STAT 231: Problem Set 2B

Brandon Kwon

due by 5 PM on Friday, March 5

Series B homework assignments are designed to help you further ingest and practice the material covered in class over the past week(s). You are encouraged to work with other students, but all code must be written by you and you must indicate below who you discussed the assignment with (if anyone).

Steps to proceed:

1. In RStudio, go to File > Open Project, navigate to the folder with the course-content repo, select the course-content project (course-content.Rproj), and click "Open"
2. Pull the course-content repo (e.g. using the blue-ish down arrow in the Git tab in upper right window)
3. Copy ps2B.Rmd from the course repo to your repo (see page 6 of the GitHub Classroom Guide for Stat231 if needed)
4. Close the course-content repo project in RStudio
5. Open YOUR repo project in RStudio
6. In the ps2B.Rmd file in YOUR repo, replace "YOUR NAME HERE" with your name
7. Add in your responses, committing and pushing to YOUR repo in appropriate places along the way
8. Run "Knit PDF"
9. Upload the pdf to Gradescope. Don't forget to select which of your pages are associated with each problem. *You will not get credit for work on unassigned pages (e.g., if you only selected the first page but your solution spans two pages, you would lose points for any part on the second page that the grader can't see).*

If you discussed this assignment with any of your peers, please list who here:

ANSWER:

MDSR Exercise 4.14 (modified)

Use the `Pitching` data frame from the `Lahman` package to identify every pitcher in baseball history who has accumulated at least 300 wins (`W`) and at least 3,000 strikeouts (`SO`).

a. How many pitchers meet this criteria?

   ANSWER: 10 pitchers meet this criteria.

```
library(Lahman)
Pitching2 <- Pitching
grouped_Pitching2 <- Pitching2 %>%
  group_by(playerID) %>%
  summarise(total_W = sum(W), total_SO = sum(SO)) %>%
  filter(total_W >= 300 & total_SO >= 3000) %>%
  select(playerID, total_W, total_SO)
grouped_Pitching2
```

```
## # A tibble: 10 x 3
##    playerID  total_W total_SO
##    <chr>       <int>    <int>
##  1 carltst01     329     4136
##  2 clemero02     354     4672
##  3 johnsra05     303     4875
##  4 johnswa01     417     3509
##  5 maddugr01     355     3371
##  6 niekrph01     318     3342
##  7 perryga01     314     3534
##  8 ryanno01      324     5714
##  9 seaveto01     311     3640
## 10 suttodo01     324     3574
```

```
nrow(grouped_Pitching2)
```

```
## [1] 10
```

b. Which of these pitchers had the most accumulated strikeouts? How many strikeouts had he accumulated? What is the most strikeouts he had in one season?

   ANSWER: ryanno01 had the most accumulated strikeouts. He had 5714 strikeouts. The most strikeouts he had in one season was 383 strikeouts.

```
highest_SO <- grouped_Pitching2 %>%
  filter(total_SO == max(grouped_Pitching2$total_SO)) %>%
select(playerID, total_W, total_SO)
highest_SO
```

```
## # A tibble: 1 x 3
##   playerID total_W total_SO
##   <chr>      <int>    <int>
## 1 ryanno01     324     5714
```

```r
just_ryan <- Pitching %>%
  filter(playerID == "ryanno01")
max(just_ryan$SO)
```

```
## [1] 383
```

MDSR Exercise 4.17 (modified)

a. The Violations data set in the `mdsr` package contains information regarding the outcome of health inspections in New York City. Use these data to calculate the median violation score by zipcode and dba for zipcodes in Manhattan. What pattern (if any) do you see between the number of inspections and the median score? Generate a visualization to support your response.

   ANSWER: According to the visualization that I created, there seems to be a logarithmic relationship between number of inspections and the median violation score. With this said, this logarithmic relationship implies that there seems to be a somewhat positive correlation between number of inspections and the median violation score. In other words, as the number of inspections rises, the median violation score rises as well (slightly).

