) |>  
 group\_by(year) |>  
 summarize(n = n(), min = min(height\_in), q25 = quantile(height\_in, 0.25),  
 median = median(height\_in), q75 = quantile(height\_in, 0.75),  
 max = max(height\_in))

* What should this produce? (Results on next slide)

## Yearly Five-Number Summaries

qsdat |>  
 filter(complete.cases(height\_in)) |>  
 group\_by(year) |>  
 summarize(n = n(), min = min(height\_in), q25 = quantile(height\_in, 0.25),  
 median = median(height\_in), q75 = quantile(height\_in, 0.75),  
 max = max(height\_in))

# A tibble: 11 × 7  
 year n min q25 median q75 max  
 <fct> <int> <dbl> <dbl> <dbl> <dbl> <dbl>  
 1 2014 40 60 64.8 68 71 73   
 2 2015 49 61 65 68 70 74   
 3 2016 64 60 64 67 70 76   
 4 2017 48 62 65 67 69 77   
 5 2018 51 60 63 66 70 73   
 6 2019 60 57 65 68 70 77.5  
 7 2020 66 59 63 66 69.8 76   
 8 2021 55 60 64.5 67.5 71 77.5  
 9 2022 54 59 66 68.5 70.4 76   
10 2023 53 60 64 65 69 77   
11 2024 54 60 64 67 69.8 76

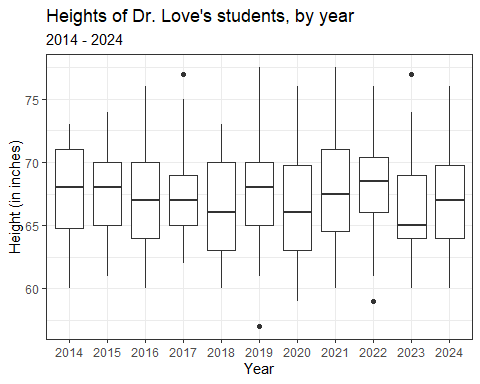
* Do these summaries change materially over time?
* What are these summaries, specifically?

## Five-Number Summary

* Key summaries based on percentiles / quantiles
  + minimum = 0th, maximum = 100th, median = 50th
  + quartiles (25th, 50th and 75th percentiles)
  + Range is maximum - minimum
  + IQR (inter-quartile range) is 75th - 25th percentile
* These summaries are generally more resistant to outliers than mean, standard deviation
* Form the elements of a boxplot (box-and-whisker plot)

## Boxplot of Heights by Year

dat1 <- qsdat |>  
 filter(complete.cases(height\_in))   
  
ggplot(data = dat1, aes(x = year, y = height\_in)) +  
 geom\_boxplot() +  
 labs(title = "Heights of Dr. Love's students, by year",  
 subtitle = "2014 - 2024", x = "Year", y = "Height (in inches)")



* How did we deal with missing data here?

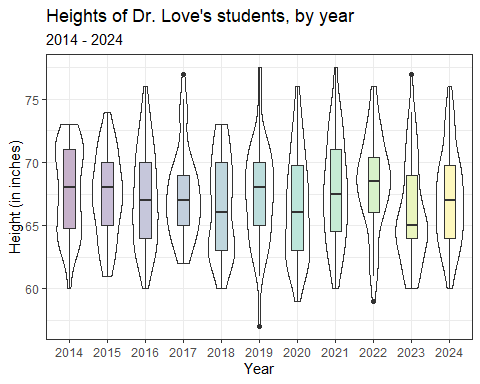
## Thinking about the Boxplot

* Box covers the middle half of the data (25th and 75th percentiles), and the solid line indicates the median
* Whiskers extend from the quartiles to the most extreme values that are not judged by **Tukey’s** “fences” method to be candidate outliers
  + Fences are drawn at 25th percentile - 1.5 IQR and 75th percentile + 1.5 IQR
* Are any values candidate outliers by this method?
* Was it important to change year to a factor earlier?

