

431 Class 04

thomaselove.github.io/431

2020-09-03

Today's Agenda

- Working with the “Simple” Survey data again
- Dealing with types of variables
 - Turning characters into factors
- Numerical summaries of quantities and categorical variables
- Visualizing distributions
 - of quantitative variables (histograms)
 - of quantities stratified by categories (violin plots, boxplots)
 - using faceting to stratify by categories
- Some Coding “Tricks” in the tidyverse
- Bonus Content: Reviewing the Photo Age Guesses from Class 01

R Packages loaded for today's class

```
library(patchwork)    # to combine plots  
library(knitr)        # for the kable function
```

Main package for today's class

```
library(tidyverse) # includes essential packages
```

-- Attaching packages -----

v ggplot2 3.3.2	v purrr 0.3.4
v tibble 3.0.3	v dplyr 1.0.2
v tidyr 1.1.2	v stringr 1.4.0
v readr 1.3.1	vforcats 0.5.0

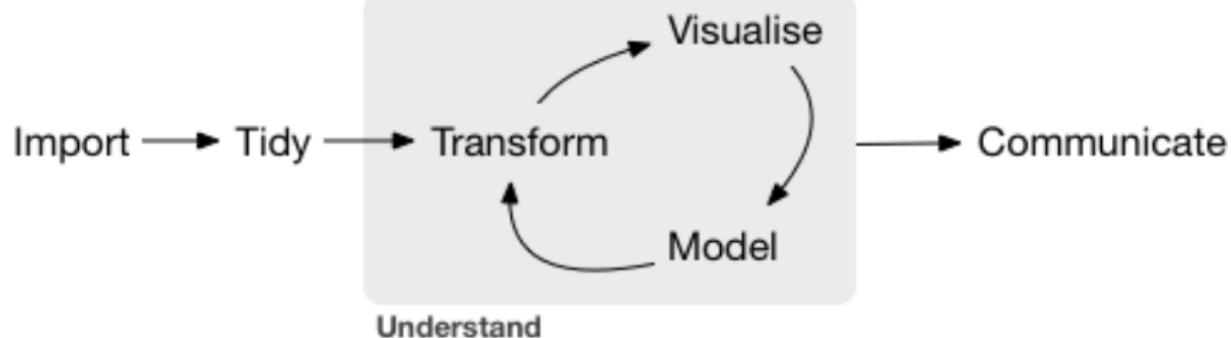
-- Conflicts -----

x dplyr::filter()	masks stats::filter()
x dplyr::lag()	masks stats::lag()

We always load the tidyverse last

It helps resolve some (but not all) conflicts.

Data Science



Program

Load in some data...

```
day1 <- read_csv("data/surveyday1_2020.csv")
```

Parsed with column specification:

```
cols(  
  .default = col_double(),  
  sex = col_character(),  
  glasses = col_character(),  
  english = col_character(),  
  favcolor = col_character()  
)
```

See `spec(...)` for full column specifications.

- `col_double()` means a double-precision (number)
- `col_character()` means a string (non-numeric)

day1

```
# A tibble: 382 x 21
  student sex  glasses english statsofar ageguess
  <dbl> <chr> <chr>   <chr>      <dbl>    <dbl>
1 202001 <NA>  y       n          NA        NA
2 202002 <NA>  y       y          NA        NA
3 202003 <NA>  y       y          NA        NA
4 202004 <NA>  y       n          NA        NA
5 202005 <NA>  y       y          NA        NA
6 202006 <NA>  n       y          NA        NA
7 202007 <NA>  y       y          NA        NA
8 202008 <NA>  n       y          NA        NA
9 202009 <NA>  y       n          NA        NA
10 202010 <NA> y       n          NA        NA
# ... with 372 more rows, and 15 more variables:
#   smoke <dbl>, h.left <dbl>, h.right <dbl>,
#   handedness <dbl>, statfuture <dbl>, haircut <dbl>,
#   lecture <dbl>, alone <dbl>, height.in <dbl>,
```

Summary of some day1 variables

```
day1 %>% select(english, lastsleep, year) %>% summary()
```

english	lastsleep	year
Length:382	Min. : 2.000	Min. :2014
Class :character	1st Qu.: 6.000	1st Qu.:2016
Mode :character	Median : 7.000	Median :2017
	Mean : 6.907	Mean :2017
	3rd Qu.: 8.000	3rd Qu.:2019
	Max. :12.000	Max. :2020
	NA's :3	

Working with Factors

Convert a “character” variable to a “factor”

```
day1 <- day1 %>% mutate(english_fac = as.factor(english))
```

```
day1 %>% select(english, english_fac) %>% summary()
```

```
english           english_fac  
Length:382        n    : 73  
Class :character  y    :306  
Mode  :character  NA's:  3
```

Convert to a factor and recode the levels

```
day1 <- day1 %>%  
  mutate(english_fac = as.factor(english)) %>%  
  mutate(english_det = fct_recode(english_fac,  
                                    "Not English" = "n",  
                                    "English" = "y"))  
  
day1 %>% select(english, english_fac, english_det) %>%  
  summary()
```

```
english          english_fac      english_det  
Length:382        n    : 73      Not English: 73  
Class :character   y    : 306     English       : 306  
Mode  :character   NA's:  3      NA's         :  3
```

Convert a numeric code to a “factor”

```
day1 <- day1 %>%  
  mutate(smoke_fac = as.factor(smoke)) %>%  
  mutate(smoke_det = fct_recode(smoke_fac,  
    Never = "1", Quit = "2", Current = "3"))  
  
day1 %>% select(smoke, smoke_fac, smoke_det) %>% summary()
```

	smoke	smoke_fac	smoke_det
Min.	:1.000	1 :358	Never :358
1st Qu.	:1.000	2 : 18	Quit : 18
Median	:1.000	3 : 4	Current: 4
Mean	:1.068	NA's: 2	NA's : 2
3rd Qu.	:1.000		
Max.	:3.000		
NA's	:2		

Numerical Summaries of Quantities

Summarizing the Sleep Times

```
mosaic::favstats(~ lastsleep, data = day1)
```

Registered S3 method overwritten by 'mosaic':

```
method                  from  
fortify.SpatialPolygonsDataFrame ggplot2
```

	min	Q1	median	Q3	max	mean	sd	n	missing
	2	6	7	8	12	6.907045	1.314816	379	3

Stop the messages about this...

with messages permitted...

```
```{r}
mosaic::favstats(~ lastsleep, data = day1)
```
```

Messages not shown...

```
```{r, message = FALSE}
mosaic::favstats(~ lastsleep, data = day1)
```
```

Breaking down Sleep Hours, by Year

```
mosaic::favstats(lastsleep ~ year, data = day1)
```

| | year | min | Q1 | median | Q3 | max | mean | sd | n |
|---|---------|-----|-----|--------|------|------|----------|----------|----|
| 1 | 2014 | 4 | 6.0 | 7.00 | 8.00 | 9.0 | 6.902439 | 1.179086 | 41 |
| 2 | 2015 | 4 | 6.5 | 7.50 | 8.00 | 9.0 | 7.316327 | 1.184509 | 49 |
| 3 | 2016 | 4 | 6.0 | 7.00 | 8.00 | 9.5 | 7.002698 | 1.197164 | 63 |
| 4 | 2017 | 2 | 6.0 | 6.75 | 8.00 | 9.0 | 6.708333 | 1.406041 | 48 |
| 5 | 2018 | 4 | 6.0 | 7.00 | 7.50 | 8.5 | 6.784314 | 1.030800 | 51 |
| 6 | 2019 | 4 | 6.5 | 7.00 | 8.00 | 12.0 | 7.113333 | 1.260679 | 60 |
| 7 | 2020 | 2 | 6.0 | 7.00 | 7.75 | 11.0 | 6.571642 | 1.652803 | 67 |
| | missing | | | | | | | | |
| 1 | | 1 | | | | | | | |
| 2 | | 0 | | | | | | | |
| 3 | | 1 | | | | | | | |
| 4 | | 0 | | | | | | | |
| 5 | | 0 | | | | | | | |
| 6 | | 1 | | | | | | | |

Using kable to specify decimal places

```
mosaic::favstats(lastsleep ~ year, data = day1) %>%  
  kable(digits = 2)
```

