STAT 231: Problem Set 2B

Mike Santos

due by 2 PM on Friday, September 11

Series B homework assignments are designed to help you futher 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:

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

ANSWER: N/A

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:

```
print("I had to comment out/leave out some code as it couldn't knit in time")
```

```
## [1] "I had to comment out/leave out some code as it couldn't knit in time"
```

```
library(Lahman)
Pitching2 = group_by(Pitching, playerID)
summary(Pitching2)
```

```
##    playerID             yearID         stint           teamID      lgID
##  Length:47628       Min.   :1871   Min.   :1.000   PHI    : 2213   AA:  657
##  Class :character   1st Qu.:1944   1st Qu.:1.000   CHN    : 2159   AL:22398
##  Mode  :character   Median :1982   Median :1.000   SLN    : 2129   FL:  173
##                     Mean   :1971   Mean   :1.082   PIT    : 2125   NA:  132
##                     3rd Qu.:2004   3rd Qu.:1.000   CIN    : 2061   NL:24114
##                     Max.   :2019   Max.   :5.000   CLE    : 2054   PL:   58
##                                                    (Other):34887   UA:   96
##        W                L                G               GS
##  Min.   : 0.000   Min.   : 0.000   Min.   :  1.00   Min.   : 0.000
##  1st Qu.: 0.000   1st Qu.: 1.000   1st Qu.:  7.00   1st Qu.: 0.000
##  Median : 2.000   Median : 3.000   Median : 21.00   Median : 2.000
##  Mean   : 4.605   Mean   : 4.605   Mean   : 23.68   Mean   : 9.262
##  3rd Qu.: 7.000   3rd Qu.: 7.000   3rd Qu.: 35.00   3rd Qu.:17.000
##  Max.   :60.000   Max.   :48.000   Max.   :106.00   Max.   :75.000
##
##        CG              SHO              SV             IPouts
##  Min.   : 0.000   Min.   : 0.000   Min.   : 0.000   Min.   :   0.0
##  1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.:  48.0
##  Median : 0.000   Median : 0.000   Median : 0.000   Median : 162.0
##  Mean   : 2.978   Mean   : 0.425   Mean   : 1.497   Mean   : 247.9
##  3rd Qu.: 2.000   3rd Qu.: 0.000   3rd Qu.: 1.000   3rd Qu.: 381.0
##  Max.   :75.000   Max.   :16.000   Max.   :62.000   Max.   :2040.0
##
##        H               ER               HR              BB
##  Min.   :  0.0   Min.   :  0.00   Min.   : 0.000   Min.   :  0.00
##  1st Qu.: 18.0   1st Qu.:  9.00   1st Qu.: 1.000   1st Qu.:  7.00
##  Median : 53.0   Median : 24.00   Median : 4.000   Median : 21.00
##  Mean   : 82.8   Mean   : 35.39   Mean   : 6.462   Mean   : 29.23
##  3rd Qu.:129.0   3rd Qu.: 56.00   3rd Qu.: 9.000   3rd Qu.: 44.00
##  Max.   :772.0   Max.   :291.00   Max.   :50.000   Max.   :289.00
##
##        SO             BAOpp            ERA              IBB
##  Min.   :  0.00   Min.   :0.000   Min.   :  0.000   Min.   : 0.000
##  1st Qu.:  8.00   1st Qu.:0.240   1st Qu.:  3.160   1st Qu.: 0.000
##  Median : 31.00   Median :0.266   Median :  4.150   Median : 1.000
##  Mean   : 46.52   Mean   :0.313   Mean   :  5.118   Mean   : 2.298
##  3rd Qu.: 68.00   3rd Qu.:0.300   3rd Qu.:  5.540   3rd Qu.: 3.000
##  Max.   :513.00   Max.   :9.990   Max.   :189.000   Max.   :23.000
```