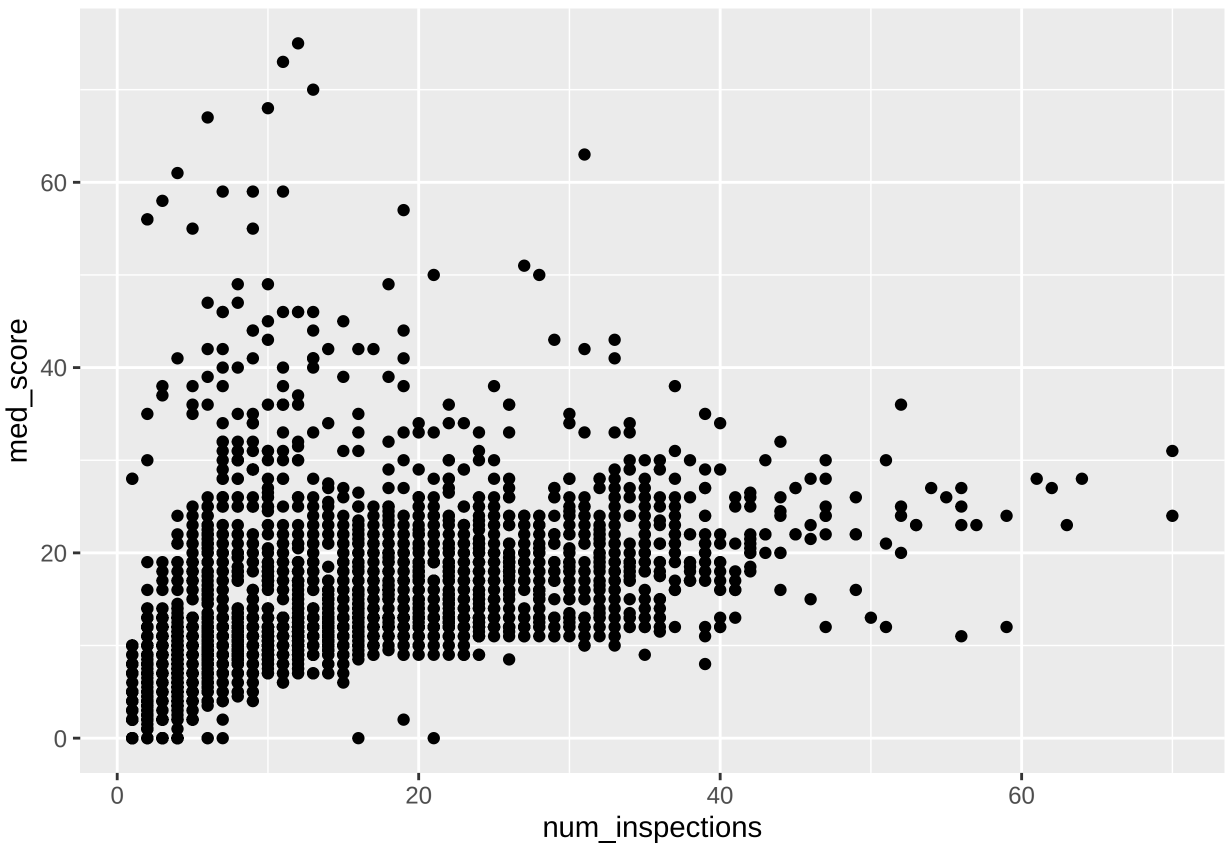
```
library(mdsr)
City_violations <- Violations %>%
  filter(boro == "MANHATTAN") %>%
  group_by(zipcode, dba) %>%
  summarize(med_score = median(score), num_inspections = n()) %>%
  drop_na()
```

```
## ‘summarise()‘ has grouped output by ’zipcode’. You can override using the ‘.groups‘ argument.
```

```
City_violations
```

```
## # A tibble: 4,321 x 4
## # Groups:   zipcode [72]
##    zipcode dba                          med_score num_inspections
##      <int> <chr>                            <dbl>           <int>
##  1   10001 16 HANDLES                           2               3
##  2   10001 5 SENSES                            32               7
##  3   10001 7 GRAMS CAFFE                        5               5
##  4   10001 876 MARKET DELI                     15              22
##  5   10001 99 CENTS BEST & FRESH PIZZA         11              12
##  6   10001 A&H DELI                            10               2
##  7   10001 AA ICHIBAN SUSHI                    16              24
##  8   10001 AARON’S CHINESE AND THAI            18              11
##  9   10001 ABACKY POTLUCK                      20              16
## 10   10001 APPETITE NYC                         8              14
## # ... with 4,311 more rows
```

```
ggplot(data = City_violations) +
  geom_point(aes(x = num_inspections, y = med_score))
```



b. In your visualization in part (a), there should be at least a few points that stand out as outliers. For *one of the outliers*, add text to the outlier identifying what business it is and an arrow pointing from the text to the observation. First, you may want to `filter` to identify the name of the business (so you know what text to add to the plot).

