STAT 231: Problem Set 2B

Evan Daisy

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: Conrad Kuklinsky (Who turned his in before me)

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)
data(Pitching)
GoodPitching <- Pitching %>%
  group_by(playerID) %>%
  summarise(TotalWins = sum(W), TotalSO = sum(SO)) %>%
  filter(TotalWins >= 300) %>%
  filter(TotalSO >= 3000)
```

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: The pitcher with player ID ryanno01 had the most accumulated strikeouts, with 5714. The most strikeouts he had in a single season was 383.

```
BestStriker <- GoodPitching %>%
  filter(TotalSO == max(TotalSO))
RyanByYear <- Pitching %>%
  filter(playerID == "ryanno01") %>%
  summarise(BestSeason = max(SO))
```

MDSR Exercise 4.17 (modified)

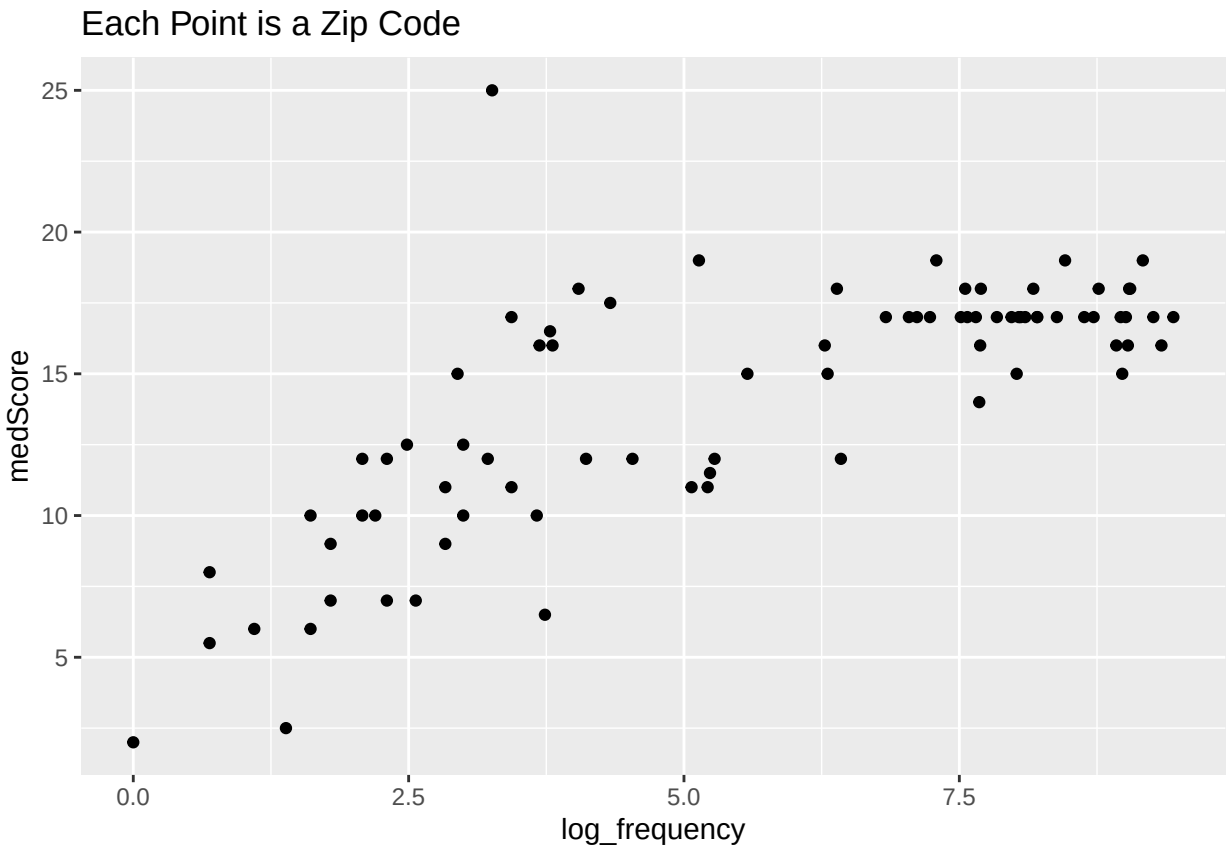
a. The Violations data set in the `mdsr` package contains information regarding the outcome of health inspections in New York City. Use these data to calculate the median violation score by zipcode and dba for zipcodes in Manhattan. What pattern (if any) do you see between the number of inspections and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

