Brilliant Cassowary

Exploratory data analysis

Nidhi Soma (ns848) Joice Chen (jc3528) Jinpeng Li (jl3496)
Stephen Syl-Akinwale (sis33)

```r
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.4.4     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.0
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
library(janitor)
```

```

Attaching package: 'janitor'

The following objects are masked from 'package:stats':

    chisq.test, fisher.test
```

```r
library(dplyr)
library(skimr)
library(stringr)
library(ggplot2)
library(usethis)
```

Inserting the dataset on Coffee

```
coffee_df<-read_csv("data/GACTT_RESULTS_ANONYMIZED_v2.csv")
```

```
Rows: 4042 Columns: 113
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (44): Submission ID, What is your age?, How many cups of coffee do you t...
dbl (13): Lastly, how would you rate your own coffee expertise?, Coffee A - ...
lgl (56): Where do you typically drink coffee? (At home), Where do you typic...

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.
```

Research question(s)

Research question(s). State your research question (s) clearly.

Data collection and cleaning

Have an initial draft of your data cleaning appendix. Document every step that takes your raw data file(s) and turns it into the analysis-ready data set that you would submit with your final project. Include text narrative describing your data collection (downloading, scraping, surveys, etc) and any additional data curation/cleaning (merging data frames, filtering, transformations of variables, etc). Include code for data curation/cleaning, but not collection.

1. After inserting the data, it is in a table format. We need to evaluate which columns are useful or pretty much empty. This section removes columns with mostly NA values, since they will not be helpful for analysis.

```
#remove NA columns
coffee_clean <- coffee_df |>
  select(-contains("flavorings")) |>
  select(-contains("Gender (please specify"))

#new names
#coffee_clean <- coffee_df |>
#  rename_with(~str_extract(.x, '(?<=\\().*?(?=\\))'))
```

2. Next, we need to evaluate which columns contain repetitive information. There are a few columns that ask a question, and the next few columns are the answer to that question, so the question itself contains repetive values that we already have in the subsequent column names. This section removes the question columns.

```r
#remove repetitive questions
coffee_clean <- coffee_clean |>
  mutate(`Where do you typically drink coffee?` = NULL) |>
  mutate(`How do you brew coffee at home?` = NULL)|>
  mutate(`On the go, where do you typically purchase coffee?` = NULL) |>
  mutate(`Do you usually add anything to your coffee?` = NULL) |>
  mutate(`What kind of diary do you add?` = NULL) |>
  mutate(`What kind of sugar or sweetener do you add?` = NULL) |>
  mutate(`Why do you drink coffee?` = NULL)
```