```
##                   NA's   :4441    NA's   :94        NA's   :14578
##        WP             HBP              BK             BFP
##  Min.   : 0.000   Min.   : 0.000   Min.   : 0.000   Min.   :   0.0
##  1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.:  73.0
##  Median : 2.000   Median : 1.000   Median : 0.000   Median : 234.0
##  Mean   : 2.603   Mean   : 2.379   Mean   : 0.296   Mean   : 353.7
##  3rd Qu.: 4.000   3rd Qu.: 3.000   3rd Qu.: 0.000   3rd Qu.: 546.0
##  Max.   :83.000   Max.   :54.000   Max.   :16.000   Max.   :2906.0
##                   NA's   :734                       NA's   :3
##        GF             R               SH             SF
##  Min.   : 0.00   Min.   :  0.00   Min.   : 0.000   Min.   : 0.000
##  1st Qu.: 0.00   1st Qu.: 10.00   1st Qu.: 0.000   1st Qu.: 0.000
##  Median : 3.00   Median : 27.00   Median : 2.000   Median : 1.000
##  Mean   : 6.28   Mean   : 42.04   Mean   : 2.728   Mean   : 2.185
##  3rd Qu.: 8.00   3rd Qu.: 66.00   3rd Qu.: 4.000   3rd Qu.: 3.000
##  Max.   :84.00   Max.   :519.00   Max.   :27.000   Max.   :17.000
##                                   NA's   :19187    NA's   :19187
##       GIDP
##  Min.   : 0.000
##  1st Qu.: 1.000
##  Median : 4.000
##  Mean   : 5.871
##  3rd Qu.: 9.000
##  Max.   :47.000
##  NA's   :20318
```

```
library(purrr)

#Pitching4 = Pitching %>% split(Pitching$playerID) %>% map(summary)

Ace = Pitching2 %>% filter('W' >= 300 & 'SO' >= 3000)
Ace
```

```
## # A tibble: 47,628 x 30
## # Groups:   playerID [9,845]
##    playerID yearID stint teamID lgID      W     L     G    GS    CG   SHO    SV
##    <chr>     <int> <int> <fct>  <fct> <int> <int> <int> <int> <int> <int> <int>
##  1 bechtge~   1871     1 PH1    NA        1     2     3     3     2     0     0
##  2 brainas~   1871     1 WS3    NA       12    15    30    30    30     0     0
##  3 fergubo~   1871     1 NY2    NA        0     0     1     0     0     0     0
##  4 fishech~   1871     1 RC1    NA        4    16    24    24    22     1     0
##  5 fleetfr~   1871     1 NY2    NA        0     1     1     1     1     0     0
##  6 flowedi~   1871     1 TRO    NA        0     0     1     0     0     0     0
##  7 mackde01   1871     1 RC1    NA        0     1     3     1     1     0     0
##  8 mathebo~   1871     1 FW1    NA        6    11    19    19    19     1     0
##  9 mcbridi~   1871     1 PH1    NA       18     5    25    25    25     0     0
## 10 mcmuljo~   1871     1 TRO    NA       12    15    29    29    28     0     0
## # ... with 47,618 more rows, and 18 more variables: IPouts <int>, H <int>,
## #   ER <int>, HR <int>, BB <int>, SO <int>, BAOpp <dbl>, ERA <dbl>, IBB <int>,
## #   WP <int>, HBP <int>, BK <int>, BFP <int>, GF <int>, R <int>, SH <int>,
## #   SF <int>, GIDP <int>
```

