Review of Data Science 1

You can download this .qmd file from here. Just hit the Download Raw File button.

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_2024` 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_2024/Data/vaccinations_2021.csv")  #<2>

# Approach 3
vaccine_data <- read_csv("https://proback.github.io/264_fall_2024/Data/vaccinations_2021.c

# Approach 4
vaccine_data <- read_csv("https://raw.githubusercontent.com/proback/264_fall_2024/main/Dat
```

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:

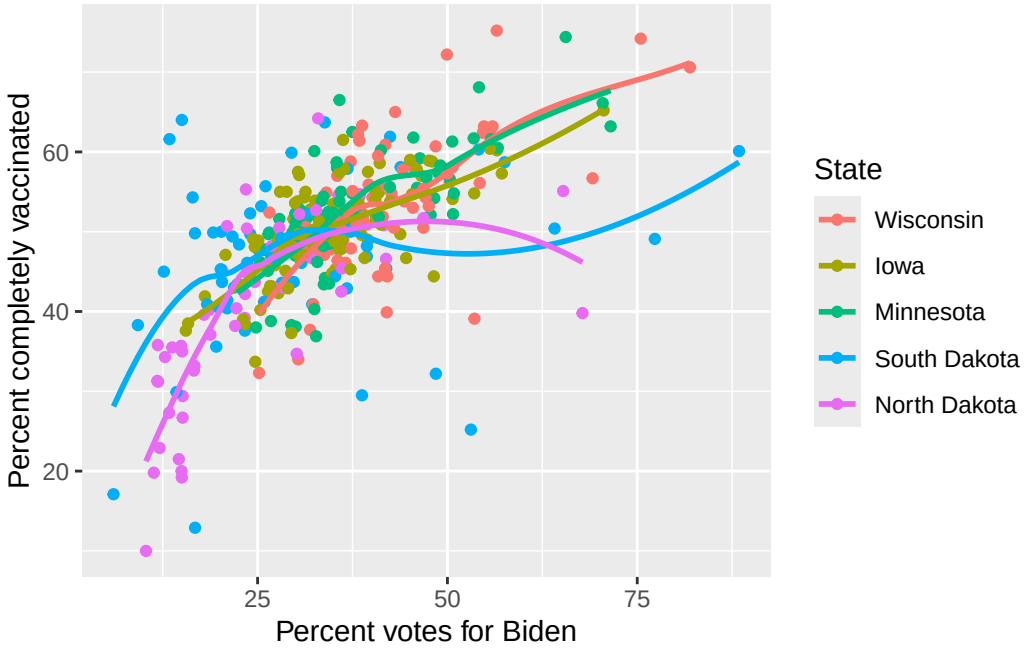
- `state` = state the county is located in
- `county` = name of the county
- `region` = region the state is located in
- `metro_status` = Is the county considered "Metro" or "Non-metro"?
- `rural_urban_code` = from 1 (most urban) to 9 (most rural)
- `perc_complete_vac` = percent of county completely 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 votes for Biden",
       y = "Percent completely vaccinated",
       color = "State")
```

