Using the Illinois Report Card Data to Teach Statistics

MMC Conference of Workshops

Frank Briody | Prospect High School | frankbriody@gmail.com

2/1/2020

Contents

1 Variables

The ISBE raw data file *rc17.txt* contains 1,471 variables. The variable definitions are in the Excel file *RC17_layout.xlsx* and have been categorized into the groups shown below. The first number represents available variables in each group while the second is the number actually imported into the processed data file. The import script produces 316 variables from 20 of the 21 categories for all 3,796 Illinois public schools. (None of the NAEP variables were imported.) Usable files will be discussed in section 4.

School information (13 variables;12 imported)
Student demographics (396;71)
ACT (44;11)
Instructional setting (92;2)
Teacher and admin statistics (78;26)
District financial (67;40)
Region and legislative (3;2)
National Assmnt. of Educ. Progress (NAEP) (184;0)
College and Career readiness (16;3) CTE (4;1)
Advanced coursework (12;3)

AP courses (168;42)
IB courses (168;42)
Dual credit (168;42)
AP exams (36;12)
Post secondary remediation (4;1)
Response rate (5E survey) (4;2)
Health and wellness (3;1)
Teacher attendance (4;1)
Teacher evaluation (2;1)
School district count (3;1)

---

2 Descriptive Statistics via State Demographics

2.1 Categorical Count (Raw)

```
school_type <- rc17 %>%
  count(SCHOOL_TYPE_NAME, sort = TRUE) %>%
  mutate(rel_freq = n/sum(n))
school_type
```

```
## # A tibble: 4 x 3
##   SCHOOL_TYPE_NAME     n rel_freq
##   <chr>            <int>    <dbl>
## 1 ELEMENTARY        2406   0.634
## 2 HIGH SCHOOL        644   0.170
## 3 MIDDLE SCHL        604   0.159
## 4 CHARTER SCH        142   0.0374
```

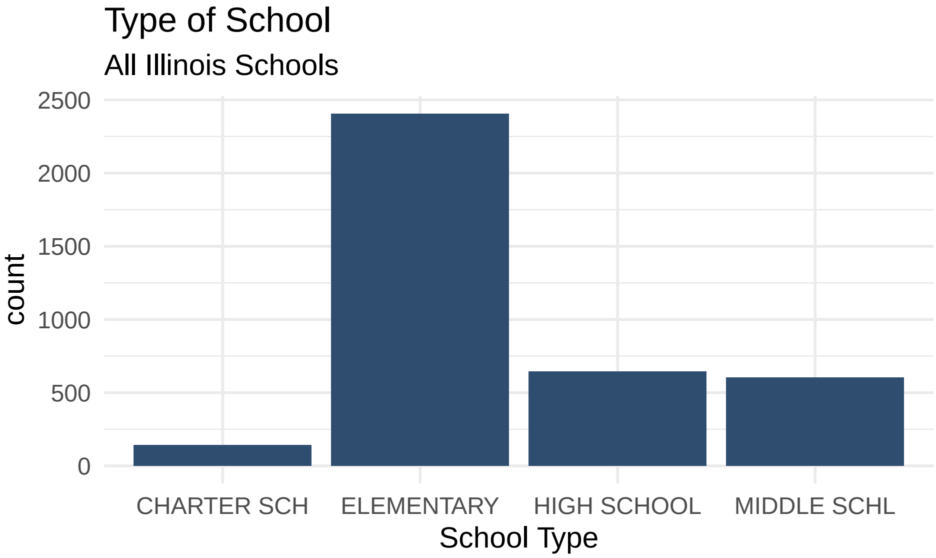
2.2 Categorical Count (Formatted)

```
kable(school_type) %>%
  kable_styling(bootstrap_options = "striped", full_width = FALSE)
```

SCHOOL_TYPE_NAME	n	rel_freq
ELEMENTARY	2406	0.6338251
HIGH SCHOOL	644	0.1696523
MIDDLE SCHL	604	0.1591149
CHARTER SCH	142	0.0374078

2.3 Categorical Plot

```
ggplot(rc17, aes(x=factor(SCHOOL_TYPE_NAME))+
  geom_bar(fill="#2F4E6F")+
  labs(title = "Type of School", x = "School Type", subtitle = "All Illinois Schools") +
  theme_minimal()
```