3. The main part of our data cleaning is fixing the column names to be in a tidy format. We go through and rename columns in the original form of “question? (response)” to “question_response”. We also manually rename some confusing results from this method.

```r
original_names <- colnames(coffee_clean)
tidy_names <- gsub(" ", "_", original_names)
tidy_names <- tolower(tidy_names)
tidy_names <- gsub("[[:punct:]]&&[^_]", "", tidy_names)

colnames(coffee_clean) <- tidy_names

coffee_clean <- coffee_clean |>
  rename(
    age = "what_is_your_age?",
    cups_of_coffee_per_day = "how_many_cups_of_coffee_do_you_typically_drink_per_day?",
    how_else_at_home = "how_else_do_you_brew_coffee_at_home?",
    where_else_purchase_coffee = "where_else_do_you_purchase_coffee?",
    favorite_coffee_drink = "what_is_your_favorite_coffee_drink?",
    favorite_coffee = "please_specify_what_your_favorite_coffee_drink_is",
    prefer_between_abc = "between_coffee_a,_coffee_b,_and_coffee_c_which_did_you_prefer?",
    other_flavoring = "what_other_flavoring_do_you_use?",
    best_described_before = "before_today's_tasting,_which_of_the_following_best_described_wh
    like_coffee = "how_strong_do_you_like_your_coffee?",
    roast_level = "what_roast_level_of_coffee_do_you_prefer?",
    caffeine = "how_much_caffeine_do_you_like_in_your_coffee?",
    own_coffee_expertise = "lastly,_how_would_you_rate_your_own_coffee_expertise?",
    prefer_between_ad = "between_coffee_a_and_coffee_d,_which_did_you_prefer?",
    favorite_overall_coffee = "lastly,_what_was_your_favorite_overall_coffee?",
    time_spent_on_equipment = "approximately_how_much_have_you_spent_on_coffee_equipment_in_
    good_value_equipment = "do_you_feel_like_you’re_getting_good_value_for_your_money_with_re
  )

colnames(coffee_clean) <- sapply(colnames(coffee_clean), function(name) {
  if (grepl("where_do_you_typically_drink_coffee", name)) {
    name <- gsub("where_do_you_typically_drink_coffee\\?_\\((.*)\\)", "drink_\\1", name)
  } else if (grepl("how_do_you_brew_coffee_at_home", name)) {
    name <- gsub("how_do_you_brew_coffee_at_home\\?_\\((.*)\\)", "at_home_\\1", name)
  } else if (grepl("on_the_go,_where_do_you_typically_purchase_coffee", name)) {
    name <- gsub("on_the_go,_where_do_you_typically_purchase_coffee\\?_\\((.*)\\)", "purchase
  } else if (grepl("do_you_usually_add_anything_to_your_coffee", name)) {
    name <- gsub("do_you_usually_add_anything_to_your_coffee\\?_\\((.*)\\)", "add_to_\\1", na
  } else if (grepl("what_kind_of_dairy_do_you_add", name)) {
    name <- gsub("what_kind_of_dairy_do_you_add\\?_\\((.*)\\)", "dairy_add_\\1", name)
  } else if (grepl("what_kind_of_sugar_or_sweetener_do_you_add", name)) {
    name <- gsub("what_kind_of_sugar_or_sweetener_do_you_add\\?_\\((.*)\\)", "sugar_sweetener
  } else if (grepl("why_do_you_drink_coffee", name)) {
    name <- gsub("why_do_you_drink_coffee\\?_\\((.*)\\)", "reason_\\1", name)
  }
  name
}
)

#manually changing some more confusing names
coffee_clean_2 <- coffee_clean |>
  rename(at_home_coffee_brewing_machine = `at_home_coffee_brewing_machine_(e.g._mr._coffee)`
         at_home_pod_or_capsule_machine = `at_home_pod/capsule_machine_(e.g._keurig/nespresso
         at_home_coffee_extract = `at_home_coffee_extract_(e.g._cometeer)`,
         purchase_national_chain = `purchase_national_chain_(e.g._starbucks,_dunkin)`,
         add_to_none = `add_to_no_-_just_black`,
         add_to_milk = `add_to_milk,_dairy_alternative,_or_coffee_creamer`,
         sugar_sweetener_add_artificial_sweeteners = `sugar_sweetener_add_artificial_sweetene
         sugar_sweetener_add_raw_sugar= `sugar_sweetener_add_raw_sugar_(turbinado)`,
         where_work = `do_you_work_from_home_or_in_person?`,
         monthly_coffee_cost = `in_total,_much_money_do_you_typically_spend_on_coffee_in_a_mo
         like_taste = `do_you_like_the_taste_of_coffee?`,
         know_where_coffee_from = `do_you_know_where_your_coffee_comes_from?`,
         most_pay = `what_is_the_most_you've_ever_paid_for_a_cup_of_coffee?`,
         most_willing_pay = `what_is_the_most_you'd_ever_be_willing_to_pay_for_a_cup_of_coffe
         good_value_money = `do_you_feel_like_you’re_getting_good_value_for_your_money_when_y
  mutate(`what_kind_of_dairy_do_you_add?` = NULL)
```

4. After renaming our columns, we noticed some of them work nicely as categorical factors. This section goes through and modify them to be factors in a logical order.

```
#change type to categorical

coffee_clean_factors <- coffee_clean_2 |>
  mutate(age = factor(age),
         monthly_coffee_cost = factor(monthly_coffee_cost))|>
  mutate(across(like_taste:political_affiliation, factor)) |>
  mutate(across(like_coffee:caffeine, factor)) |>
  mutate(cups_of_coffee_per_day = as_factor(cups_of_coffee_per_day))|>
  mutate(best_described_before = factor(best_described_before))
```

```
#add category

coffee_clean_factors <- coffee_clean_factors |>
  mutate(age = fct_relevel(age, c("<18 years old",
                                  "18-24 years old",
                                  "25-34 years old",
                                  "35-44 years old",
                                  "45-54 years old",
                                  "55-64 years old",
                                  ">65 years old")))|>
  mutate(monthly_coffee_cost = fct_relevel(monthly_coffee_cost, c(
    "<$20",
    "$20-$40",
    "$40-$60",
    "$60-$80",
    "$80-$100",
    ">$100"))) |>
  mutate(most_pay = fct_relevel(
    most_pay,
    c("Less than $2",
    "$2-$4",
    "$4-$6",
    "$6-$8",
    "$8-$10",
    "$10-$15",
    "$15-$20",
    "More than $20"
```

```
   ))) |>
  mutate(most_willing_pay = fct_relevel(
    most_willing_pay,
    c("Less than $2",
    "$2-$4",
    "$4-$6",
    "$6-$8",
    "$8-$10",
    "$10-$15",
    "$15-$20",
    "More than $20"
  ))) |>
  mutate(cups_of_coffee_per_day = fct_relevel(cups_of_coffee_per_day,
                                              c("Less than 1",
                                              "1",
                                              "2",
                                              "3",
                                              "4",
                                              "More than 4"))) |>
  mutate(caffeine = fct_relevel(caffeine,
                                c("Decaf", "Half caff", "Full caffeine"))) |>
  mutate(like_coffee = fct_relevel(like_coffee,
                                   c("Weak",
                                     "Somewhat light",
                                     "Medium",
                                     "Somewhat strong",
                                     "Very strong")))



#glimpse(coffee_clean_factors)
  # mutate(`what_is_the_most_you've_ever_paid_for_a_cup_of_coffee?` = fct_relevel(
  #   "Less than $2",
  #   "$2-$4",
  #   "$4-$6",
  #   "$6-$8",
  #   "$8-$10",
  #   "$10-$15",
  #   "$15-$20",
  #   "More than $20"
  # )) |>
  #   mutate(`what_is_the_most_you'd_ever_be_willing_to_pay_for_a_cup_of_coffee?`) = fct_rel
  #   "Less than $2",
  #   "$2-$4",
  #   "$4-$6",
  #   "$6-$8",
  #   "$8-$10",
  #   "$10-$15",
  #   "$15-$20",
  #   "More than $20"
  # )
```

