431 Class 04

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

2024-09-05

## Today’s Agenda

* Build on where we left off in Class 03 in studying the 15-item Quick Survey.
* Demonstrate techniques and code that might be helpful for Lab 1, due next Wednesday 2024-09-11 at noon.

## Load packages and set theme

library(janitor)  
library(patchwork)  
library(rstanarm) ## new today  
library(easystats)  
library(tidyverse)  
  
theme\_set(theme\_bw())  
knitr::opts\_chunk$set(comment = NA)  
  
source("c04/data/Love-431.R") ## new today

We’re sourcing in the Love-431.R R script, which contains a function we’ll use.

## Ingest, manage data

Read in data from .csv (comma-separated version) file.

quicksur\_raw <-   
 read\_csv("c04/data/quick\_survey\_2024.csv", show\_col\_types = FALSE) |>  
 janitor::clean\_names()

Select variables, make categories into factors

qsdat <- quicksur\_raw |>  
 select(student, year, english, smoke, pulse, height\_in, haircut) |>  
 mutate(year = as\_factor(year),  
 smoke = as\_factor(smoke),  
 english = as\_factor(english),  
 student = as.character(student))

## Resulting tibble

qsdat

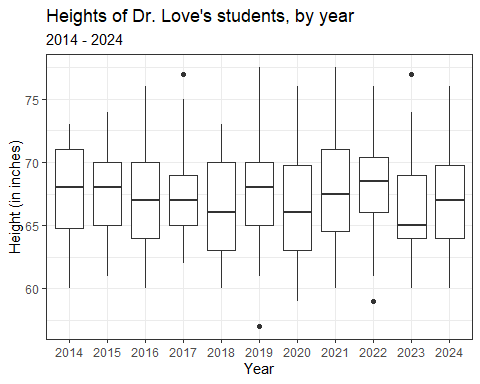
# A tibble: 603 × 7  
 student year english smoke pulse height\_in haircut  
 <chr> <fct> <fct> <fct> <dbl> <dbl> <dbl>  
 1 202401 2024 y 1 75 63 50  
 2 202402 2024 y 1 74 69 30  
 3 202403 2024 y 1 90 66 80  
 4 202404 2024 n 1 70 68.5 10  
 5 202405 2024 y 1 80 62 45  
 6 202406 2024 y 1 99 64 45  
 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

## Today’s Questions

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

## Are there differences in student height 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)")

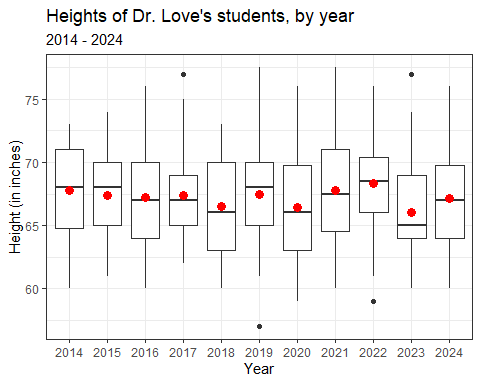


## What does the boxplot show?

* Median = thick line inside the box
* Quartiles (25th and 75th percentiles) form the edges of the box
* Whiskers extend out to the most extreme non-outliers
* Candidate outliers identified through Tukey’s fences…
  + Q1 - 1.5 IQR and Q3 + 1.5 IQR, where IQR = inter-quartile range is Q3 - Q1
  + IQR = length of the “box” in the boxplot

## Adding Means to the Boxplot

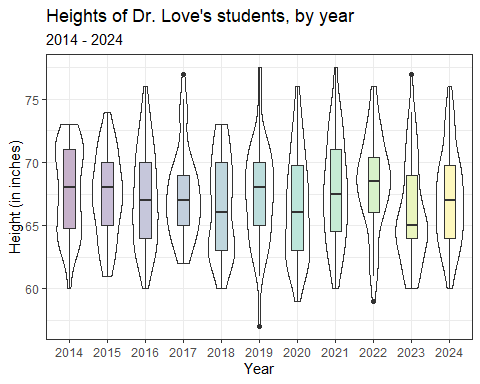
dat1 <- qsdat |>  
 filter(complete.cases(height\_in))   
  
ggplot(data = dat1, aes(x = year, y = height\_in)) +  
 geom\_boxplot() +  
 stat\_summary(fun = mean, geom = "point",   
 shape = 16, size = 3, col = "red") +  
 labs(title = "Heights of Dr. Love's students, by year",  
 subtitle = "2014 - 2024", x = "Year", y = "Height (in inches)")



## 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?

## Summary Statistics for Student Height, 2014-2024

qsdat |> describe\_distribution(height\_in)

