431 Class 06

github.com/THOMASELOVE/2019-431

2019-09-12

Today's Agenda

1. On Jeff Leek's *Elements of Data Analytic Style*
2. NHANES Example
   - See the related example in the Course Notes Chapters 3-6
3. Course Project Discussion and "Meetup"

Leek Chapter 5: Exploratory Analysis

- EDA To understand properties of the data and discover new patterns
- Visualize and inspect qualitative features rather than a huge table of raw data

1. Make big data as small as possible as quickly as possible
2. Plot as much of the actual data as you can
3. For large data sets, subsample before plotting
4. Use log transforms for ratio measurements
5. Missing values can have a mighty impact on conclusions

Leek: Chapter 9 Written Analyses

Elements: title, introduction/motivation, description of statistical tools used, results with measures of uncertainty, conclusions indicating potential problems, references

1. What is the question you are answering?
2. Lead with a table summarizing your tidy data set (critical to identify data versioning issues)
3. For each parameter of interest report an estimate and measure of uncertainty on the scientific scale of interest
4. Summarize the importance of reported estimates
5. Do not report every analysis you performed

Leek: Chapter 10 Creating Figures

Communicating effectively with figures is non-trivial. The goal is clarity.

> *When viewed with an appropriately detailed caption, (a figure should) stand alone without any further explanation as a unit of information.*

1. Humans are best at perceiving position along a single axis with a common scale
2. Avoid chartjunk (gratuitous flourishes) in favor of high-density displays
3. Axis labels should be large, easy to read, in plain language
4. Figure titles should communicate the plot's message
5. Use a palette (like `viridis`) that color-blind people can see (and distinguish) well

Check out Karl Broman's excellent presentation on displaying data badly at https://github.com/kbroman/Talk_Graphs

Leek Chapter 13: A Few Matters of Form

- Variable names should always be reported in plain language.
- If measurements are only accurate to the tenths digit, don't report estimates with more digits.
- Report estimates followed by parentheses that hold a 95% CI or other measure of uncertainty.
- When reporting p values, censor small values ($p < 0.0001$, not $p = 0$ or $p = 1.6$ x 10^{-25})

Upcoming Reading

Leek *Elements of Data Analytic Style* (finish by Oct 1)

- Chapters 2-4 should be very helpful for project (Data analytic question, Tidying data, Checking data)
- 6-8 are more about Parts B and C of the course
- 11-12 on Presenting Data and Reproducibility
- 14 is a Data Analysis Checklist

Nate Silver *The Signal and the Noise* for Tuesday

- Introduction: Is increased access to information a good thing?
- Chapter 1: The failure to predict the 2008 housing bubble and recession
- Chapters 2-3 on forecasting politics and baseball by 2019-09-24.

What about R for Data Science?

I'd be trying to get through *Explore* (sections 2-8) before our first Quiz.

- Section 11 on Data import
- Section 18 on Pipes
- Section 27 on R Markdown and maybe 28 on Graphics for communication

https://r4ds.had.co.nz/

Back to our NHANES Example

Today's Packages

The R packages we're using today are NHANES, magrittr, janitor and tidyverse.

```r
library(NHANES); library(magrittr)
library(janitor); library(tidyverse)
```

CWRU Colors

```r
cwru.blue <- '#0a304e'
cwru.gray <- '#626262'
```

Our nh2 data set, again

```
set.seed(20190910) # so we can get the same sample again

nh2 <- NHANES %>%
    filter(SurveyYr == "2011_12") %>%
    select(ID, SurveyYr, Age, Height, Weight, BMI, Pulse,
           SleepHrsNight, BPSysAve, BPDiaAve, Gender,
           PhysActive, SleepTrouble, Smoke100,
           Race1, HealthGen, Depressed) %>%
    rename(SleepHours = SleepHrsNight, Sex = Gender,
           SBP = BPSysAve, DBP = BPDiaAve) %>%
    filter(Age > 20 & Age < 80) %>% ## ages 21-79 only
    drop_na() %>% # removes all rows with NA
    sample_n(., size = 1000) %>% # sample 1000 rows
    clean_names() # from the janitor package (snake case)
```

Codebook for nh2 (ID and Quantitative Variables)

Name	Description
id	Identifying code for each subject
survey_yr	2011_12 for all, indicates administration date
age	Age in years at screening of subject (must be 21-79)
height	Standing height in cm
weight	Weight in kg
bmi	Body mass index ($\frac{weight}{(height_{meters})^2}$ in $\frac{kg}{m^2}$)
pulse	60 second pulse rate
sleep_hrs	Self-reported hours (usually gets) per night
sbp	Systolic Blood Pressure (mm Hg)
dbp	Diastolic Blood Pressure (mm Hg)

Codebook for nh2 (Categorical Variables)

Binary Variables

Name	Levels	Description
sex	F, M	Sex of study subject
phys_active	No, Yes	Moderate or vigorous sports/recreation?
sleep_trouble	No, Yes	Has told a provider about trouble sleeping?
smoke100	No, Yes	Smoked at least 100 cigarettes in lifetime?

Multi-Categorical Variables

Name	Levels	Description
race1	5	Self-reported Race/Ethnicity
health_gen	5	Self-reported overall general health
depressed	3	How often subject felt depressed in last 30d

A Look at Body-Mass Index

Let's look at the *body-mass index*, or BMI. The definition of BMI for adult subjects (which is expressed in units of kg/m^2) is:

$$\text{BMI} = \frac{\text{weight in kg}}{(\text{height in meters})^2} = 703 \times \frac{\text{weight in pounds}}{(\text{height in inches})^2}$$

BMI is, essentially, a measure of a person's *thinnness* or *thickness*.