Data description

Have an initial draft of your data description section. Your data description should be about your analysis-ready data.

What are the observations (rows) and the attributes (columns)? The observations represent an individual respondent to the survey. The columns are questions that they answered, ranging from demographic data to coffee preferences.

Why was this dataset created? To understand the general public's preferences as consumers for coffee.

Who funded the creation of the dataset? James Hoffman and Cometeer.

What processes might have influenced what data was observed and recorded and what was not? The survey quickly was sold out, and Hoffman's audience in general is coffee specialists. That will likely skew the population surveyed to be people who likely prefer specialty coffee, so it may be a biased sample. Additionally, this survey was conducted through people ordering tasting kits online, which were then sent to the participants to prepare and complete voluntarily, so there may have been differences in that. One example is that because participants were following a livestream to demonstrate how to do their taste test, their coffees may have been out long enough to have cooled, which could be another unaccounted variable that affected their taste preferences.

What preprocessing was done, and how did the data come to be in the form that you are using? Zip codes and geographic data seemed to have been removed. Participants were anonymized to protect their privacy. It wasn't disclosed how Hoffmann and his team collected all the taste test results that participants filled out, but once they got that data, they made it into a spreadsheet to be shared with the public.

If people are involved, were they aware of the data collection and if so, what purpose did they expect the data to be used for? People were involved, and they were made aware of the data collection by a YouTube video stating that the purpose of this taste test was to understand coffee preferences in the USA. The participants had to order the coffee tasting kit on their own in order to participate, showing their willingness to accept these terms. Hoffmann also made his intentions clear in his video with why he wanted to collect the data, and that he was planning to publicize the raw data later on.

Data limitations

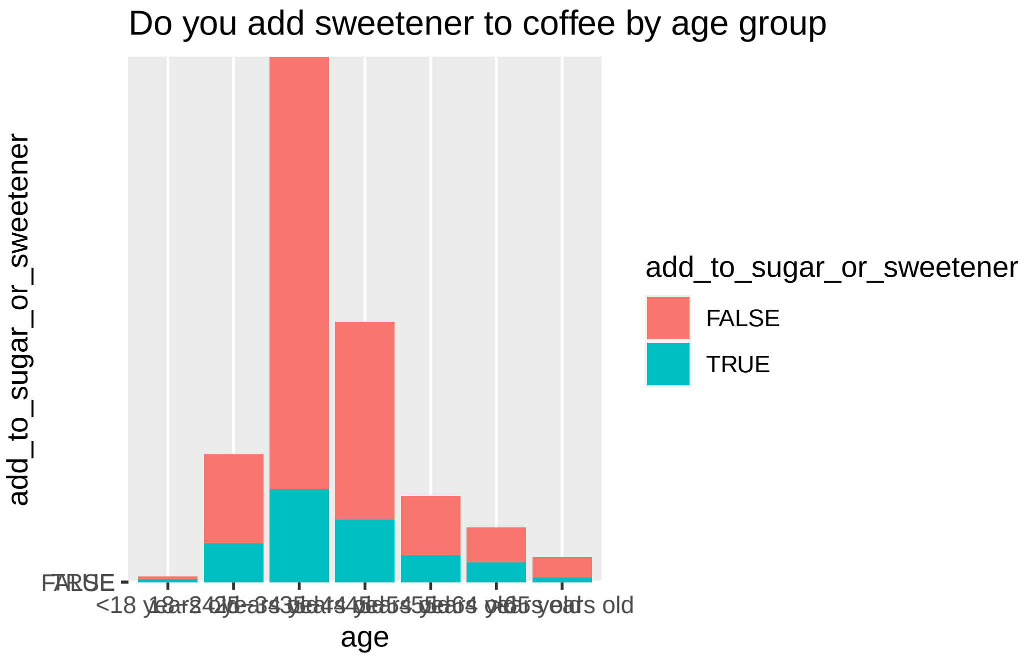
Identify any potential problems with your dataset.