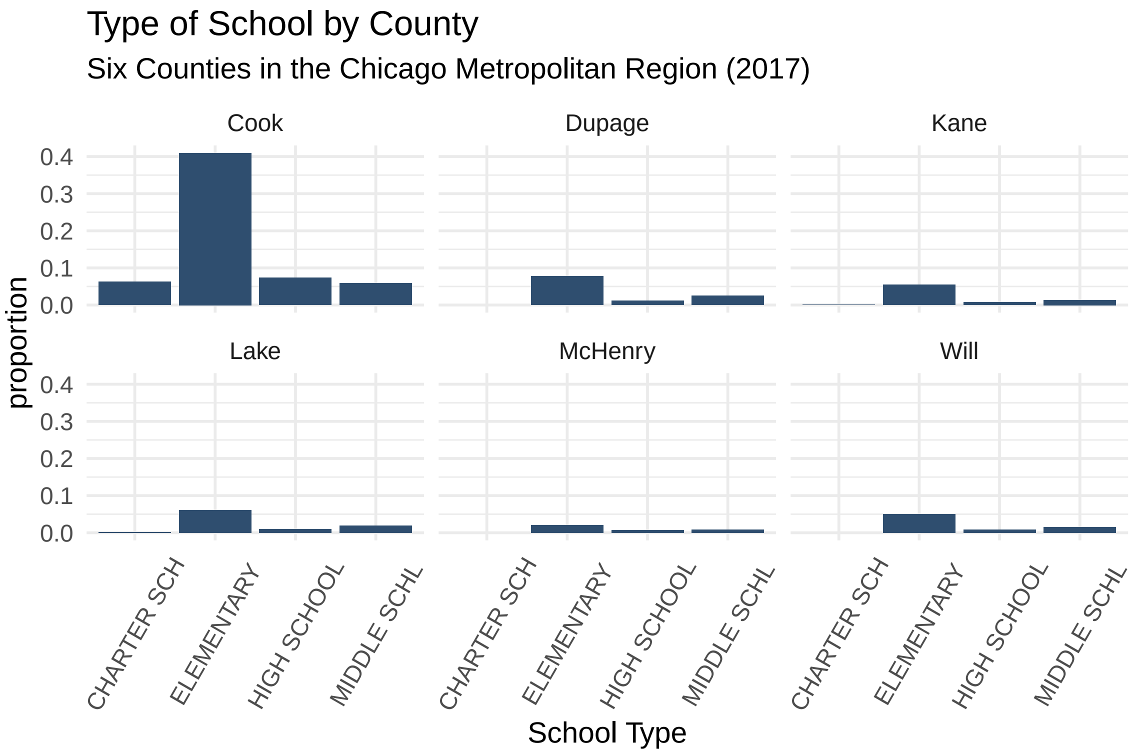
2.4 Categorical Analysis

Example 1

Write a short analysis for the types of schools in the state of Illinois.

Example 2

```
rc17 %>%
filter(COUNTY == "Dupage" | COUNTY == "Will" | COUNTY == "Kane" |
         COUNTY == "Lake" | COUNTY == "Cook"| COUNTY == "McHenry") %>%
ggplot(aes(x=factor(SCHOOL_TYPE_NAME), y = (..count..)/sum(..count..))) +
  geom_bar(fill="#2F4E6F")+
  facet_wrap(~COUNTY, nrow = 2) +
  labs(title = "Type of School by County",
       x = "School Type",
       y = "proportion",
       subtitle = "Six Counties in the Chicago Metropolitan Region (2017)") +
  theme_minimal()  +
  theme(axis.text.x = element_text(angle = 60, vjust = 0.5))
```



Write a short analysis for the types of schools in the six county region.

3 Software

3.1 R and RStudio

R is open source, free, industry-standard statistical analysis software that was first introduced in 1993. (R is an adaptation of the S languge that was invented at Bell Labs in 1976.) RStudio, which arrived in 2011, is a free development environment for using R. Both R and RStudio have a bit of an initial learing curve, but mastering a few basic commands opens up a world of analysis options. The RStudio installation page https://rstudio.com/products/rstudio/download/#download provides instructions for setting up both R and RStudio. Suggestions for learning R are in the Appendix.

3.2 Fathom

Key Curriculum Press, the creators of The Geometer's Sketchpad software, sells Fathom ($39 USD) at https://fathom.concord.org. Fathom is graphical statistical analysis software which accomplishes most tasks through a drag and drop interface. Its primary audience is teachers and students, but has reached end-of-life status. It can still be purchased and used but no further developments are expected.

3.3 Others

Excel, JMP, and SAS are other software platforms that can be used for analysis but will not be discussed here. The report card data will be provided in .csv format which can be imported by these programs.

4 Data Import

4.1 Data Files

1. **Super Easy Method** All graphs from this presentation are available for download at https://github.com/fbriody/MMC2020/tree/master/EDA__present__files/figure-html.
2. **Easy Method** A reasonably sized (2.7MB) data file containing 316 variables for the 2,049 schools in the Chicagoland six county region is avaialble at http://frankbriody.com/rc17__data.zip. The six counties are Cook, Dupage, Kane, Lake, McHenry and Will. This data file can be imported into RStudio, Fathom, Excel, etc.
3. **Some Variables for All Schools** I used the import script `import_rc17.txt` to select 316 variables for all 3,796 Illinois schools. A 5MB data file is available at http://frankbriody.com/rc17__data.zip. Again, this data file can be imported into RStudio, Fathom, Excel, etc.
4. **Starting from Scratch** The original raw data is available on the ISBE Report Card Data Library web page https://www.isbe.net/Pages/Illinois-State-Report-Card-Data.aspx. You will need to download both the fixed width data file (rc17.txt 2.4MB becomes 35.4MB) and the variable definitions (RC17__layout.xlsx). Use and/or modify import_rc17.txt https://github.com/fbriody/MMC2020 to get a subset of the data into RStudio.

A note about files and file names:

- The original ISBE data is in a file named `rc17.txt`.
- An import command is used in RStudio to produce a dataframe named `rc17`. This dataframe is a container within RStudio and is not a separate external file. This import command is available as `rc17_import.txt` and is available at https://github.com/fbriody/MMC2020. Filtering or modifying the dataframe within RStudio does not write changes to the original `rc17.txt` file.
- After a subset of the original `rc17.txt` datafile was imported into RStudio, a subset was exported as `rc17.csv`. It is important to note that this .csv file does **NOT** contain all of the original ISBE variables - only 316 variables for all Illinois schools. This subset is also available at http://frankbriody.com/rc17__data.zip.
- RStudio (or other software) can be used to export a dataframe or a filtered subset of a dataframe to .csv. A six county subset, `sixco.csv` is included in the data file linked above.