- BMI from 18.5 to 25 indicates optimal weight
- BMI below 18.5 suggests person is underweight
- BMI above 25 suggests overweight.
- BMI above 30 suggests obese.

A First Set of Exploratory Questions

Variables of Interest: `bmi`, `phys_active`, `health_gen`, `pulse`

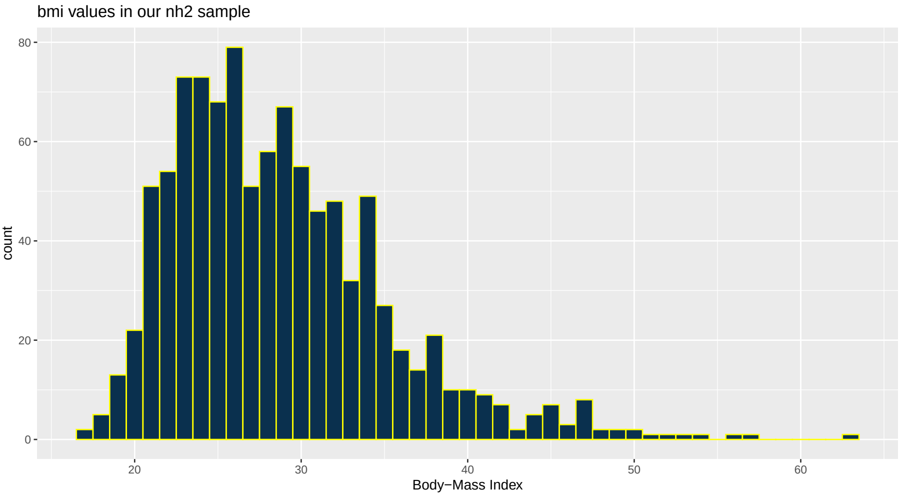
1. What is the distribution of `bmi` in our `nh2` sample of adults?
2. How does the distribution of `bmi` vary by whether the subject is physically active?
3. How does the distribution of `bmi` vary by the subject's self-reported general health?
4. What is the association between `bmi` and the subject's pulse rate?
5. Does that `bmi-pulse` association differ in subjects who are physically active, and those who are not?

Note: These are NOT what anyone would call research questions, which involve generating scientific hypotheses, among other things. These are merely triggers for visualizations and (small) analyses.

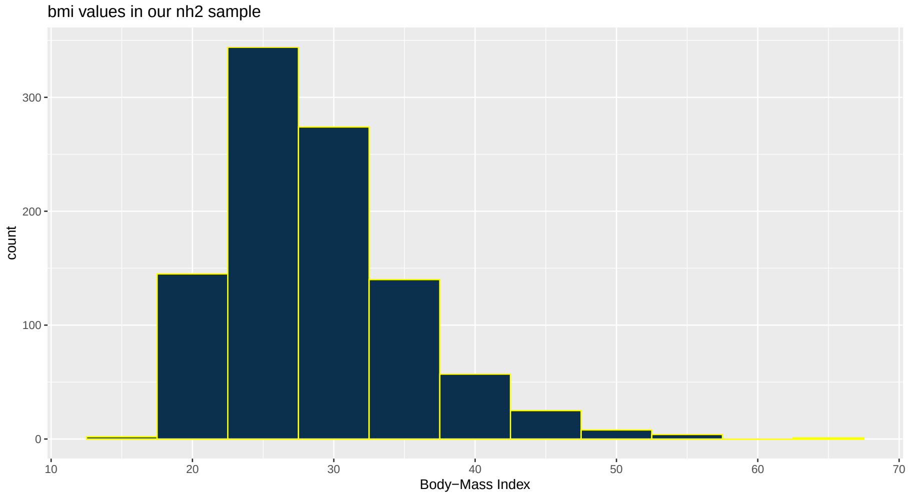
Histogram of BMI in nh2 with binwidth = 1

```
ggplot(nh2, aes(x = bmi)) +
    geom_histogram(binwidth = 1, fill = cwru.blue,
                   col = "yellow") +
    labs(title = "bmi values in our nh2 sample",
         x = "Body-Mass Index")
```

Histogram of BMI in `nh2` with binwidth = 1



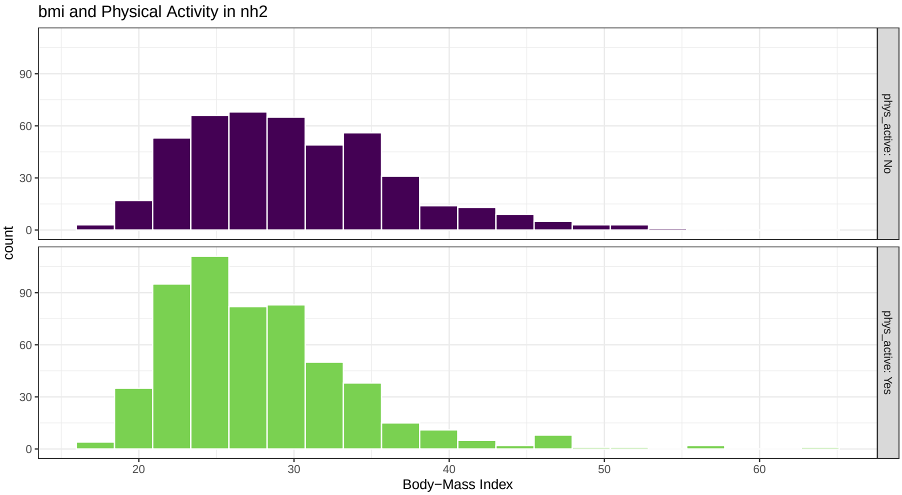
Histogram of BMI in `nh2` with binwidth = 5