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

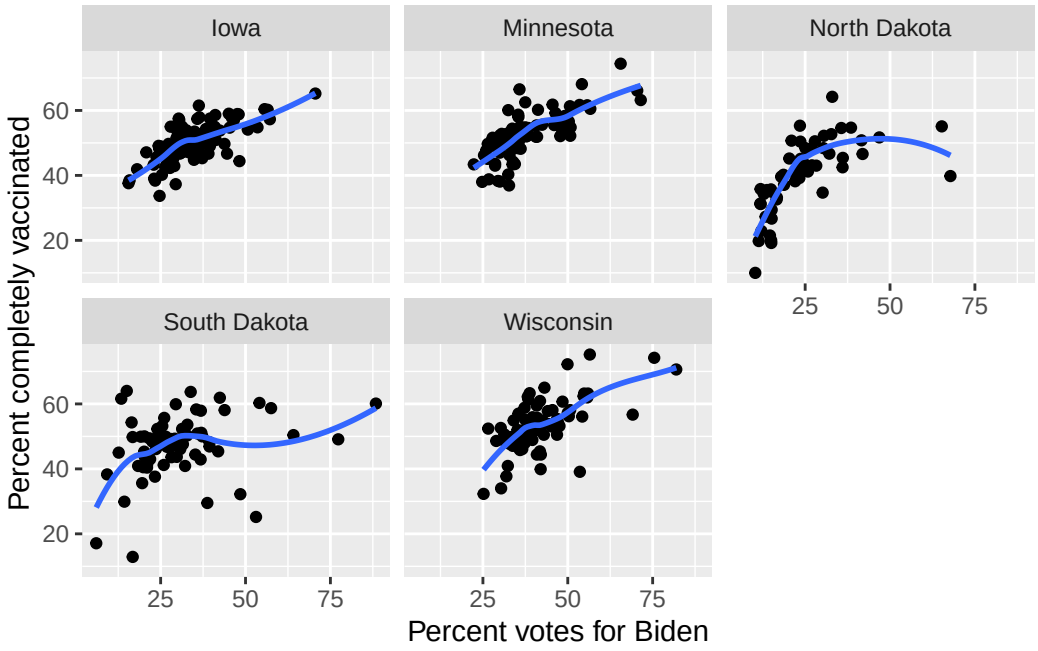


b) One plot per state containing a scatterplot and a smoother.

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)) +
  facet_wrap(~state)+
  geom_point()+
  geom_smooth(se = FALSE)+
  labs(x = "Percent votes for Biden",
       y = "Percent completely vaccinated",
       color = "State")
```

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



I prefer the second plot because I can see each state more clearly so it's easier to compare them and understand their individual patterns. Iowa, Minnesota, and Wisconsin have similar trends (all gradually trending positive), but ND and SD both have different trends. SD has a bit of a dip in percent of county completely vaccinated between 50% and 75% votes for Biden, and ND has a negative trend in percent of county completely vaccinated after 50% votes for Biden. ND also only has two points past 50% percent votes for Biden, which is likely influencing the shape of the trend.

2. We wish to compare the proportions of counties in each region with median household income above the national median ($69,560).

a) Fill in the blanks below to produce a segmented bar plot with regions ordered from highest proportion above the median to lowest.

b) Create a table of proportions by region to illustrate that your bar plot in (a) is in the correct order (you should find two regions that are *really* close when you just try to eyeball differences).

c) Explain why we can replace `fct_relevel(region, FILL IN CODE)` with

```
mutate(region_sort = fct_reorder(region, median_HHincome_2019 < 69560, .fun =
mean))
```

but not

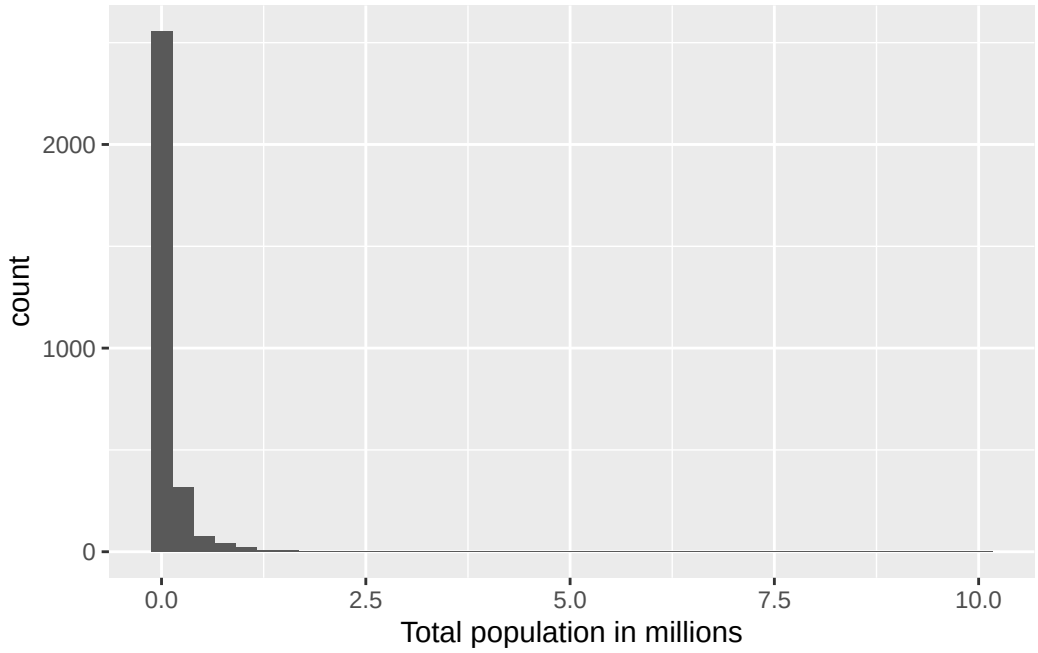
```
mutate(region_sort = fct_reorder(region, median_HHincome_2019 < 69560))
```

