Practice2S24

BilalTariq

2024-02-15

Practice2 - Due Thursday, 2/15 by midnight to Gradescope

Reminder: Practice assignments may be completed working with other individuals.

Reading

The associated reading for the week is Chapter 4, Chapter 5, Chapter 6 (skip 6.4), and Sections 8.3 and 8.4.

Other Notes

When displaying anything to the screen, be it table or plot, make it look nice! That is, use nice variable names, clean up axes, make the display nice, etc. This means you need to load library(kableExtra) with your other libraries. This is a good habit to get into now that we've seen commands for these tasks.

Practicing Academic Integrity

If you worked with others or used resources outside of provided course material (anything besides our textbook, course materials in the repo, labs, R help menu) to complete this assignment, please acknowledge them below using a bulleted list.

I acknowledge the following individuals with whom I worked on this assignment:

Name(s) and corresponding problem(s)

-

I used the following sources to help complete this assignment:

Source(s) and corresponding problem(s)

-

1 - Hardest Concept

We've covered many different data wrangling concepts and associated verbs during this unit. This problem will help you identify ways to get support on concepts you find challenging, beyond what we have in class and in the textbook.

> part a - What concept or data wrangling verb did you find most challenging to work with during this unit?

Solution: Pivoting has been pretty hard to understand for me. Also, generally knowing what to use and what syntax has been hard especially when it came to joining tables and manipulating that data.

> part b - Look in our Resources folder at the tidyr and data-transformation cheat sheets. Can you find information related to your selected concept or verb? If so, what sheet is it in? What if any insights do you get from the cheatsheet?

(If you picked a concept or verb not on these cheatsheets, try to find it on a different one, or ask me where it is likely to be. These are just the two most common cheatsheets to reference for these chapters, but not the only ones you might need.)

Solution: Thank you! It's in both the tdyr and data transformation cheatsheets. From them, I understand that pivot longer allows us to collapse several columns into two and pivot wider does the opposite where it widen's the data. The data transformation file also allows me to at a quick glance understand the syntax to use.

> part c - Most of the packages we use have vignettes that have been created for them. Vignettes are designed to show how functions are used. Identify either a function related to your concept or your selected verb (which is a function), and find what package it is in. Then look for a package vignette. What package did you look for a vignette for? Is your concept or verb illustrated in the vignette?

(Searching with Google or within R are possible.)

Solution: pivot wider is in the tidyr package

> part d - Many people blog examples of different R functions. Search for an R example of your concept or verb using Google. Look over the search results and identify one that demonstrates correct use of the concept or verb. List the URL.

Solution: https://tidyr.tidyverse.org/reference/pivot_longer.html

2 - MDSR 5.2

Use the `Batting`, `Pitching`, and `People` tables in the *Lahman* package to answer the following questions. Remember that you are responsible for loading packages in the setup chunk.

> part a - List the name of every player in baseball history who has accumulated at least 300 home runs (HR) and at least 300 stolen bases (SB). You can find the first and last name of the player in the `People` data frame. Join this to your result along with the total home runs and total bases stolen for each of these elite players.

Solution:

```
BattingJoined <- inner_join(People,Batting, by = c("playerID" = "playerID"))

newBatting <- BattingJoined %>%
  select(HR, SB, nameFirst, nameLast) %>%
  filter(HR > 30, SB > 30)

glimpse(newBatting)
```

```
Rows: 47
Columns: 4
$ HR        <int> 44, 31, 41, 43, 31, 33, 34, 33, 42, 40, 32, 39, 37, 33, 40, ~
$ SB        <int> 31, 36, 37, 31, 31, 52, 39, 31, 40, 37, 45, 43, 41, 33, 32, ~
$ nameFirst <chr> "Hank", "Bobby", "Ronald", "Jeff", "Dante", "Barry", "Barry"~
$ nameLast  <chr> "Aaron", "Abreu", "Acuna", "Bagwell", "Bichette", "Bonds", "~
```

> part b - Similarly, list the names of every pitcher in baseball history who has accumulated at least 300 wins (W) and at least 3,000 strikeouts (SO).