BMI Histograms faceted by Physical Activity Status

```r
ggplot(nh2, aes(x = bmi, fill = phys_active)) +
    geom_histogram(bins = 20, col = "white") +
    labs(title = "bmi and Physical Activity in nh2",
         x = "Body-Mass Index") +
    scale_fill_viridis_d(end = 0.8) +
    guides(fill = FALSE) +
    theme_bw() +
    facet_grid(phys_active ~ ., labeller = "label_both")
```

BMI Histograms faceted by Physical Activity Status



Average BMI by Physical Activity Status, I

Create a tibble that helps us answer:

- What is the “average” BMI in each activity group?
- How many people fall into each activity group?

```
nh2 %>%
    group_by(phys_active) %>%
    summarize(count = n(), mean(bmi), median(bmi))
```

```
# A tibble: 2 x 4
  phys_active count `mean(bmi)` `median(bmi)`
  <fct>       <int>       <dbl>         <dbl>
1 No            456        30.0          28.9
2 Yes           544        27.7          26.4
```

Average BMI by Physical Activity Status, II

Making this look a bit more presentable as a table...

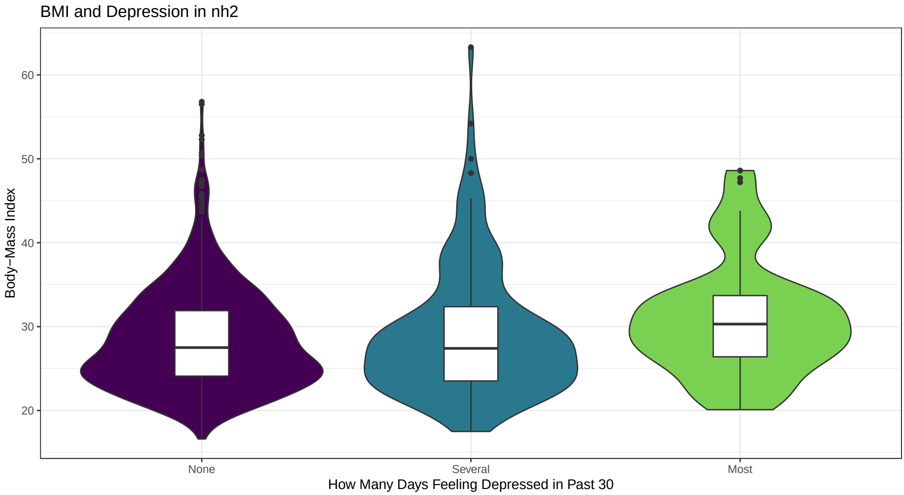
```
nh2 %>%
    group_by(phys_active) %>%
    summarize("Count" = n(),
              "Mean(BMI)" = round(mean(bmi),2),
              "Median(BMI)" = median(bmi)) %>%
    knitr::kable()
```

phys_active	Count	Mean(BMI)	Median(BMI)
No	456	29.98	28.90
Yes	544	27.73	26.45

BMI by Depression Status: Violin Plot

```r
ggplot(nh2, aes(x = depressed, y = bmi, fill = depressed)) +
    geom_violin() +
    geom_boxplot(width = 0.2, fill = "white") +
    labs(title = "BMI and Depression in nh2",
         y = "Body-Mass Index",
         x = "How Many Days Feeling Depressed in Past 30") +
    scale_fill_viridis_d(end = 0.8) +
    guides(fill = FALSE) +
    theme_bw()
```

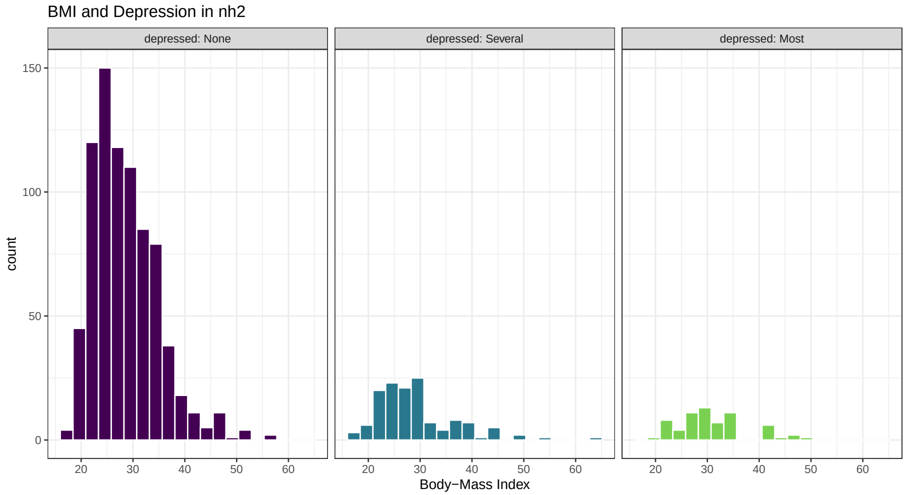
BMI by Depression Status: Violin Plot



BMI by Depression Status, Faceted Histograms

```r
ggplot(nh2, aes(x = bmi, fill = depressed)) +
    geom_histogram(bins = 20, col = "white") +
    labs(title = "BMI and Depression in nh2",
         x = "Body-Mass Index") +
    scale_fill_viridis_d(end = 0.8) +
    guides(fill = FALSE) +
    theme_bw() +
    facet_wrap(~ depressed, labeller = "label_both")
```