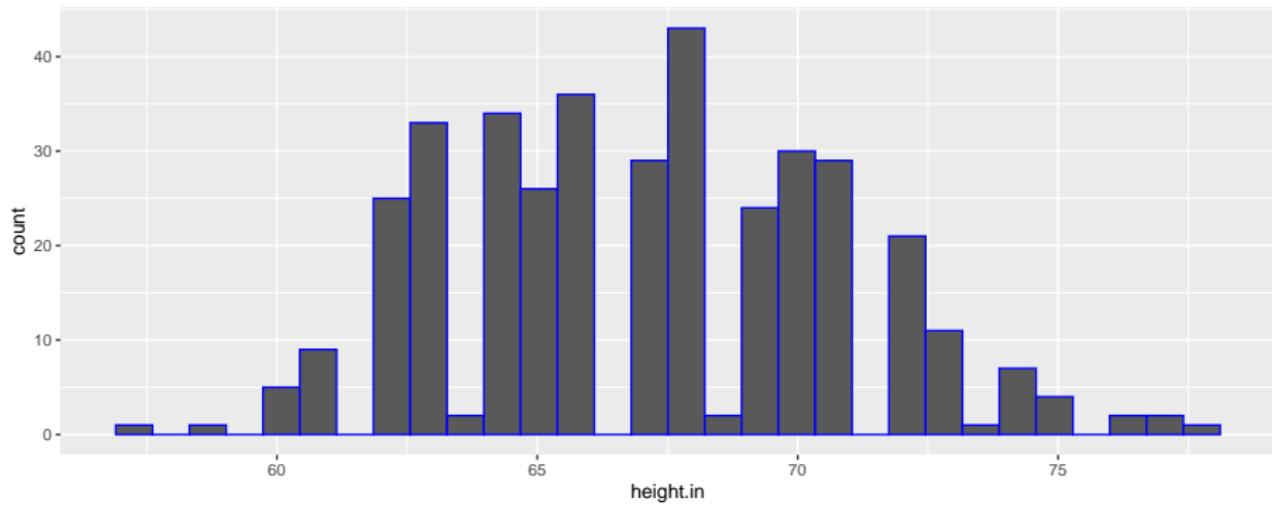
| year | min | Q1 | median | Q3 | max | mean | sd | n | missing |
|------|-----|-----|--------|------|------|------|------|----|---------|
| 2014 | 4 | 6.0 | 7.00 | 8.00 | 9.0 | 6.90 | 1.18 | 41 | 1 |
| 2015 | 4 | 6.5 | 7.50 | 8.00 | 9.0 | 7.32 | 1.18 | 49 | 0 |
| 2016 | 4 | 6.0 | 7.00 | 8.00 | 9.5 | 7.00 | 1.20 | 63 | 1 |
| 2017 | 2 | 6.0 | 6.75 | 8.00 | 9.0 | 6.71 | 1.41 | 48 | 0 |
| 2018 | 4 | 6.0 | 7.00 | 7.50 | 8.5 | 6.78 | 1.03 | 51 | 0 |
| 2019 | 4 | 6.5 | 7.00 | 8.00 | 12.0 | 7.11 | 1.26 | 60 | 1 |
| 2020 | 2 | 6.0 | 7.00 | 7.75 | 11.0 | 6.57 | 1.65 | 67 | 0 |

Visualizing the distribution of a variable with a Histogram

Plot 1. Initial Attempt

```
day1 %>% filter(complete.cases(height.in)) %>%  
  ggplot(data = ., aes(x = height.in)) +  
  geom_histogram(col = "blue")
```

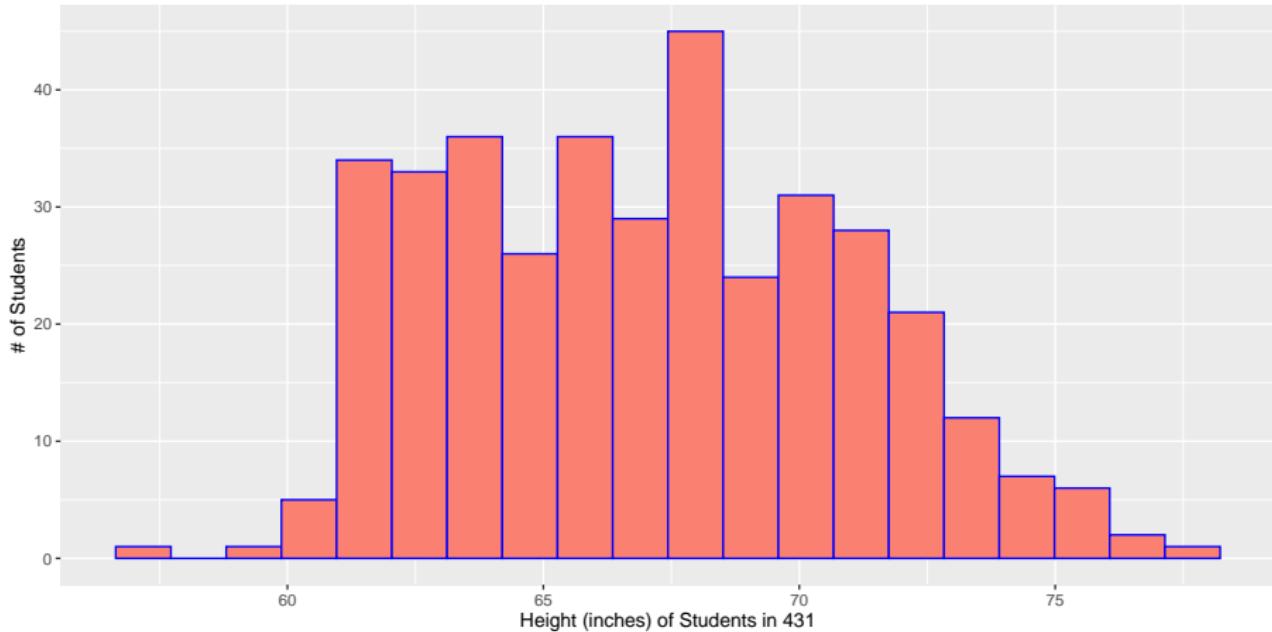
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.



Plot 2. New Version: fill is *salmon*, using 20 bins.

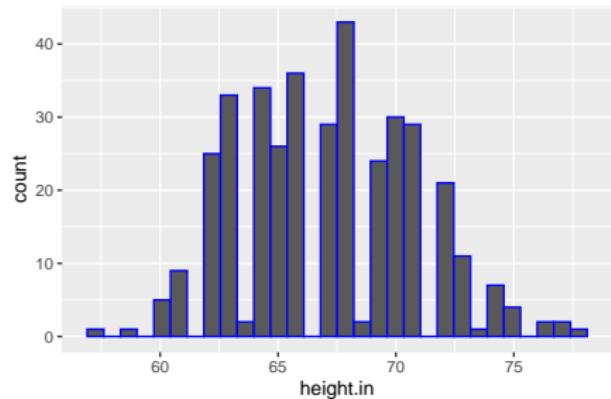
Heights of 378 students in 431

4 students had missing heights



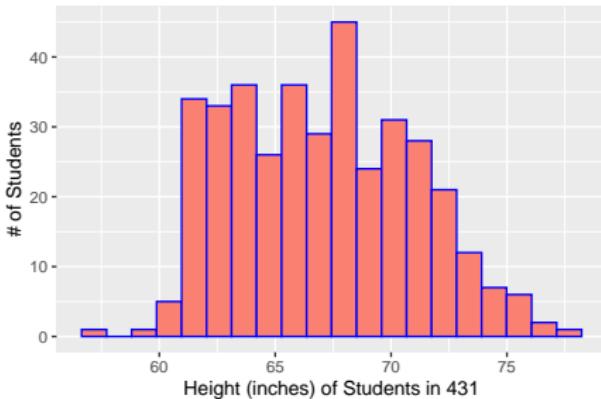
How do we get from plot 1 to plot 2?

1



2

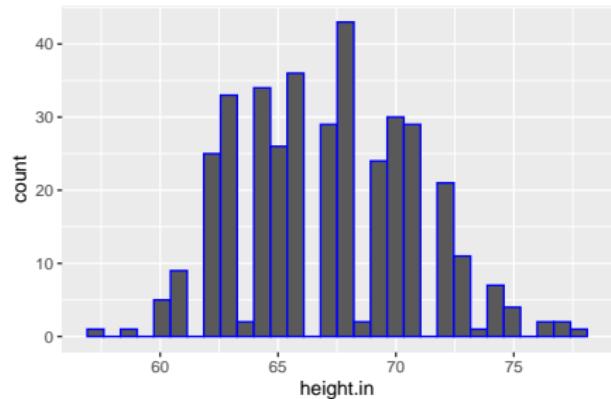
Heights of 378 students in 431
4 students had missing heights



- What do we need to change?

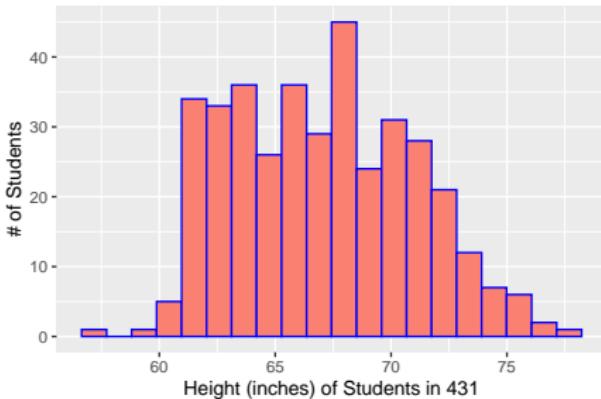
How do we get from plot 1 to plot 2?