## Adding a Violin to the Boxplot

* When we’d like to better understand the shape of a distribution, we can amplify the boxplot.

dat1 <- qsdat |>  
 filter(complete.cases(height\_in))  
  
ggplot(data = dat1, aes(x = year, y = height\_in)) +  
 geom\_violin() +  
 geom\_boxplot(aes(fill = year), width = 0.3) +  
 guides(fill = "none") +  
 scale\_fill\_viridis\_d(alpha = 0.3) +  
 labs(title = "Heights of Dr. Love's students, by year",  
 subtitle = "2014 - 2024", x = "Year", y = "Height (in inches)")



## Boxplot with Violin

* How did we change the boxplot when we added the violin?
* What would happen if we added the boxplot first and the violin second?
* What does guides(fill = "none") do?
* What does scale\_fill\_viridis\_d(alpha = 0.3) do?

## Key Numerical Summaries

qsdat |>  
 filter(complete.cases(height\_in)) |>  
 group\_by(year) |>  
 summarize(n = n(), mean = mean(height\_in), sd = sd(height\_in),  
 median = median(height\_in), mad = mad(height\_in))

# A tibble: 11 × 6  
 year n mean sd median mad  
 <fct> <int> <dbl> <dbl> <dbl> <dbl>  
 1 2014 40 67.8 3.46 68 4.45  
 2 2015 49 67.3 3.32 68 2.97  
 3 2016 64 67.2 3.86 67 4.45  
 4 2017 48 67.4 3.46 67 2.97  
 5 2018 51 66.5 3.81 66 4.45  
 6 2019 60 67.4 3.83 68 4.45  
 7 2020 66 66.4 4.09 66 4.45  
 8 2021 55 67.8 4.13 67.5 5.19  
 9 2022 54 68.4 3.74 68.5 3.71  
10 2023 53 66.0 4.00 65 2.97  
11 2024 54 67.2 3.70 67 4.45

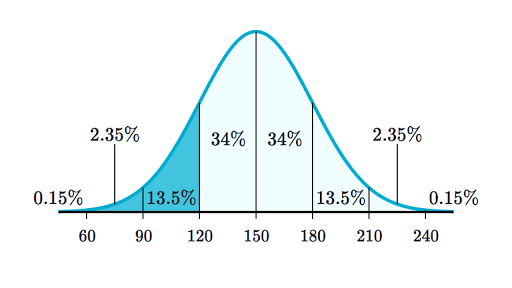
## Question 3

Are the data on student heights in 2024 well described by a Normal distribution?

* Can we use a mean and standard deviation to describe the center and spread of the data effectively?

## A Normal distribution

This is a Normal (or Gaussian) distribution with mean 150 and standard deviation 30.



* A Normal distribution is completely specified by its mean and standard deviation. The “bell shape” doesn’t change.

## Summarizing Quantitative Data

If the data followed a Normal model,

* we would be justified in using the sample **mean** to describe the center, and
* in using the sample **standard deviation** to describe the spread (variation.)

But it is often the case that these measures aren’t robust enough, because the data show meaningful skew (asymmetry), or the data have lighter or heavier tails than a Normal model would predict.

## The Empirical Rule for Approximately Normal Distributions

If the data followed a Normal distribution,

* approximately 68% of the data would be within 1 SD of the mean,
* approximately 95% of the data would be within 2 SD of the mean, while
* essentially all (99.7%) of the data would be within 3 SD of the mean.

## Empirical Rule & 2024 Student Heights

dat2 <- qsdat |>  
 filter(complete.cases(height\_in),  
 year == "2024")  
  
describe\_distribution(dat2$height\_in)

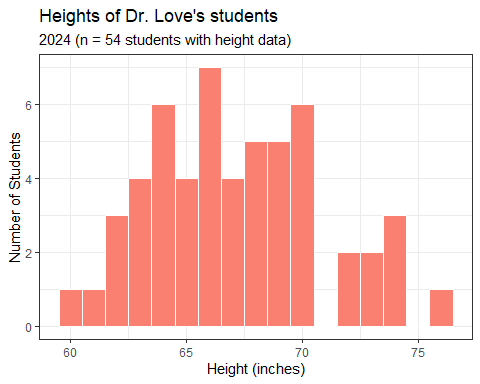
Mean | SD | IQR | Range | Skewness | Kurtosis | n | n\_Missing  
--------------------------------------------------------------------------  
67.17 | 3.70 | 6 | [60.00, 76.00] | 0.35 | -0.42 | 54 | 0

In 2024, we had 54 students whose height\_in was available, with mean 67.2 inches (170.7 cm) and standard deviation 3.7 inches (9.4 cm).