Solution:

```
pitcherJoined <- inner_join(People,Pitching, by = c("playerID" = "playerID"))

newPitcher <- pitcherJoined %>%
  select(nameFirst,nameLast, W, SO) %>%
  filter(W > 30, SO > 300)

glimpse(newPitcher)
```

```
Rows: 23
Columns: 4
```

```
$ nameFirst <chr> "Lady", "Charlie", "John", "John", "Pud", "Guy", "Bill", "Wa~
$ nameLast  <chr> "Baldwin", "Buffinton", "Clarkson", "Clarkson", "Galvin", "H~
$ W         <int> 42, 48, 53, 36, 46, 52, 36, 33, 41, 37, 35, 34, 41, 36, 48, ~
$ SO        <int> 323, 417, 308, 313, 369, 385, 314, 303, 359, 334, 335, 302, ~
```

> part c - Finally, list the name and year of every player who has hit at least 50 home runs in a single season. Which player had the lowest batting average in that season?

Note: Batting average is calculated as the number of hits (H) divided by the number of at bats (AB).

Solution:

```r
BattingJoined <- inner_join(People,Batting, by = c("playerID" = "playerID"))

newBatting50 <- BattingJoined %>%
  select(nameFirst, nameLast, HR, H, AB) %>%
  filter(HR > 50) %>%
  mutate(BattingAverage = H/AB)



glimpse(newBatting50)
```

```
Rows: 40
Columns: 6
$ nameFirst      <chr> "Pete", "Jose", "Barry", "Chris", "Cecil", "George", "J~
$ nameLast       <chr> "Alonso", "Bautista", "Bonds", "Davis", "Fielder", "Fos~
$ HR             <int> 53, 54, 73, 53, 51, 52, 58, 57, 58, 56, 56, 58, 51, 52,~
$ H              <int> 155, 148, 156, 167, 159, 197, 213, 198, 175, 185, 180, ~
$ AB             <int> 597, 569, 476, 584, 573, 615, 585, 609, 556, 608, 633, ~
$ BattingAverage <dbl> 0.2596315, 0.2601054, 0.3277311, 0.2859589, 0.2774869, ~
```

3 - MDSR 4.11 (modified)

The `Violations` data set in the **mdsr** package contains information regarding the outcome of health inspections of restaurants in New York City. Note that higher inspection scores indicate worse violations: "restaurants with an inspection score between 0 and 13 points earn an A, those with 14 to 27 points receive a B and those with 28 or more a C" (nyc.gov).

> part a - Use these data to calculate the median violation score by zip code for zip codes 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.

Solution: There seems to be a positive correlation between the number of inspections and the median score.