BMI by Depression Status, Faceted Histograms



BMI by Depression Status, Numerically

```r
nh2 %>%
    group_by(depressed) %>%
    summarize("Count" = n(),
              "Mean(BMI)" = round(mean(bmi),2),
              "Median(BMI)" = median(bmi)) %>%
    knitr::kable()
```

depressed	Count	Mean(BMI)	Median(BMI)
None	801	28.53	27.5
Several	134	29.12	27.4
Most	65	30.89	30.3

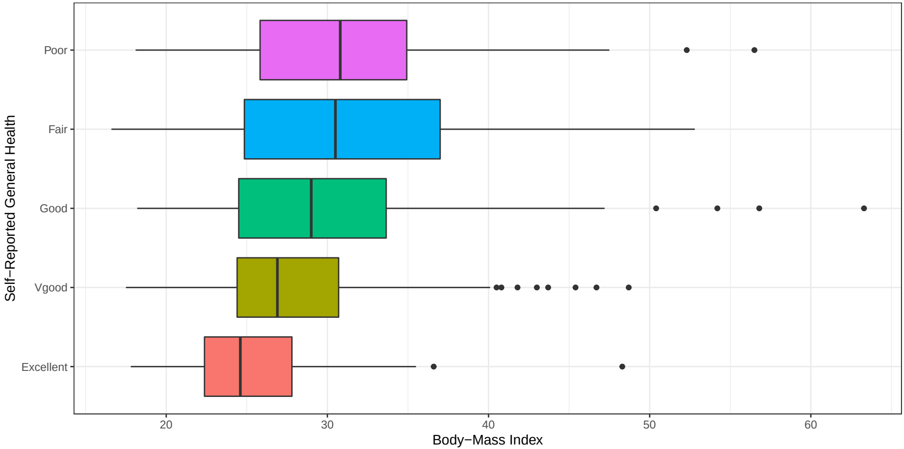
BMI by Self-Reported Health Status

```r
ggplot(nh2, aes(x = health_gen, y = bmi,
                fill = health_gen)) +
    geom_boxplot() +
    theme_bw() +
    coord_flip() +
    guides(fill = FALSE) +
    labs(title = "BMI by Self-Reported General Health",
         subtitle = "1,000 NHANES Subjects in nh2",
         x = "Self-Reported General Health",
         y = "Body-Mass Index")
```

BMI by Self-Reported Health Status

BMI by Self−Reported General Health

1,000 NHANES Subjects in nh2



BMI by Self-Reported Health Status

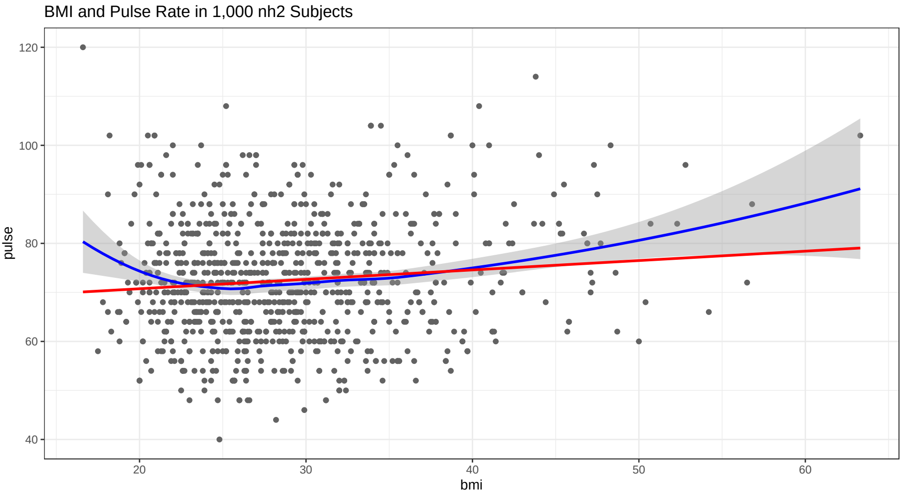
```
nh2 %>%
    group_by(health_gen) %>%
    summarize(count = n(), mean(bmi),
              median(bmi), sd(bmi)) %>%
    knitr::kable(digits = 2)
```

health_gen	count	mean(bmi)	median(bmi)	sd(bmi)
Excellent	144	25.47	24.6	4.51
Vgood	329	27.86	26.9	5.14
Good	383	29.62	29.0	6.76
Fair	124	31.69	30.5	7.83
Poor	20	32.56	30.8	9.80

Association of BMI and Pulse Rate

```r
ggplot(nh2, aes(x = bmi, y = pulse)) +
    geom_point(col = cwru.gray) +
    geom_smooth(method = "loess", se = TRUE, col = "blue") +
    geom_smooth(method = "lm", se = FALSE, col = "red") +
    theme_bw() +
    labs(title = "BMI and Pulse Rate in 1,000 nh2 Subjects")
```

Association of BMI and Pulse Rate



Correlation Coefficient to Summarize Association?