- There are many NA values across the dataset.
- Demographics are only limited to people in the US, and most respondents are likely coffee specialists, which introduces a bias in the data, so it cannot be generalized to the US public's coffee preferences.
- In the other columns, there is no clear order in the answers, so it is hard to sort through and find extra patterns.

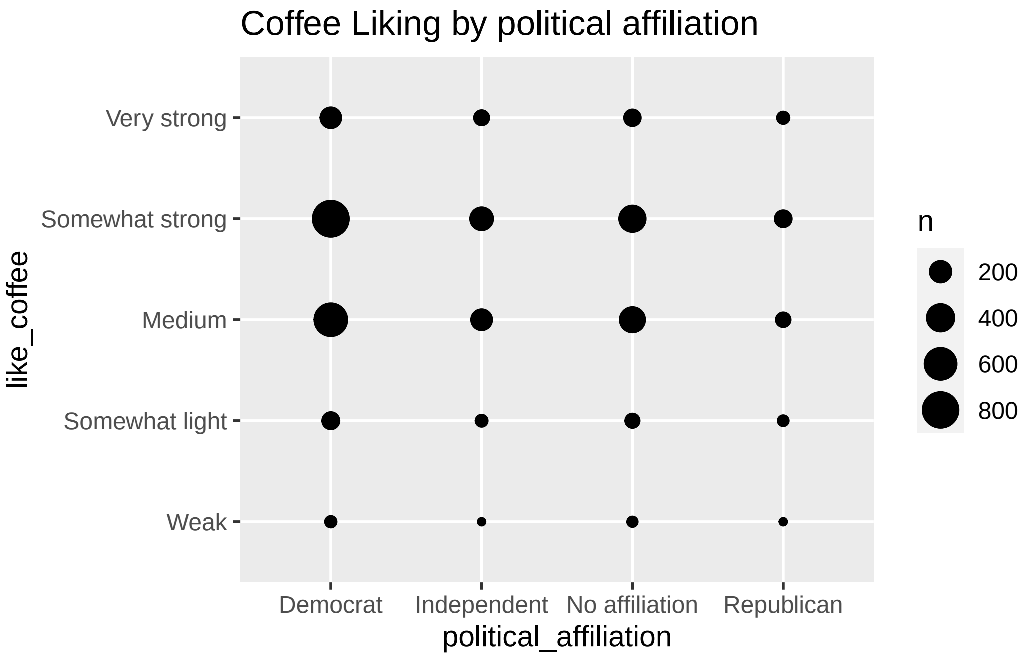
Exploratory data analysis

Perform an (initial) exploratory data analysis.

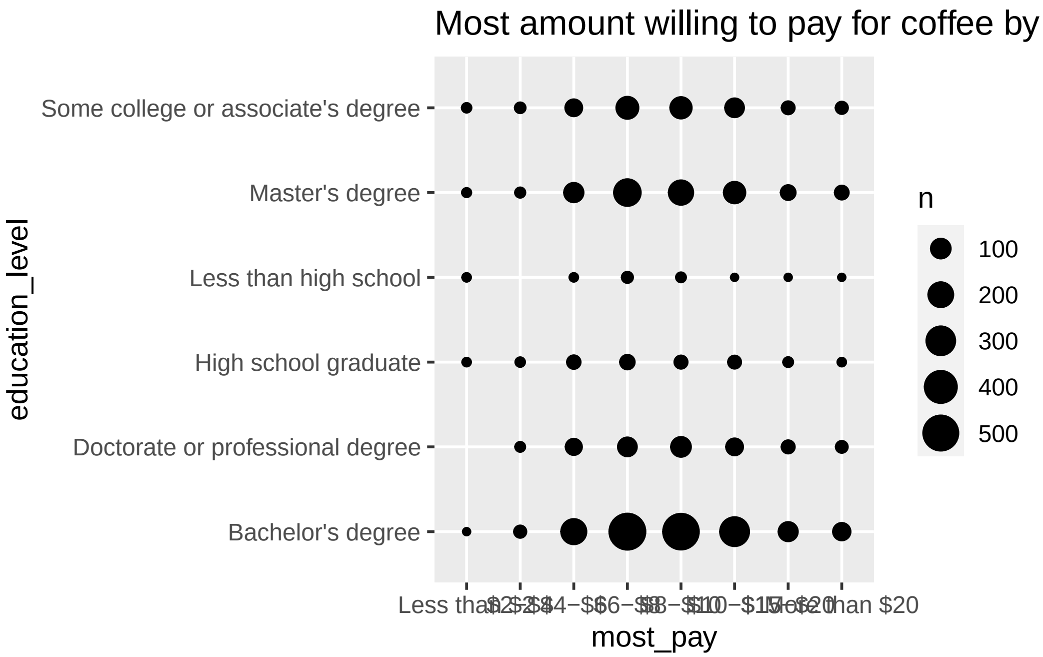
```r
#Joice graphs
#sweetener by age
coffee_clean_factors |>
  select(age, add_to_sugar_or_sweetener)|>
  drop_na() |>
  ggplot(aes(x = age,
             y = add_to_sugar_or_sweetener,
             fill = add_to_sugar_or_sweetener)) +
  geom_col() +
  labs(title = "Do you add sweetener to coffee by age group")
```