Subsetting and exporting is a two step process. First, use R to create the subset:

```
sixco <- rc17 %>%
  filter(COUNTY %in% c("Cook", "Lake", "Will", "Kane", "McHenry", "Dupage"))
```

Then, export:

```
write.csv(sixco,"sixco.csv", row.names = FALSE)
```

The resulting .csv can be imported by another software platform. (Note there is no need to export if you are staying in RStudio. Just refer to your new dataframe subset, in this case `sixco`.) You can customize the above command(s) to suit your needs.

To get a .csv file into R, either use the File menu and Import Dataset, or send the command

```
newdata <- read.csv(file = 'datafile.csv')
```

which creates a `newdata` datframe within RStudio.

4.2 Importing into Other Software

Fathom

Import one of the .csv data files into Fathom by choosing `File -> Import -> Import from File...` and then navigate to the file location.

Excel

Use Excel to import rc17.csv.

5 Numeric Summaries

5.1 Lists

To get the number of High Schools per county in the Six County Region:

```
sixco %>%
  filter(SCHOOL_TYPE_NAME == "HIGH SCHOOL") %>%
  group_by(COUNTY) %>%
  summarise(count = n())
```

```
## # A tibble: 6 x 2
##   COUNTY  count
##   <chr>   <int>
## 1 Cook      151
## 2 Dupage     23
## 3 Kane       16
## 4 Lake       21
## 5 McHenry    14
## 6 Will       17
```

McHenry county seems like a good candidate for small-set data that can be analyzed with a graphing calculator. Subsetting based on a criteria produces a single list of scores.

```
mchenry_act <- rc17 %>%
  filter(COUNTY == "McHenry", is.na(ACT_COMP_SCHOOL) == FALSE ) #don't include missing
mchenry_act$ACT_COMP_SCHOOL
```

```
##  [1] 22.4 19.7 18.1 23.1 22.6 23.8 22.7 24.0 21.1 19.9 22.9 22.9 21.4 21.2
```

Adding `sort()` orders the scores. Remove the comma and the `decreasing` option to produce an increasing list.

```
sort(mchenry_act$ACT_COMP_SCHOOL, decreasing = TRUE)
```

```
##  [1] 24.0 23.8 23.1 22.9 22.9 22.7 22.6 22.4 21.4 21.2 21.1 19.9 19.7 18.1
```

Lake County is a little larger, but a boxplot can be quickly made from an ordered and formatted table of ACT Scores. Create a boxplot for Lake County ACT scores. How could you compare to DuPage county?

```
lake_ACT <- rc17 %>%
  filter(SCHOOL_TYPE_NAME == "HIGH SCHOOL", COUNTY == "Lake") %>%
  arrange(desc(ACT_COMP_SCHOOL)) %>%
  select(COUNTY, SCHOOL_NAME, ACT = ACT_COMP_SCHOOL)
kable(lake_ACT)
```

COUNTY	SCHOOL_NAME	ACT
Lake	Adlai E Stevenson High School	26.9
Lake	Deerfield High School	26.4
Lake	Lake Forest High School	26.3
Lake	Libertyville High School	25.9
Lake	Highland Park High School	25.2
Lake	Vernon Hills High School	25.1
Lake	Lake Zurich High School	24.9
Lake	Barrington High School	24.8
Lake	Grayslake Central High School	23.3
Lake	Lakes Community High School	22.6
Lake	Grayslake North High School	22.4
Lake	Warren Township High School	22.1
Lake	Wauconda High School	21.8
Lake	Antioch Comm High School	21.7
Lake	Mundelein Cons High School	21.4
Lake	Grant Community High School	21.3
Lake	New Tech High - Zion-Benton East	20.1
Lake	Zion-Benton Twnshp Hi Sch	18.6
Lake	Waukegan High School	17.9
Lake	Round Lake Senior High School	17.8
Lake	North Chicago Community High Sch	17.5

5.2 Two-Way Tables

```r
district_type <- rc17 %>%
  filter(COUNTY == "Lake" | COUNTY == "Dupage") %>%
  group_by(COUNTY)

two_way <- with(district_type, table(DISTRICT_SIZE_NAME, COUNTY))

kable(two_way, caption = "Types of School Districts") %>%
  kable_styling(bootstrap_options = "striped", full_width = F, latex_options = "hold_position")
```

Table 1: Types of School Districts

	Dupage	Lake
LARGE	182	150
MEDIUM	52	38
SMALL	0	4

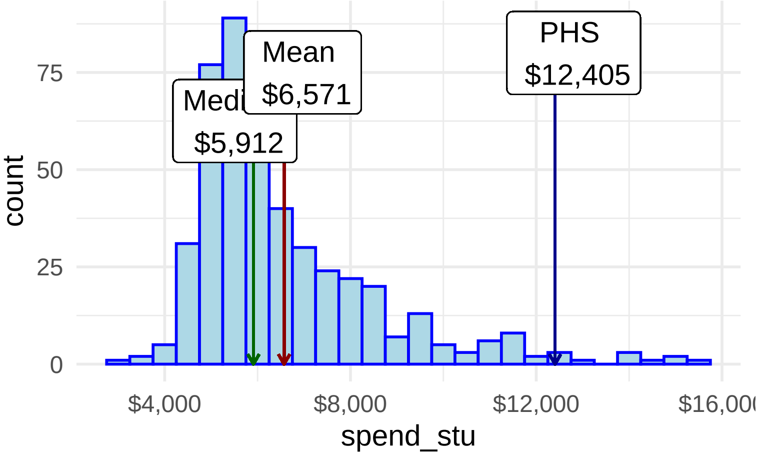
5.3 Mean vs Median

A numeric summary for instructional spending per pupil by district:

```
rc17 %>%
  filter(SCHOOL_TYPE_NAME == "HIGH SCHOOL") %>%
  group_by(DISTRICT_NAME) %>%
  summarize(spend_stu = mean(INSTRUCT_EXPEND_PER_PUPIL_DISTRICT201516, na.rm = TRUE)) %>%
  summary()
```

```
##  DISTRICT_NAME        spend_stu
##  Length:473         Min.   : 2975
##  Class :character   1st Qu.: 5263
##  Mode  :character   Median : 5912
##                     Mean   : 6571
##                     3rd Qu.: 7315
##                     Max.   :15535
##                     NA's   :1
```