```
vaccine_data |>
#  mutate(HHincome_vs_national = ifelse(median_HHincome_2019 < 69560, FILL IN CODE)) |>
#  mutate(region_sort = fct_relevel(region, FILL IN CODE)) |>
  ggplot(mapping = aes(x = region_sort, fill = HHincome_vs_national)) +
    geom_bar(position = "fill")
```

3. We want to examine the distribution of total county populations and then see how it's related to vaccination rates.

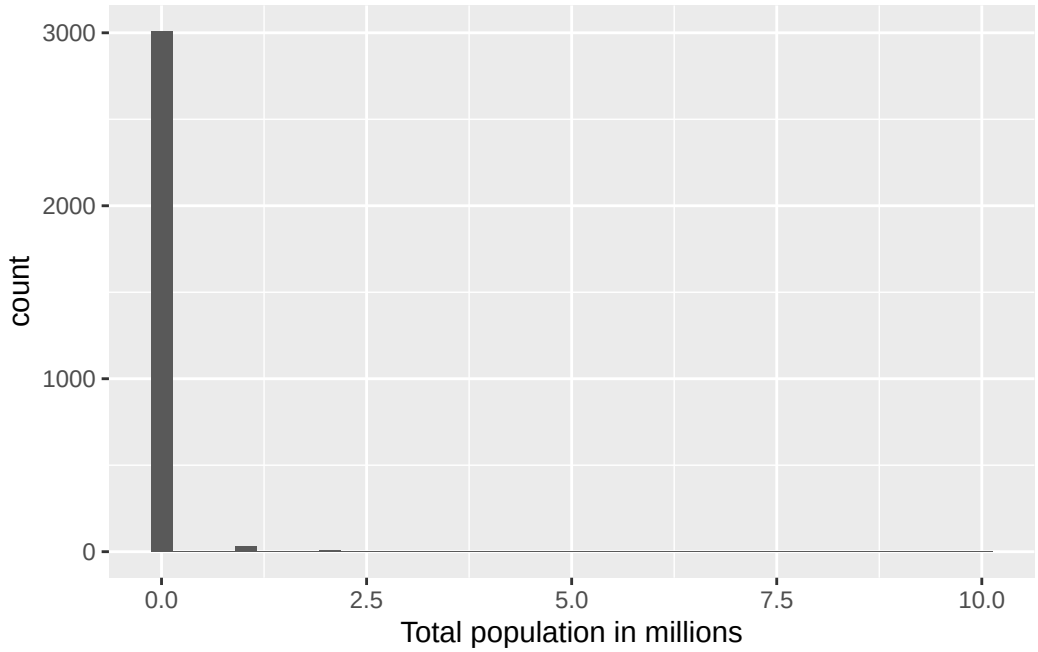
a) *Carefully and thoroughly* explain why the two histograms below provide different plots.

```
vaccine_data |>
  mutate(tot_pop_millions = tot_pop / 1000000) |>
  ggplot(mapping = aes(x = tot_pop_millions)) +
    geom_histogram(bins = 40) +
    labs(x = "Total population in millions")
```