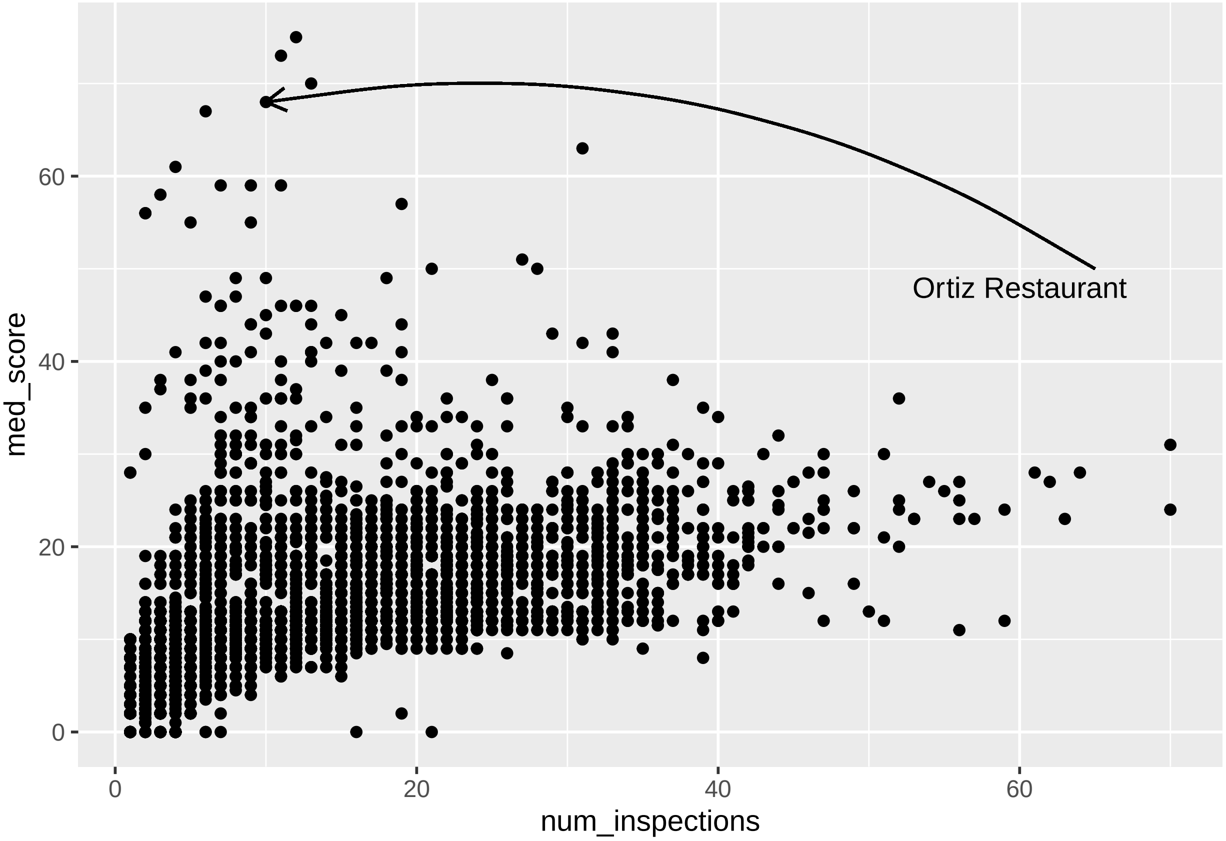
(Can't remember how to create a curved arrow in `ggplot`? The answers to this question on Stack Exchange may help. Can't remember how to add text to the plot in `ggplot`? Check out the text examples with `annotate` here, or answers to this question that use `geom_text`.)

```
City_violations %>%
  arrange(desc(med_score)) %>%
  head(n=4)
```

```
## # A tibble: 4 x 4
## # Groups:   zipcode [4]
##   zipcode dba                    med_score num_inspections
##     <int> <chr>                      <dbl>           <int>
## 1   10014 SUSHI DOJO EXPRESS            75              12
## 2   10012 BY CHLOE                      73              11
## 3   10010 BAO BAO CAFE                  70              13
## 4   10032 ORTIZ RESTAURANT              68              10
```

```
ggplot(data = City_violations) +
  geom_point(aes(x = num_inspections, y = med_score)) +
  geom_curve(
    aes(x = 65, y = 50, xend = 10, yend = 68), data = City_violations, curvature = 0.2, arrow = arrow(l

  annotate("text", x = 60, y = 48, label = "Ortiz Restaurant")
```