The Pearson correlation coefficient is a very limited measure. It only describes the degree to which a **linear** relationship is present in the data. But we can look at it.

```
nh2 %$% cor(bmi, pulse)
```

```
[1] 0.1076127
```

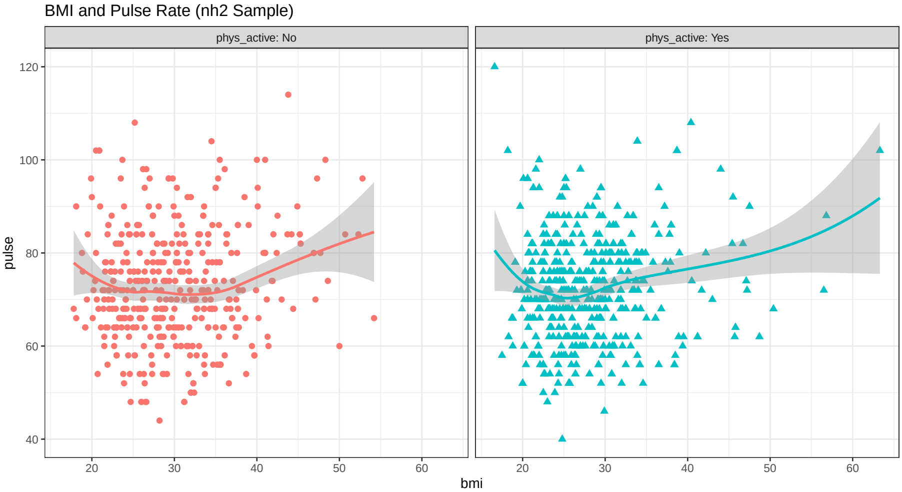
- The Pearson correlation ranges from -1 (perfect negative [as x rises, y falls] linear relationship) to +1 (perfect positive [as x rises, y rises] linear relationship.)
- Our correlation is pretty close to zero. This implies we have a very weak linear association in this case, across the entire sample.

Does Physical Activity affect the Pulse-BMI Association?

Let's change the shape and color of the points based on physical activity status.

```r
ggplot(data = nh2, aes(x = bmi, y = pulse,
                       color = phys_active,
                       shape = phys_active)) +
    geom_point(size = 2) +
    geom_smooth(method = "loess") +
    guides(color = FALSE, shape = FALSE) +
    labs(title = "BMI and Pulse Rate (nh2 Sample)") +
    facet_wrap(~ phys_active, labeller = "label_both") +
    theme_bw()
```

Does Physical Activity affect the Pulse-BMI Association?



Correlation(BMI, pulse) by Physical Activity?

- The Pearson correlation coefficient for the relationship between bmi and pulse in the full sample was quite weak, specifically, it was 0.108.
- Grouped by physical activity status, do we get a different story?

```
nh2 %>%
    group_by(phys_active) %>%
    summarize(cor(bmi, pulse)) %>%
    knitr::kable(digits = 3)
```

phys_active	cor(bmi, pulse)
No	0.101
Yes	0.114

Working with a Categorical Outcome (Self-Reported General Health) in NHANES

General Health Status

Here's a Table of the General Health Status results. This is a self-reported rating of each subject's health on a five point scale (Excellent, Very Good, Good, Fair, Poor.)

```
nh2 %>%
    select(health_gen) %>%
    table() %>%
    addmargins()
```

```
.
Excellent     Vgood      Good      Fair      Poor
      144       329       383       124        20
      Sum
     1000
```

The `health_gen` data are categorical, which means that summarizing them with averages isn't as appealing as looking at percentages, proportions and rates.

Using `tabyl` instead

I actually prefer to use `tabyl` from the `janitor` package, whenever I can.

```
nh2 %>%
    tabyl(health_gen)
```

```
 health_gen   n percent
  Excellent 144   0.144
      Vgood 329   0.329
       Good 383   0.383
       Fair 124   0.124
       Poor  20   0.020
```

This produces a tibble of the information, which can then be manipulated.

Neatening Up the tabyl

```r
nh2 %>%
    tabyl(health_gen) %>%
    adorn_pct_formatting() %>%
    knitr::kable()
```

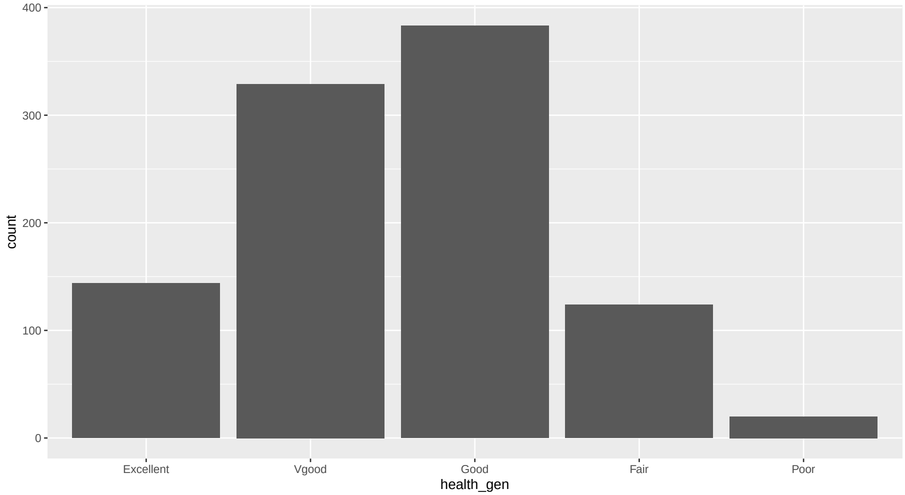
health_gen	n	percent
Excellent	144	14.4%
Vgood	329	32.9%
Good	383	38.3%
Fair	124	12.4%
Poor	20	2.0%

