Homework 1

Determinants of COVID vaccination rates

First, a little detour to describe several alternatives for reading in data:

If you navigate to my Github account, and find the `264_fall_2025` repo, there is a Data folder inside. You can then click on `vacc_Mar21.csv` to see the data we want to download. This link should also get you there, but it's good to be able to navigate there yourself.

```
# Approach 1
vaccine_data <- read_csv("Data/vaccinations_2021.csv")                    ①

# Approach 2
vaccine_data <- read_csv("~/264_fall_2025/Data/vaccinations_2021.csv")    ②

# Approach 3
vaccine_data <- read_csv("https://proback.github.io/264_fall_2025/Data/vaccinations_2021.csv

# Approach 4
vaccine_data <- read_csv("https://raw.githubusercontent.com/proback/264_fall_2025/refs/heads/
```

① Approach 1: create a Data folder in the same location where this .qmd file resides, and then store vaccinations_2021.csv in that Data folder
② Approach 2: give R the complete path to the location of vaccinations_2021.csv, starting with Home (~)
③ Approach 3: link to our course webpage, and then know we have a Data folder containing all our csvs
④ Approach 4: navigate to the data in GitHub, hit the Raw button, and copy that link

A recent Stat 272 project examined determinants of covid vaccination rates at the county level. Our data set contains 3053 rows (1 for each county in the US) and 14 columns; here is a quick description of the variables we'll be using:

- `state` = state the county is located in

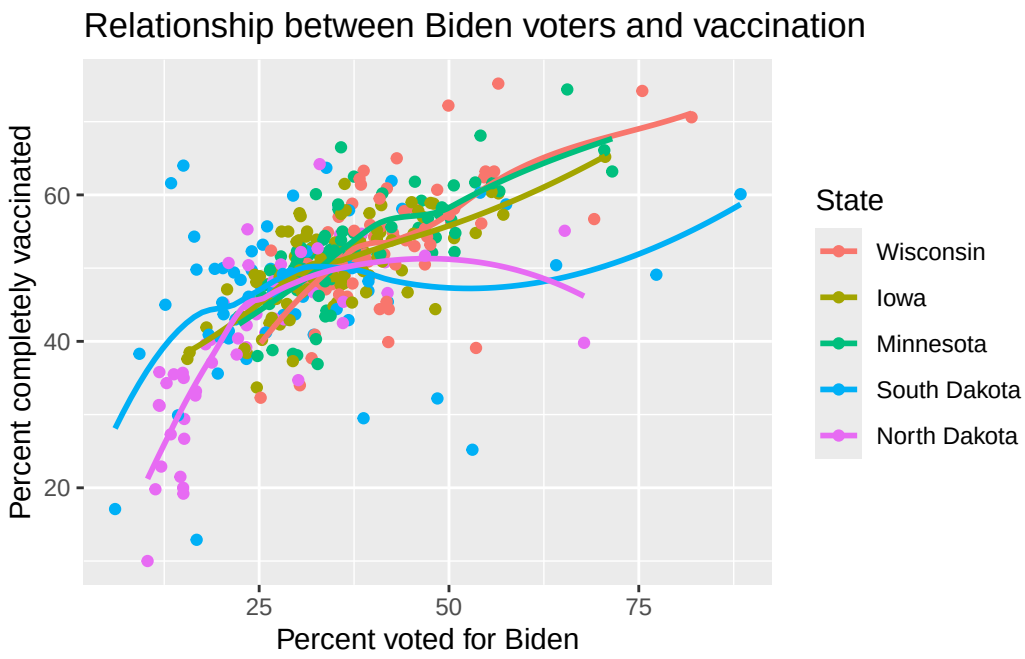
- `county` = name of the county
- `region` = region the state is located in
- `metro_status` = Is the county considered "Metro" or "Non-metro"?
- `rural_urban_code` = from 1 (most urban) to 9 (most rural)
- `perc_complete_vac` = percent of county completely vaccinated as of 11/9/21
- `tot_pop` = total population in the county
- `votes_Trump` = number of votes for Trump in the county in 2020
- `votes_Biden` = number of votes for Biden in the county in 2020
- `perc_Biden` = percent of votes for Biden in the county in 2020
- `ed_somecol_perc` = percent with some education beyond high school (but not a Bachelor's degree)
- `ed_bachormore_perc` = percent with a Bachelor's degree or more
- `unemployment_rate_2020` = county unemployment rate in 2020
- `median_HHincome_2019` = county's median household income in 2019