Variable | Mean | SD | IQR | Range | Skewness | Kurtosis | n | n\_Missing  
---------------------------------------------------------------------------------------  
height\_in | 67.20 | 3.82 | 6 | [57.00, 77.50] | 0.20 | -0.53 | 594 | 9

qsdat |> describe\_distribution(height\_in, centrality = "median")

Variable | Median | MAD | IQR | Range | Skewness | Kurtosis | n | n\_Missing  
----------------------------------------------------------------------------------------  
height\_in | 67 | 4.45 | 6 | [57.00, 77.50] | 0.20 | -0.53 | 594 | 9

## Use my lovedist() function from the Love-431.R script…

qsdat |> reframe(lovedist(height\_in)) |> print\_md(digits = 2)

| n | miss | mean | sd | med | mad | min | q25 | q75 | max |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 603 | 9 | 67.20 | 3.82 | 67 | 4.45 | 57 | 64 | 70 | 77.50 |

* n = sample size, miss = missing values
* mean = arithmetic average (sum of values / number of values)
* sd = standard deviation (measures dispersion)
* med = median (50th percentile)
* mad = median absolute deviation (scaled measure of dispersion)
* min, q25, q75, max = 0th, 25th, 75th and 100th percentile

## lovedist() results by year

qsdat |> group\_by(year) |> reframe(lovedist(height\_in))

# A tibble: 11 × 11  
 year n miss mean sd med mad min q25 q75 max  
 <fct> <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
 1 2014 42 2 67.8 3.46 68 4.45 60 64.8 71 73   
 2 2015 49 0 67.3 3.32 68 2.97 61 65 70 74   
 3 2016 64 0 67.2 3.86 67 4.45 60 64 70 76   
 4 2017 48 0 67.4 3.46 67 2.97 62 65 69 77   
 5 2018 51 0 66.5 3.81 66 4.45 60 63 70 73   
 6 2019 61 1 67.4 3.83 68 4.45 57 65 70 77.5  
 7 2020 67 1 66.4 4.09 66 4.45 59 63 69.8 76   
 8 2021 58 3 67.8 4.13 67.5 5.19 60 64.5 71 77.5  
 9 2022 54 0 68.4 3.74 68.5 3.71 59 66 70.4 76   
10 2023 53 0 66.0 4.00 65 2.97 60 64 69 77   
11 2024 56 2 67.2 3.70 67 4.45 60 64 69.8 76

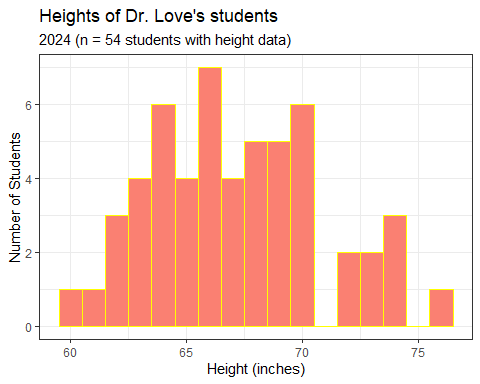
## Question 3

Are the data on student heights in 2024 well described by a Normal distribution? How about haircut prices?

* Can we use a mean and standard deviation to describe the center and spread of the data effectively?
* Can we estimate a population mean accurately using our sample of data?

## Histogram of 2024 Student Heights

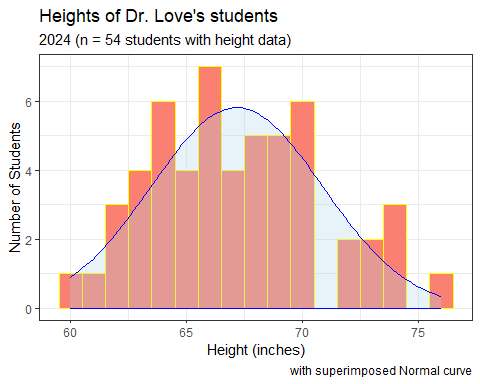
dat2 <- qsdat |>  
 filter(complete.cases(height\_in)) |>  
 filter(year == "2024")  
  
ggplot(data = dat2, aes(x = height\_in)) +  
 geom\_histogram(fill = "salmon", col = "yellow", binwidth = 1) +  
 labs(title = "Heights of Dr. Love's students",  
 subtitle = "2024 (n = 54 students with height data)",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