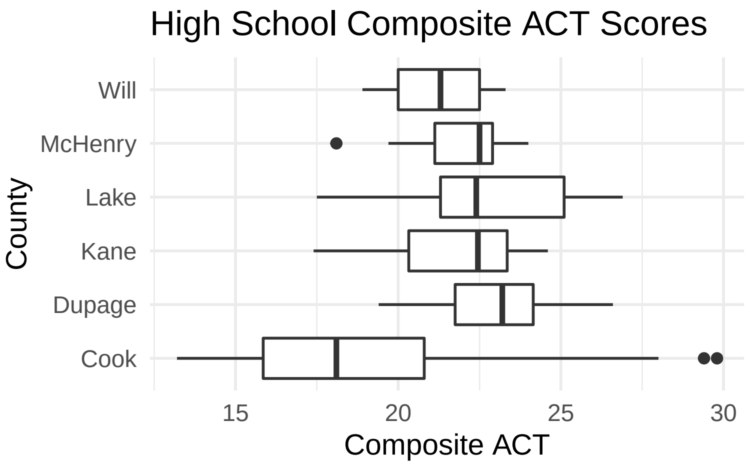
Putting summary numbers on a plot (code available on GitHub):



Boxplots

```
sixco  %>%
  filter(SCHOOL_TYPE_NAME == "HIGH SCHOOL") %>%
  ggplot(mapping = aes(x = COUNTY, y = ACT_COMP_SCHOOL)) +
  geom_boxplot() +
  theme_minimal() +
  coord_flip() + #horizontal boxplots preferred
  labs(x = "County", y = "Composite ACT",
       title = "High School Composite ACT Scores")
```

```
## Warning: Removed 4 rows containing non-finite values (stat_boxplot).
```



Comment on the distribution of ACT scores both within and between counties. For which counties would you expect the mean to be close to the median? Which county or counties would prefer to report the median instead of the mean?

6 Single Values

6.1 Finding a School

```
rc17 %>%
  filter(str_detect(SCHOOL_NAME, "Morton")) %>%
  select(SCHOOL_ID, SCHOOL_NAME, ACT_COMP_SCHOOL)
```

```
## # A tibble: 7 x 3
##   SCHOOL_ID       SCHOOL_NAME                      ACT_COMP_SCHOOL
##   <chr>           <chr>                                      <dbl>
## 1 060162010170001 J Sterling Morton East High Sch             18.4
## 2 060162010170002 J Sterling Morton West High Sch             18.7
## 3 060162010170003 J Sterling Morton Freshman Cntr             NA
## 4 070161450022004 Morton Gingerwood Elem School               NA
## 5 150162990252844 Morton Elem Career Academy                  NA
## 6 530907090260006 Morton High School                          23.3
## 7 530907090261005 Morton Jr High School                       NA
```

6.2 Using a Filter

```
prospect <- rc17 %>%
  filter(str_detect(SCHOOL_NAME, "Prospect High School"))
prospect_act <- prospect$ACT_COMP_SCHOOL
prospect_act
```

```
## [1] 25
```

6.3 Using a Function

If you put this right after the data import step you can always find single values for a specific school quickly.

```
phs_value <- function(unk) {
  x <- rc17 %>%
    filter(SCHOOL_ID == "050162140170005")
  x[unk]
}

phs_value("ACT_COMP_SCHOOL")
```

```
## # A tibble: 1 x 1
##   ACT_COMP_SCHOOL
##             <dbl>
## 1              25
```

6.4 Analysis in Fathom

Drag a table from the shelf, drag a Graph from the shelf then drag a variable (or variables) onto the graph.

Fathom tries to auto detect **variable types**, but you can force a change by holding down `shift` or `option`. Missing values are often stored as character, so creating a scatterplot may require holding down `option`. If a categorical variable is coded as a number, holding down `shift` coerces into categorical.

Double-clicking a value opens up Fathom's **inspecing** box.

6.5 Filters in Excel

Excel has quite powerful filter tools. For reasonably sized data files, it may be efficient to filter and export a .csv then open the subset of data in RStudio.

7 Correlation and Regression

7.1 Guess the Correlation

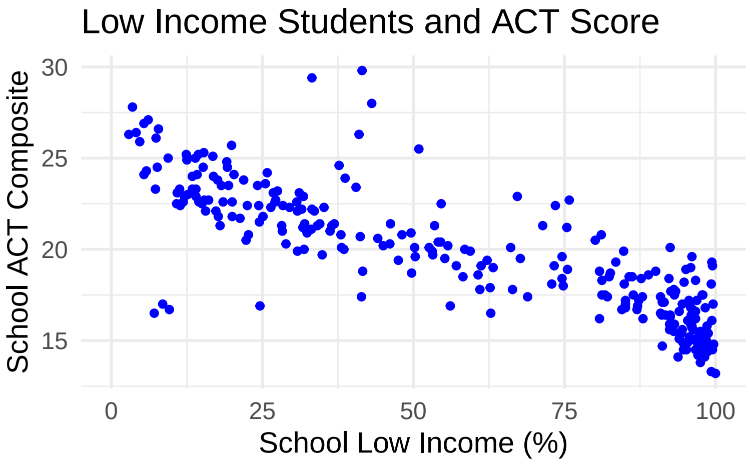
- ACT Composite vs Chronically Truant (#) Guess:________________ Actual:________________
- ACT Composite vs Chronically Truant (%) Guess:________________ Actual:________________
- ACT Composite vs Student Mobility Guess:________________ Actual:________________
- ACT Composite vs Attendance rate (%) Guess:________________ Actual:________________