Consider a picture of the data…

## Histogram of 2024 Student Heights

dat2 <- qsdat |>  
 filter(complete.cases(height\_in)) |>  
 filter(year == "2024")  
  
ggplot(data = dat2, aes(x = height\_in)) +  
 geom\_histogram(fill = "salmon", col = "white", binwidth = 1) +  
 labs(title = "Heights of Dr. Love's students",  
 subtitle = "2024 (n = 54 students with height data)",  
 y = "Number of Students", x = "Height (inches)")



* How did we use the two filter() statements?
* Why might I have changed from specifying bins to binwidth here?

## Checking the 1-SD Empirical Rule

* Of the 54 students in 2024 with heights, how many were within 1 SD of the mean?
  + Mean = 67.2, SD = 3.7.
  + 67.2 - 3.7 = 63.5 inches and 67.2 + 3.7 = 70.9 inches

qsdat |> filter(complete.cases(height\_in)) |>  
 filter(year == "2024") |>  
 count(height\_in >= 63.5 & height\_in <= 70.9)

# A tibble: 2 × 2  
 `height\_in >= 63.5 & height\_in <= 70.9` n  
 <lgl> <int>  
1 FALSE 17  
2 TRUE 37

37/(37+17)

[1] 0.6851852

## 2-SD Empirical Rule

* How many of the 54 height\_in values gathered in 2024 were between 67.2 - 2(3.7) = 59.8 and 67.2 + 2(3.7) = 74.6 inches?

qsdat |> filter(complete.cases(height\_in)) |>  
 filter(year == "2024") |>  
 count(height\_in >= 59.8 & height\_in <= 74.6)

# A tibble: 2 × 2  
 `height\_in >= 59.8 & height\_in <= 74.6` n  
 <lgl> <int>  
1 FALSE 1  
2 TRUE 53

53/(53+1)

[1] 0.9814815

## 3-SD Empirical Rule

* How many of the 54 height\_in values gathered in 2024 were between 67.2 - 3(3.7) = 56.1 and 67.2 + 3(3.7) = 78.3 inches?

qsdat |> filter(complete.cases(height\_in)) |>  
 filter(year == "2024") |>  
 count(height\_in >= 56.1 & height\_in <= 78.3)

# A tibble: 1 × 2  
 `height\_in >= 56.1 & height\_in <= 78.3` n  
 <lgl> <int>  
1 TRUE 54

54/(54+0)

[1] 1

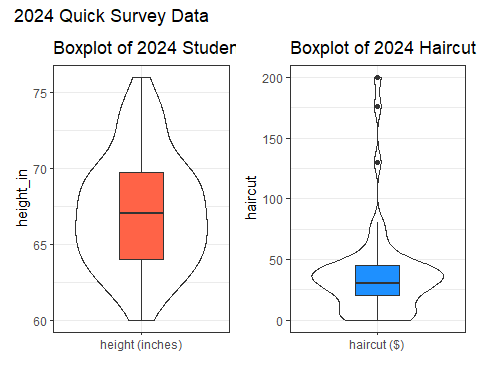
## Empirical Rule Table for 2024 data

* = sample mean, = sample SD
* For height\_in: = 54 with data,
* For haircut: = 53 with data,

| Range | “Normal” | height\_in | haircut |
| --- | --- | --- | --- |
|  | ~68% | = 68.5% | = 92.4% |
|  | ~95% | = 98.1% | = 94.3% |
|  | ~99.7% | = 100% | = 96.2% |

## Boxplots of Height and Haircut Prices

dat2 <- qsdat |> filter(complete.cases(height\_in), year == "2024")  
  
p2 <- ggplot(data = dat2, aes(x = "height (inches)", y = height\_in)) +  
 geom\_violin() + geom\_boxplot(width = 0.3, fill = "tomato") +  
 labs(title = "Boxplot of 2024 Student Heights", x = "")  
  
dat3 <- qsdat |> filter(complete.cases(haircut), year == "2024")  
  
p3 <- ggplot(data = dat3, aes(x = "haircut ($)", y = haircut)) +  
 geom\_violin() + geom\_boxplot(width = 0.3, fill = "dodgerblue") +  
 labs(title = "Boxplot of 2024 Haircut Prices", x = "")  
  
p2 + p3 +   
 plot\_annotation(title = "2024 Quick Survey Data")



* What is width = 0.3 doing? How about the x options?
* What am I doing with p2 + p3 + plot\_annotation?
* What should this look like?