1. Consider only Minnesota and its surrounding states (Iowa, Wisconsin, North Dakota, and South Dakota). We want to examine the relationship between the percentage who voted for Biden and the percentage of complete vaccinations by state. Generate two plots to examine this relationship:

a) A scatterplot with points and smoothers colored by state. Make sure the legend is ordered in a meaningful way, and include good labels on your axes and your legend. Also leave off the error bars from your smoothers.

```
vaccine_data |>
  filter(state %in% c("Minnesota", "Iowa", "Wisconsin",
                      "North Dakota", "South Dakota")) |>
  ggplot(aes(x = perc_Biden, y = perc_complete_vac,
             color = fct_reorder2(state, perc_Biden, perc_complete_vac))) +
  geom_point() +
  geom_smooth(se = FALSE) +
  labs( x = "Percent voted for Biden",
        y = "Percent completely vaccinated",
        title = "Relationship between Biden voters and vaccination",
        color = "State")
```

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

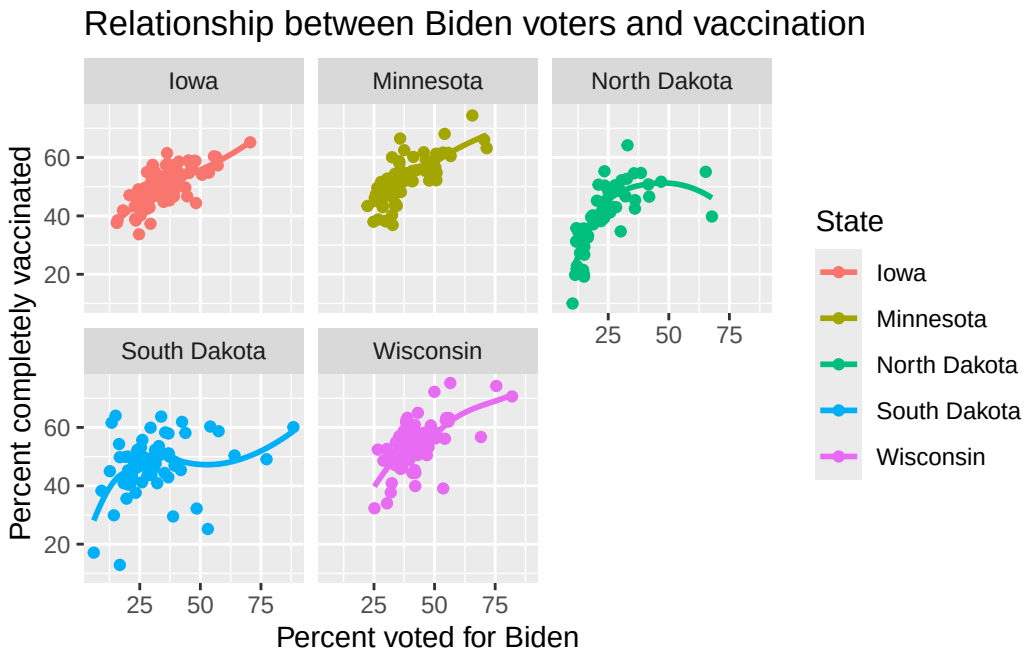


b) One plot per state containing a scatterplot and a smoother.

Describe which plot you prefer and why. What can you learn from your preferred plot?

```
vaccine_data |>
  filter(state %in% c("Minnesota", "Iowa", "Wisconsin",
                      "North Dakota", "South Dakota")) |>
  ggplot(aes(x = perc_Biden, y = perc_complete_vac, color = state)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  labs( x = "Percent voted for Biden",
        y = "Percent completely vaccinated",
        title = "Relationship between Biden voters and vaccination",
        color = "State") +
  facet_wrap(~state)
```