```
vaccine_data |>
  mutate(tot_pop_millions = tot_pop %/% 1000000) |>
```

```
ggplot(mapping = aes(x = tot_pop_millions)) +
  geom_histogram(bins = 40) +
  labs(x = "Total population in millions")
```



b) Find the top 5 counties in terms of total population.

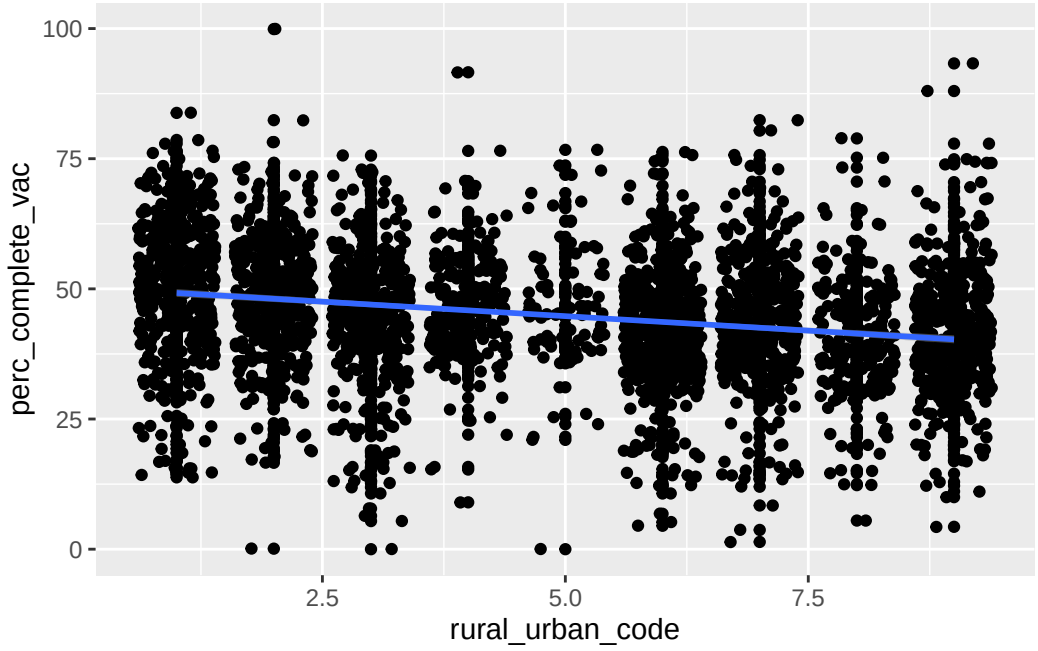
c) Plot a histogram of logged population and describe this distribution.

d) Plot the relationship between log population and percent vaccinated using separate colors for Metro and Non-metro counties (be sure there's no 3rd color used for NAs). Reduce the size and transparency of each point to make the plot more readable. Describe what you can learn from this plot.

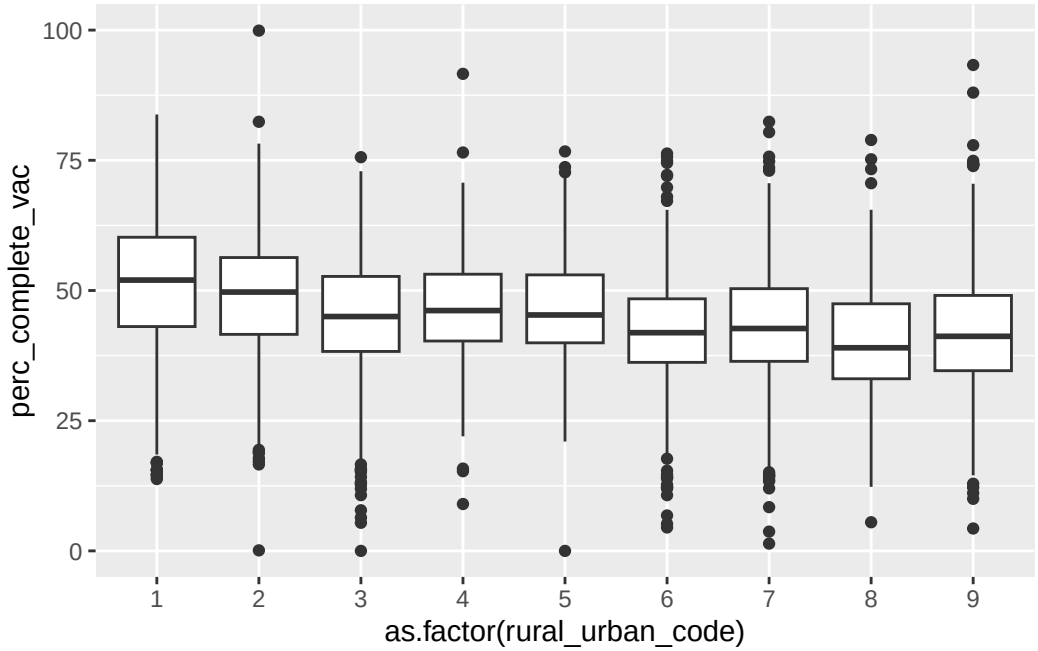
4. Produce 3 different plots for illustrating the relationship between the rural_urban_code and percent vaccinated. Hint: you can sometimes turn numeric variables into categorical variables for plotting purposes (e.g. `as.factor()`, `ifelse()`).