## Mean/SD vs. Median/MAD

If the data are approximately Normally distributed (like height\_in and pulse) we can safely use the sample mean and standard deviation as summaries. If not “Normal”, then …

* The median is a more robust summary of the center.
* For spread, try the median absolute deviation (scaled to equal the standard deviation if the data are Normal)

| Measure | Median | Mean | MAD | Std. Dev. |
| --- | --- | --- | --- | --- |
| height\_in | 67 | 67.2 | 4.5 | 3.7 |
| haircut | 30 | 37.9 | 22.2 | 38.2 |

## Question 4

Do tall people pay less for haircuts?

* Why might we think that they do, before we see the data?
* Convert our student heights from inches to centimeters…

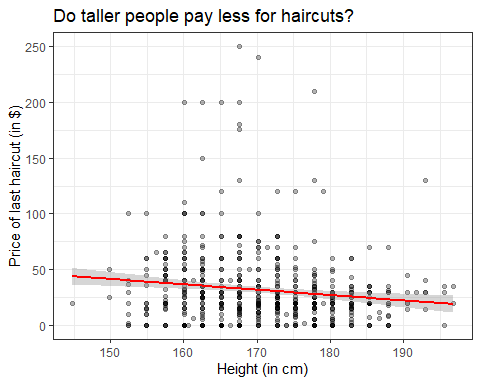
qsdat <- qsdat |> mutate(height\_cm = height\_in \* 2.54)  
  
qsdat |> select(student, height\_in, height\_cm) |> head()

# A tibble: 6 × 3  
 student height\_in height\_cm  
 <chr> <dbl> <dbl>  
1 202401 63 160.  
2 202402 69 175.  
3 202403 66 168.  
4 202404 68.5 174.  
5 202405 62 157.  
6 202406 64 163.

## A First Scatterplot

* We’ll include the straight line from a linear model, in red.

dat4 <- qsdat |> filter(complete.cases(height\_cm, haircut))   
  
ggplot(dat4, aes(x = height\_cm, y = haircut)) +  
 geom\_point(alpha = 0.3) +   
 geom\_smooth(method = "lm", col = "red",  
 formula = y ~ x, se = TRUE) +  
 labs(x = "Height (in cm)",  
 y = "Price of last haircut (in $)",  
 title = "Do taller people pay less for haircuts?")



## What is the (Pearson) correlation of height and haircut price?

dat4 <- qsdat |> filter(complete.cases(height\_cm, haircut))   
  
dat4 |>   
 select(height\_in, height\_cm, haircut) |>  
 cor()

height\_in height\_cm haircut  
height\_in 1.0000000 1.0000000 -0.1394674  
height\_cm 1.0000000 1.0000000 -0.1394674  
haircut -0.1394674 -0.1394674 1.0000000

## What is the straight line regression model?

dat4 <- qsdat |> filter(complete.cases(height\_cm, haircut))   
  
mod1 <- lm(haircut ~ height\_cm, data = dat4)  
  
mod1

Call:  
lm(formula = haircut ~ height\_cm, data = dat4)  
  
Coefficients:  
(Intercept) height\_cm   
 113.46 -0.48

## Summarizing our model mod1

model\_parameters(mod1)

Parameter | Coefficient | SE | 95% CI | t(587) | p  
---------------------------------------------------------------------  
(Intercept) | 113.46 | 24.06 | [66.22, 160.71] | 4.72 | < .001  
height cm | -0.48 | 0.14 | [-0.76, -0.20] | -3.41 | < .001