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

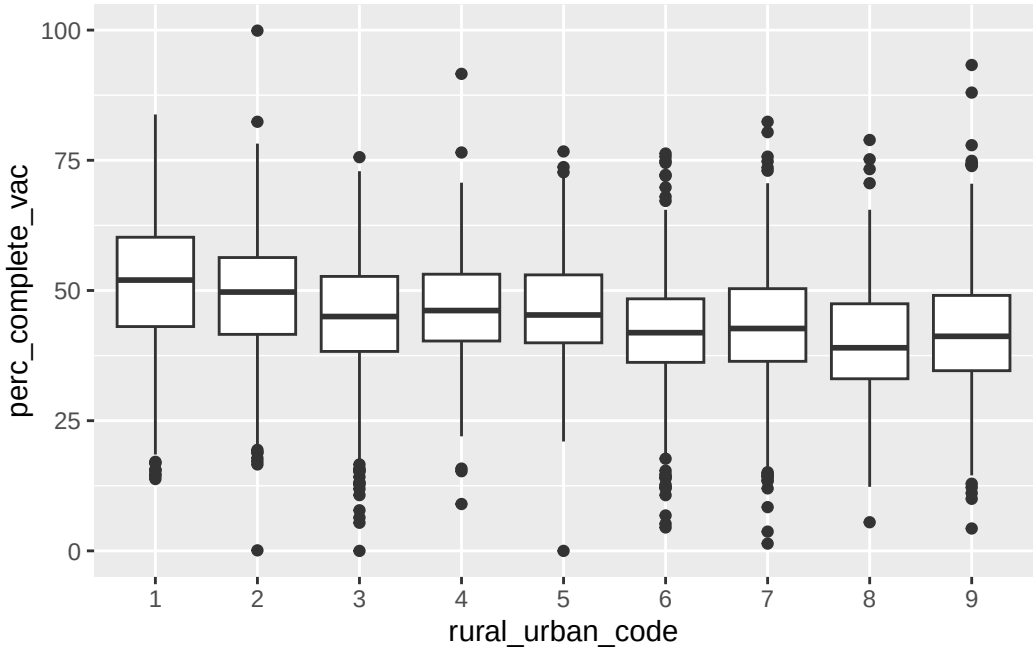


I prefer the first plot because it is easier to compare between states when on the same plot since they are fairly similar, but the second is nice if you are trying to just see what the trend is in every state and are doing less strict comparing.

4. Produce 3 different plots for illustrating the relationship between the rural_urban_code and percent vaccinated. Hint: you can sometimes turn numeric variables into categorical variables for plotting purposes (e.g. `as.factor()`, `ifelse()`).

State your favorite plot, why you like it better than the other two, and what you can learn from your favorite plot. Create an alt text description of your favorite plot, using the Four Ingredient Model. See this link for reminders and references about alt text.

```
vaccine_data |>
  mutate(rural_urban_code = as.factor(rural_urban_code)) |>
  ggplot(aes(x = rural_urban_code, y = perc_complete_vac)) +
  geom_boxplot()
```

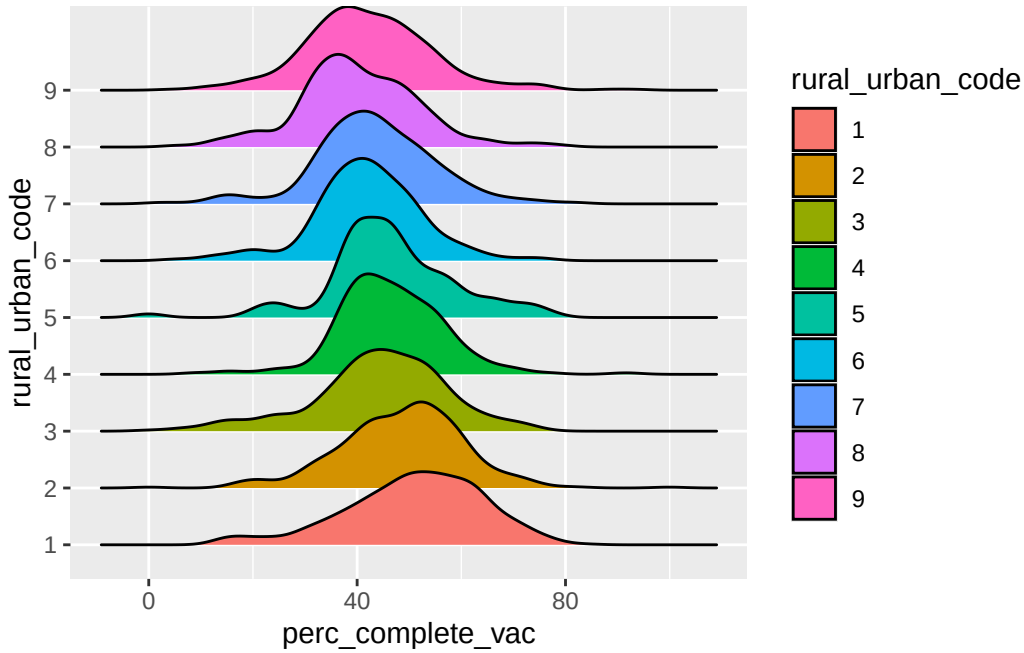



```
library(ggridges)
```

```
Warning: package 'ggridges' was built under R version 4.4.3
```