Bar Chart for Categorical Data

Usually, a **bar chart** is the best choice for a graphing a variable made up of categories.

```
ggplot(data = nh2, aes(x = health_gen)) +
    geom_bar()
```

Original Bar Chart of General Health

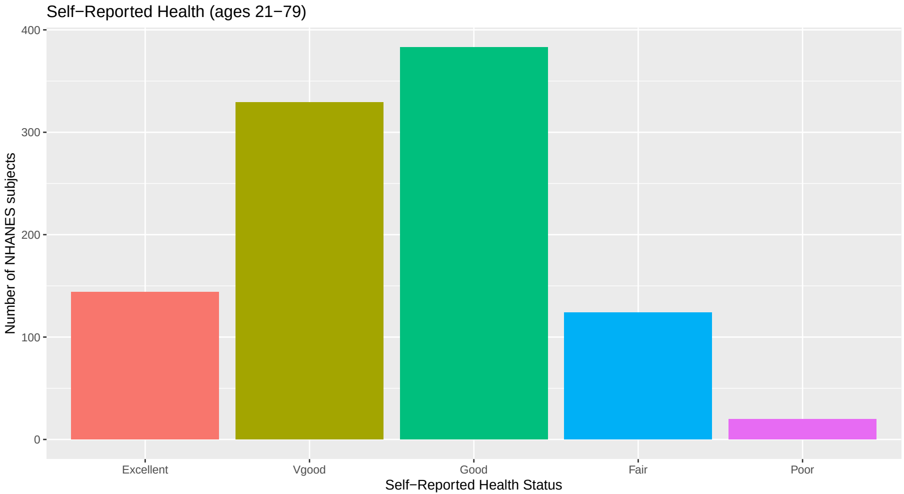


Improving the Bar Chart

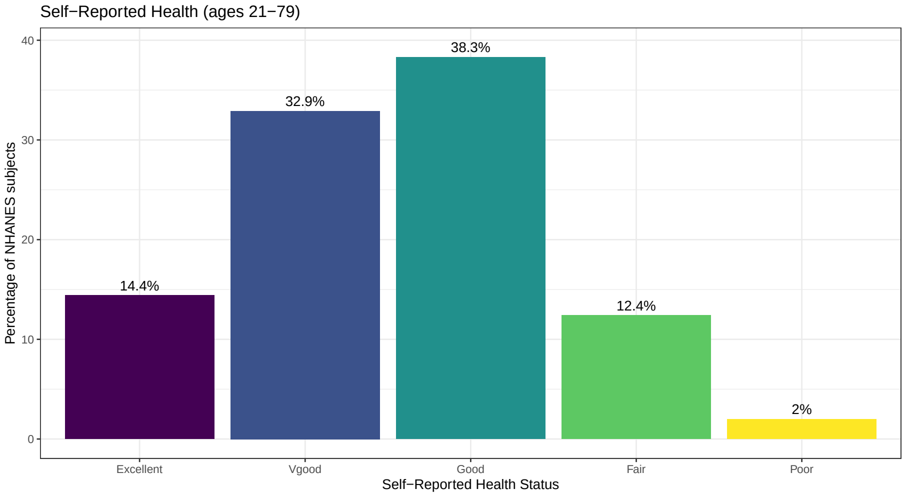
There are lots of things we can do to make this plot fancier.

```
ggplot(data = nh2,
       aes(x = health_gen, fill = health_gen)) +
    geom_bar() +
    guides(fill = FALSE) +
    labs(x = "Self-Reported Health Status",
         y = "Number of NHANES subjects",
         title = "Self-Reported Health (ages 21-79)")
```

The Improved Bar Chart



Or, we can really go crazy... (code on next slide)



What crazy looks like. . .

```r
nh2 %>%
    count(health_gen) %>%
    ungroup() %>%
    mutate(pct = round(prop.table(n) * 100, 1)) %>%
    ggplot(aes(x = health_gen, y = pct, fill = health_gen)) +
    geom_bar(stat = "identity", position = "dodge") +
    scale_fill_viridis_d() +
    guides(fill = FALSE, col = FALSE) +
    geom_text(aes(y = pct + 1,    # nudge above top of bar
                  label = paste0(pct, '%')),  # prettify
              position = position_dodge(width = .9),
              size = 4) +
    labs(x = "Self-Reported Health Status",
         y = "Percentage of NHANES subjects",
         title = "Self-Reported Health (ages 21-79)") +
    theme_bw()
```

Working with Tables

We can add a marginal total, and compare subjects by sex, as follows...

```
nh2 %>%
    select(sex, health_gen) %>%
    table() %>%
    addmargins() %>%
    knitr::kable()
```

	Excellent	Vgood	Good	Fair	Poor	Sum
female	73	165	179	48	12	477
male	71	164	204	76	8	523
Sum	144	329	383	124	20	1000

Or use `tabyl`

```r
nh2 %>%
  tabyl(sex, health_gen)
```

```
    sex Excellent Vgood Good Fair Poor
 female        73   165  179   48   12
   male        71   164  204   76    8
```