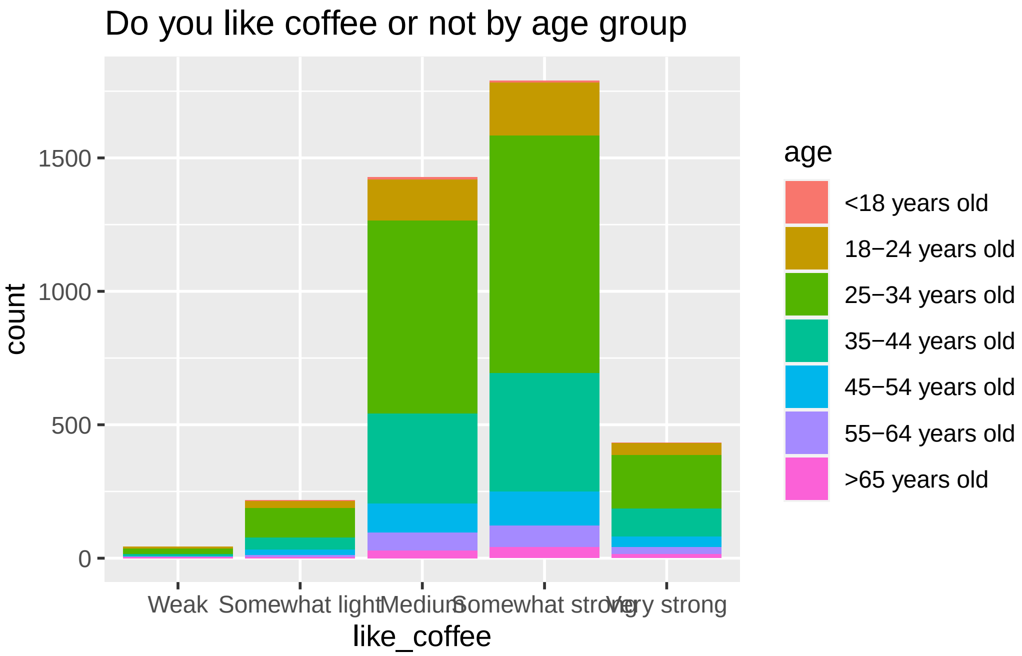
```
#coffee payment political affiliation
coffee_clean_factors |>
  select(political_affiliation, like_coffee) |>
  drop_na() |>
  ggplot(aes(x = political_affiliation, y = like_coffee)) +
  geom_count() +
  labs(title = "Coffee Liking by political affiliation")
```



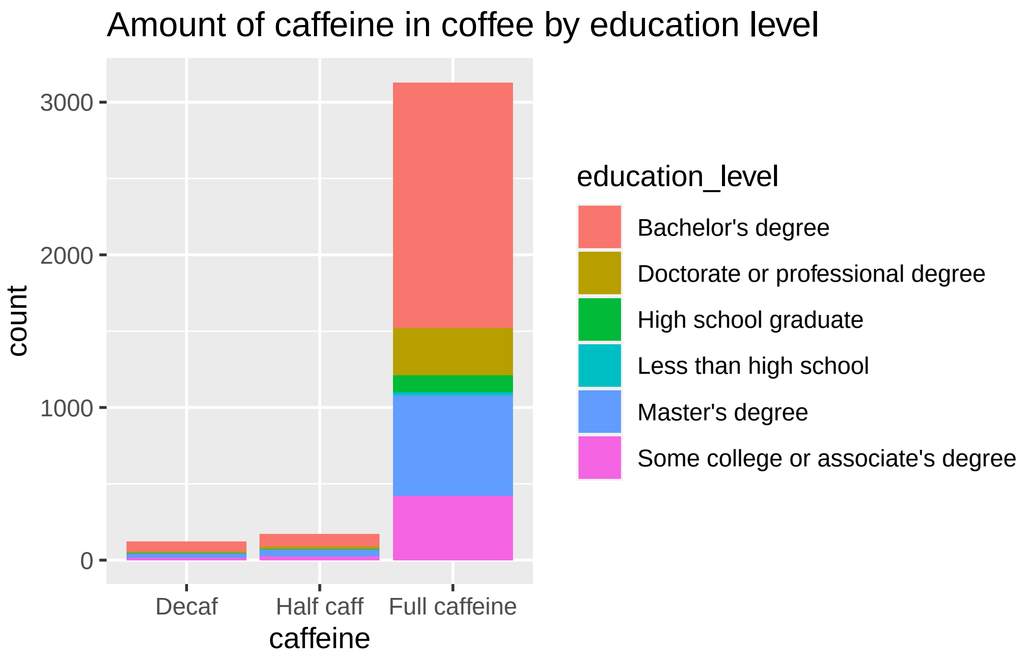
```
#coffee expected and actual pay
coffee_clean_factors |>
  select(most_pay, education_level) |>
  drop_na() |>
  ggplot(aes(x = most_pay, y = education_level)) +
  geom_count() +
  labs(title = "Most amount willing to pay for coffee by education level")
```



```
#coffee like or not by age
coffee_clean_factors |>
  select(like_coffee, age) |>
  drop_na() |>
  ggplot(aes(x = like_coffee, fill = age)) +
  geom_bar() +
  labs(title = "Do you like coffee or not by age group")
```