7.2 Scatterplot Analysis

The analysis of scatterplots should lead to a discussion about outliers, influentials and regression details.

```
ACTvLI <- sixco %>%
  ggplot(mapping = aes(x = LOWINCOME_SCHOOL_per, y = ACT_COMP_SCHOOL)) +
  geom_point(color="Blue", size = 1) +
  labs(title = "Low Income Students and ACT Score", x = "School Low Income (%)", y = "School ACT Composite" )
ACTvLI
```

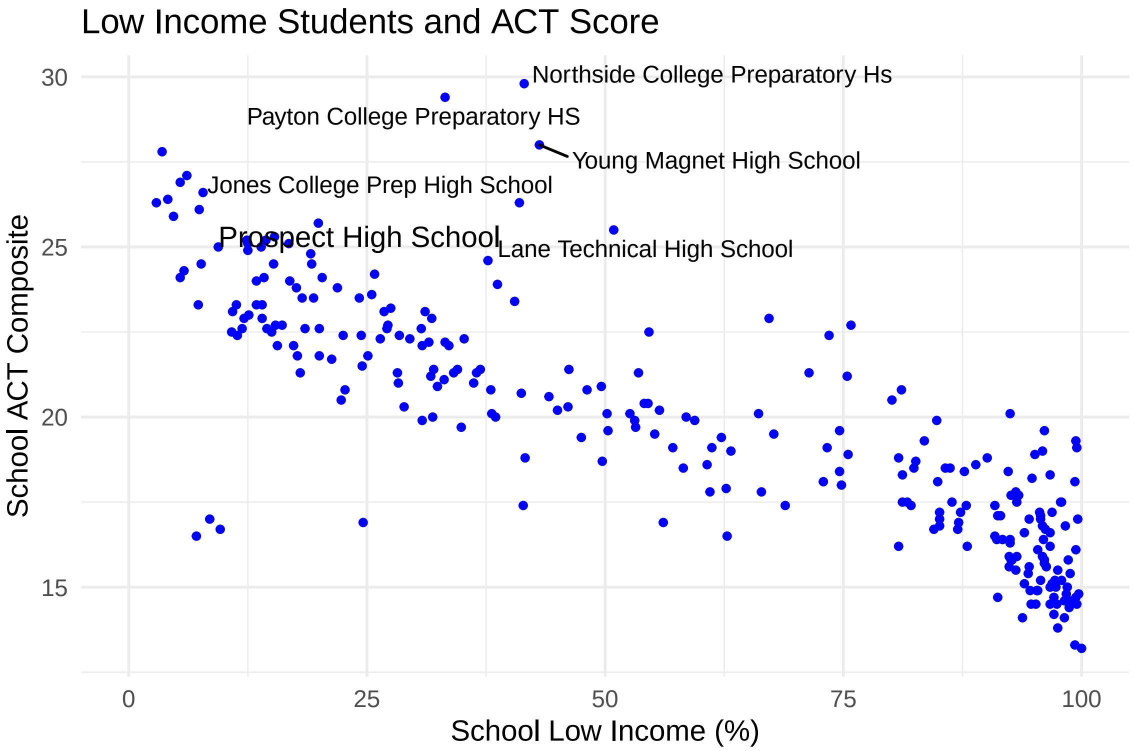
```
## Warning: Removed 1779 rows containing missing values (geom_point).
```



Add some labels using the `ggrepel` package. (The warning about missing values has been removed. Also notice the layering of information.)

```
library(ggrepel)
```

```
ACTvLI +
  geom_text(aes(label=ifelse(SCHOOL_ID =="050162140170005",
                           as.character(SCHOOL_NAME),'')),hjust=0,vjust=0) +
  geom_text_repel(aes(LOWINCOME_SCHOOL_per, ACT_COMP_SCHOOL,
                    label = ifelse(ACT_COMP_SCHOOL >25 & #label criteria
                                   LOWINCOME_SCHOOL_per>25, #label criteria
                                   SCHOOL_NAME, "")), size = 3)
```

