Problem Set 0

Due September 6, 2023, 10am on Canvas

To successfully complete this problem set, please follow these steps:

1. Download the contents of the assignment Dropbox link in a folder (directory) on your computer designated for this problem set only. Follow the directory structure, e.g. putting the `data` folder containing the datasets as a subdirectory of the project directory.

2. Insert your answers in the yellow boxes using Microsoft Word, and prepare a single `.R` script for what you produce. Save the word document as a `.PDF`.

3. Please submit the PDF to the designated `PS-XX: pdf` link and your R Script to the `PS-XX: R` link.

(1) Your name:

Nikhilla Bhuvana Sundar

(2) Group members, if any:

N/A

(3) Compliance with the Academic Code on problem set[1] (sign with an X below)

X

[1] You may use the same code from classmates, Ed Discussion Board, instructors, and generative AI. However, you must hand in your own unique written work and code in all cases. Any copy/paste of another's work is plagiarism. In other words, you can work with your classmate(s), sitting side-by-side and going through the problem set question-by-question, or use generative AI to provide potential code for you, but you must each type your own answers and your own code.

The goal of this problem set is to get you started with data visualization, help you learn correct syntax, and help you to understand how we will grade code. We have provided the solution code for you at the bottom. You should still try to solve the questions by yourself first.

This assignment will be assigned points based on whether the R code runs on the autograder, and whether you have provided the correct figures in the Word Document. However, those points will not count toward your course grade unless you submit something.

R Primers

Before you start PS-01 due Monday, September, 11, please go complete all of the following R tutorials (primers). If you are new to R, you probably want to start this first before PS-00 or interchange between the two. If you are already familiar with tidyverse, this will be a fast review.

1. Visualization Basics (`ggplot2`)
2. Programming Basics (objects and functions)
3. Work with Tibbles
4. Isolating Data with `dplyr` and the pipe
5. Creating Variables and Dataframes with `dplyr`

As you write, be conscious of whether you are following tidyverse code style.

Project Setup

The rest of the semester you will be using R on your personal computers. Please use the latest versions and projects settings we discuss on the first two classes. To recap, those are:

1. To download the correct versions of R and RStudio:https://vimeo.com/743587308 (5 minutes)
2. To configure packages for reproducible workflow: https://vimeo.com/743589249 (8 minutes)

Problem 1: Read and Clean Data

Only start this section once you have set up your Rstudio Project as shown in the previous page.

`states-data.xlsx` in the data folder is an Excel file with 50 rows, one for each state. It contains the following variables.

Variable Name	Description
State Abbrevation	Two-letter abbreviation for state
total_pop	Total population (2020 Census)
white_pop	Total population that is non-hispanic White
D 2020	Votes for the Democratic Presidentical candidate in the 2020 election (Biden)
R 2020	Votes for the Republican Presidentical candidate in the 2020 election (Trump)

1.1

Read in the dataset by the function `readxl::read_excel()`[2] and assign it to the object `dat_orig`.

1.2

Create a new dataset called `dat` that has the following modifications:

1. The variables in this dataset are formatted for reading, but they do not conform to tidyverse syntax and are tedious to type up in code. Change all the variable names to "snake case" as in the style guide. An easy way to do this is to use the function `janitor::clean_names()`.

2. Create a new variable called `biden_vshare` that is the two party voteshare of Joe Biden in the state, defined as

$$V_i = \frac{D_i^{2020}}{D_i^{2020} + R_i^{2020}}$$

   where D_i^{2020} is the total votes obtained by Biden in the 2020 election, and R_i^{2020} is Donald Trump's.

3. Change the long variable names to the following shorter versions that follow tidyverse code style:

   - "State (Abbreviated)" should be `st`
   - "Income per capita" should be `inc`

---

[2] In R, we use the double colon notation `readxl::read_excel` where the word on the left hand side of the :: is the package and the word on the right hand side of the :: is the function. If you are having trouble or if this is your first time reading a file, you can use the menu and go to `File` (on the toolbar), `Import Dataset`, then `From Excel` and follow the instructions.