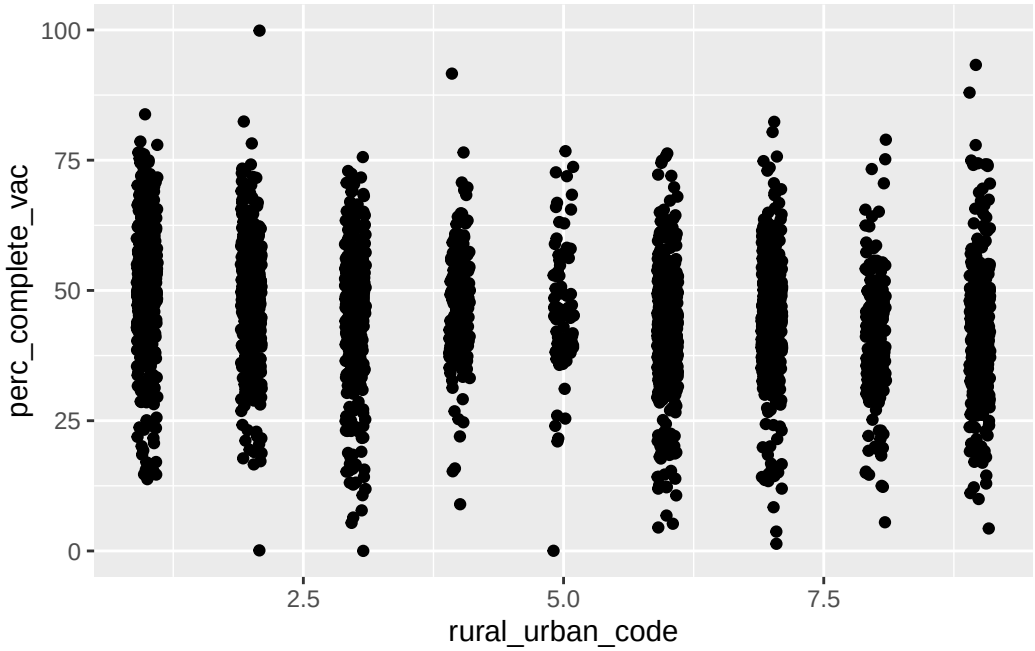
```
vaccine_data |>
  mutate(rural_urban_code = as.factor(rural_urban_code)) |>
  ggplot(aes(x = perc_complete_vac, y = rural_urban_code, fill = rural_urban_code)) +
  geom_density_ridges()
```

```
Picking joint bandwidth of 3.04
```




```
#violin plot

#scatterplot if we pretend its two numerics (with jitter)
vaccine_data |>
  ggplot(aes(x = rural_urban_code, y = perc_complete_vac)) +
  geom_jitter(width = 0.1)
```