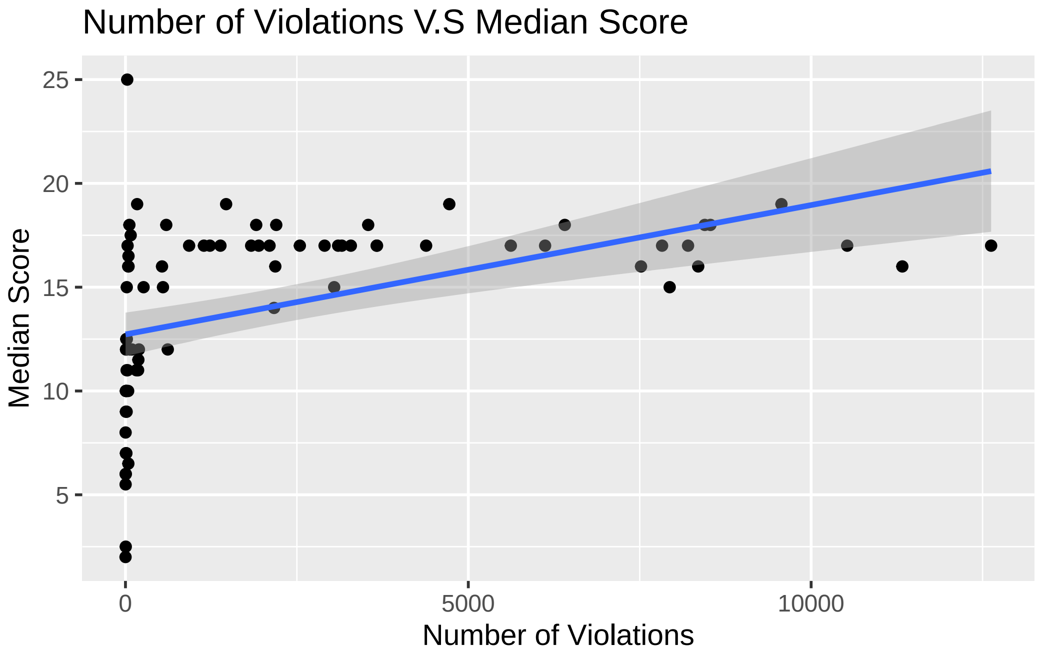
```
# You will likely need to drop NA values in this process.
# Do so AFTER getting the variables you need into the data set.
# If you drop NAs too early, you will lose observations that had missing values
# for variables you don't care about, and thus, lose information!
# By getting just Manhattan, your final data set should have 81 obs.

newViolations <- Violations %>%
  select(boro, zipcode, score) %>%
  filter(boro == "MANHATTAN") %>%
  group_by(zipcode) %>%
  na.omit() %>%

  summarise(
    NumberViolations = n(),
    MedianScore = median(score),
  )

 ggplot(newViolations, aes(x = NumberViolations, y = MedianScore)) +
  geom_point() +
  geom_smooth(method = "lm") + labs(title = "Number of Violations V.S Median Score", x = "
```

```
`geom_smooth()` using formula = 'y ~ x'
```



part b - In your visualization above, there are several potential outliers but there is one zipcode in particular that does not seem to fall along the general trend. Add text to the outlier identifying what zipcode it is, and add an arrow pointing from the text to the observation. Note: first, you may want to `filter()` to identify the zipcode (so you know what text to add to the plot).

Solution:

4 - MDSR 6.5

Generate the code to convert the data frame from the starting point to the results.

Figures available in text online in Section 6.6.

The starting data frame is provided. Hint (from text): Use *pivot_longer()* in conjunction with *pivot_wider()*. Hint (from Prof. Wagaman): There is also a way to do this just with *pivot_wider()* but you need to explore some of its options.

```
OrigData <- 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))

pivotedData <- OrigData %>%
  pivot_wider(names_from = sex, values_from = c(meanL, sdL, meanR, sdR), names_glue = "{se

glimpse(pivotedData)
```

```
Rows: 2
Columns: 9
$ grp     <chr> "A", "B"
$ F.meanL <dbl> 0.22, 0.33
$ M.meanL <dbl> 0.47, 0.55
$ F.sdL   <dbl> 0.11, 0.11
$ M.sdL   <dbl> 0.33, 0.31
$ F.meanR <dbl> 0.34, 0.40
$ M.meanR <dbl> 0.57, 0.65
$ F.sdR   <dbl> 0.08, 0.07
$ M.sdR   <dbl> 0.33, 0.27
```

Solution:

5 - Combining your Wrangling and Visualization Skills

When we looked at our first UN votes visual, some wrangling was required to get the data into a format appropriate for the visual. Now that we've examined both visualization and wrangling, you can combine the skills too! (And you did a little of this above).

We will be looking at a data set on high school students in Portugal. We have information on their performance in a Math course and a Portugeuse course (think of this as your natural language course, i.e. English for English speakers, etc.), as well as a host of demographic variables. Detailed information about the data set is provided on the following pages - you should look it over as you tackle this problem. (Feel free to remove the info when knitting to the final version of your assignment.)