ANSWER: There is a moderately positive and almost linear association between log(number of inspections) and median score when grouping by zip code, indicating a positive relationship between number of inspections and median score. When grouping by restaurant, there are too many points to see any kind of relationship.

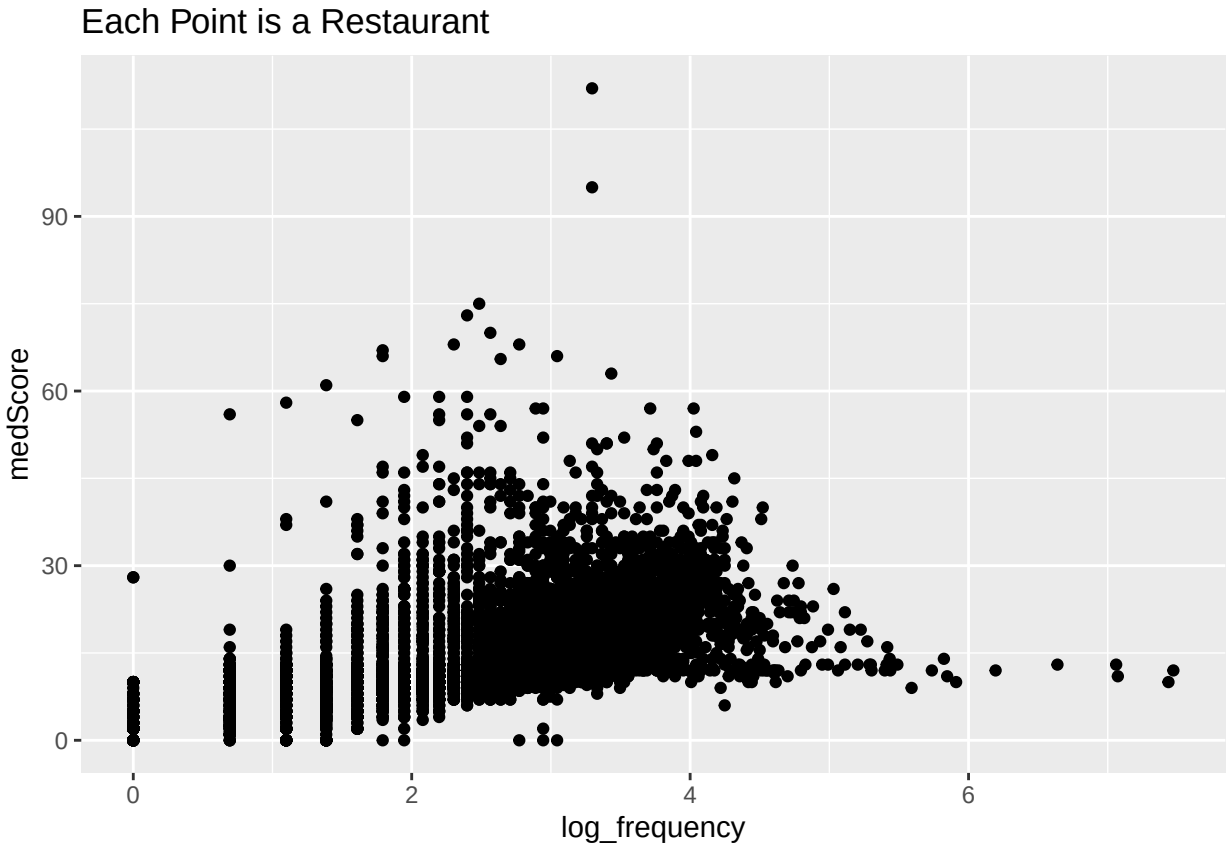
```
data(Violations)
ManViolations <- Violations %>%
  filter(boro == "MANHATTAN") %>%
  filter(is.na(score) == FALSE)
ByPlace <- ManViolations %>%
  group_by(dba) %>%
  summarise(medScore = median(score), frequency = n()) %>%
  mutate(log_frequency = log(frequency))
ByZip <- ManViolations %>%
  group_by(zipcode) %>%
  summarise(medScore = median(score), frequency = n()) %>%
  mutate(log_frequency = log(frequency))
ByBoth <- ManViolations %>%
  group_by(zipcode, dba) %>%
  summarise(medScore = median(score), frequency = n()) %>%
  mutate(log_frequency = log(frequency)) %>%
  mutate(log_medScore = log(medScore))
```

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

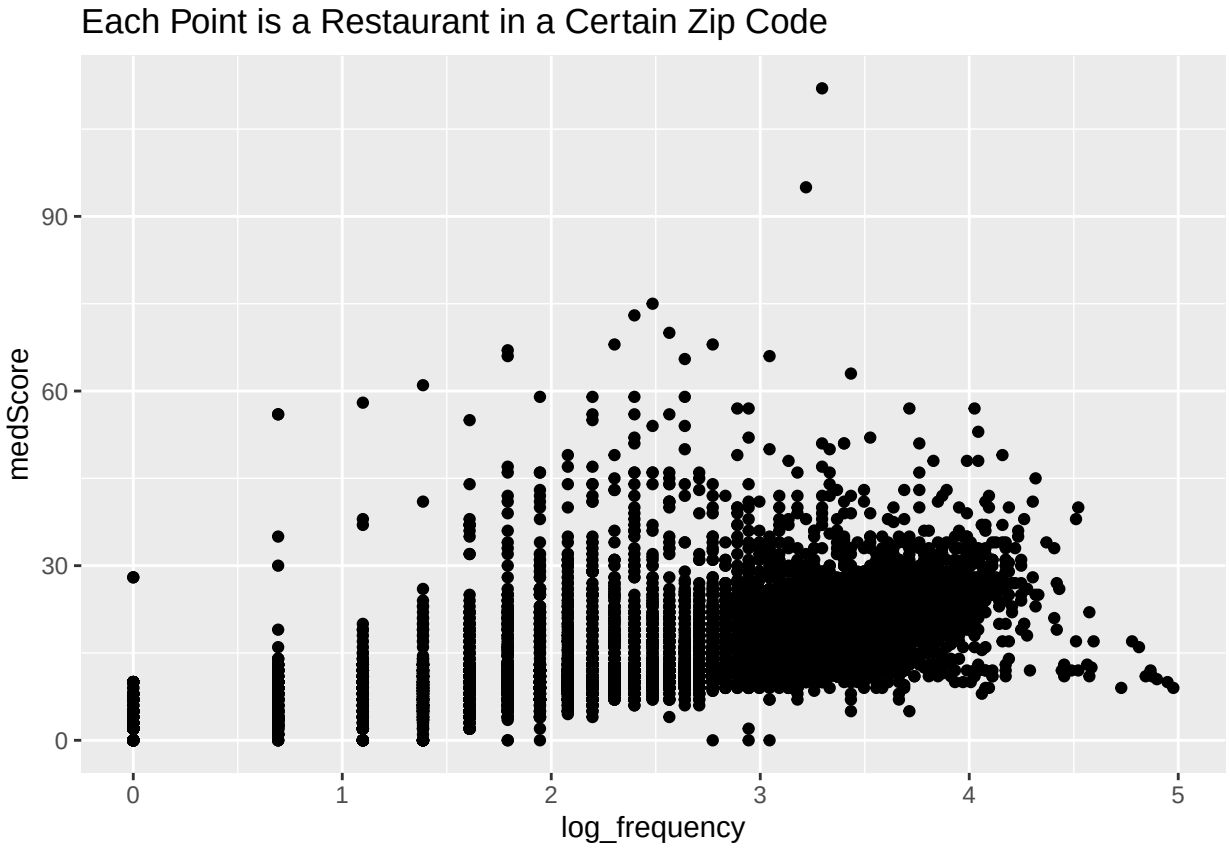
```
g <- ggplot(data = ByZip, aes(x = log_frequency, y = medScore))
g2 <- ggplot(data = ByPlace, aes(x = log_frequency, y = medScore))
g3 <- ggplot(data = ByBoth, aes(x = log_frequency, y = medScore))
g + geom_point() + labs(title = "Each Point is a Zip Code")
```