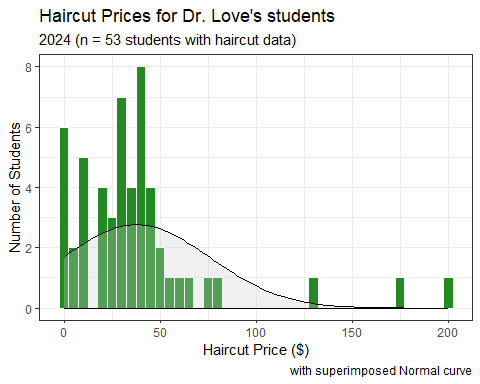
## Add Normal curve to our histogram?

bw = 1 # specify width of bins in histogram  
  
ggplot(dat2, aes(height\_in)) +  
 geom\_histogram(binwidth = bw,   
 fill = "salmon", col = "yellow") +  
 stat\_function(fun = function(x)  
 dnorm(x, mean = mean(dat2$height\_in, na.rm = TRUE),  
 sd = sd(dat2$height\_in, na.rm = TRUE)) \*  
 length(dat2$height\_in) \* bw,   
 geom = "area", alpha = 0.3, fill = "lightblue", col = "blue") +  
 labs(title = "Heights of Dr. Love's students",  
 subtitle = "2024 (n = 54 students with height data)",  
 y = "Number of Students", x = "Height (inches)",  
 caption = "with superimposed Normal curve")



## Histogram of Haircut Prices (2024)

dat3 <- qsdat |> filter(complete.cases(haircut), year == "2024")  
  
bw = 5 # specify width of bins in histogram  
  
ggplot(dat3, aes(haircut)) +  
 geom\_histogram(binwidth = bw,   
 fill = "forestgreen", col = "white") +  
 stat\_function(fun = function(x)  
 dnorm(x, mean = mean(dat3$haircut, na.rm = TRUE),  
 sd = sd(dat3$haircut, na.rm = TRUE)) \*  
 length(dat3$haircut) \* bw,   
 geom = "area", alpha = 0.3, fill = "grey80", col = "black") +  
 labs(title = "Haircut Prices for Dr. Love's students",  
 subtitle = "2024 (n = 53 students with haircut data)",  
 y = "Number of Students", x = "Haircut Price ($)",  
 caption = "with superimposed Normal curve")



## 2024 Student Data: Key Summaries

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

# 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 54 0 67.2 3.70 67 4.45 60 64 69.8 76

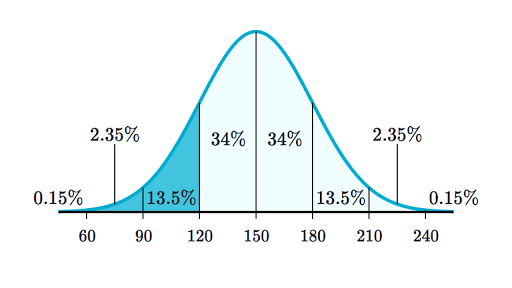
dat3 <- qsdat |> filter(complete.cases(haircut), year == "2024")  
  
dat3 |> reframe(lovedist(haircut))

# 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 53 0 37.9 38.2 30 22.2 0 20 45 200

* Which measure’s center is reasonably well summarized by its mean?

## 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.

## 2024 Student Heights

dat2 <- qsdat |> filter(complete.cases(height\_in), year == "2024")  
  
nrow(dat2); mean(dat2$height\_in); sd(dat2$height\_in)

[1] 54

[1] 67.1713

[1] 3.703588

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).

## 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), 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

## Empirical Rule Table for 2024 data

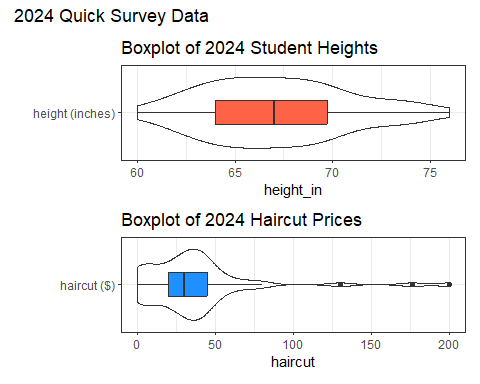
* = sample mean, = sample SD
* For height\_in: = 54 with data,
* For haircut: = 53 with data,

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

|  |
| --- |
| Note |
| I calculated these fractions offline. Those calculations aren’t included in the R code for these slides. |

## Boxplots of Height and Haircut Prices

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



* What is width = 0.3 doing? How about the y 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 |

## Making Estimates from our Sample

One estimate of the average height of all students who study at CWRU could come from our samples of data from my classes.

dat1 <- qsdat |> filter(complete.cases(height\_in))  
  
dat1 |> reframe(lovedist(height\_in))

# A tibble: 1 × 10  
 n miss mean sd med mad min q25 q75 max  
 <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
1 594 0 67.2 3.82 67 4.45 57 64 70 77.5

Our **point estimate** for the mean of the population of all CWRU students is the mean from our sample of 594 students, which is 67.2 inches. How much uncertainty is there in that estimate?