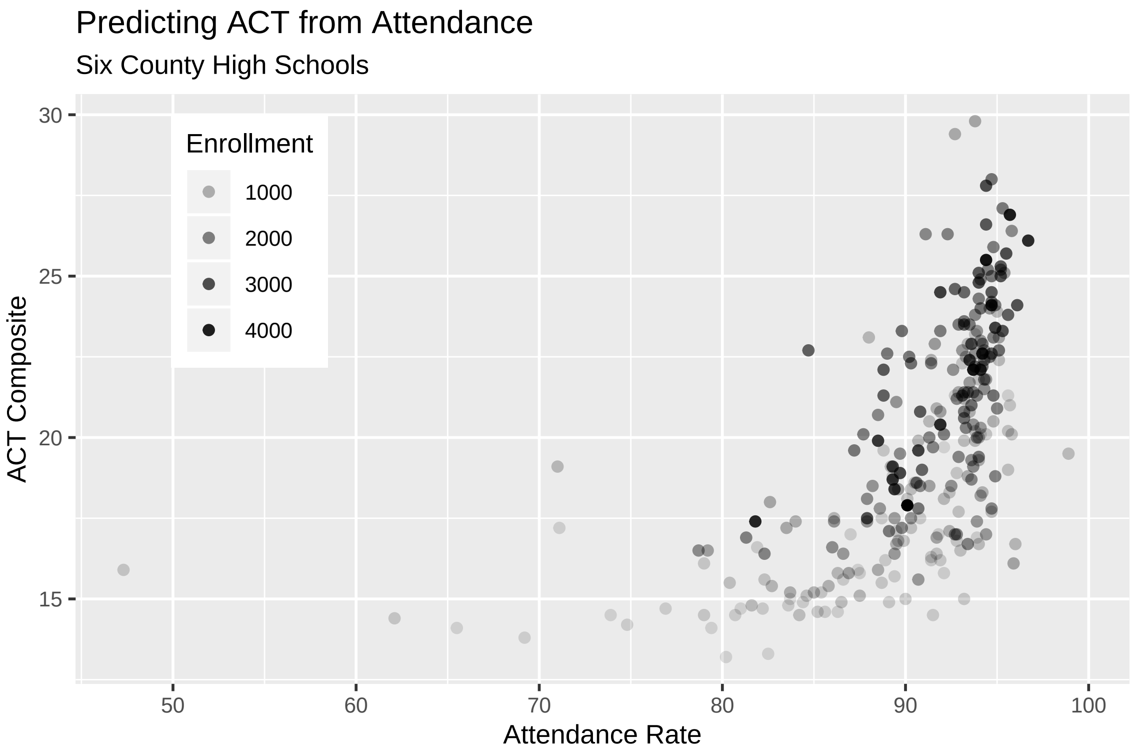


Example: Predicting ACT Scores from Attendance

Suppose you choose 3 variables (Composite ACT Score, Enrollment and Attendance Rate) for all schools in the Six County region. What question(s) and display(s) would you explore?

Adding a third variable introduces another layer of analysis.

```
sixco %>% ggplot(mapping = aes(x= ATTENDANCE_RATE_SCHOOL_perALL, y = ACT_COMP_SCHOOL)) +
  geom_point(aes(alpha = sixco$SCHOOL_TOTAL_ENROLLMENT)) + #alpha is transparency
  labs(alpha = "Enrollment", x = "Attendance Rate",
       y = "ACT Composite", title = "Predicting ACT from Attendance",
       subtitle = "Six County High Schools") +
  theme(legend.position = c(.165, .75), text = element_text(size=10))
```



7.3 Regression Output

Predicting ACT from attendance for six-county schools:

```
summary(lm(sixco$ACT_COMP_SCHOOL~sixco$ATTENDANCE_RATE_SCHOOL_perALL))
```

```
##
## Call:
## lm(formula = sixco$ACT_COMP_SCHOOL ~ sixco$ATTENDANCE_RATE_SCHOOL_perALL)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7088 -2.2108 -0.2343  1.8209 10.9398
##
## Coefficients:
##                                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        -11.26875    2.61192  -4.314 2.25e-05 ***
## sixco$ATTENDANCE_RATE_SCHOOL_perALL   0.34311    0.02887  11.886  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.844 on 268 degrees of freedom
##   (1779 observations deleted due to missingness)
## Multiple R-squared:  0.3452, Adjusted R-squared:  0.3427
## F-statistic: 141.3 on 1 and 268 DF,  p-value: < 2.2e-16
```

Removing an influential point:

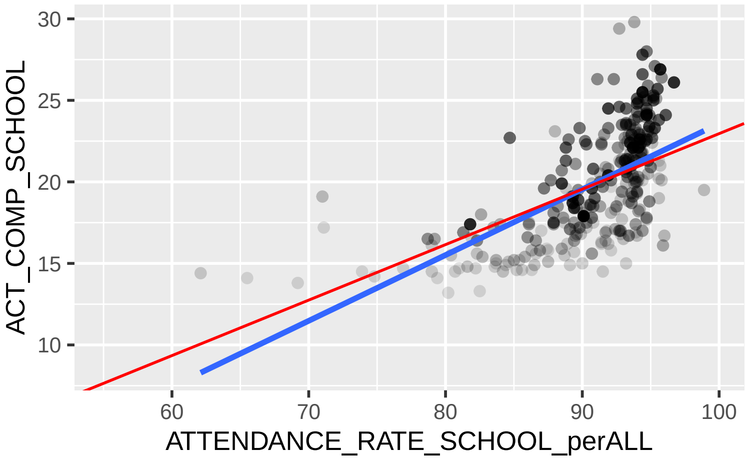
```
sixco_removed <- sixco %>%
  filter(ATTENDANCE_RATE_SCHOOL_perALL>50)

  summary(lm(sixco_removed$ACT_COMP_SCHOOL~sixco_removed$ATTENDANCE_RATE_SCHOOL_perALL))
```

```
##
## Call:
## lm(formula = sixco_removed$ACT_COMP_SCHOOL ~ sixco_removed$ATTENDANCE_RATE_SCHOOL_perALL)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.8340 -2.1147 -0.1728  1.6627  8.7676
##
## Coefficients:
##                                              Estimate Std. Error t value
## (Intercept)                                 -16.74644    2.81182  -5.956
## sixco_removed$ATTENDANCE_RATE_SCHOOL_perALL   0.40322    0.03104  12.993
##                                             Pr(>|t|)
## (Intercept)                                 8.13e-09 ***
## sixco_removed$ATTENDANCE_RATE_SCHOOL_perALL  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.75 on 267 degrees of freedom
##   (1778 observations deleted due to missingness)
## Multiple R-squared:  0.3873, Adjusted R-squared:  0.385
## F-statistic: 168.8 on 1 and 267 DF,  p-value: < 2.2e-16
```