```
g2 + geom_point() + labs(title = "Each Point is a Restaurant")
```



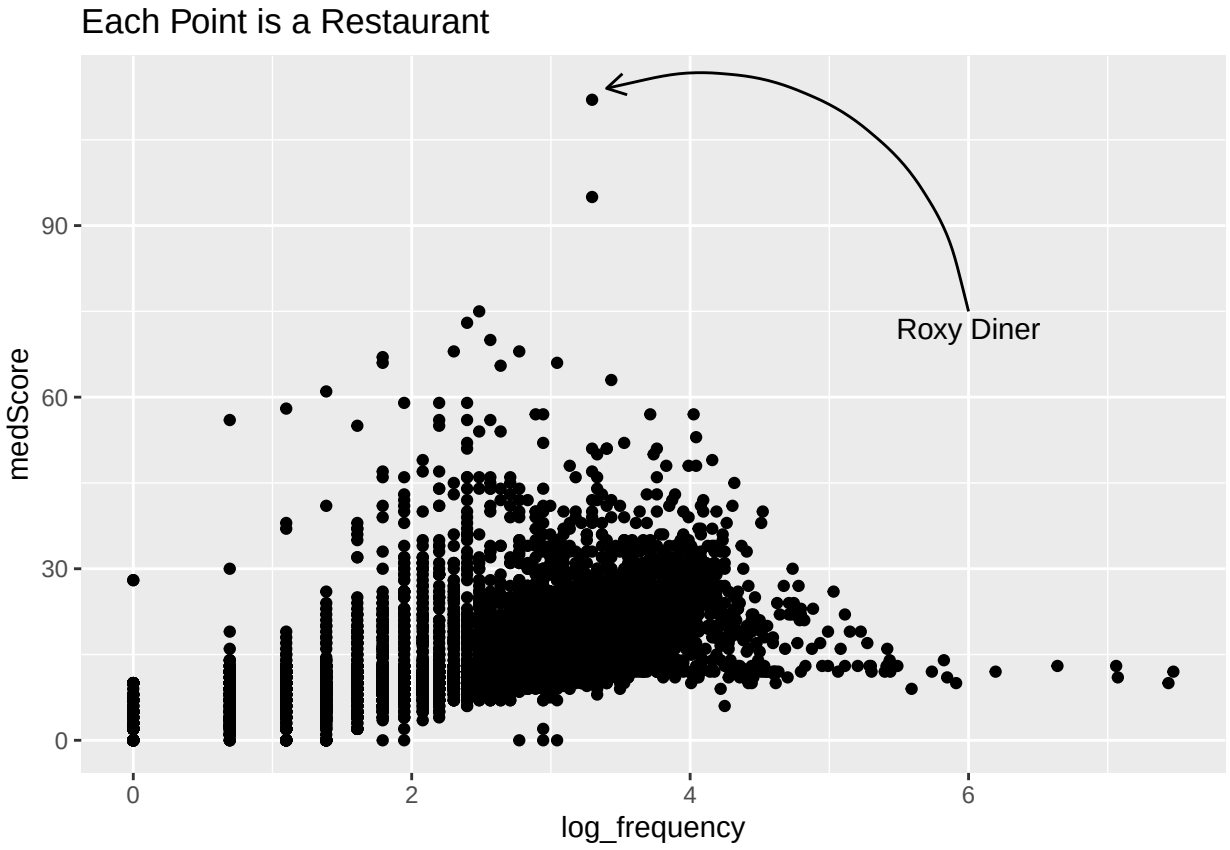