1



2

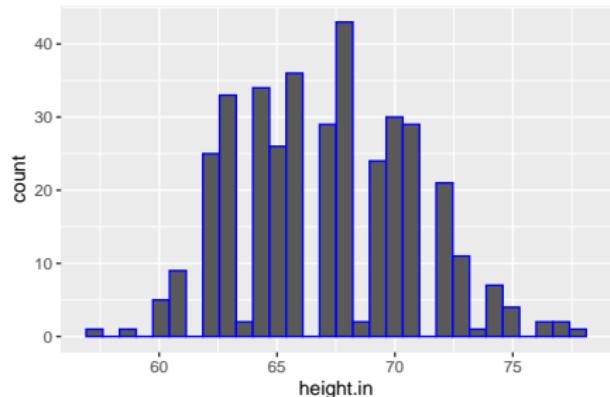
Heights of 378 students in 431
4 students had missing heights



- What do we need to change?
- salmon color for the “fill” of the histogram’s bars

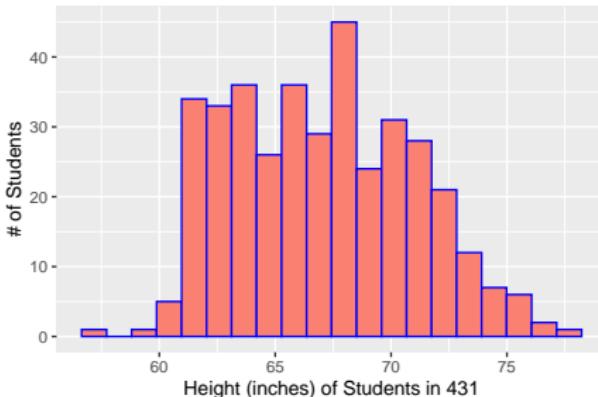
How do we get from plot 1 to plot 2?

1



2

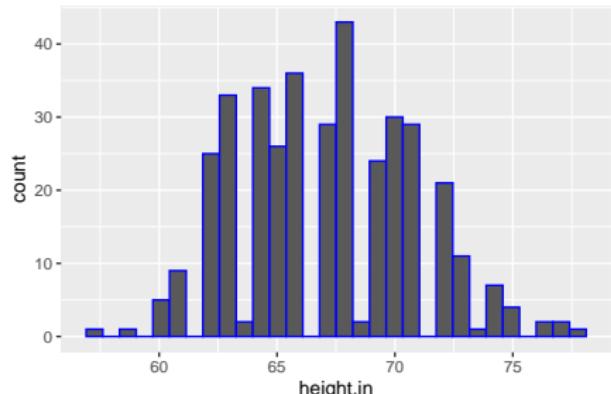
Heights of 378 students in 431
4 students had missing heights



- What do we need to change?
- salmon color for the “fill” of the histogram’s bars
- specify 20 bins in the histogram

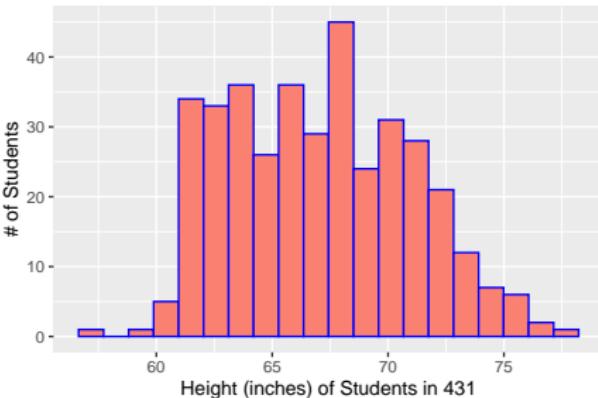
How do we get from plot 1 to plot 2?

1



2

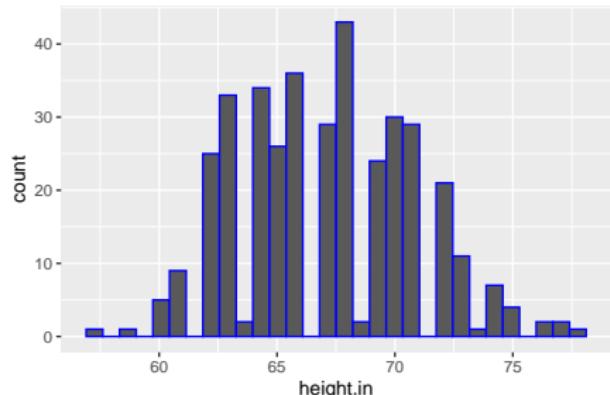
Heights of 378 students in 431
4 students had missing heights



- What do we need to change?
- salmon color for the “fill” of the histogram’s bars
- specify 20 bins in the histogram
- relabel the x and y axes

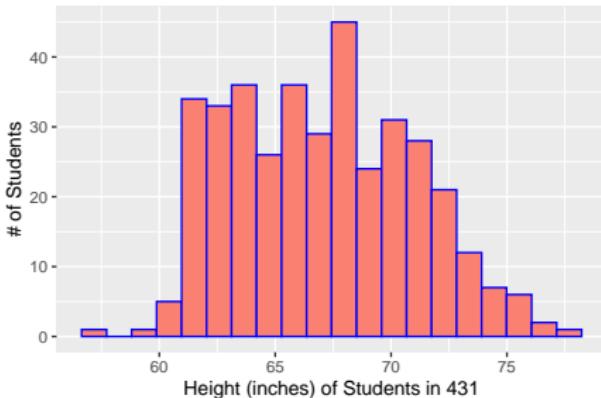
How do we get from plot 1 to plot 2?

1



2

Heights of 378 students in 431
4 students had missing heights

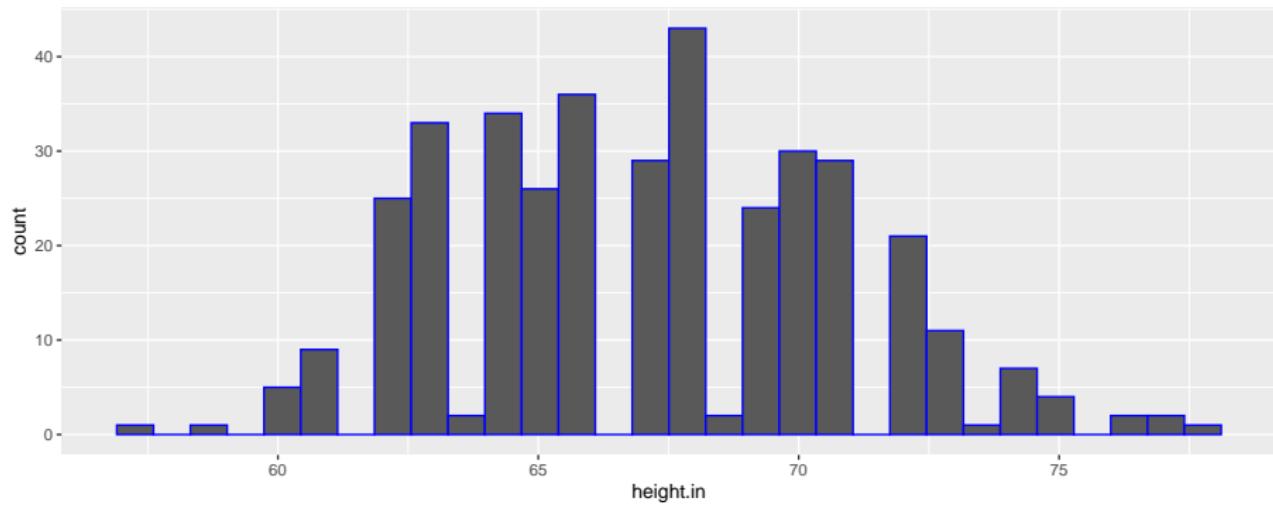


- What do we need to change?
- salmon color for the “fill” of the histogram’s bars
- specify 20 bins in the histogram
- relabel the x and y axes
- add title and subtitle

Starting with Plot 1. Initial Attempt

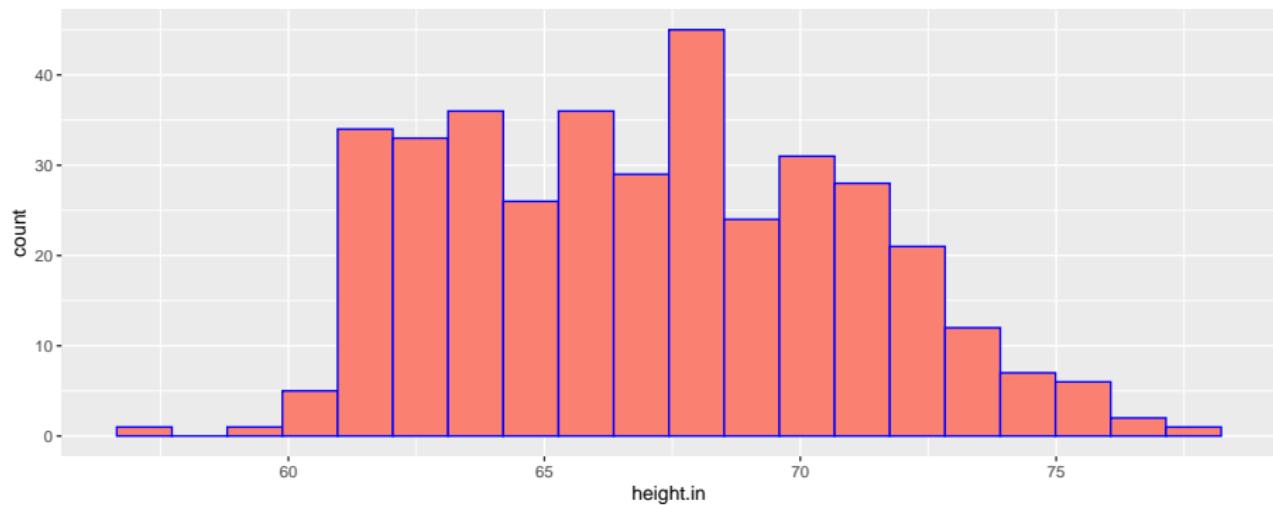
```
day1 %>% filter(complete.cases(height.in)) %>%  
  ggplot(data = ., aes(x = height.in)) +  
  geom_histogram(col = "blue")
```

`stat_bin()` using `bins = 30`.
Pick better value with `binwidth`.



