Lab 4

Math 241, Week 4

```
# Put all necessary libraries here
library(tidyverse)
```

Due: Friday, February 23rd at 8:30am

Goals of this lab

1. Practice creating and interpreting visualizations.
2. Practice wrangling data.

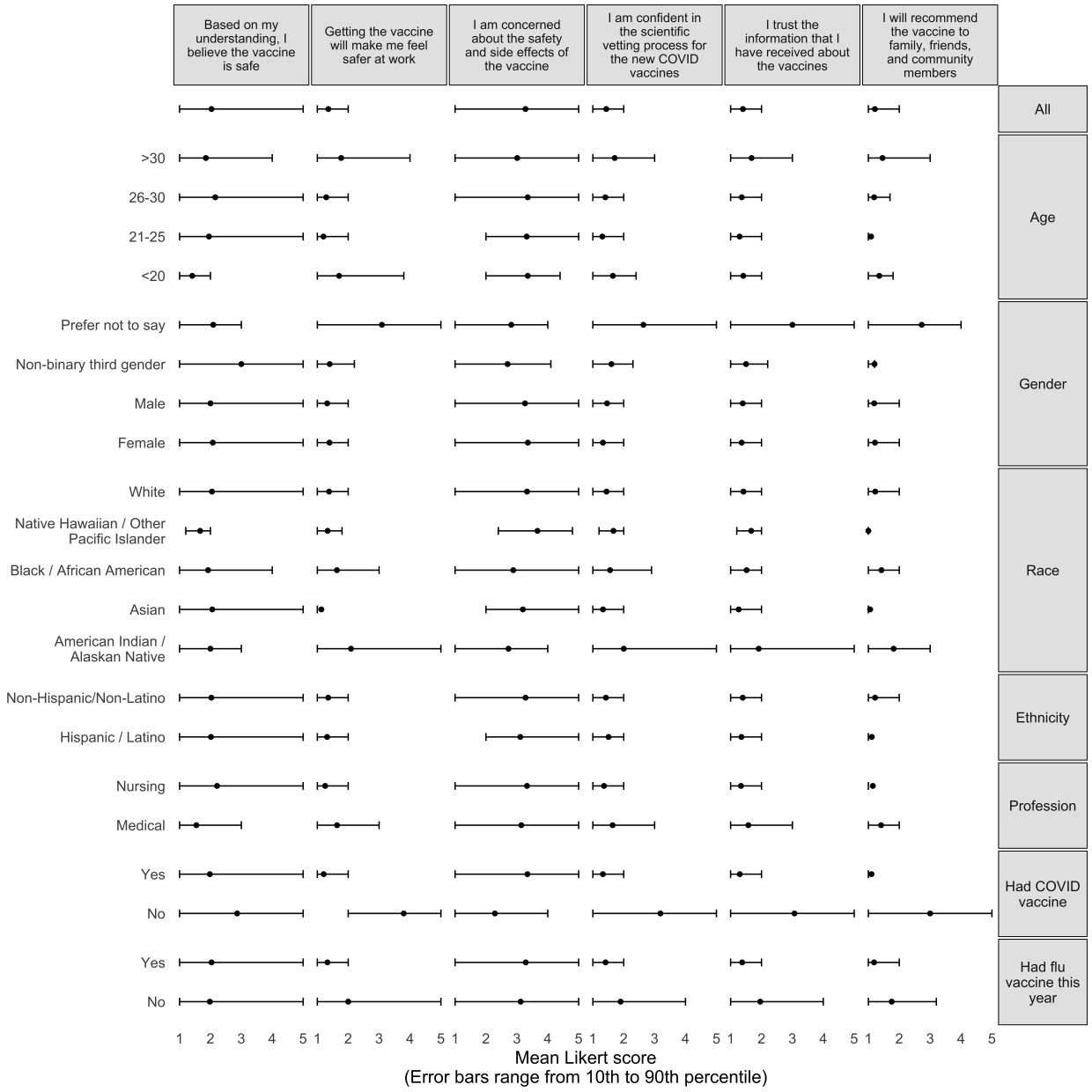
Problem 1: COVID survey - interpretation

A survey was conducted on the attitudes and opinions towards COVID-19 vaccination held by medical and nursing students across the US. The data were collected by Johns Hopkins School of Medicine. The following visualization was created based on the survey results.

Each facet in the visualization represents a response-explanatory variable combination. The response variables are in the columns and the explanatory variables are in the rows of the grid. For each response variable the survey takers indicated their level of agreement with the given statement. The responses were on a Likert scale: 1 - Strongly Agree, 2 - Somewhat Agree , 3 - Neither Agree Nor Disagree, 4 - Somewhat Disagree, 5 - Strongly Disagree.