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

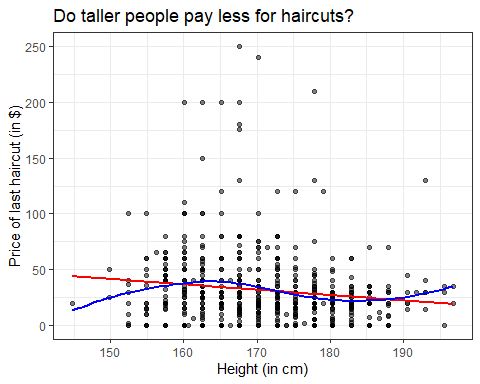
Regression Equation is:

Our predicted haircut price for someone who is 170 cm (about 5 feet 7 inches) tall is…

$$
haircut = 113.46 - 0.48 (170) = $31.86
$$

## lm fit vs. loess smooth curve?

dat4 <- qsdat |> filter(complete.cases(height\_cm, haircut))   
  
ggplot(dat4, aes(x = height\_cm, y = haircut)) +  
 geom\_point(alpha = 0.5) +   
 geom\_smooth(method = "lm", col = "red",  
 formula = y ~ x, se = FALSE) +  
 geom\_smooth(method = "loess", col = "blue",  
 formula = y ~ x, se = FALSE) +  
 labs(x = "Height (in cm)",  
 y = "Price of last haircut (in $)",  
 title = "Do taller people pay less for haircuts?")



* Does a linear model appear to fit these data well?
* Do taller people pay less for their haircuts?

## Question 5

Do students have a more substantial tobacco history if they prefer to speak English or a language other than English?

qsdat |> count(english, smoke)

# A tibble: 9 × 3  
 english smoke n  
 <fct> <fct> <int>  
1 y 1 436  
2 y 2 30  
3 y 3 6  
4 y <NA> 1  
5 n 1 116  
6 n 2 3  
7 n 3 4  
8 <NA> 1 6  
9 <NA> <NA> 1

smoke codes (tobacco use): 1 = Never, 2 = Former, 3 = Current

## Restrict ourselves to 2024 data

* Do students in the 2024 class have a more substantial history of tobacco use if they prefer to speak a language other than English?

dat5 <- qsdat |>   
 filter(year == "2024") |>  
 filter(complete.cases(english, smoke)) |>  
 select(student, year, english, smoke)  
summary(dat5)

student year english smoke   
 Length:54 2024 :54 y:46 1:51   
 Class :character 2014 : 0 n: 8 2: 2   
 Mode :character 2015 : 0 3: 1   
 2016 : 0   
 2017 : 0   
 2018 : 0   
 (Other): 0

## Tabulating the categorical variables individually

dat5 |> tabyl(english)

english n percent  
 y 46 0.8518519  
 n 8 0.1481481

dat5 |> tabyl(smoke) |> adorn\_pct\_formatting()

smoke n percent  
 1 51 94.4%  
 2 2 3.7%  
 3 1 1.9%

* What does adorn\_pct\_formatting() do?

## Cross-Classification (2 rows 3 columns)

dat5 |> tabyl(english, smoke)

english 1 2 3  
 y 43 2 1  
 n 8 0 0

## Recode the smoke levels to more meaningful names in tobacco

dat5 <- dat5 |>   
 mutate(tobacco = fct\_recode(smoke,   
 "Never" = "1", "Quit" = "2", "Current" = "3"))

### Check our work?

dat5 |> count(smoke, tobacco)

# A tibble: 3 × 3  
 smoke tobacco n  
 <fct> <fct> <int>  
1 1 Never 51  
2 2 Quit 2  
3 3 Current 1

* Everyone with smoke = 1 has tobacco as Never, etc.

## Restate the cross-tabulation

Now we’ll use this new variable, and this time, add row and column totals.

dat5 |> tabyl(english, tobacco) |>   
 adorn\_totals(where = c("row", "col"))

english Never Quit Current Total  
 y 43 2 1 46  
 n 8 0 0 8  
 Total 51 2 1 54

* What can we conclude about this association?

## How about in 2014-2024?

dat6 <- qsdat |>   
 filter(complete.cases(english, smoke)) |>  
 mutate(tobacco = fct\_recode(smoke,   
 "Never" = "1", "Quit" = "2", "Current" = "3"))  
  
dat6 |>   
 tabyl(english, tobacco) |>   
 adorn\_totals(where = c("row", "col"))

english Never Quit Current Total  
 y 436 30 6 472  
 n 116 3 4 123  
 Total 552 33 10 595

* Now, what is your conclusion?