We can "adorn" the tabyl

```r
nh2 %>%
  tabyl(sex, health_gen) %>%
  adorn_totals(where = c("row", "col"))
```

```
    sex Excellent Vgood Good Fair Poor Total
 female        73   165  179   48   12   477
   male        71   164  204   76    8   523
  Total       144   329  383  124   20  1000
```

We can "adorn" the `tabyl` in several ways

```
nh2 %>%
  tabyl(sex, health_gen) %>%
  adorn_totals() %>% # note default is row totals only
  adorn_title()
```

```
       health_gen
    sex  Excellent Vgood Good Fair Poor
 female         73   165  179   48   12
   male         71   164  204   76    8
  Total        144   329  383  124   20
```

Getting Percentages by Column in each Row

```r
nh2 %>%
  tabyl(sex, health_gen) %>%
  adorn_totals(where = "row") %>%
  adorn_percentages(denominator = "row") %>%
  adorn_pct_formatting(digits = 1) %>%
  adorn_title()
```

```
        health_gen
    sex  Excellent Vgood  Good  Fair Poor
 female      15.3% 34.6% 37.5% 10.1% 2.5%
   male      13.6% 31.4% 39.0% 14.5% 1.5%
  Total      14.4% 32.9% 38.3% 12.4% 2.0%
```

Getting Percentages by Row in each Column

```
nh2 %>%
  tabyl(sex, health_gen) %>%
  adorn_totals(where = "col") %>%
  adorn_percentages(denominator = "col") %>%
  adorn_pct_formatting(digits = 1) %>%
  adorn_title()
```

```
        health_gen                                   
    sex  Excellent Vgood  Good  Fair  Poor Total
 female      50.7% 50.2% 46.7% 38.7% 60.0% 47.7%
   male      49.3% 49.8% 53.3% 61.3% 40.0% 52.3%
```

Percentages and Counts by Column in each Row

```r
nh2 %>%
  tabyl(sex, health_gen) %>%
  adorn_totals(where = "row") %>%
  adorn_percentages(denominator = "row") %>%
  adorn_pct_formatting(digits = 1) %>%
  adorn_ns(position = "front")
```

```
    sex   Excellent       Vgood        Good
 female  73 (15.3%) 165 (34.6%) 179 (37.5%)
   male  71 (13.6%) 164 (31.4%) 204 (39.0%)
  Total 144 (14.4%) 329 (32.9%) 383 (38.3%)
        Fair       Poor
  48 (10.1%) 12 (2.5%)
  76 (14.5%)  8 (1.5%)
 124 (12.4%) 20 (2.0%)
```

Old Way to get Row Proportions

We'll use prop.table and get the row proportions by feeding it a 1.

```
nh2 %>%
    select(sex, health_gen) %>%
    table() %>%
    prop.table(.,1) %>%
    round(.,2) %>%
    knitr::kable()
```

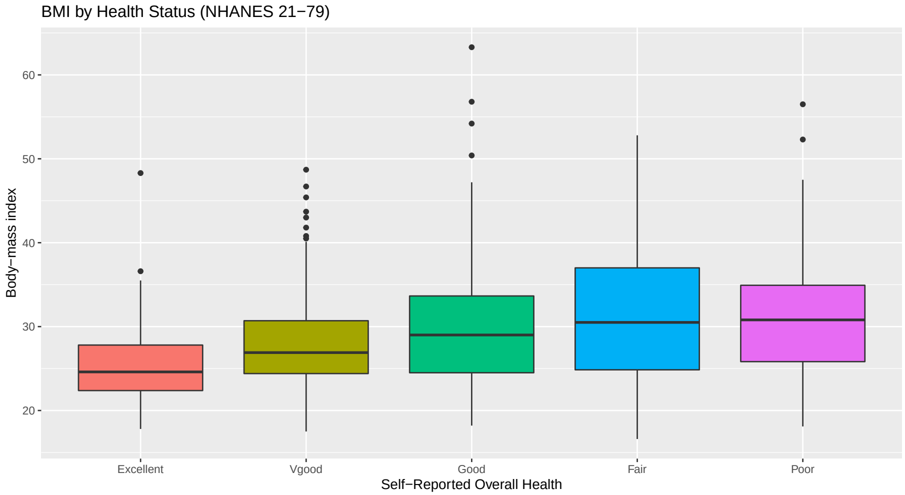
	Excellent	Vgood	Good	Fair	Poor
female	0.15	0.35	0.38	0.10	0.03
male	0.14	0.31	0.39	0.15	0.02

BMI by General Health Status

Let's consider now the relationship between self-reported overall health and body-mass index.

```r
ggplot(data = nh2,
       aes(x = health_gen, y = bmi, fill = health_gen)) +
    geom_boxplot() +
    labs(title = "BMI by Health Status (NHANES 21-79)",
         y = "Body-mass index",
         x = "Self-Reported Overall Health") +
    guides(fill = FALSE)
```

What happens with the `Poor` category?



Summary Table of BMI distribution by health_gen

```r
nh2 %>%
    group_by(health_gen) %>%
    summarize("BMI n" = n(),
              "Mean" = round(mean(bmi),1),
              "SD" = round(sd(bmi),1),
              "min" = round(min(bmi),1),
              "Q25" = round(quantile(bmi, 0.25),1),
              "median" = round(median(bmi),1),
              "Q75" = round(quantile(bmi, 0.75),1),
              "max" = round(max(bmi),1)) %>%
    knitr::kable()
```

- Resulting table is shown in the next slide.