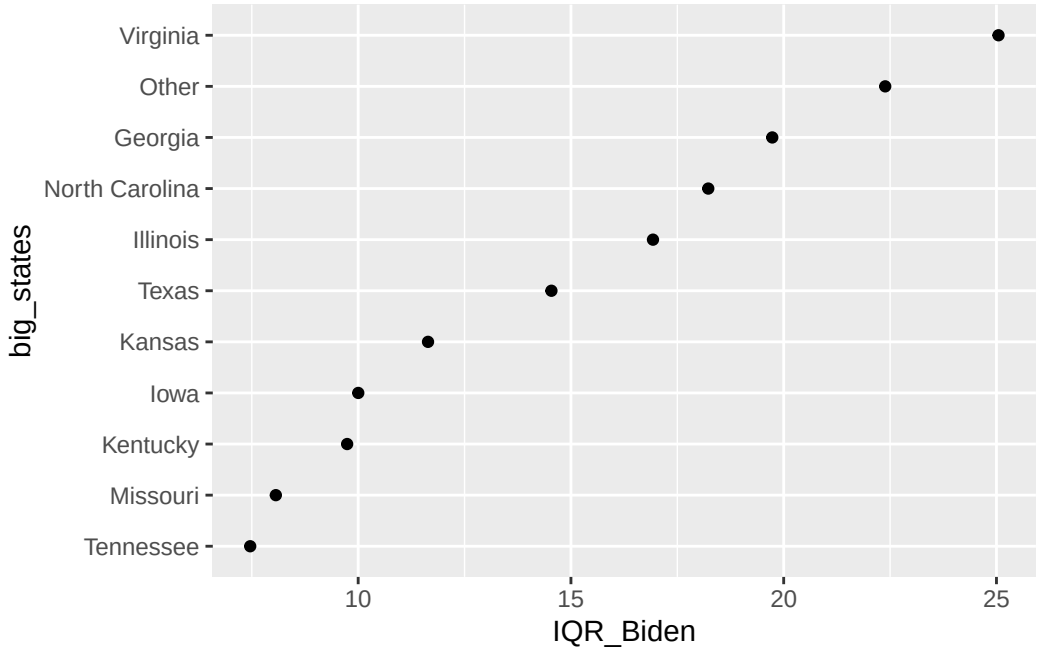

```
#could even treat as two categorical variables (cut functions or ifelse)
```

I prefer the density ridges plot, as it makes it very clear based on looking at the peaks what the trend between the codes is, and I find the color coding and stacking easier to read and understand compared to the box plot.

Alt text: This is a density ridges plot of the percentage of a county completely vaccinated from 0 to 100% on the x-axis organized by the rural-urban code of the county from 1 to 9 on the y-axis and as the color coding. The plot indicates that more rural counties have a lower vaccination percentage than urban counties, peaking around 55% for the most urban counties (code 1) and as low as 40% for the most rural counties (code 9).

5. BEFORE running the code below, sketch the plot that will be produced by R. AFTER running the code, describe what conclusion(s) can we draw from this plot?

```
vaccine_data |>
  filter(!is.na(perc_Biden)) |>
  mutate(big_states = fct_lump(state, n = 10)) |>
  group_by(big_states) |>
  summarize(IQR_Biden = IQR(perc_Biden)) |>
  mutate(big_states = fct_reorder(big_states, IQR_Biden)) |>
  ggplot() +
    geom_point(aes(x = IQR_Biden, y = big_states))
```



We can determine which of the states with the most counties have the largest variability amongst their counties in what percentage voted for Biden. Virginia has the largest variability and Tennessee has the least. States not in the top ten as a whole also have large variability when considered together.

6. In this question we will focus only on the 12 states in the Midwest (i.e. where region == "Midwest").

a) Create a tibble with the following information for each state. Order states from least to greatest state population.

- number of different `rural_urban_codes` represented among the state's counties (there are 9 possible)
- total state population
- proportion of Metro counties
- median unemployment rate

```
state_stuff <- vaccine_data |>
  filter(region == "Midwest") |>
  group_by(state) |>
  summarise(
    num_r_u_codes = n_distinct(rural_urban_code),
    tot_state_pop = sum(tot_pop),
   prop_metro = mean(metro_status == "Metro"),
```