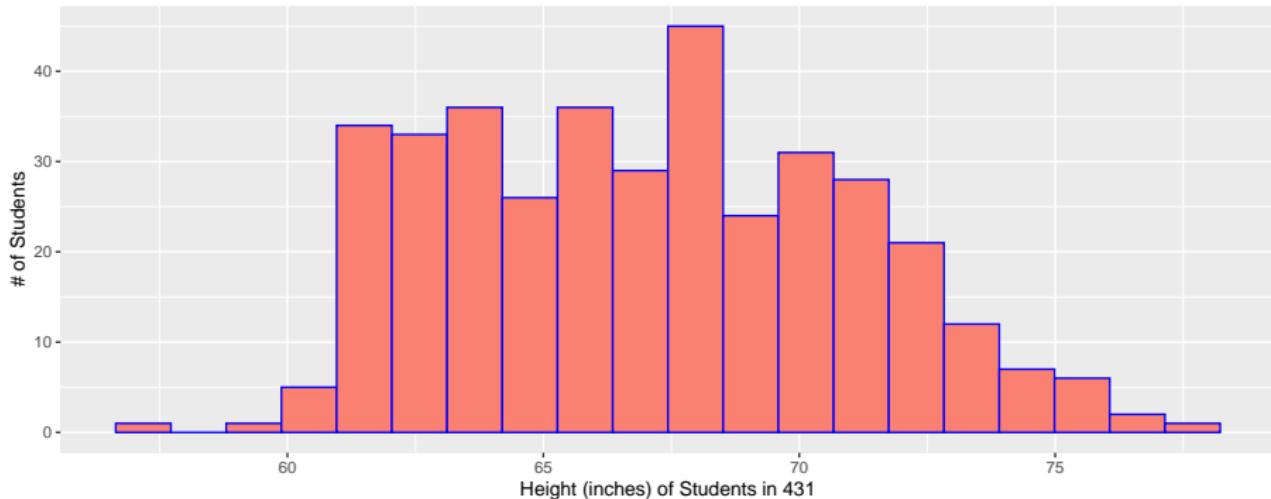
Add salmon fill, use 20 bins

```
day1 %>% filter(complete.cases(height.in)) %>%
  ggplot(data = ., aes(x = height.in)) +
  geom_histogram(bins = 20, col = "blue", fill = "salmon")
```



Now relabel x and y axes

```
day1 %>% filter(complete.cases(height.in)) %>%  
  ggplot(data = ., aes(x = height.in)) +  
  geom_histogram(bins = 20, col = "blue", fill = "salmon") +  
  labs(x = "Height (inches) of Students in 431",  
       y = "# of Students")
```



How many students provided their height?

```
nrow(day1)
```

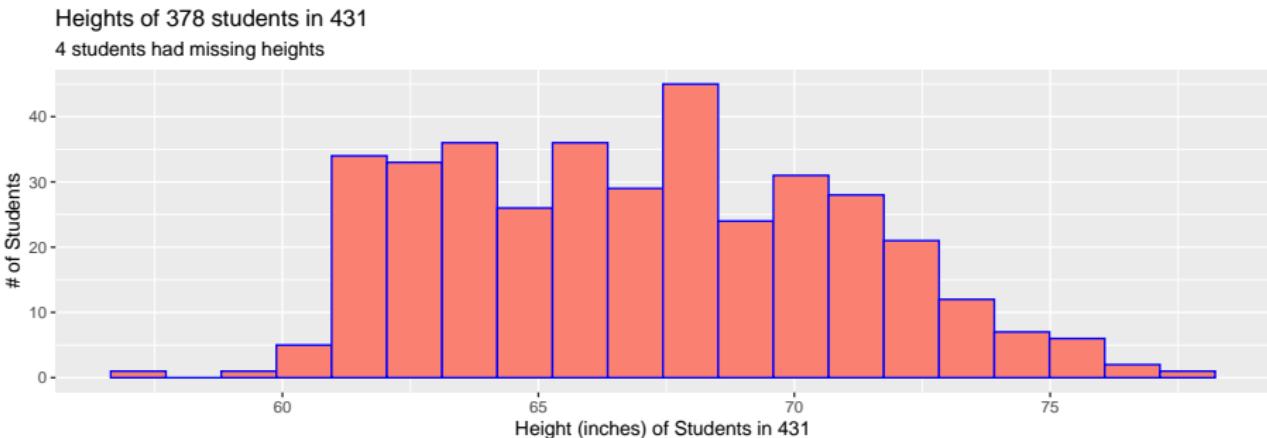
```
[1] 382
```

```
mosaic::favstats(~ height.in, data = day1)
```

| min | Q1 | median | Q3 | max | mean | sd | n | missing |
|-----|----|--------|----|------|----------|--------|-----|---------|
| 57 | 64 | 67 | 70 | 77.5 | 67.11905 | 3.7355 | 378 | 4 |

Actual Code for Plot 2 (adds title and subtitle)

```
day1 %>% filter(complete.cases(height.in)) %>%
  ggplot(data = ., aes(x = height.in)) +
  geom_histogram(bins = 20, col = "blue", fill = "salmon") +
  labs(x = "Height (inches) of Students in 431",
       y = "# of Students",
       title = "Heights of 378 students in 431",
       subtitle = "4 students had missing heights")
```

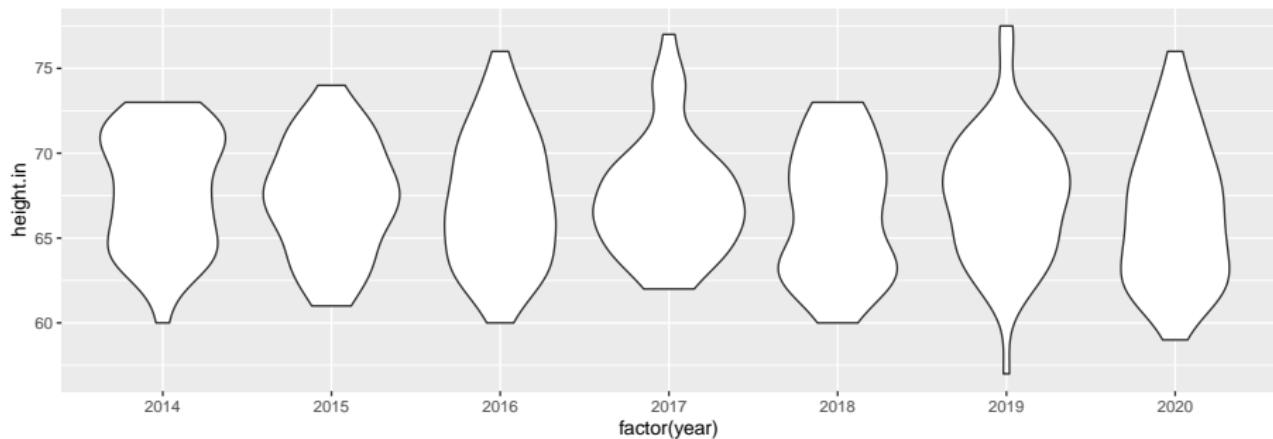


Violin Plots and Boxplots to Compare Distributions of a Quantitative Variable across levels of a Categorical Variable

Can we look at the Height Distributions by Year?

```
ggplot(day1, aes(x = factor(year), y = height.in)) +  
  geom_violin()
```

Warning: Removed 4 rows containing non-finite values
(stat_ydensity).