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:

```r
colMax <- function(data) sapply(data, max, na.rm = TRUE)
max(Pitching$W, na.rm = TRUE)
```

```
## [1] 60
```

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:

```
library(mdsr)
Violations
```

```
## # A tibble: 480,621 x 16
##     camis dba   boro  building street zipcode  phone inspection_date     action
##     <int> <chr> <chr>    <int> <chr>    <int>  <dbl> <dttm>              <chr>
##  1 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2015-02-09 00:00:00 Viola~
##  2 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2014-03-03 00:00:00 Viola~
##  3 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2013-10-10 00:00:00 No vi~
##  4 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2013-09-11 00:00:00 Viola~
##  5 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2013-09-11 00:00:00 Viola~
##  6 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2013-08-14 00:00:00 Viola~
##  7 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2013-08-14 00:00:00 Viola~
##  8 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2013-08-14 00:00:00 Viola~
##  9 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2013-08-14 00:00:00 Viola~
## 10 3.01e7 MORR~ BRONX     1007 MORRI~   10462 7.19e9 2013-08-14 00:00:00 Viola~
## # ... with 480,611 more rows, and 7 more variables: violation_code <chr>,
## #   score <int>, grade <chr>, grade_date <dttm>, record_date <dttm>,
## #   inspection_type <chr>, cuisine_code <dbl>
```

```
zip = group_by(Violations, zipcode)
dba = group_by(Violations, dba)
summary(Violations)
```

```
##      camis              dba                boro              building
##  Min.   :30075445   Length:480621      Length:480621      Min.   :       0
##  1st Qu.:41104568   Class :character   Class :character   1st Qu.:     202
##  Median :41475322   Mode  :character   Mode  :character   Median :     790
##  Mean   :42930615                                         Mean   :   63776
##  3rd Qu.:41694265                                         3rd Qu.:    3720
##  Max.   :50045372                                         Max.   :94179419
##                                                           NA's   :10094
##     street             zipcode          phone
##  Length:480621      Min.   :10001   Min.   :6.463e+08
##  Class :character   1st Qu.:10021   1st Qu.:2.127e+09
##  Mode  :character   Median :10466   Median :7.183e+09
##                     Mean   :10667   Mean   :5.185e+09
##                     3rd Qu.:11229   3rd Qu.:7.186e+09
##                     Max.   :11697   Max.   :7.186e+10
##                                     NA's   :334
##  inspection_date                 action          violation_code
##  Min.   :1900-01-01 00:00:00   Length:480621      Length:480621
##  1st Qu.:2013-07-08 00:00:00   Class :character   Class :character
##  Median :2014-05-30 00:00:00   Mode  :character   Mode  :character
##  Mean   :2014-02-06 19:36:27
##  3rd Qu.:2015-03-19 00:00:00
##  Max.   :2016-01-04 00:00:00
```

```
##
##      score             grade              grade_date
##  Min.   : -2.00   Length:480621      Min.   :2011-07-21 00:00:00
##  1st Qu.: 11.00   Class :character   1st Qu.:2013-07-17 00:00:00
##  Median : 17.00   Mode  :character   Median :2014-06-18 00:00:00
##  Mean   : 20.04                      Mean   :2014-05-18 01:40:45
##  3rd Qu.: 25.00                      3rd Qu.:2015-03-26 00:00:00
##  Max.   :156.00                      Max.   :2016-01-04 00:00:00
##  NA's   :31242                       NA's   :261498
##   record_date          inspection_type     cuisine_code
##  Min.   :2016-01-06   Length:480621      Min.   : 1.00
##  1st Qu.:2016-01-06   Class :character   1st Qu.: 5.00
##  Median :2016-01-06   Mode  :character   Median :26.00
##  Mean   :2016-01-06                      Mean   :31.51
##  3rd Qu.:2016-01-06                      3rd Qu.:51.00
##  Max.   :2016-01-06                      Max.   :84.00
##
```

```
summary(Violations$score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   -2.00   11.00   17.00   20.04   25.00  156.00   31242
```

```
#zip2 = Violations %>% split(Violations$zipcode)
#dba2 = Violations %>% split(Violations$dba)

library(ggplot2)
plot = ggplot(data = Violations, aes(x = score)) +
geom_bar(stat = "identity", aes(y = inspection_date)
, fill = "#b2d7e9", color = "white")
plot
```

```
## Warning: Removed 31242 rows containing missing values (position_stack).
```

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`? Can't remember how to add text to the plot in `ggplot`? Check out the answers to questions #5 and #8, respectively, in the Moodle R Q&A forum!)

```
print("Data cannot load quickly enough to plot")
```

```
## [1] "Data cannot load quickly enough to plot"
```

```
#Here I would filter by whatever number is the outlier, then the business is identified
#I couldn't find the questions in the Moodle Q&A
#plot +
 # annotate("text", x=0, y=0, label= "Business") +
 # geom_curve(...)
```

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 conjuction with `spread()`; OR `pivot_longer()` in conjuction 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))
#install.packages("data.table")           # Install and load data.table
#library("data.table")

#setDT(FakeDataLong)
#FakeDataLong
```

PUG Post

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. In your PUG team discussion forum on GitHub, start a thread called "Brainstorming" (or, if another team member has already started the thread, reply to their post) with your ideas.

> ANSWER: Do not write anything here. Write down your ideas in your PUG team's discussion thread titled "Brainstorming" on GitHub.