- "Disposable Income per Capita" should be `disp`
- "Median Household Income" should be `inc_med`

Problem 2: Visualize

2.1

Create a scatterplot with per capita income on the x-axis and 2020 Biden voteshare on the y-axis, with the following adjustments.

- Label the y-axis title `"2020 Democratic Presidential Voteshare"`
- Label the x-axis title `"Household Income Per Capita"`

In the class, any graph that you present should have, at a minimum, readable titles like this.

2.2 Label Axes

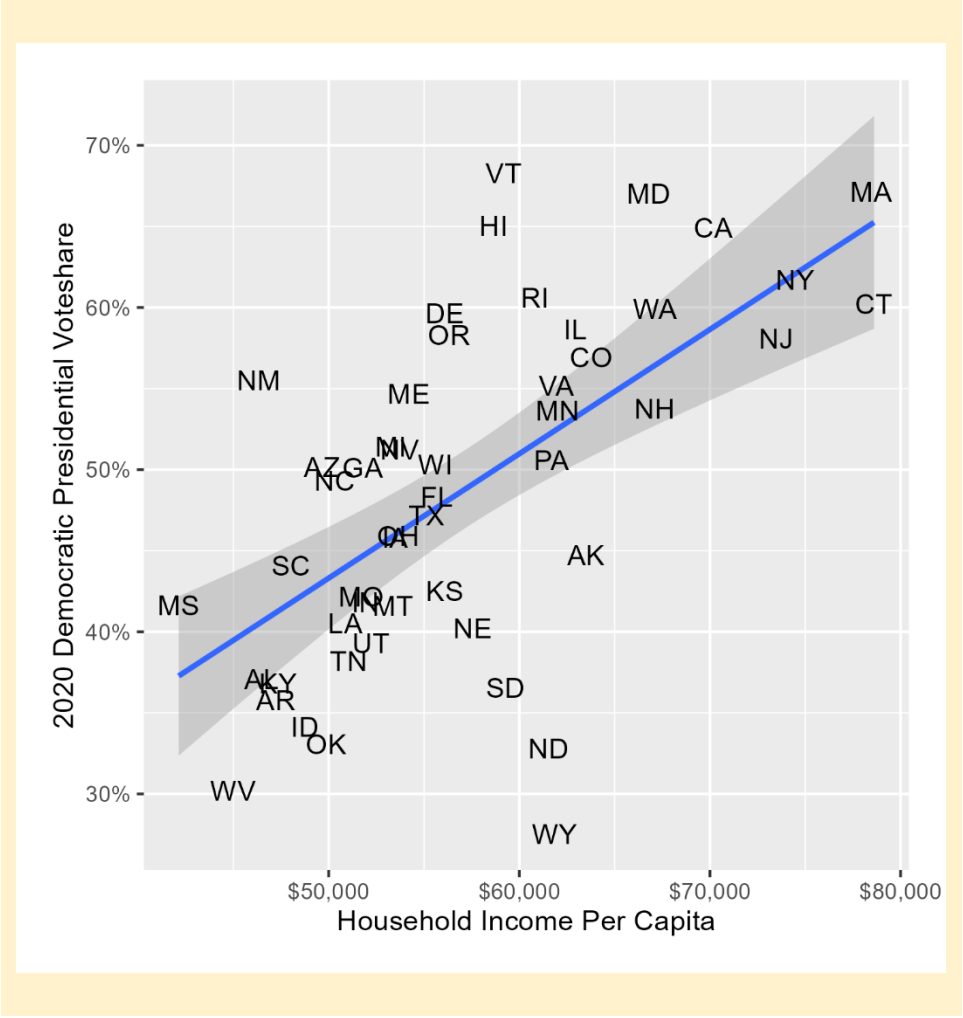
Make a new figure that makes the following modifications to the above figure. Assign this ggplot to an R object called `gg_scatter`