Remove missing values, revise axis labels

```
day1 %>% filter(complete.cases(year, height.in)) %>%  
  ggplot(., aes(x = factor(year), y = height.in)) +  
  geom_violin() +  
  labs(x = "Year", y = "Height (inches)")
```

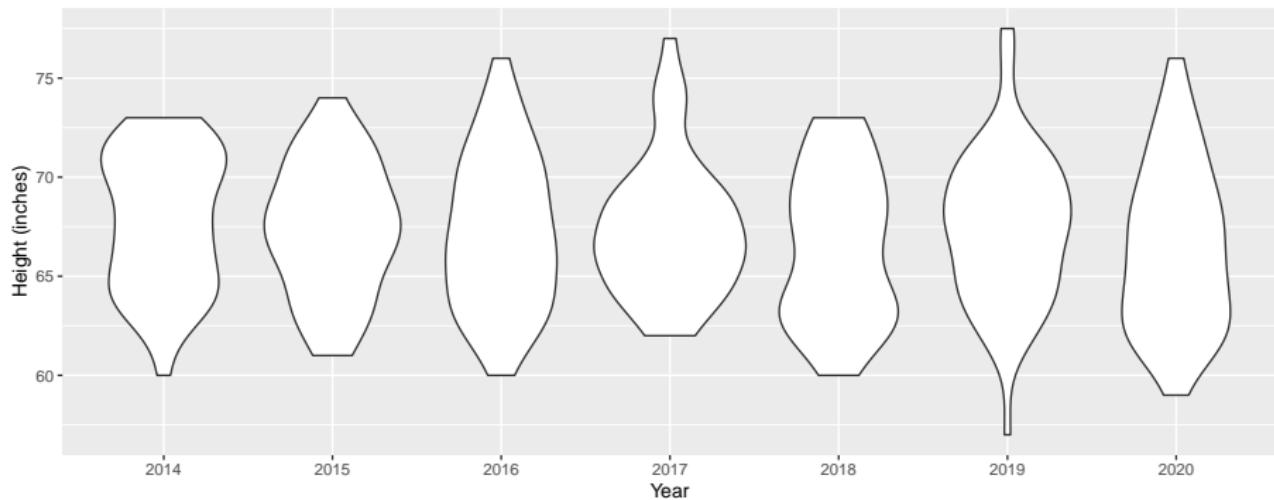


Table summarizing Student Heights, by Year

```
mosaic::favstats(height.in ~ year, data = day1) %>%  
  kable(digits = 1)
```

| year | min | Q1 | median | Q3 | max | mean | sd | n | missing |
|------|-----|------|--------|------|------|------|-----|----|---------|
| 2014 | 60 | 64.8 | 68 | 71.0 | 73.0 | 67.8 | 3.5 | 40 | 2 |
| 2015 | 61 | 65.0 | 68 | 70.0 | 74.0 | 67.3 | 3.3 | 49 | 0 |
| 2016 | 60 | 64.0 | 67 | 70.0 | 76.0 | 67.2 | 3.9 | 64 | 0 |
| 2017 | 62 | 65.0 | 67 | 69.0 | 77.0 | 67.4 | 3.5 | 48 | 0 |
| 2018 | 60 | 63.0 | 66 | 70.0 | 73.0 | 66.5 | 3.8 | 51 | 0 |
| 2019 | 57 | 65.0 | 68 | 70.0 | 77.5 | 67.4 | 3.8 | 60 | 1 |
| 2020 | 59 | 63.0 | 66 | 69.8 | 76.0 | 66.4 | 4.1 | 66 | 1 |

What's in a Boxplot

- The central box in a boxplot indicates the 25th (Q1), 50th (median) and 75th (Q3) percentiles.
- It uses the interquartile range ($IQR = Q3 - Q1$) as a measure of spread.
- Points more than 1.5 IQR below Q1 or more than 1.5 IQR above Q3 are candidate **outliers**.
- The “whiskers” in the plot extend from Q1 down to the smallest non-outlier and from Q3 up to the largest non-outlier.

Who invented the boxplot (or box-and-whiskers plot)? John **Tukey**.

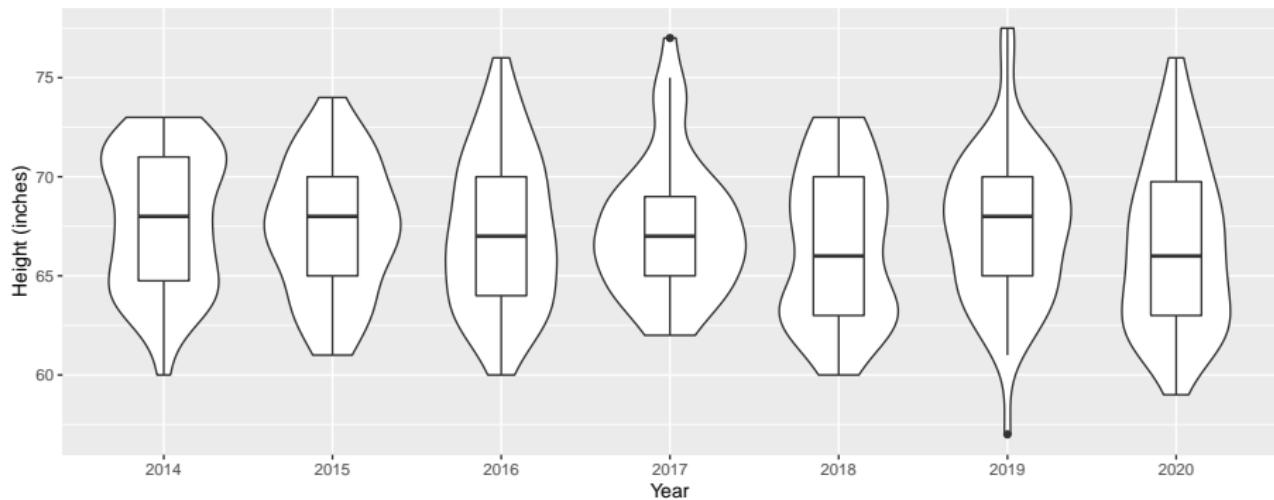


Numerical quantities focus on expected values, graphical summaries on unexpected values.

-- John **Tukey**

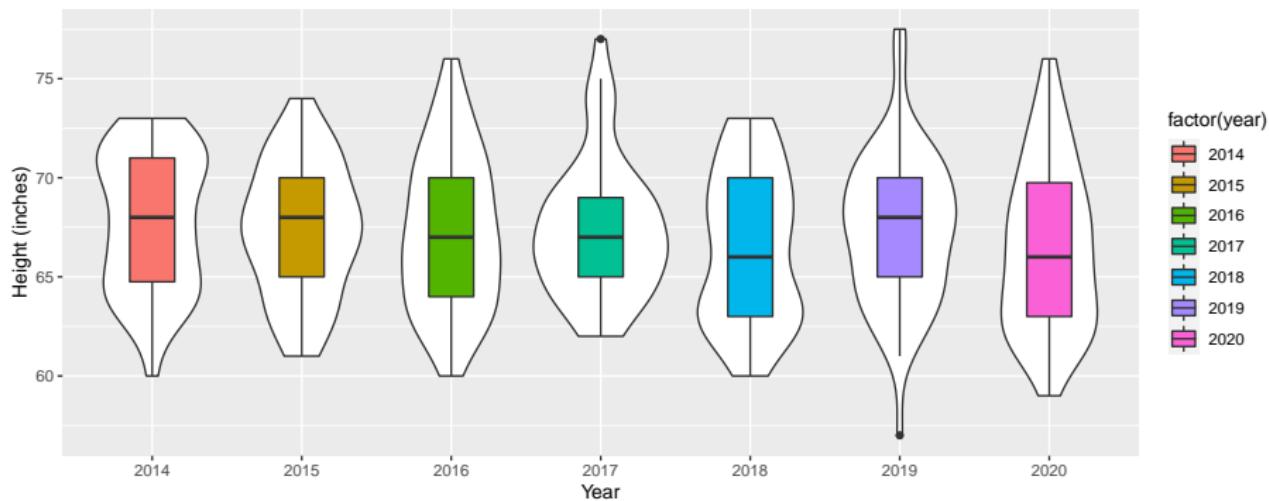
Add Boxplots: Height Distributions by Year?

```
day1 %>% filter(complete.cases(year, height.in)) %>%
  ggplot(., aes(x = factor(year), y = height.in)) +
  geom_violin() +
  geom_boxplot(width = 0.3) +
  labs(x = "Year", y = "Height (inches)")
```



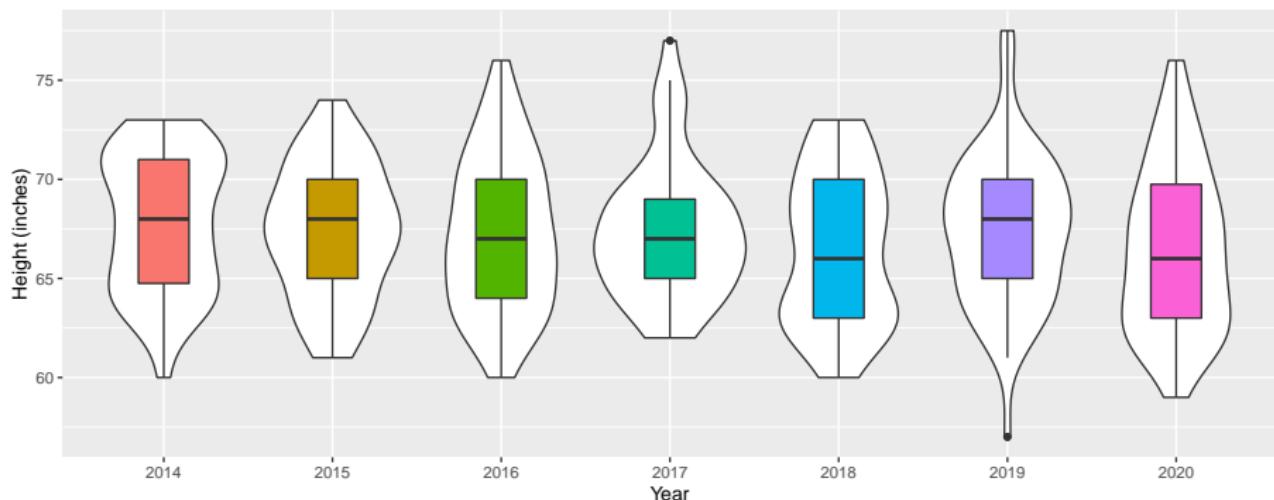
Add fill for the boxes?

```
day1 %>% filter(complete.cases(year, height.in)) %>%
  ggplot(., aes(x = factor(year), y = height.in)) +
  geom_violin() +
  geom_boxplot(aes(fill = factor(year)), width = 0.3) +
  labs(x = "Year", y = "Height (inches)")
```