```
g3 + geom_point() + labs(title = "Each Point is a Restaurant in a Certain Zip Code")
```



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`.)

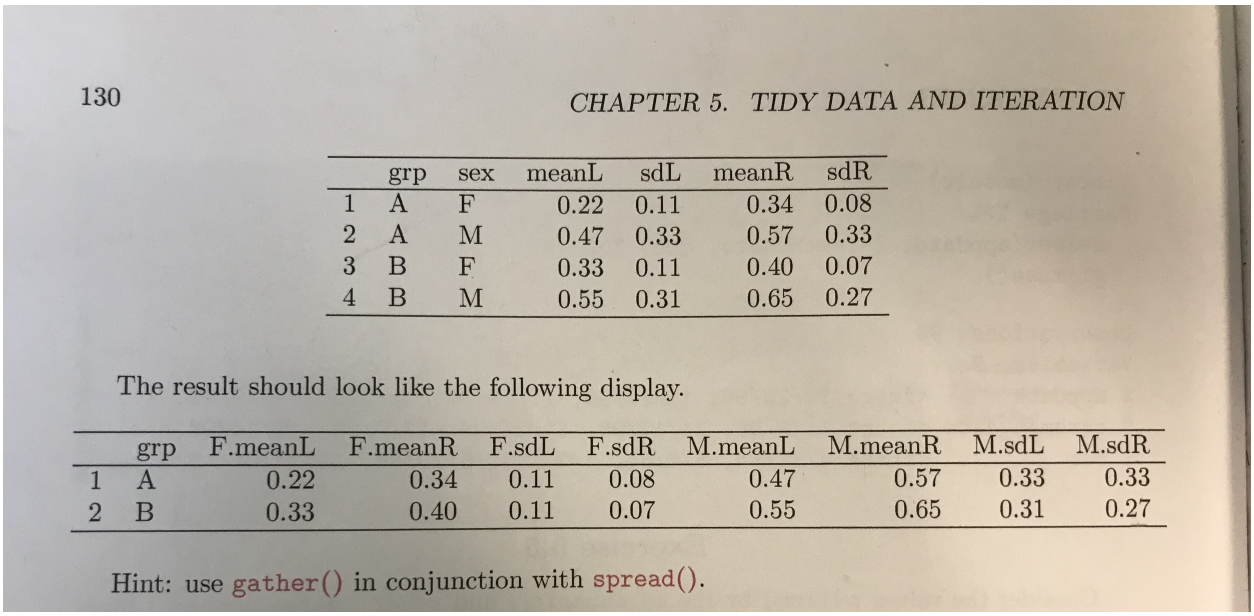
```
CrazyPlaces <- ByPlace %>%
  filter(medScore > 90)
arrow <- data.frame(x1 = 6, y1 = 75, x2 = 3.4, y2 = 114)
g2 <- ggplot(data = ByPlace, aes(x = log_frequency, y = medScore))
g2 + geom_point() +
  labs(title = "Each Point is a Restaurant") +
  geom_curve(aes(x = x1, y = y1, xend = x2, yend = y2),
             data = arrow, arrow = arrow(length = unit(0.03, "npc"))) +
  annotate("text", x = 6, y = 72, label = "Roxy Diner")
```