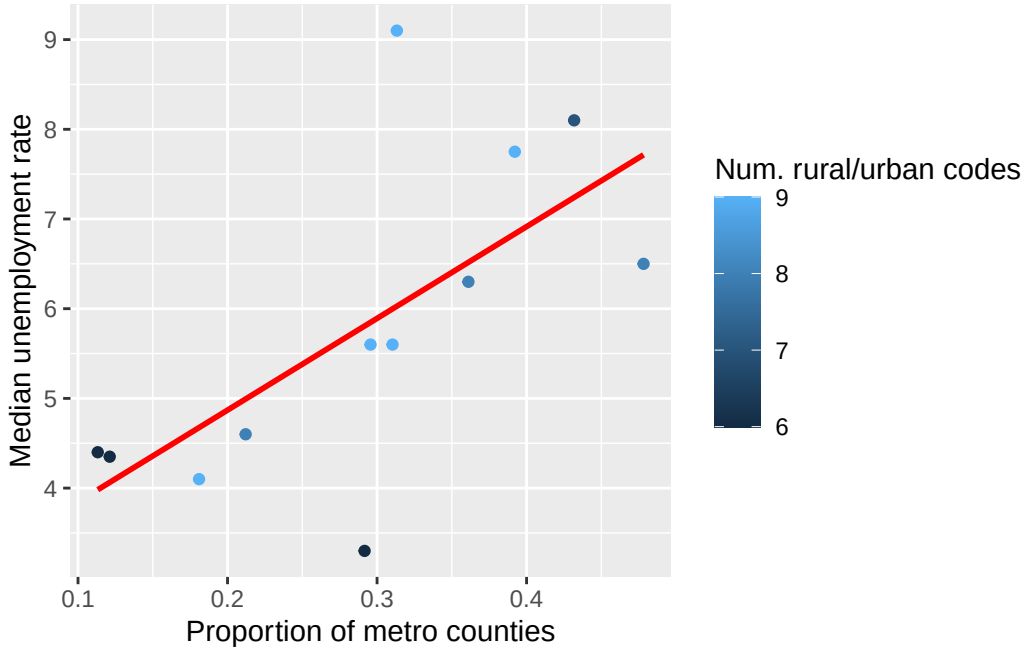
```
    med_unemp = median(unemployment_rate_2020)) |>
  arrange(tot_state_pop)
state_stuff
```

```
# A tibble: 12 x 5
   state        num_r_u_codes tot_state_pop prop_metro med_unemp
   <chr>                <int>         <dbl>      <dbl>     <dbl>
 1 North Dakota             6        762062      0.113      4.4
 2 South Dakota             6        884659      0.121      4.35
 3 Nebraska                 6       1261262      0.292      3.3
 4 Kansas                   9       2913314      0.181      4.1
 5 Iowa                     8       3155070      0.212      4.6
 6 Minnesota                9       5639632      0.310      5.6
 7 Wisconsin                8       5822434      0.361      6.3
 8 Missouri                 9       6137428      0.296      5.6
 9 Indiana                  8       6732219      0.478      6.5
10 Michigan                 9       9986857      0.313      9.1
11 Ohio                     7      11689100      0.432      8.1
12 Illinois                 9      12671821      0.392      7.75
```

b) Use your tibble in (a) to produce a plot of the relationship between proportion of Metro counties and median unemployment rate. Points should be colored by the number of different `rural_urban_codes` in a state, but a single linear trend should be fit to all points. What can you conclude from the plot?

```
state_stuff |>
  ggplot(aes(x = prop_metro, y = med_unemp)) +
  geom_point(aes(color = num_r_u_codes)) +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(color = "Num. rural/urban codes",
       x = "Proportion of metro counties",
       y = "Median unemployment rate")
```

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



8. Hypothetical R chunk #1:

```
# Hypothetical R chunk 1
temp <- vaccine_data |>
  mutate(new_perc_vac = ifelse(perc_complete_vac > 95, NA, perc_complete_vac),
         MD_group = cut_number(people_per_MD, 3)) |>
  group_by(MD_group) |>
  summarise(n = n(),
            mean_perc_vac = mean(new_perc_vac, na.rm = TRUE),
            mean_white = mean(perc_white, na.rm = TRUE))
```

a) Describe the tibble `temp` created above. What would be the dimensions? What do rows and columns represent? It would have 3 rows and 4 columns; each row represents an MD_group, aka whether or not there is a low, medium, or high number of residents per doctor. One of the columns (MD_group) represents the MD_group, n represents how many counties fall into that group, mean_perc_vac represents the mean percentage of residents completely vaccinated, and mean_white represents the mean percentage of white residents.