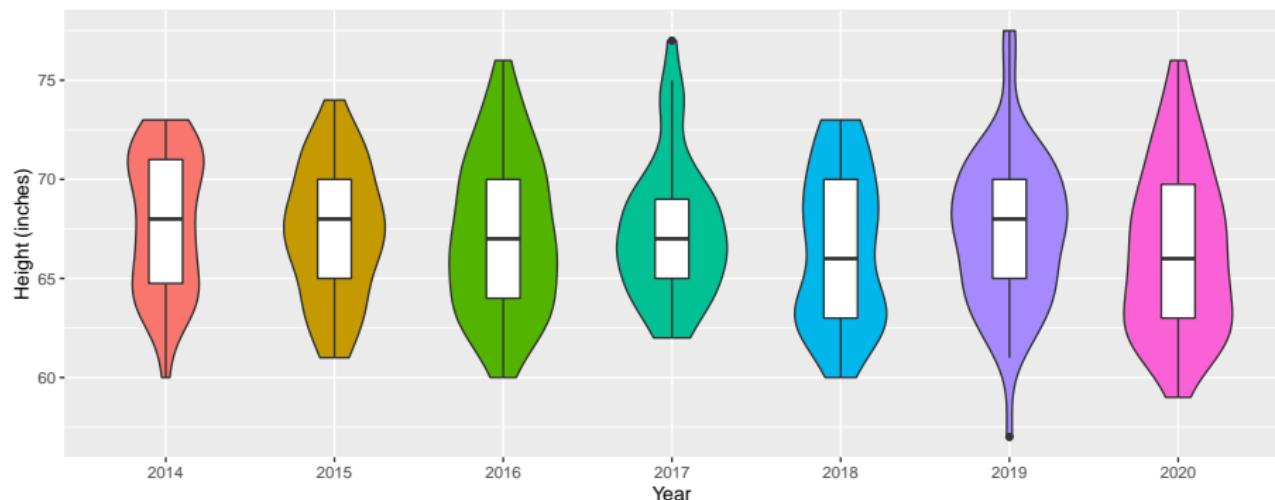
Drop unnecessary “guides” (legend)

```
day1 %>% filter(complete.cases(year, height.in)) %>%  
  ggplot(., aes(x = factor(year), y = height.in)) +  
  geom_violin() +  
  geom_boxplot(aes(fill = factor(year)), width = 0.3) +  
  guides(fill = FALSE) +  
  labs(x = "Year", y = "Height (inches)")
```



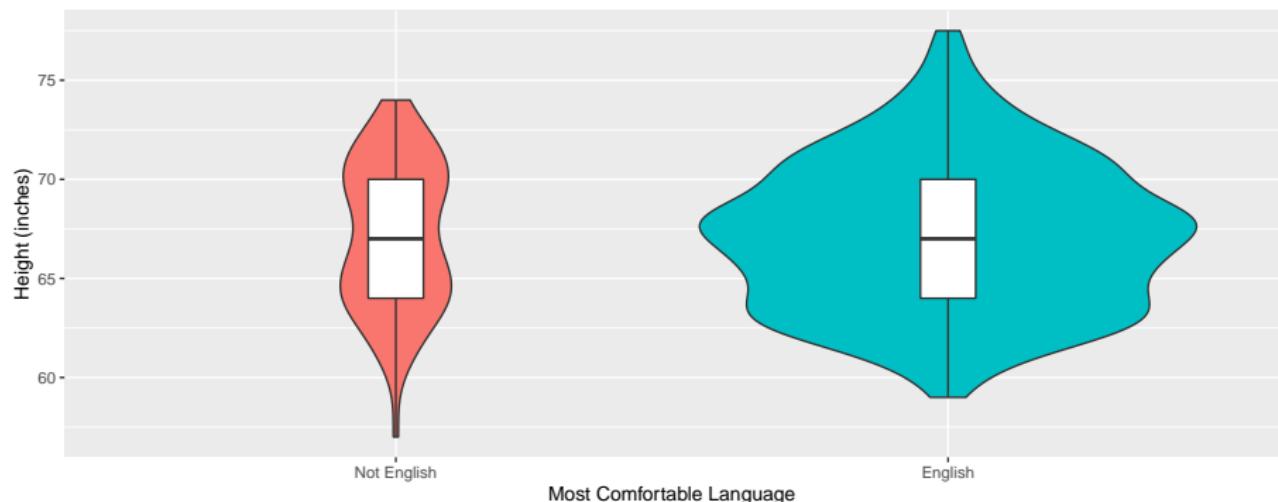
Scale violins by sample size and add fill

```
day1 %>% filter(complete.cases(year, height.in)) %>%  
  ggplot(., aes(x = factor(year), y = height.in)) +  
  geom_violin(aes(fill = factor(year)), scale = "count") +  
  geom_boxplot(width = 0.2) +  
  guides(fill = FALSE) +  
  labs(x = "Year", y = "Height (inches)")
```



Association of height and language?

```
day1 %>% filter(complete.cases(english_det, height.in)) %>%
  ggplot(., aes(x = english_det, y = height.in)) +
  geom_violin(aes(fill = english_det), scale = "count") +
  geom_boxplot(width = 0.1) +
  guides(fill = FALSE) +
  labs(x = "Most Comfortable Language", y = "Height (inches)")
```



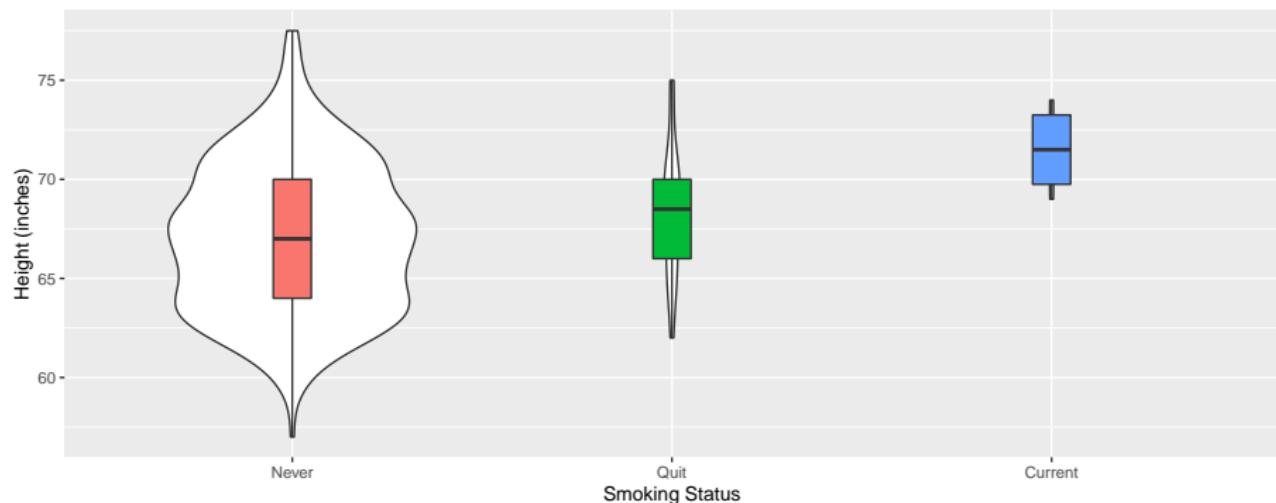
Height by Language

```
mosaic::favstats(height.in ~ english_det, data = day1) %>%  
  kable(digits = 1)
```

| english_det | min | Q1 | median | Q3 | max | mean | sd | n | missing |
|-------------|-----|----|--------|----|------|------|-----|-----|---------|
| Not English | 57 | 64 | 67 | 70 | 74.0 | 67.1 | 3.7 | 72 | 1 |
| English | 59 | 64 | 67 | 70 | 77.5 | 67.2 | 3.8 | 303 | 3 |

Association of height with smoking status?

```
day1 %>% filter(complete.cases(smoke_det, height.in)) %>%
  ggplot(., aes(x = smoke_det, y = height.in)) +
  geom_violin(scale = "count") +
  geom_boxplot(aes(fill = smoke_det), width = 0.1) +
  guides(fill = FALSE) +
  labs(x = "Smoking Status", y = "Height (inches)")
```



