ggplot2, pt. 2

Sarah Moore

2022-10-05

Getting Started

We are going to go through a workflow today to practice working with version control and saving from R directly. First, start a new project in your computer's local folder for this class. Call the R project anes_survey_work.

When you are in the project let's open an R script. Your working directory *should* be now set to whatever file you set as the path for your R project. So, it should be the file for this class. We can check this. So long as it's what we expected, we can just move along.

What's next when we *VERY* first set out an R script?

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.0      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
## Warning: One or more parsing issues, see `problems()` for details
```

```
## Rows: 8280 Columns: 1771
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (20): version, V203001, V203054, V203056, V203078, V203079, V203080, V...
## dbl (1751): V200001, V160001_orig, V200002, V200003, V200004, V200005, V2000...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Let's pick at max six variables that we want to work with just so that we don't have to get used to too much at once.

How about these:

a. **V200001**: Case ID

b. **V201028**: Did the respondent vote for president

-9. Refused

-1. Inapplicable

1. Yes
2. No

c. **V202024**: Has respondent gotten into a political argument in past 12 months

-9. Refused

-7. Incomplete interview

-6. No post-election interview

1. Yes
2. No

d. **V202025**: Has respondent joined a protest march, rally, or demonstration in past 12 months

-9. Refused

-7. Incomplete interview

-6. No post-election interview

1. Yes
2. No

e. **V201014e**: Party registration in state of voter registration

-1. Inapplicable

0. No party registration
1. Party registration

f. **V201115**: How hopeful respondent feels about how things are going in the country

-9. Refused

-8. Don't know

1. Not at all
2. A little
3. Somewhat
4. Very
5. Extremely

g. **V201600**: Respondent sex

-9. Refused

1. Male
2. Female

Based on these variables we want to work with, subset them using `select` and get rid of all the unwanted values.

```
anes2020 %>%
  select(resp_id = V200001, resp_pres_vote = V201028, resp_pol_argument = V202024,
         resp_part_protest = V202025, resp_state_reg = V201014e,
         resp_hopeful = V201115, resp_sex = V201600) -> anes_sub


anes_sub
```

```
## # A tibble: 8,280 x 7
##    resp_id resp_pres_vote resp_pol_argument resp_part_~1 resp_~2 resp_~3 resp_~4
##      <dbl>          <dbl>             <dbl>        <dbl>   <dbl>   <dbl>   <dbl>
##  1  200015             -1                 1            2       1       3       1
##  2  200022             -1                 2            2       1       1       2
##  3  200039             -1                 1            1       0       1       2
##  4  200046             -1                 2            2       1       2       1
##  5  200053             -1                 2            2       1       4       1
##  6  200060             -1                 1            2       0       1       2
##  7  200084             -1                 1            2       1       4       2
##  8  200091             -1                 1            2      -2       4       2
##  9  200107              1                 2            2       1       3       2
## 10  200114             -1                 1            2       1       1       1
## # ... with 8,270 more rows, and abbreviated variable names
## #   1: resp_part_protest, 2: resp_state_reg, 3: resp_hopeful, 4: resp_sex
```

```
bad_values <- c(-1, -2, -5, -6, -7, -8, -9)

anes_sub %>%
  mutate(across(everything(), na_if_in, bad_values))-> anes_clean_sub

any(anes_clean_sub[anes_clean_sub<0])
```

```
## [1] FALSE
```

```
any(bad_values %in% anes_clean_sub)
```

```
## [1] FALSE
```

What kinds of questions might motivate looking at these variables?

What kinds of variables are these???

- Out of the `geom` functions here, which make sense to use for univariate description?
- What about bivariate or multivariate?

Using one of these ideas let's build from the bottom up.

```
# let's start with the data.

# now we map our x (and/or) y aesthetics, sometimes more if we want

# let's set a geom_ layer

#now run the code together!!
```