```
vaccine_data |>
  ggplot(aes(x = rural_urban_code, y = perc_complete_vac)) +
  geom_point()+
  geom_jitter() +
  geom_smooth(method = "lm")
```

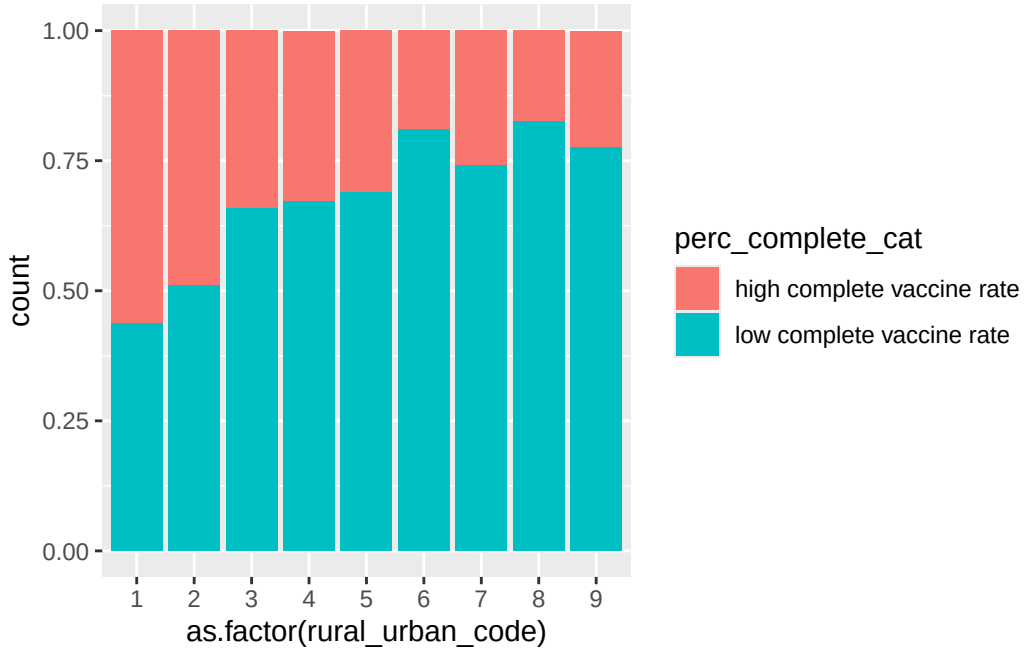
```
`geom_smooth()` using formula = 'y ~ x'
```




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




```
vaccine_data |>
  mutate(perc_complete_cat = ifelse(perc_complete_vac > 50, "high complete vaccine rate",
 ggplot(aes(x = as.factor(rural_urban_code), fill= perc_complete_cat))+
  geom_bar(position = "fill")
```



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.

I like the boxplot the best because it provides the most information, including the spread and the pattern of medians of each rural-urban code.

This is a boxplot showing the relationship between a county's rural/urban code and the percent of the county completely vaccinated. On the x-axis is the rural-urban code, which ranges from 1 to 9, with 1 as most urban and 9 as most rural. On the y-axis is the percent of the county completely vaccinated. This graph demonstrates that the spread of percent completely vaccinated is fairly similar across county rural-urban code, however, there is a pattern of slight decrease in percent completely vaccinated the more rural the county.

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))
```

The plot shows us the variability of percent of votes for Biden from county to county within the top 10 states with the most counties. By looking at this graph, we can see that states like Tennessee and Missouri have much smaller ranges in the count-to-county variability in percent votes for Biden, while states such as Virginia and those included in "Other" have greater variability.