## Uncertainty Interval for a Mean

We’ll run a linear model (using only an intercept term) to obtain a 95% uncertainty (confidence) interval for the population mean using our data…

mod1 <- lm(height\_in ~ 1, data = dat1)  
  
model\_parameters(mod1, ci = 0.95)

Parameter | Coefficient | SE | 95% CI | t(593) | p  
-------------------------------------------------------------------  
(Intercept) | 67.20 | 0.16 | [66.89, 67.50] | 428.83 | < .001

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

Our 95% uncertainty interval (confidence interval) is (66.9, 67.5) inches for the true mean height across the entire population of CWRU students.

## Haircuts in 2024?

The problem with the haircut price data in 2024 is that it doesn’t follow a Normal distribution, so it’s more difficult to describe the center of the data using a sample mean…

dat3 <- qsdat |> filter(complete.cases(haircut), year == "2024")  
  
mod\_haircut <- lm(haircut ~ 1, data = dat3)  
  
model\_parameters(mod\_haircut, ci = 0.95)

Parameter | Coefficient | SE | 95% CI | t(52) | p  
------------------------------------------------------------------  
(Intercept) | 37.91 | 5.25 | [27.37, 48.44] | 7.22 | < .001

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

* How wide is this interval compared to the one for heights?

## 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…

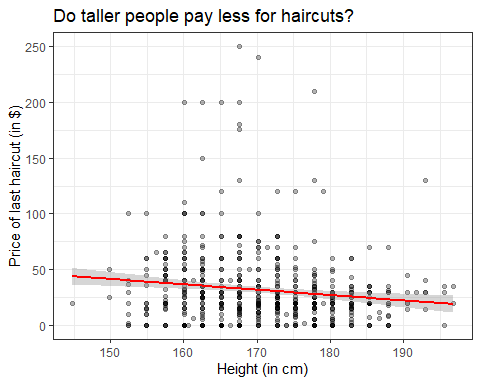
qsdat <- qsdat |> mutate(height\_cm = height\_in \* 2.54)  
  
qsdat |> select(student, height\_in, height\_cm) |> head()

# 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 straight line regression model?

dat4 <- qsdat |> filter(complete.cases(height\_cm, haircut))   
  
mod4 <- lm(haircut ~ height\_cm, data = dat4)  
  
mod4

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

## Summarizing our model mod1

model\_parameters(mod4)

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…

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

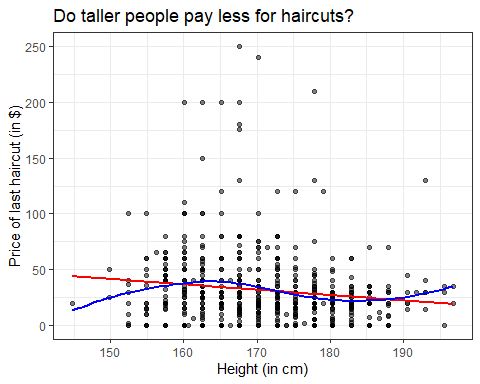
## Interpreting the Model

Again, our regression Equation is:

* The intercept suggests that a student who is 0 cm tall has an average haircut price of $113.46, although we don’t have any students in the data whose height is anywhere near as small as 0 cm, so this isn’t very useful.
* The slope is more important. When comparing two students who differ by 1 cm in height, we observe a haircut price that is, on average, $0.48 smaller for the student with the larger (taller) height.

## 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?

## 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

## 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

|  |
| --- |
| Note |
| By now, I’ve shown you everything you need to complete all Tasks for Lab 1. |

## 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 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   
 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   
 ggridges\_0.5.6 glue\_1.7.0 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 janitor\_2.2.0 jquerylib\_0.1.4   
 jsonlite\_1.8.8 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\_1.3.0   
 memoise\_2.0.1 methods\_4.4.1 mgcv\_1.9-1   
 mime\_0.12 miniUI\_0.1.1.1 minqa\_1.2.8   
 modelbased\_0.8.8 modelr\_0.1.11 multcomp\_1.4-26   
 munsell\_0.5.1 mvtnorm\_1.3-0 nlme\_3.1-164   
 nloptr\_2.1.1 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 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   
 QuickJSR\_1.3.1 R6\_2.5.1 ragg\_1.3.2   
 rappdirs\_0.3.3 RColorBrewer\_1.1.3 Rcpp\_1.0.13   
 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 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\_0.8.5 selectr\_0.4.2   
 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 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   
 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