b) What would happen if we replaced `new_perc_vac = ifelse(perc_complete_vac > 95, NA, perc_complete_vac)` with `new_perc_vac = ifelse(perc_complete_vac > 95, perc_complete_vac, NA)`? The tibble would only include counties with a vaccination rate above 95% because any ones below 95% would become an NA and then later get removed in the means.

c) What would happen if we replaced `mean_white = mean(perc_white, na.rm = TRUE)` with `mean_white = mean(perc_white)`? It would possibly end up as NA if any of the counties do not have this data, which is pretty likely since we made anything above 95% into an NA.

d) What would happen if we removed `group_by(MD_group)`? The summarize would summarize everything into a single line with 3 columns for n, mean_perc_vac, and mean_white

9. Hypothetical R chunk #2:

```
# Hypothetical R chunk 2
ggplot(data = vaccine_data) +
  geom_point(mapping = aes(x = perc_over_65, y = perc_complete_vac,
                           color = HR_party)) +
  geom_smooth()

temp <- vaccine_data |>
  group_by(HR_party) |>
  summarise(var1 = n()) |>
  arrange(desc(var1)) |>
  slice_head(n = 3)

vaccine_data |>
  ggplot(mapping = aes(x = fct_reorder(HR_party, perc_over_65, .fun = median),
                       y = perc_over_65)) +
    geom_boxplot()
```

a) Why would the first plot produce an error? The first plot would produce an error because the mapping/aesthetic has been placed in the geom_point, meaning that there is no mapping/aesthetic for the geom_smooth (as opposed to making it a global mapping/aesthetic by putting it in the ggplot function), so it will not know what to plot and will run an error.

b) Describe the tibble `temp` created above. What would be the dimensions? What do rows and columns represent? The tibble would have 3 rows representing whichever parties had the largest number of observations in the original dataset (so the largest number of counties with a US Rep. of that party). It would have two columns; one representing the party and one for the number of observations/counties with a US Rep. of that party.

c) What would happen if we replaced `fct_reorder(HR_party, perc_over_65, .fun = median)` with `HR_party`? If we replaced this code, the boxplots would not be ordered by median but by whatever the original ordering of HR_party was, so likely just alphabetically.

10. Hypothetical R chunk #3:

```
# Hypothetical R chunk 3
vaccine_data |>
  filter(!is.na(people_per_MD)) |>
  mutate(state_lump = fct_lump(state, n = 4)) |>
  group_by(state_lump, rural_urban_code) |>
  summarise(mean_people_per_MD = mean(people_per_MD)) |>
  ggplot(mapping = aes(x = rural_urban_code, y = mean_people_per_MD,
      colour = fct_reorder2(state_lump, rural_urban_code, mean_people_per_MD))) +
    geom_line()
```

a) Describe the tibble piped into the ggplot above. What would be the dimensions? What do rows and columns represent? The tibble will have an unknown number of rows determined by the number of rural urban codes in the four states with the most observations (counties) plus the "other" category. It will be the number of state-code pairs. There will be 3 columns, one for the state (or "other"), one for the rural urban code, and one for the mean number of residents per doctor under that state and rural urban code.

b) Carefully describe the plot created above. The plot will be a line plot with the rural urban code on the x axis and mean number of residents per doctor on the y axis, colored by which of the five state options it represents. The color of the lines will be ordered by the value of the final observation in each line using fct_reorder2 (so the line highest on the right side of the dataset will be top of the color order in the legend).

c) What would happen if we removed `filter(!is.na(people_per_MD))`? If we removed this filter, than counties that do not have the residents per doctor data will remain in the dataset, which will cause any of the five state categories that includes one of these counties to have its mean also become NA, which will probably ruin the plot and conclusions being drawn.

d) What would happen if we replaced `fct_reorder2(state_lump, rural_urban_code, mean_people_per_MD)` with `state_lump`? If we replaced this code, the coloring of the lines would not be ordered by the value of the final observation but by whatever the original ordering of state_lump was, so likely just alphabetically or by number of counties.