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

```
vaccine_data_1 <- vaccine_data |>
  filter(region == "Midwest") |>
  group_by(state) |>
  summarize(n_rural_urban = n_distinct(rural_urban_code),
            population = sum(tot_pop),
            prop_metro = mean(metro_status == "Metro"),
            median_unemp = median(unemployment_rate_2020)) |>
arrange(population)


vaccine_data_1
```

```
# A tibble: 12 x 5
   state        n_rural_urban population prop_metro median_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
```

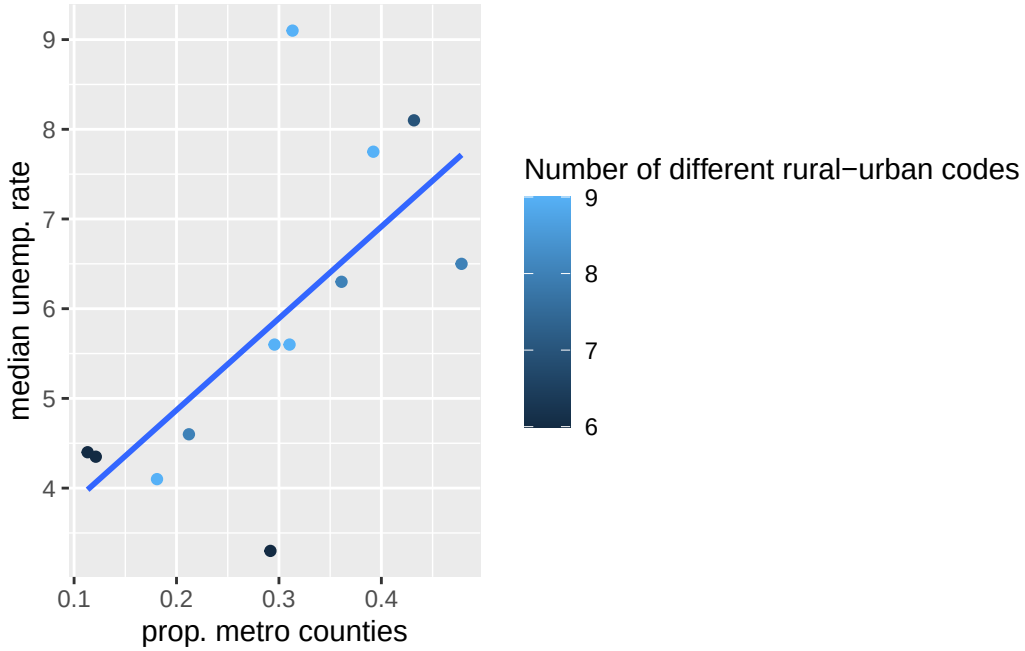
```
 7 Wisconsin                8    5822434      0.361         6.3
 8 Missouri                 9    6137428      0.296         5.6
 9 Indiana                  8    6732219      0.478         6.5
10 Michigan                 9    9986857      0.313         9.1
11 Ohio                     7   11689100      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?

```
vaccine_data_1 |>
  ggplot(aes(x = prop_metro, y = median_unemp, color = n_rural_urban)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)+
  labs(x = "prop. metro counties",
       y = "median unemp. rate",
       color = "Number of different rural-urban codes")
```

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

```
Warning: The following aesthetics were dropped during statistical transformation:
colour.
i This can happen when ggplot fails to infer the correct grouping structure in
  the data.
i Did you forget to specify a `group` aesthetic or to convert a numerical
  variable into a factor?
```



From this plot, we can conclude that the greater the proportion of metro counties in a state, the higher the median unemployment rate. There does not seem to be a distinguishable pattern regarding the role of number of different rural-urban codes in the relationship between proportion of metro counties and median unemployment rate.

7. Generate an appropriate plot to compare vaccination rates between two subregions of the US: New England (which contains the states Maine, Vermont, New Hampshire, Massachusetts, Connecticut, Rhode Island) and the Upper Midwest (which, according to the USGS, contains the states Minnesota, Wisconsin, Michigan, Illinois, Indiana, and Iowa). What can you conclude from your plot?

In this next section, we consider a few variables that could have been included in our data set, but were NOT. Thus, you won't be able to write and test code, but you nevertheless should be able to use your knowledge of the tidyverse to answer these questions.

Here are the hypothetical variables:

- HR_party = party of that county's US Representative (Republican, Democrat, Independent, Green, or Libertarian)
- people_per_MD = number of residents per doctor (higher values = fewer doctors)
- perc_over_65 = percent of residents over 65 years old
- perc_white = percent of residents who identify as white

8. Hypothetical R chunk #1:

```r
# 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?