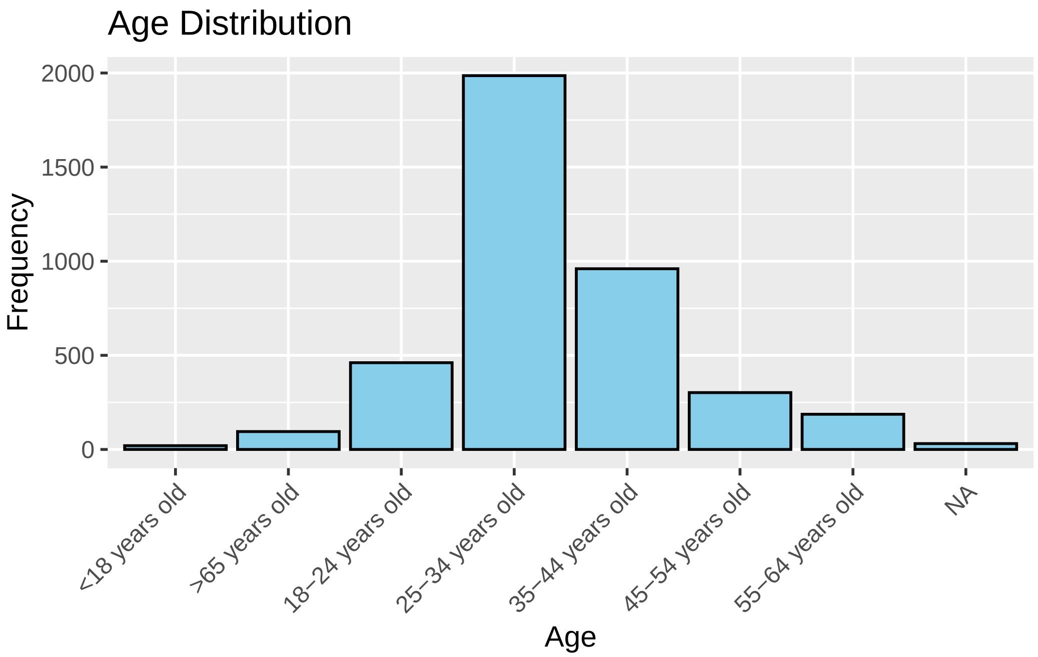


```r
#amount of caffeine by education level
coffee_clean_factors |>
  select(caffeine, education_level) |>
  drop_na() |>
  ggplot(aes(x = caffeine, fill = education_level)) +
  geom_bar() +
  labs(title = "Amount of caffeine in coffee by education level")
```