- Modify the tick labels of the y-axis to display decimals as percentages. This avoids reader confusion of whether "0.5" means 0.5% or 50%. Use the `scales::percent` function in the `labels` argument of the `scale_y_continuous` layer
- Modify the tick labels of the x-axis to display dollar signs. Apply it the same way as the y-axis, but using the `scales::dollar` function instead of percent.
- Do not use points to label the data points. Instead, use the two-letter abbreviation of the state using `geom_text`.
- Show the best-fit straight line. Change the smoothing method from the default to a linear regression (`lm`), or ordinary least squares regression.

2.3 Saving Figures

Write code that will save the figure into the following file. Then, copy and paste (or drag) the file into the yellow box to show your output.

- The filename should be `"inc-partisan-2020.png"`
- Make a new folder called `"figures"` in the project directory and save the file there.
- Save it 5 inches wide and 5 inches high
- Add a 0.2 inch whitespace around the figure so that the axis labels are not cut off. Use the following line of code: `theme(plot.margin = margin(0.2, 0.2, 0.2, 0.2, "in"))`.


70%
60%
50%
40%
30%
2020 Democratic Presidential Voteshare
$50,000
$60,000
$70,000
$80,000
Household Income Per Capita
VT
MD
MA
HI
CA
NY
RI
WA
CT
DE
OR
IL
NJ
CO
NM
VA
ME
MN
NH
PA
WI
NC
SC
KS
MS
MT
NE
LA
UT
TN
SD
AR
ID
OK
ND
WV
WY
AK

3: Teachly Profile

To help me keep track of attendance and classroom participation patterns, I will be using a too called Teachly. For this to work, please take 3-5 minutes to clicking on the link below, signing up with your yale.edu address, and fill in your Student Profile:

https://bit.ly/47MARzT

Your answers can be very brief. The purpose of collecting this information is so the teaching team can incorporate student's relevant experience in some of the lectures. If have any relevant work experience (e.g., being an research assistant for J-PAL), you should note them. The only reason we ask your race, gender, and language is to track whether classroom participation is represented (e.g., whether men are talking more than women).[3]

X	Enter a X here if you have completed the Teachly Profile

[3] The information in your Student Profile will be used for teaching-related purposes only (see Teachly's privacy policy here), and you can edit your student profile at any time.

Solution Code

The code that follows is the recommended solution for this problem set. Note that in your own script, you will need to specify the *loading* of necessary libraries for any of the functions to run. However, both here and as general practice, you should **not** specify the installation of those packages in your scripts.

The autograder will only have the following packages installed, so attempting to load any other package in your script will throw an error.

```
tidyverse
readxl
janitor
scales
```

The recommended script follows:

```
dat_orig <- read_excel("data/states-data.xlsx")


## 1 ----
dat <- dat_orig |>
  clean_names() |>
  mutate(biden_vshare = d_2020 / (d_2020 + r_2020),
         obama_vshare = d_2012 / (d_2012 + r_2012)) |>
  rename(st = st_abbrev,
         inc = income_percapita,
         disp = disposable_income_percapita,
         inc_med = median_household_income)

## 2 ----
dat |>
  ggplot(aes(x = inc, y = biden_vshare)) +
  geom_point() +
  labs(
    x = "Household Income Per Capita",
    y = "2020 Democratic Presidential Voteshare"
  )

gg_scatter <- dat |>
  ggplot(aes(x = inc, y = biden_vshare)) +
  geom_smooth(method = "lm") +
  geom_text(aes(label = st)) +
  scale_x_continuous(labels = dollar) +
  scale_y_continuous(labels = percent) +
  labs(
    x = "Household Income Per Capita",
    y = "2020 Democratic Presidential Voteshare"
  ) +
  theme(plot.margin = margin(0.2, 0.2, 0.2, 0.2, "in"))
```

```
# 3 ----
ggsave("figures/inc-partisan-2020.png", width = 5, height = 5)
```