Both models on the same plot. A best fit line can be added by including

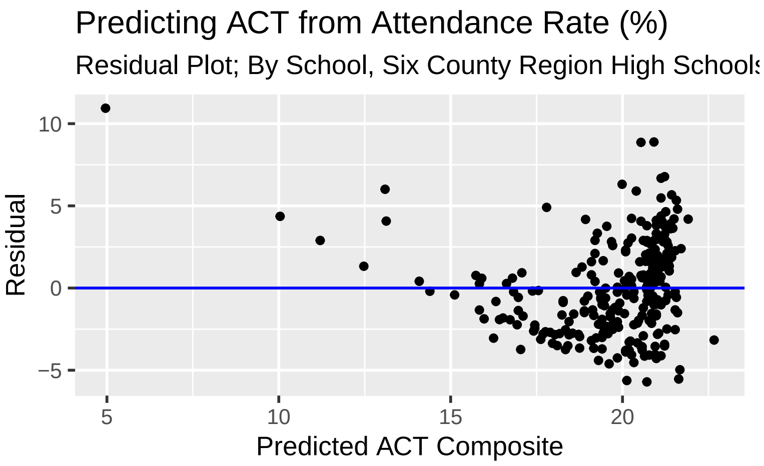
```
geom_smooth(method = lm, na.rm = TRUE, se = FALSE) +
```

in the `ggplot` command. Is there a ***significant*** change?



7.3.1 Residual Plot

Notice that residual plots are residuals vs. predicted and not residuals vs. explanatory.



7.4 Regression in Fathom

Force a variable to be numeric (option) or categorical (shift). Adding a Third Variable: Are charter schools different?

8 Random Selection and Simulation

8.1 Rolling a die

```
set.seed(2020)
one_die <- sample(1:6, 10, replace = TRUE)
one_die
```

```
##  [1] 4 3 4 3 1 1 1 3 1 4
```

8.2 Random Selection

The sample command can be used to generate a random selection. Adding a statistic command (like mean(variable)) and repeating quickly generates a sampling distribution.

```
four_schools <-  sample_n(rc17, 4)
four_schools[c("SCHOOL_NAME", "SCHOOL_TOTAL_ENROLLMENT")]
```

```
## # A tibble: 4 x 2
##   SCHOOL_NAME             SCHOOL_TOTAL_ENROLLMENT
##   <chr>                                     <dbl>
## 1 McKinley Elem School                        374
## 2 Stanton School                              295
## 3 Harvard High School                         678
## 4 Hernandez Middle School                    1044
```

```
#alternate form of dataframe$variable is dataframe[variable]
mean(four_schools$SCHOOL_TOTAL_ENROLLMENT)
```

```
## [1] 597.75
```

8.3 Stratified Sample

```
strat_samp <- sixco %>%
  filter(SCHOOL_TYPE_NAME == "HIGH SCHOOL") %>%
  group_by(COUNTY) %>% #stratify by county
  sample_n(3)
strat_samp[c("SCHOOL_NAME", "COUNTY", "SCHOOL_TOTAL_ENROLLMENT")]
```

```
## # A tibble: 18 x 3
## # Groups:   COUNTY [6]
##    SCHOOL_NAME                      COUNTY  SCHOOL_TOTAL_ENROLLMENT
##    <chr>                            <chr>                     <dbl>
##  1 Bogan High School                Cook                        769
##  2 Mather High School               Cook                       1472
##  3 Ogden Int High School            Cook                        715
##  4 Westmont High School             Dupage                      449
##  5 Lake Park High School            Dupage                     2599
##  6 Glenbard South High School       Dupage                     1171
##  7 Bartlett High School             Kane                       2487
##  8 Larkin High School               Kane                       2087
##  9 Central High School              Kane                       1047
## 10 North Chicago Community High Sch Lake                        767
## 11 Libertyville High School         Lake                       1935
## 12 Highland Park High School        Lake                       2040
## 13 Crystal Lake Central High School McHenry                    1545
## 14 McHenry  East High School        McHenry                     795
## 15 Cary-Grove Community High School McHenry                    1746
## 16 Peotone High School              Will                        530
## 17 Bolingbrook High School          Will                       3469
## 18 Crete-Monee High School          Will                       1634
```

8.4 Confidence Interval Simulation

A function that samples then calculates confidence interval bounds and stores results in a matrix. Each student can verify the interval bounds. (Use t* = 2.064)

```
rand_samp <- function(samples, vari, samp_size) {
  sixco_hs_nona <- sixco_hs[!is.na(sixco_hs[vari]), ] #remove schools with no value
  a <- matrix(ncol = 7, nrow = samples)
  for (k in 1:samples){
    dat_fra <- sample_n(sixco_hs_nona, samp_size)
    t_star <- qt(.975, df = samp_size - 1)
    x_bar <- mean(dat_fra[[vari]])
    stan_dev <- sd(dat_fra[[vari]])
    lower_b <- x_bar  - t_star*stan_dev/(samp_size)**.5
    upper_b <- x_bar + t_star*stan_dev/(samp_size)**.5
  v1 <- k
  v2 <- samp_size
  v3 <- x_bar
  v4 <- stan_dev
  v5 <- lower_b
  v6 <- upper_b
  v7 <- mean(sixco_hs[[vari]], na.rm = TRUE)

  a[k,] <- c(v1, v2, v3, v4, v5, v6, v7)} #row k and all columns of matrix a
  colnames(a) <- c("Sample", "Sample_Size", "Mean", "StdDev", "L_Bound", "U_Bound", "Parameter")
  return(a)
  }
```