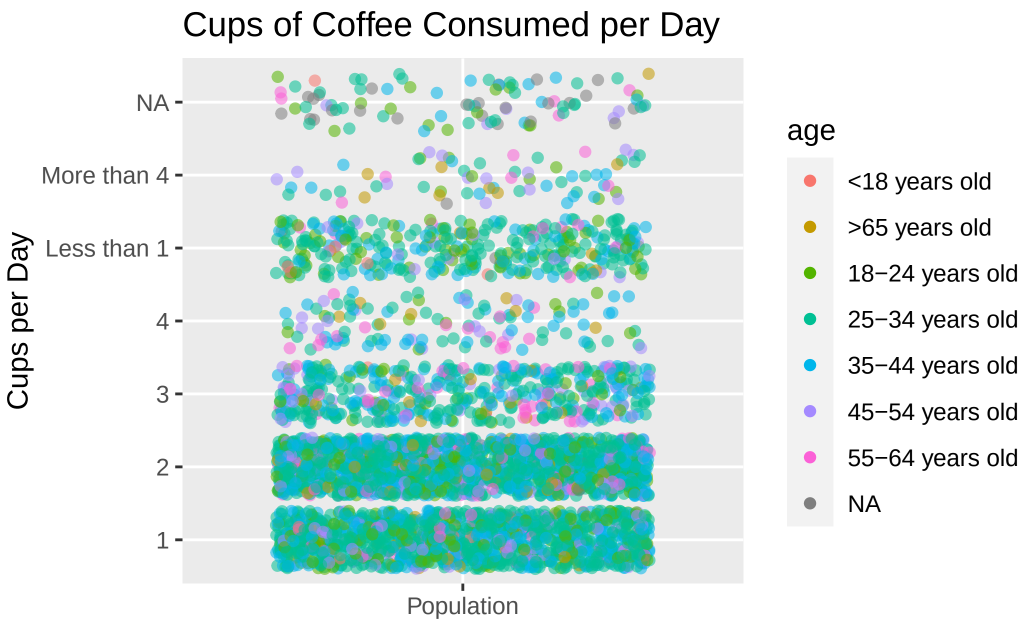


```r
#Stephen Graphs

# Histogram of Age Distribution
ggplot(coffee_clean, aes(x = age)) +
  geom_histogram(stat = "count",binwidth = 5, fill = "skyblue", color = "black") +
  labs(title = "Age Distribution",
       x = "Age",
       y = "Frequency")+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

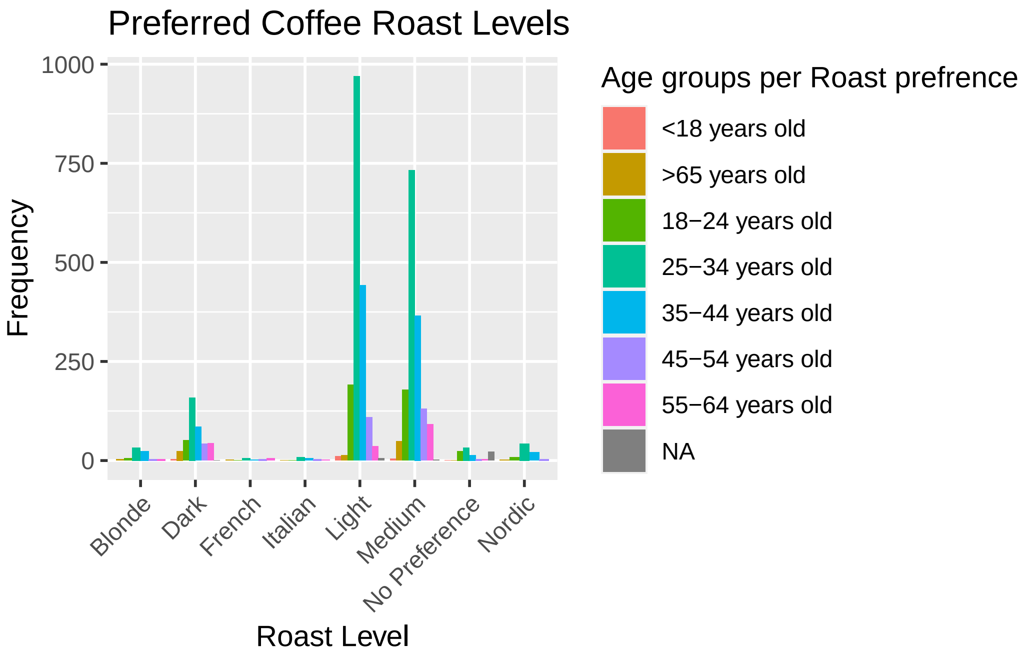
```
Warning in geom_histogram(stat = "count", binwidth = 5, fill = "skyblue", :
Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```



```
# Plot of Cups of Coffee Consumed per Day
ggplot(coffee_clean, aes(x = "Population", y = cups_of_coffee_per_day, color= age, alpha= 0.5
  geom_jitter() +
  labs(title = "Cups of Coffee Consumed per Day",
       x = "",
       y = "Cups per Day")+
  guides(alpha = "none")
```



```r
#Preferred roast levels
# preprocess the data
coffee_clean_roast <- coffee_clean %>%
  mutate(roast_level = ifelse(is.na(roast_level), "No Preference", roast_level))

ggplot(coffee_clean_roast, aes(x = roast_level, fill= age)) +
  geom_bar(position = "dodge") +
  labs(title = "Preferred Coffee Roast Levels",
       x = "Roast Level",
       y = "Frequency",
       fill= "Age groups per Roast prefrence") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

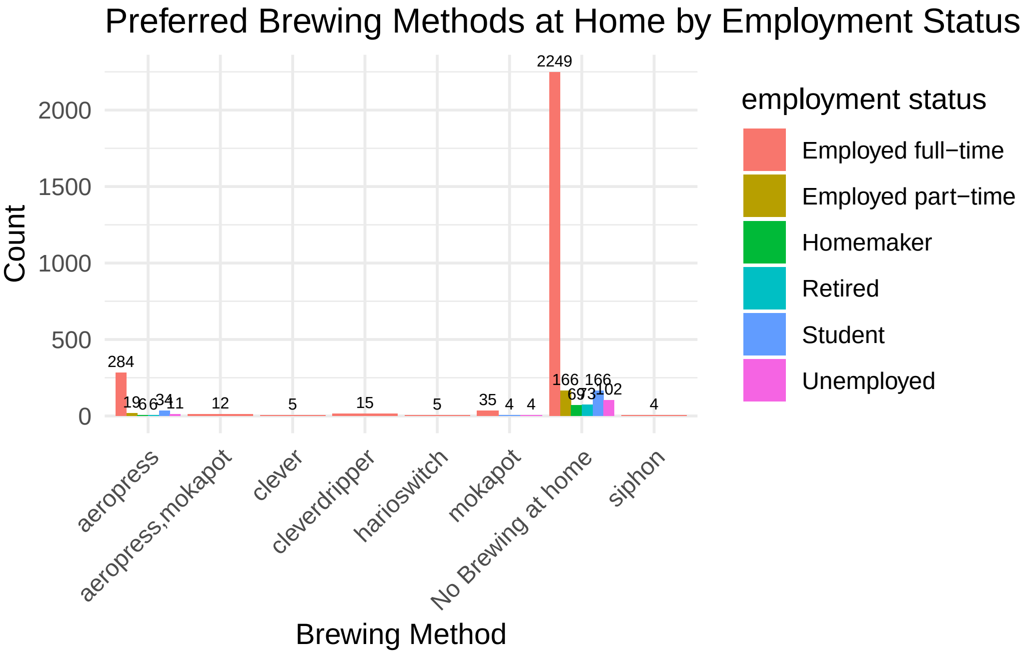