- The first line if this code is creating a new tibble,"temp", which will be derived from the "vaccine_data" dataset. The mutate function is used to create two new variables "new_perc_vac" and "MD_group". "New_perc_vac" is created in order to label any counties with reported complete vaccination rates greater than 95% as "na" so that we can remove them when we create the variable "mean_perc_vac" in the summarise function. "MD_group" is created in order to form three different groups with approximately equal numbers of observations based on the number of residents per doctor in a given county. Because this data is grouped by "MD_group" whch has been specified to have 3 different categories, there will be 3 rows based on counties with high, medium, or low resident to doctor ratios. The tibble will have 4 columns: "MD_group", "n" indicating the number of counties in each category of resident-doctor ratio, "mean_perc_vac", the mean of the percent of residents in the county fully vaccinated (using the "new_perc_vac" variable), and "mean white" which is the mean percent of residents who identify as white.

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)`?

- If we made this replacement, then any counties with percent complete vaccination rates less than 95% would be labeled as "na" within the variable "new_perc_vac."

c) What would happen if we replaced `mean_white = mean(perc_white, na.rm = TRUE)` with `mean_white = mean(perc_white)`?

- Then the mean would simply be "na" because we can't take the mean of something when it contains any na values.

d) What would happen if we removed `group_by(MD_group)`?

- Then this information would be for each county rather than by the three different MD_groups.

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?

- Since the aes function is in the geom_point function and not in the ggplot function, the aesthetics mapping information is not extending to the geom_smooth function. geom_smooth requires this information about the aesthetics of the plot in order to create the line/smoother.

b) Describe the tibble `temp` created above. What would be the dimensions? What do rows and columns represent?

- This tibble is derived from the "vaccine_data" dataset. The information is grouped by HR_party which indicates the party of that county's US Representative. HR_party is a categorical variable with 5 levels. The summarize function is creating a variable "var1" which indicates the number of counties in each level of HR_party. The arrange function with the desc function orders the data from the HR_party level with the greatest number of counties to the level with the least. Slice_head selects the first rows, meaning the top three HR_party levels with the most counties. This tibble will have 5 rows for each of the 5 different parties included in the HR_party variable, and it will have 2 columns: HR_party and var1.

c) What would happen if we replaced `fct_reorder(HR_party, perc_over_65, .fun = median)` with `HR_party`?

- In the current plot, the graph is organized such that the order of how HR_party appears is determined by the median percent over 65 from smallest median percent to greatest

median percent. If we simply had HR_party as the x, then the boxplots would just be in alphabetical order.

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 code for this tibble begins by filtering out the "na" vaules in the variable "people_per_MD". Then the variable "state_lump" is created to lump together all states except for the 4 with the most counties. Group_by groups the data first by unique combinations of the 5 state_lump values and the and 9 rural_urban code values. Summarize is creates a variable "mean_people_per_MD" which is the mean of the ratio of residents to doctors in a county. This tibble could have up to 45 rows (4 of the most common states plus "other" representing all states * the 9 possible rural-urban codes). It probably has fewer rows because not all of the states have all of the rural-urban codes represented in their counties. There will be 3 columns: "state_lump", "rural_urban_code", and "mean_people_per_MD". The rows represent a unique combination of state and rural-urban code. "mean_people_per_MD" will specify the mean number of residents per doctor for each state and rural-urban code combination represented.

b) Carefully describe the plot created above.

- This graph is a color-coded line plot. On the x-axis is the rural-urban codes, ranging from 1 to 9, with 1 as the most urban and 9 as the most rural. On the y-axis is the mean ratio of people per doctor. Each line will be colored to represent a unique combination of state (only including the top 4 states with the most counties plus a 5th representing all other states) and urban-rural code. The legend for the data will be organized to match the ending location of the lines based on their rural-urban code and mean_people_per_MD.

c) What would happen if we removed `filter(!is.na(people_per_MD))`?

- If we removed this, then the na values would not be filtered out. If "people_per_MD" had any na values, this would result in our "mean_people_per_MD" returning na.

d) What would happen if we replaced `fct_reorder2(state_lump, rural_urban_code, mean_people_per_MD)` with `state_lump`?

- Then the legend would not match the ending location of the lines.