Height by Smoking Status

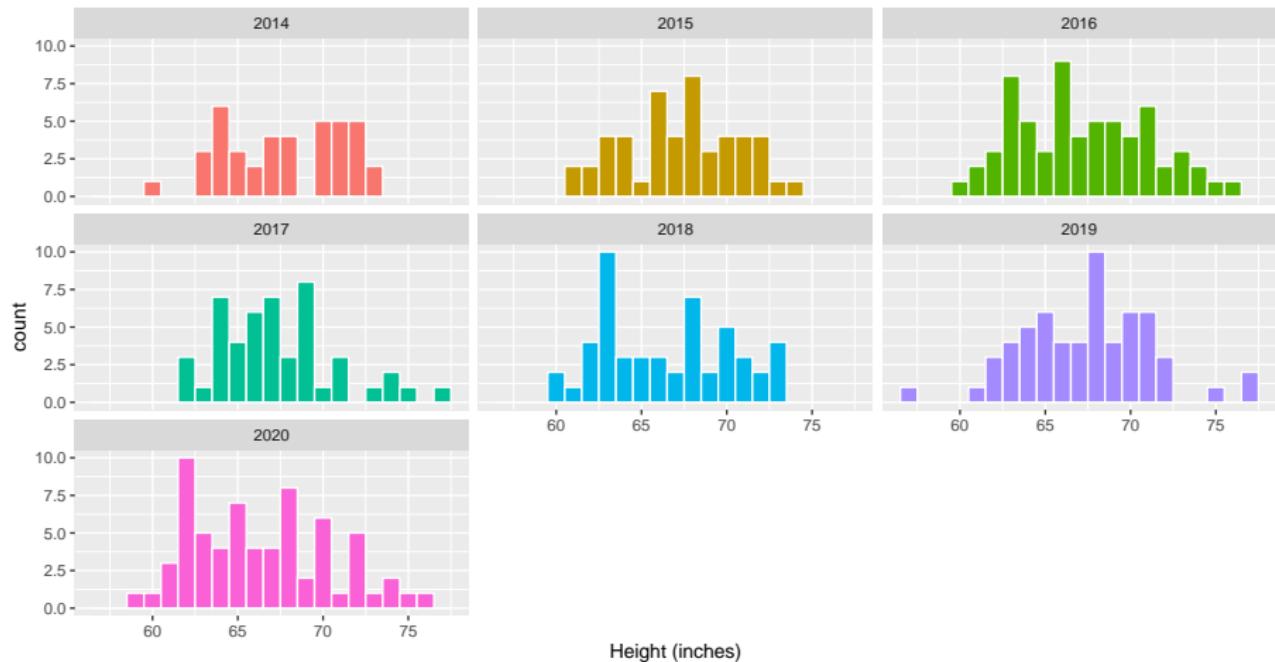
```
mosaic::favstats(height.in ~ smoke_det, data = day1) %>%  
  kable(digits = 1)
```

| smoke_det | min | Q1 | median | Q3 | max | mean | sd | n | missing |
|-----------|-----|------|--------|------|------|------|-----|-----|---------|
| Never | 57 | 64.0 | 67.0 | 70.0 | 77.5 | 67.0 | 3.7 | 355 | 3 |
| Quit | 62 | 66.0 | 68.5 | 70.0 | 75.0 | 68.4 | 3.4 | 18 | 0 |
| Current | 69 | 69.8 | 71.5 | 73.2 | 74.0 | 71.5 | 2.4 | 4 | 0 |

Working with Facets

Height Distributions, Faceted by Year

Student Height Distribution, by Year



Code for plot on previous slide

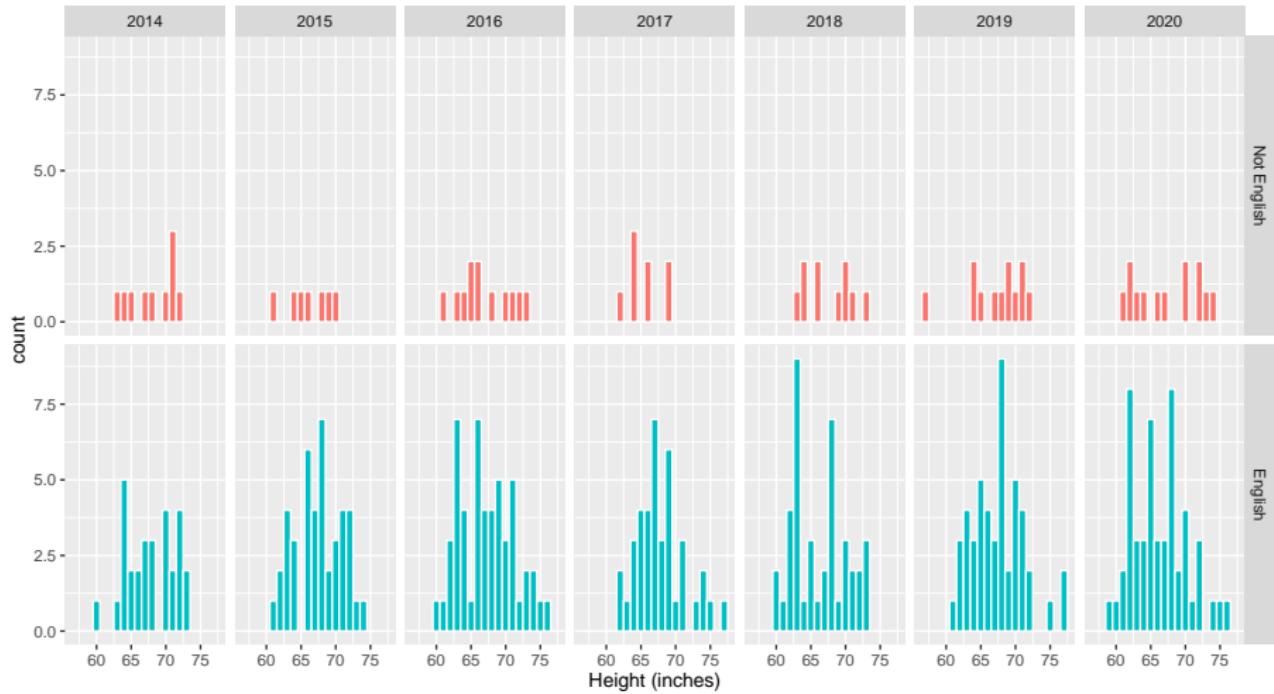
```
day1 %>% filter(complete.cases(height.in)) %>%
  ggplot(data = ., aes(x = height.in, fill = factor(year))) +
  geom_histogram(binwidth = 1, col = "white") +
  facet_wrap(~ year) +
  guides(fill = FALSE) +
  labs(title = "Student Height Distribution, by Year",
       x = "Height (inches)")
```

What do you think this will do?

```
day1 %>% filter(complete.cases(height.in, english_det, year))  
ggplot(data = .,  
       aes(x = height.in, fill = factor(english_det))) +  
  geom_histogram(binwidth = 1, col = "white") +  
  facet_grid(english_det ~ year) +  
  guides(fill = FALSE) +  
  labs(x = "Height (inches)")
```

Result on next slide...

Faceted Histograms of Height by Language, Year

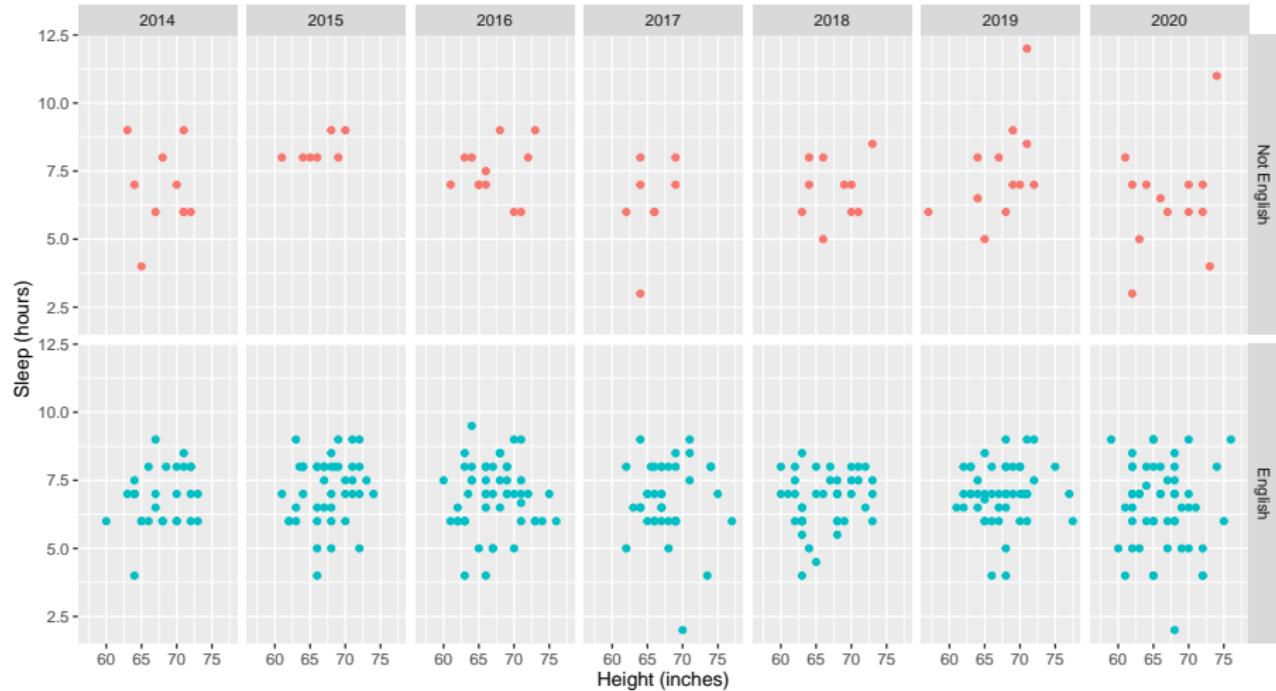


What do you think this will do?

```
day1 %>%
  filter(complete.cases(height.in, lastsleep,
                        english_det)) %>%
  ggplot(data = .,
         aes(x = height.in, y = lastsleep,
              color = english_det)) +
  geom_point() +
  facet_grid(english_det ~ year) +
  guides(color = FALSE) +
  labs(x = "Height (inches)", y = "Sleep (hours)")
```

Result on next slide...