We want to visualize the relationship between final Math and final Portugeuse grade for students who were in both courses. In addition, we want to be sure all students in the visual were under 20 years old, and had fewer than 10 absences in either course (not total). We also want to factor in weekend alcohol use and travel time as reported in the Math data set in our examination of the relationship, treating these as appropriate group variables (categorical). Students filled out the survey twice (once per course) and not all responses match between them, even for the same student.

1. Wrangle the data you need into an appropriate format, and save it as a new data set with the variables you need for your visual.

Solution:

```
MathPort <- inner_join(math_data, port_data, by = c("school", "sex", "age", "address", "fa
    filter(age < 20) %>%
  filter(absences.y < 10) %>%
  filter(absences.x < 10) %>%
  select(G3.x, G3.y, Walc.x, traveltime.x)
```

```
Warning in inner_join(math_data, port_data, by = c("school", "sex", "age", : Detected an une
i Row 79 of `x` matches multiple rows in `y`.
i Row 79 of `y` matches multiple rows in `x`.
i If a many-to-many relationship is expected, set `relationship =
  "many-to-many"` to silence this warning.
```

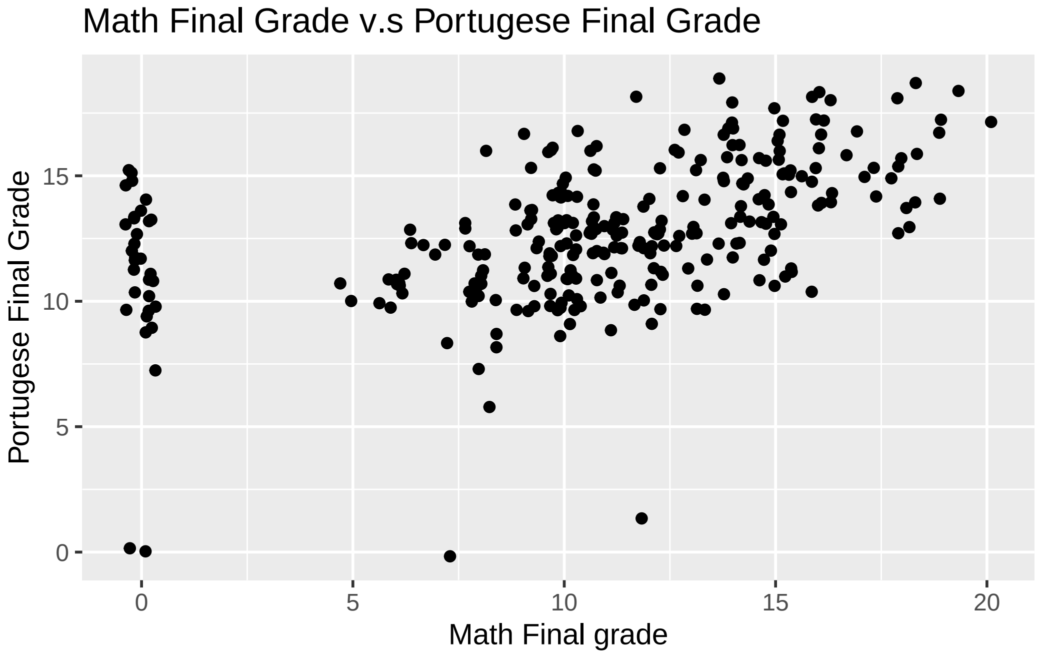
```
glimpse(MathPort)
```

```
Rows: 292
Columns: 4
$ G3.x         <int> 6, 6, 15, 10, 11, 6, 19, 15, 9, 12, 14, 11, 16, 14, 10, 1~
$ G3.y         <int> 11, 11, 14, 13, 13, 13, 17, 13, 14, 13, 12, 13, 15, 17, 1~
$ Walc.x       <int> 1, 1, 1, 2, 1, 1, 1, 1, 2, 1, 3, 2, 1, 2, 1, 3, 1, 1, 3, ~
$ traveltime.x <int> 2, 1, 1, 1, 1, 2, 1, 1, 1, 3, 1, 2, 1, 1, 3, 1, 1, 1, 1, ~
```