## Next Time

Analyzing a (small) health dataset

## Cleaning up the temporary objects

rm(mod1, p2, p3, dat1, dat2, dat2, dat3, dat4, dat5, dat6)

Warning in rm(mod1, p2, p3, dat1, dat2, dat2, dat3, dat4, dat5, dat6): object  
'dat2' not found

## this just leaves  
## qsdat and quicksur\_raw in my Global Environment

## 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:  
 askpass\_1.2.0 backports\_1.5.0 base64enc\_0.1.3   
 bayestestR\_0.14.0 bit\_4.0.5 bit64\_4.0.5   
 blob\_1.2.4 broom\_1.0.6 bslib\_0.8.0   
 cachem\_1.1.0 callr\_3.7.6 cellranger\_1.1.0   
 cli\_3.6.3 clipr\_0.8.0 coda\_0.19-4.1   
 codetools\_0.2-20 colorspace\_2.1-1 compiler\_4.4.1   
 conflicted\_1.2.0 correlation\_0.8.5 cpp11\_0.5.0   
 crayon\_1.5.3 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 digest\_0.6.37 dplyr\_1.1.4   
 dtplyr\_1.3.1 easystats\_0.7.3 effectsize\_0.8.9   
 emmeans\_1.10.4 estimability\_1.5.1 evaluate\_0.24.0   
 fansi\_1.0.6 farver\_2.1.2 fastmap\_1.2.0   
 fontawesome\_0.5.2 forcats\_1.0.0 fs\_1.6.4   
 gargle\_1.5.2 generics\_0.1.3 ggplot2\_3.5.1   
 glue\_1.7.0 googledrive\_2.1.1 googlesheets4\_1.1.1  
 graphics\_4.4.1 grDevices\_4.4.1 grid\_4.4.1   
 gtable\_0.3.5 haven\_2.5.4 highr\_0.11   
 hms\_1.1.3 htmltools\_0.5.8.1 httr\_1.4.7   
 ids\_1.0.1 insight\_0.20.4 isoband\_0.2.7   
 janitor\_2.2.0 jquerylib\_0.1.4 jsonlite\_1.8.8   
 knitr\_1.48 labeling\_0.4.3 lattice\_0.22-6   
 lifecycle\_1.0.4 lubridate\_1.9.3 magrittr\_2.0.3   
 MASS\_7.3-61 Matrix\_1.7-0 memoise\_2.0.1   
 methods\_4.4.1 mgcv\_1.9-1 mime\_0.12   
 modelbased\_0.8.8 modelr\_0.1.11 multcomp\_1.4-26   
 munsell\_0.5.1 mvtnorm\_1.3-0 nlme\_3.1-164   
 numDeriv\_2016.8.1.1 openssl\_2.2.1 parallel\_4.4.1   
 parameters\_0.22.1 patchwork\_1.2.0 performance\_0.12.3   
 pillar\_1.9.0 pkgconfig\_2.0.3 prettyunits\_1.2.0   
 processx\_3.8.4 progress\_1.2.3 ps\_1.7.7   
 purrr\_1.0.2 R6\_2.5.1 ragg\_1.3.2   
 rappdirs\_0.3.3 RColorBrewer\_1.1.3 readr\_2.1.5   
 readxl\_1.4.3 rematch\_2.0.0 rematch2\_2.1.2   
 report\_0.5.9 reprex\_2.1.1 rlang\_1.1.4   
 rmarkdown\_2.28 rstudioapi\_0.16.0 rvest\_1.0.4   
 sandwich\_3.1-0 sass\_0.4.9 scales\_1.3.0   
 see\_0.8.5 selectr\_0.4.2 snakecase\_0.11.1   
 splines\_4.4.1 stats\_4.4.1 stringi\_1.8.4   
 stringr\_1.5.1 survival\_3.7-0 sys\_3.4.2   
 systemfonts\_1.1.0 textshaping\_0.4.0 TH.data\_1.1-2   
 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 utf8\_1.2.4   
 utils\_4.4.1 uuid\_1.2.1 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   
 yaml\_2.3.10 zoo\_1.8-12