Not many self-identify in the `Poor` category

health_gen	BMI n	Mean	SD	min	Q25	median	Q75	max
Excellent	144	25.5	4.5	17.8	22.4	24.6	27.8	48.3
Vgood	329	27.9	5.1	17.5	24.4	26.9	30.7	48.7
Good	383	29.6	6.8	18.2	24.5	29.0	33.7	63.3
Fair	124	31.7	7.8	16.6	24.8	30.5	37.0	52.8
Poor	20	32.6	9.8	18.1	25.8	30.8	34.9	56.5

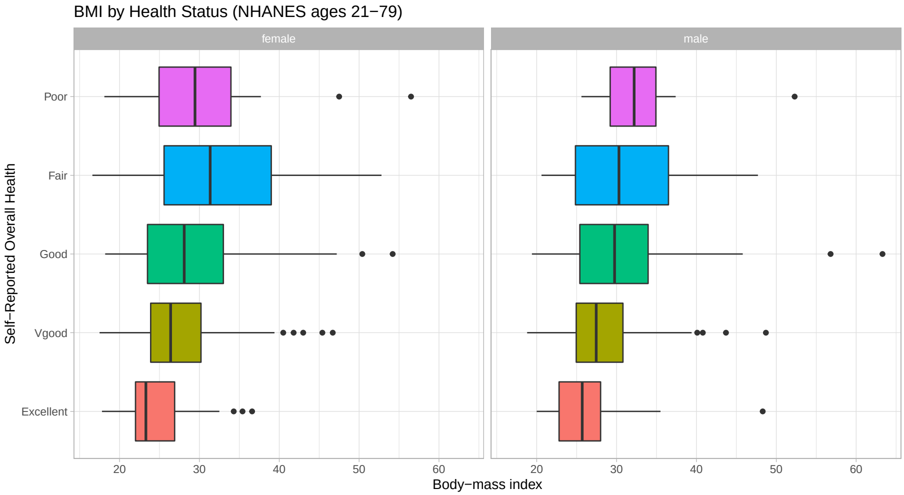
BMI by Sex and General Health Status

We'll start with two panels of boxplots to try to understand the relationships between BMI, General Health Status and Sex

```
ggplot(data = nh2,
       aes(x = health_gen, y = bmi, fill = health_gen)) +
    geom_boxplot() +
    guides(fill = FALSE) +
    facet_wrap(~ sex) +
    coord_flip() +
    theme_light() +
    labs(title = "BMI by Health Status (NHANES ages 21-79)",
         y = "Body-mass index",
         x = "Self-Reported Overall Health")
```

- Note the use of `coord_flip` to rotate the graph 90 degrees.
- Note the use of a new theme, called `theme_light()`.

BMI by Sex and General Health Status Boxplots



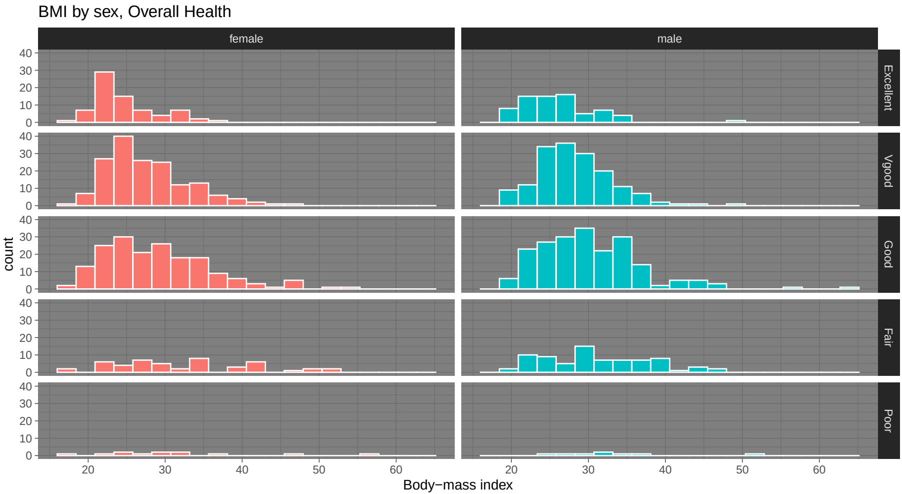
Histograms of BMI by Health and sex

Here are doubly faceted histograms, which can help address similar questions.

```
ggplot(data = nh2,
       aes(x = bmi, fill = sex)) +
    geom_histogram(color = "white", bins = 20) +
    labs(title = "BMI by sex, Overall Health",
         x = "Body-mass index") +
    guides(fill = FALSE) +
    facet_grid(health_gen ~ sex) +
    theme_dark()
```

- Note the use of `facet_grid` to specify rows and columns.
- Note the use of a new theme, called `theme_dark()`.

Histograms of BMI by Health and sex



Conclusions

This is just a small piece of the toolbox for visualizations that we'll create in this class. Many additional tools are on the way, but the main idea won't change. Using the `ggplot2` package, we can accomplish several critical tasks in creating a visualization, including:

- Identifying (and labeling) the axes and titles
- Identifying a type of `geom` to use, like a point, bar or histogram
- Changing fill, color, shape, size to facilitate comparisons
- Building "small multiples" of plots with faceting

Good data visualizations make it easy to see the data, and `ggplot2`'s tools make it relatively difficult to make a really bad graph.