MDSR Exercise 5.7

Generate the code to convert the data frame shown with this problem in the textbook (on page 130, and shown below) to wide format (i.e., the result table). Hint: use `gather()` in conjunction with `spread()`; OR `pivot_longer()` in conjunction with `pivot_wider()`.

```
FakeDataLong <- data.frame(grp = c("A","A","B", "B")
                           , sex = c("F", "M", "F", "M")
                           , meanL = c(0.22, 0.47, 0.33, 0.55)
                           , sdL = c(0.11, 0.33, 0.11, 0.31)
                           , meanR = c(0.34, 0.57, 0.40, 0.65)
                           , sdR = c(0.08, 0.33, 0.07, 0.27))

long_data <- FakeDataLong %>%
  gather(key = "column", value = "data", c(-grp, -sex))
long_data
```

```
##    grp sex column data
## 1    A   F  meanL 0.22
## 2    A   M  meanL 0.47
## 3    B   F  meanL 0.33
## 4    B   M  meanL 0.55
## 5    A   F    sdL 0.11
## 6    A   M    sdL 0.33
## 7    B   F    sdL 0.11
## 8    B   M    sdL 0.31
## 9    A   F  meanR 0.34
## 10   A   M  meanR 0.57
## 11   B   F  meanR 0.40
## 12   B   M  meanR 0.65
## 13   A   F    sdR 0.08
## 14   A   M    sdR 0.33
## 15   B   F    sdR 0.07
## 16   B   M    sdR 0.27
```

```
long_data$combined_column <- paste0(long_data$sex, ".", long_data$column)
long_data
```

```
##    grp sex column data combined_column
## 1    A   F  meanL 0.22         F.meanL
## 2    A   M  meanL 0.47         M.meanL
## 3    B   F  meanL 0.33         F.meanL
## 4    B   M  meanL 0.55         M.meanL
## 5    A   F    sdL 0.11           F.sdL
## 6    A   M    sdL 0.33           M.sdL
## 7    B   F    sdL 0.11           F.sdL
## 8    B   M    sdL 0.31           M.sdL
## 9    A   F  meanR 0.34         F.meanR
## 10   A   M  meanR 0.57         M.meanR
## 11   B   F  meanR 0.40         F.meanR
## 12   B   M  meanR 0.65         M.meanR
## 13   A   F    sdR 0.08           F.sdR
## 14   A   M    sdR 0.33           M.sdR
## 15   B   F    sdR 0.07           F.sdR
## 16   B   M    sdR 0.27           M.sdR
```

```
long_data <- subset(long_data, select = -c(sex, column))
long_data
```

```
##    grp data combined_column
## 1    A 0.22         F.meanL
## 2    A 0.47         M.meanL
## 3    B 0.33         F.meanL
## 4    B 0.55         M.meanL
## 5    A 0.11           F.sdL
## 6    A 0.33           M.sdL
## 7    B 0.11           F.sdL
## 8    B 0.31           M.sdL
## 9    A 0.34         F.meanR
## 10   A 0.57         M.meanR
## 11   B 0.40         F.meanR
## 12   B 0.65         M.meanR
## 13   A 0.08           F.sdR
## 14   A 0.33           M.sdR
## 15   B 0.07           F.sdR
## 16   B 0.27           M.sdR
```

```
wide_data <- long_data %>%
  spread(key = "combined_column", value = "data")
wide_data
```

```
##   grp F.meanL F.meanR F.sdL F.sdR M.meanL M.meanR M.sdL M.sdR
## 1   A    0.22    0.34  0.11  0.08    0.47    0.57  0.33  0.33
## 2   B    0.33    0.40  0.11  0.07    0.55    0.65  0.31  0.27
```

PUG Brainstorming

What topics or questions are you interested in exploring related to your PUG theme? Dream big here. Don't worry about whether there is data out there that's available and accessible that you could use to address your questions/topics. Just brainstorm some ideas that get you excited. Then, email your PUG team with your ideas. Title the email "PS2B Brainstorming: PUG [#] [Topic]" and CC me (kcorreia@amherst.edu) on the email. If another PUG member already initiated the email, reply all to their email.

If you don't remember your PUG # and Topic, please see the file "PUGs" on the Moodle page under this week.

If you don't know your PUG members email address, go to the class's Google group conversations (e.g., by clicking the link "Link to Google group conversations" at the top of our Moodle course page). Then, on the navigation panel (left hand side), select "Members".

> ANSWER: Do not write anything here. Email your ideas to your PUG team and me in a message titled "PS2B Brainstorming: PUG [#] [Topic]".