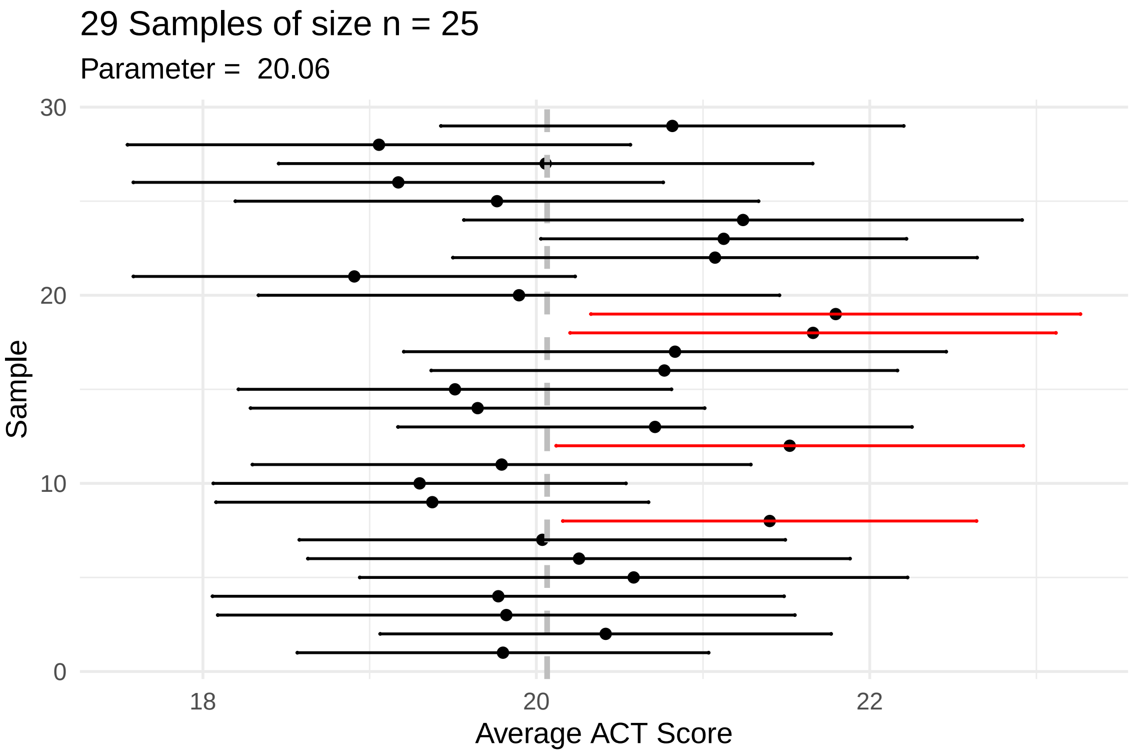
Produce and converting the matrix to a table of output.

```
set.seed(2020)
confid_ints <- as_tibble(rand_samp(29, "ACT_COMP_SCHOOL", 25))
confid_ints
```

```
## # A tibble: 29 x 7
##    Sample Sample_Size  Mean StdDev L_Bound U_Bound Parameter
##     <dbl>       <dbl> <dbl>  <dbl>   <dbl>   <dbl>     <dbl>
##  1      1          25  19.8   2.99    18.6    21.0      20.1
##  2      2          25  20.4   3.28    19.1    21.8      20.1
##  3      3          25  19.8   4.19    18.1    21.6      20.1
##  4      4          25  19.8   4.15    18.1    21.5      20.1
##  5      5          25  20.6   3.98    18.9    22.2      20.1
##  6      6          25  20.3   3.94    18.6    21.9      20.1
##  7      7          25  20.0   3.53    18.6    21.5      20.1
##  8      8          25  21.4   3.01    20.2    22.6      20.1
##  9      9          25  19.4   3.14    18.1    20.7      20.1
## 10     10          25  19.3   3.00    18.1    20.5      20.1
## # ... with 19 more rows
```

Plotting confidence intervals and comparing to parameter:

```
ggplot(confid_ints, mapping = aes(x=L_Bound, xend = U_Bound, y = Sample)) +
  geom_point(aes(x=Mean, y=Sample)) +
    geom_vline(xintercept = mean(confid_ints$Parameter),
               linetype="dashed",
                color = "grey",
               size=1) +
    geom_dumbbell(size_xend=0,size_x=0,
                color = ifelse(confid_ints$U_Bound < confid_ints$Parameter |
                               confid_ints$L_Bound > confid_ints$Parameter,
                               "red", "black")) +
    labs(x = "Average ACT Score",
         title = paste(max(confid_ints$Sample),
                       "Samples of size n =",
                       max(confid_ints$Sample_Size)
                      ),
         subtitle = paste("Parameter = ",round(confid_ints$Parameter,2))
         ) +
  theme_minimal()
```



9 Appendix

Learning More

- R and RStudio
  - R for Data Science by Garrett Grolemund and Hadley Wickham
- Fathom
  - https://fathom.concord.org/help/HelpFiles/index.html
- GitHub (https://github.com/fbriody)
  - Start with README.md (bottom of https://github.com/fbriody/MMC2020).
  - Code for handout is `handout.Rmd`.
  - Code for presentation is `EDA_present.Rmd`.
- Statistics
  - DePaul
  - Udacity
  - CodeAcademy
  - DataCamp

Contact Info

- email: frankbriody@gmail.com
- GitHub: https://github.com/fbriody
- Website: http://frankbriody.com
- Twitter: @frankbriody
- LinkedIn: https://www.linkedin.com/in/frank-briody-910700199