```
#I chose to use the graph grouping by only dba rather than both dba and ZipCode
#because it left more space with which to label an outlier.
```

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()`.



130 CHAPTER 5. TIDY DATA AND ITERATION

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

The result should look like the following display.

	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

Hint: use gather() in conjunction with spread().

```
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))
head(FakeDataLong)
```

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

```
FakeDataWide <- FakeDataLong %>%
  pivot_longer(c(meanL, meanR, sdL, sdR), names_to = "measurements", values_to = "number") %>%
  mutate(sex.measure = paste(sex,measurements, sep = ".")) %>%
  select(-measurements, -sex) %>%
  pivot_wider(names_from = sex.measure, values_from = number)
  #spread(names_from = sex.measure, values_from = number)
  #mutate(sex2 = sex) %>%
  #spread(key = sex2, value = meanL)
  #rename(F.meanL = F, M.meanL = M) %>%
  #mutate(sex3 = sex) %>%
  #spread(key = sex3, value = meanR) %>%
  #rename(F.meanR = F, M.meanR = M) %>%
  #mutate(sex4 = sex) %>%
```

```
  #spread(key = sex4, value = sdL) %>%
  #rename(F.sdL = F, M.sdL = M) %>%
  #spread(key = sex, value = sdR) %>%
  #rename(F.sdR = F, M.sdR = M)
#Above is an earlier method I tried that almost worked, but I couldn't get rid of the NA's
FakeDataWide
```

```
## # A tibble: 2 x 9
##   grp   F.meanL F.meanR F.sdL F.sdR M.meanL M.meanR M.sdL M.sdR
##   <chr>   <dbl>   <dbl> <dbl> <dbl>   <dbl>   <dbl> <dbl> <dbl>
## 1 A        0.22    0.34  0.11  0.08    0.47   0.570  0.33  0.33
## 2 B        0.33    0.4   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]".