2. Then generate an appropriate visual. Make sure your graphic has appropriate labels, legends (as needed), and a title.

Solution:

```
ggplot(MathPort, aes(x = G3.x, y = G3.y)) + geom_jitter() + labs(title = "Math Final Grade
```



3. Finally, in a few sentences, describe what you find.

Solution:

Data Set Information for Problem 5

The data set is from a paper called “Using Data Mining To Predict Secondary School Student Alcohol Consumption” by Fabio Pagnotta and Hossain Mohammad Amran of the Department of Computer Science, University of Camerino, and the data set is hosted online in UCI’s machine learning repository.

The information below was copied from the provided codebook online.

Attributes for both student-mat.csv (Math course) and student-por.csv (Portuguese language course) datasets:

1. school - student’s school (binary: ‘GP’ - Gabriel Pereira or ‘MS’ - Mousinho da Silveira)
2. sex - student’s sex (binary: ‘F’ - female or ‘M’ - male)
3. age - student’s age (numeric: from 15 to 22)
4. address - student’s home address type (binary: ‘U’ - urban or ‘R’ - rural)
5. famsize - family size (binary: ‘LE3’ - less or equal to 3 or ‘GT3’ - greater than 3)
6. Pstatus - parent’s cohabitation status (binary: ‘T’ - living together or ‘A’ - apart)
7. Medu - mother’s education (numeric: 0 - none, 1 - primary education (4th grade), 2 - 5th to 9th grade, 3 - secondary education or 4 - higher education)
8. Fedu - father’s education (numeric: 0 - none, 1 - primary education (4th grade), 2 - 5th to 9th grade, 3 - secondary education or 4 - higher education)
9. Mjob - mother’s job (nominal: ‘teacher’, ‘health’ care related, civil ‘services’ (e.g. administrative or police), ‘at_home’ or ‘other’)
10. Fjob - father’s job (nominal: ‘teacher’, ‘health’ care related, civil ‘services’ (e.g. administrative or police), ‘at_home’ or ‘other’)
11. reason - reason to choose this school (nominal: close to ‘home’, school ‘reputation’, ‘course’ preference or ‘other’)
12. guardian - student’s guardian (nominal: ‘mother’, ‘father’ or ‘other’)
13. traveltime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
14. studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
15. failures - number of past class failures (numeric: n if 1<=n<3, else 4)
16. schoolsup - extra educational support (binary: yes or no)
17. famsup - family educational support (binary: yes or no)
18. paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
19. activities - extra-curricular activities (binary: yes or no)
20. nursery - attended nursery school (binary: yes or no)
21. higher - wants to take higher education (binary: yes or no)
22. internet - Internet access at home (binary: yes or no)
23. romantic - with a romantic relationship (binary: yes or no)
24. famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)

25. freetime - free time after school (numeric: from 1 - very low to 5 - very high)
26. goout - going out with friends (numeric: from 1 - very low to 5 - very high)
27. Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
28. Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
29. health - current health status (numeric: from 1 - very bad to 5 - very good)
30. absences - number of school absences (numeric: from 0 to 93)

Finally, the grades are related with the course subject, Math or Portuguese:
31. G1 - first period grade (numeric: from 0 to 20)
32. G2 - second period grade (numeric: from 0 to 20)
33. G3 - final grade (numeric: from 0 to 20, output target)
Thus, these variables appear in each data set, but have different meaning in each.

The data was provided as two different .csv files online. I obtained some errors trying to work with them, so ended up saving them as .txt files, which are in the data subfolder for the Practice. Be sure you obtained the files in this folder! Many of the students were in both courses, but not all.