Within each facet the data are represented with points (indicating the mean Likert score) and error bars spanning from 10th to 90th percentiles of the values of the response variable for each level of the explanatory variable. The first row is the only exception to this. This row of panes is noted as "All", meaning that the it's the distribution of each of the response variables without conditioning on any of the explanatory variables.

Your task for this question is to take a close look at this plot and interpret it. There is a lot going on here, which is customary for plots that go into scientific publications – they tend to be very information dense, for better or worse... As you interpret it, discuss if there are any results that agree or disagree with your intuition. There is a lot you can say, but we don't need you to be exhaustive. Please provide three concrete examples.



Problem 2: COVID survey - reconstruct

In this exercise you will reconstruct the plot provided in Problem 1. You can find the raw data in `data/survey.csv`. Additional information on the survey can be found in `data/covid_survey.pdf`.

- Load the data using `read_csv()`. View the result and decide if any rows on top need to be `skip`ped. If so, reload again with the `skip` argument. Print the dimensions of the resulting data frame with `dim()`.
- Next, you'll do a bit of data cleanup. Eliminate any rows where **all** values aside from `response_id` are missing; these do not contain any useful information. *Hint:* There are many ways you can do this. One succinct way is using a combination of `filter()` and `if_any()` or `if_all()`. But feel free to use whatever works for your intuition, just make sure you're only eliminating rows where **all** values (not

any values!) aside from `response_id` are missing. Print the dimensions of the resulting data frame with `dim()`.

- Relabel the survey response values according to the information in `data/covid_survey.pdf`. Peeking at the plot you're working towards reconstructing can also be helpful to identify how exactly to recode the variables. Print the dimensions of the resulting data frame with `dim()`.
  - `exp_already_vax` and `exp_flu_vax`: 0 - No, 1 - Yes
  - `exp_profession`: 0 - Medical, 1 - Nursing
  - `exp_gender`: 0 - Male, 1 - Female, 3 - Non-binary third gender, 4 - Prefer not to say
  - `exp_race`: 1 - American Indian / Alaskan Native, 2 - Asian, 3 - Black / African American, 4 - Native Hawaiian / Other Pacific Islander, 5 - White
  - `exp_ethnicity`: 1 - Hispanic / Latino, 2 - Non-Hispanic/Non-Latino
  - `exp_age_bin`: 0 - <20, 20 - 21-25, 25 - 26-30, 30 - >30
- In our data we have response variables (the ones that start with `resp_`: `resp_safety`, `resp_confidence_science`, `resp_concern_safety`, `resp_feel_safe_at_work`, `resp_will_recommend`, and `resp_trust_info`) and explanatory variables (the ones that start with `exp_`: `exp_profession`, `exp_flu_vax`, `exp_gender`, `exp_race`, `exp_ethnicity`, `exp_age_bin`, and `exp_already_vax`). For each response variable and each explanatory variable, we're interested in how the values of the response variable change across values of the explanatory variable, so we calculate the 10th percentile, mean, and 90th percentile of each of the response variables for each level of each explanatory variable. There are a variety of ways we can accomplish this. One of them is to pivot the data longer, twice (!), so that each row represents a combination of each response variable with explanatory variable and its level. We provide you with the code for this task, you need to run the code and confirm that you get the same result. Make sure to print the resulting tibble to make comparison easier as you (and we!) check your work. Also write a sentence or two explaining what each `pivot_longer()` statement is doing in this code.

```
covid_survey_longer <- covid_survey %>%
  pivot_longer(
    cols = starts_with("exp_"),
    names_to = "explanatory",
    values_to = "explanatory_value"
  ) %>%
  filter(!is.na(explanatory_value)) %>%
  pivot_longer(
    cols = starts_with("resp_"),
    names_to = "response",
    values_to = "response_value"
  )
```

```
covid_survey_longer
```

```
## # A tibble: 43,428 x 5
##    response_id explanatory    explanatory_value response          response_value
##          <int> <chr>          <chr>             <chr>                     <int>
##  1           1 exp_profession Nursing           resp_safety                   5
##  2           1 exp_profession Nursing           resp_confidence_~             2
##  3           1 exp_profession Nursing           resp_concern_saf~             2
##  4           1 exp_profession Nursing           resp_feel_safe_a~             1
##  5           1 exp_profession Nursing           resp_will_recomm~             1
```

```
##  6           1 exp_profession Nursing           resp_trust_info               1
##  7           1 exp_flu_vax    Yes               resp_safety                   5
##  8           1 exp_flu_vax    Yes               resp_confidence_~             2
##  9           1 exp_flu_vax    Yes               resp_concern_saf~             2
## 10           1 exp_flu_vax    Yes               resp_feel_safe_a~             1
## # i 43,418 more rows
```

- Group the data (`covid_survey_longer`) by `explanatory`, `explanatory_value`, and `response`, and then calculate the following summary statistics:
  - `mean`: mean of the `response_value`
  - `low`: 10th percentile of the `response_value`
  - `high`: 90th percentile of the `response_value`