Faceted Scatterplots from Previous Slide



Next Time

That's the end of the slides for Class 03.

- More intricate data work in R
- Visualizing associations
- Linking visualizations to models
- A new example

The remainder of this PDF looks at the age guessing activity we did in Class 01, but now you can compare 2020 results to previous years.

Age Guessing Activity from Class 01



#1 Age 21



#2 Age 64



#3 Age 28



#4 Age 14



#5 Age 54



#6 Age 74



#7 Age 44



#8 Age 83



#9 Age 24



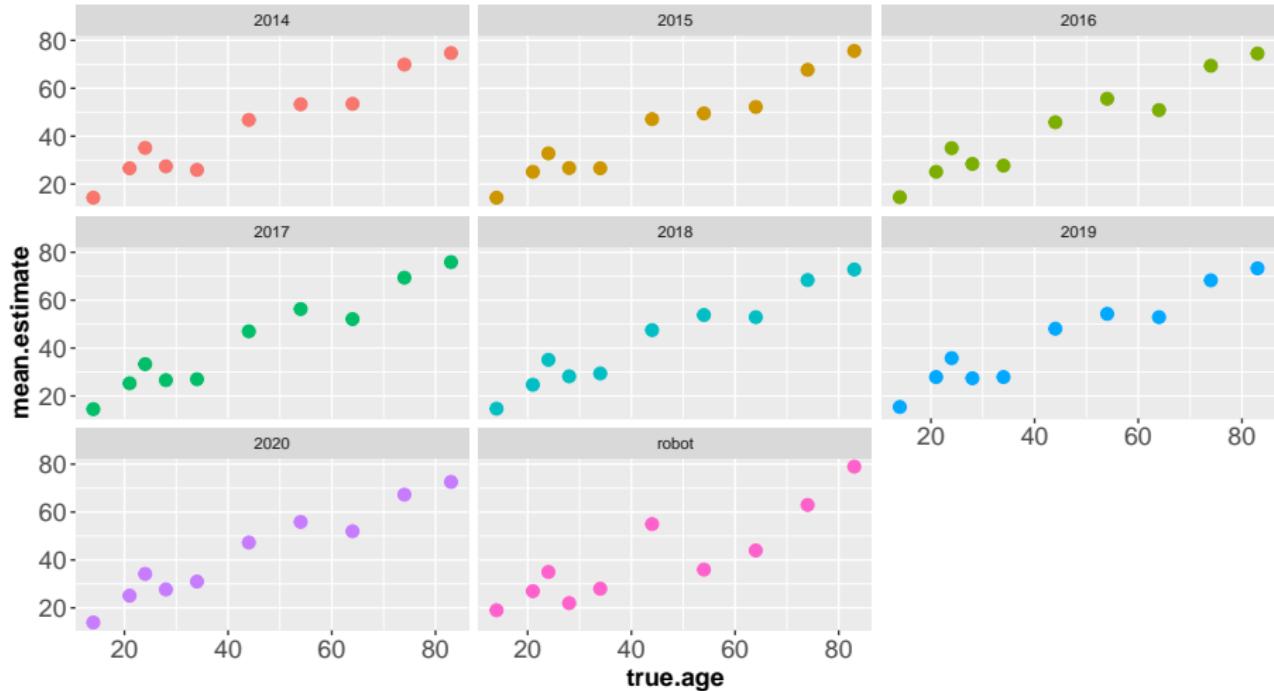
#10 Age 34

Age Guessing (including 2020)

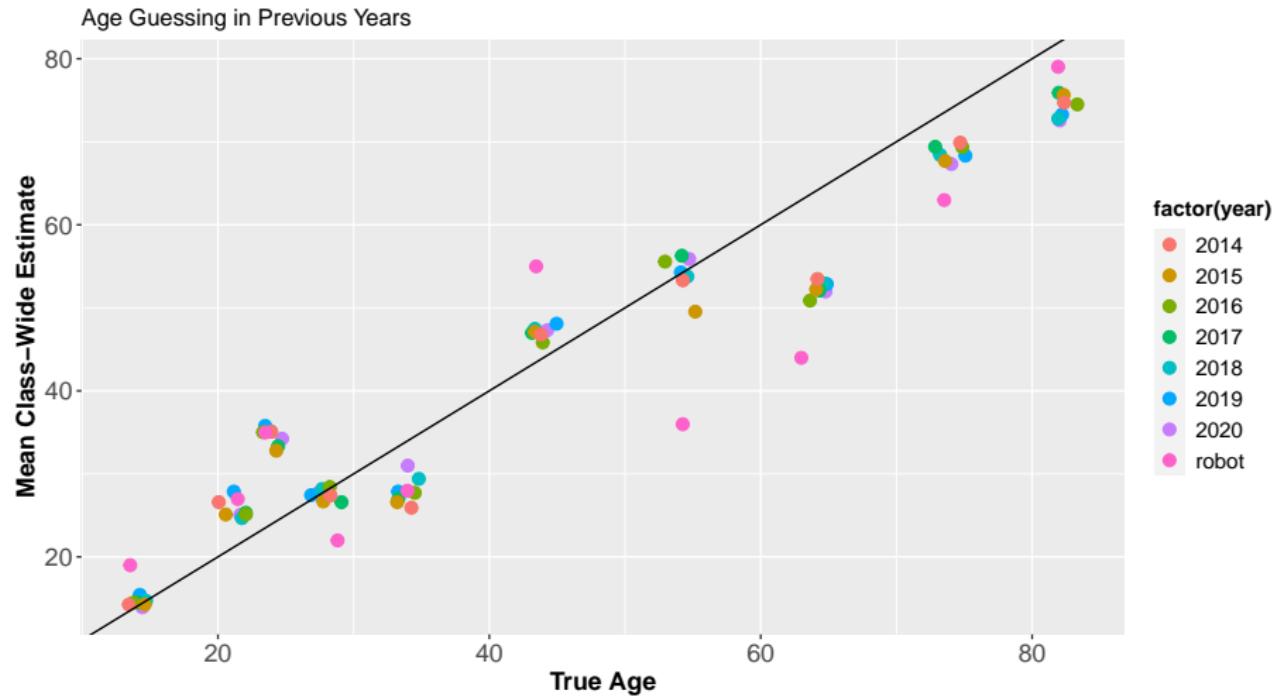
```
ageguess <- read_csv("data/photo-age-history-2020.csv")
```

| card | label | true.age | sex | facing | year | mean.estimate | error |
|------|-----------|----------|-----|--------|------|---------------|-------|
| 1 | Chong | 21 | M | R | 2020 | 25.1 | -4.1 |
| 2 | Archuleta | 64 | F | L | 2020 | 52.0 | 12.0 |
| 3 | Mayfield | 28 | F | L | 2020 | 27.7 | 0.3 |
| 4 | Love | 14 | M | L | 2020 | 13.9 | 0.1 |
| 5 | McGinn | 54 | F | R | 2020 | 55.9 | -1.9 |
| 6 | Chaney | 74 | M | L | 2020 | 67.3 | 6.7 |
| 7 | Storm | 44 | M | R | 2020 | 47.3 | -3.3 |
| 8 | Glantz | 83 | F | L | 2020 | 72.6 | 10.4 |
| 9 | Honey | 24 | M | L | 2020 | 34.2 | -10.2 |
| 10 | Lawson | 34 | F | R | 2020 | 31.0 | 3.0 |

Scatterplot of Results by Year, 1



Scatterplot of Results by Year, 2



Mean Class-Wide Guesses (2014-2020 combined)



| | | | | | | | | | |
|--------------|------|-----------|------|-----------|------|-----------|------|------------|------|
| #1 Age 21 | 25.7 | #2 Age 64 | 52.4 | #3 Age 28 | 27.5 | #4 Age 14 | 14.5 | #5 Age 54 | 54.1 |
| 2014-2020 | | | | | | | | | |
| Mean Guesses | 68.6 | | 47.1 | | 74.2 | | 34.5 | | 27.9 |
| #6 Age 74 | | #7 Age 44 | | #8 Age 83 | | #9 Age 24 | | #10 Age 34 | |



Scatterplot of 2020 Results with Labels

Errors in 2020 Age Guessing, by Subject's Sex