Name this new data frame `covid_survey_summary_stats_by_group`. It should look like the following:

```
covid_survey_summary_stats_by_group
```

```
## # A tibble: 126 x 6
## # Groups:   explanatory, explanatory_value [21]
##    explanatory explanatory_value response                mean   low  high
##    <chr>       <chr>             <chr>                  <dbl> <dbl> <dbl>
##  1 exp_age_bin 21-25             resp_concern_safety     3.32     2     5
##  2 exp_age_bin 21-25             resp_confidence_science 1.31     1     2
##  3 exp_age_bin 21-25             resp_feel_safe_at_work  1.20     1     2
##  4 exp_age_bin 21-25             resp_safety             1.95     1     5
##  5 exp_age_bin 21-25             resp_trust_info         1.29     1     2
##  6 exp_age_bin 21-25             resp_will_recommend     1.09     1     1
##  7 exp_age_bin 26-30             resp_concern_safety     3.35     1     5
##  8 exp_age_bin 26-30             resp_confidence_science 1.40     1     2
##  9 exp_age_bin 26-30             resp_feel_safe_at_work  1.29     1     2
## 10 exp_age_bin 26-30             resp_safety             2.16     1     5
## # i 116 more rows
```

- Now group the data (`covid_survey_longer`) again, this time only by `response`, in order to calculate the same summary statistics for each response variable (mean, 10th percentile, and 90th percentile), not conditioned on the explanatory variables. Name this new data frame `covid_survey_summary_stats_all`. It should look like the following:

```
covid_survey_summary_stats_all
```

```
## # A tibble: 6 x 6
##   response                mean   low  high explanatory explanatory_value
##   <chr>                  <dbl> <dbl> <dbl> <chr>       <chr>
## 1 resp_concern_safety     3.28     1     5 All         ""
## 2 resp_confidence_science 1.43     1     2 All         ""
## 3 resp_feel_safe_at_work  1.36     1     2 All         ""
## 4 resp_safety             2.03     1     5 All         ""
## 5 resp_trust_info         1.40     1     2 All         ""
## 6 resp_will_recommend     1.21     1     2 All         ""
```

- Bind the two data frames of summary statistics you created `covid_survey_summary_stats_all` and `covid_survey_summary_stats_by_group` together by row. Name the resulting data frame `covid_survey_summary_stats`. It should look like the following:

```
covid_survey_summary_stats
```

```
## # A tibble: 132 x 6
##    response                mean   low  high explanatory explanatory_value
##    <chr>                  <dbl> <dbl> <dbl> <chr>       <chr>
##  1 resp_concern_safety     3.28     1     5 All         ""
##  2 resp_confidence_science  1.43     1     2 All         ""
##  3 resp_feel_safe_at_work   1.36     1     2 All         ""
##  4 resp_safety             2.03     1     5 All         ""
##  5 resp_trust_info          1.40     1     2 All         ""
##  6 resp_will_recommend      1.21     1     2 All         ""
##  7 resp_concern_safety      3.32     2     5 exp_age_bin "21-25"
##  8 resp_confidence_science  1.31     1     2 exp_age_bin "21-25"
##  9 resp_feel_safe_at_work   1.20     1     2 exp_age_bin "21-25"
## 10 resp_safety             1.95     1     5 exp_age_bin "21-25"
## # i 122 more rows
```

- Using the data frame you created in the previous step (`covid_survey_summary_stats`), recreate the visualization from Exercise 2. The following hints should help you along the way:
  - The survey prompts used for the response variables are as follows:
    - "resp_safety" = "Based on my understanding, I believe the vaccine is safe",
    - "resp_confidence_science" = "I am confident in the scientific vetting process for the new COVID vaccines",
    - "resp_feel_safe_at_work" = "Getting the vaccine will make me feel safer at work",
    - "resp_will_recommend" = "I will recommend the vaccine to family, friends, and community members",
    - "resp_trust_info" = "I trust the information that I have received about the vaccines",
    - "resp_concern_safety" = "I am concerned about the safety and side effects of the vaccine"
  - The variable names represented on the plot for the explanatory variables are as follows:
    - "exp_age_bin" = "Age",
    - "exp_gender" = "Gender",
    - "exp_race" = "Race",
    - "exp_ethnicity" = "Ethnicity",
    - "exp_profession" = "Profession",
    - "exp_already_vax" = "Had COVID vaccine",
    - "exp_flu_vax" = "Had flu vaccine this year"
  - The `facet_grid()` function, which you will need to use to reconstruct this plot, has a `labeller` argument, which you can use to define how you want to add line breaks to facet labels. For example, `labeller = labeller = labeller(explanatory = label_wrap_gen(15))` places a line break after 15 characters on the labels of the facets defined by the `explanatory` variable.
  - Don't worry about matching the exact color for the background of the facet labels. But, if you want to match, the color is `gray90`.