```r
# Plot for top 10 brewing methods
# processing brewing methods
brewing_method_counts <- coffee_clean %>%
  filter(!is.na(employment_status)) %>%
  mutate(how_else_at_home = str_replace_all(str_trim(tolower(how_else_at_home)), "\\s+", ""))
  mutate(how_else_at_home = ifelse(is.na(how_else_at_home), "No Brewing at home", how_else_a
  count(how_else_at_home, employment_status) %>%
  ungroup() %>%
  arrange(how_else_at_home, desc(n))

t10_brews <- top_n(brewing_method_counts,20)
```

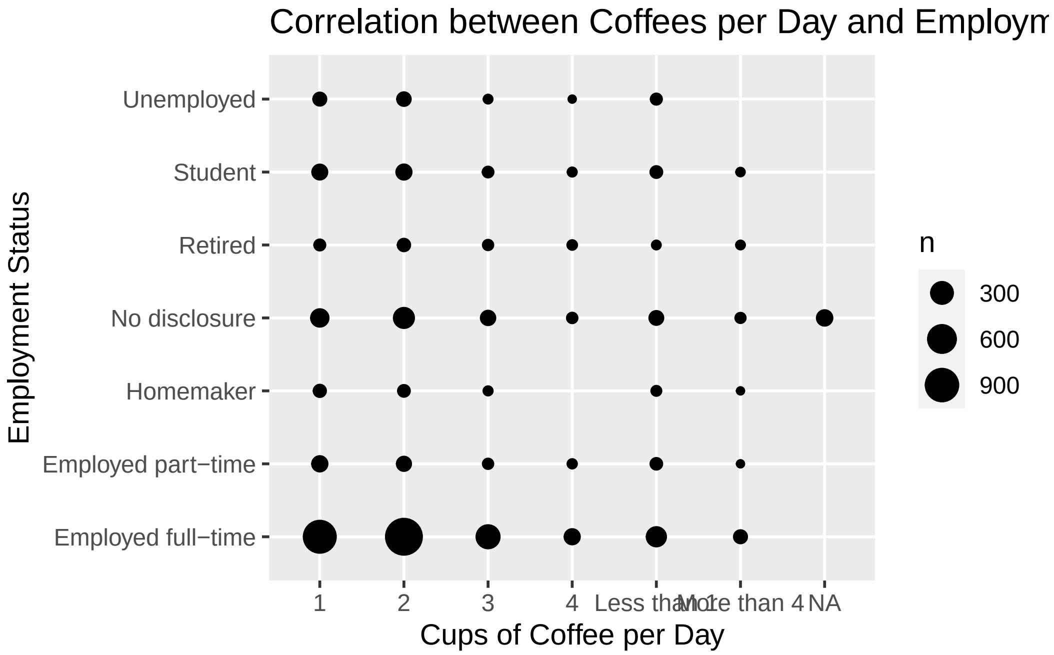
```
Selecting by n
```

```r
ggplot(t10_brews, aes(x = t10_brews$how_else_at_home,
                      y= n,
                          fill = t10_brews$employment_status)) +
  geom_col(position = "dodge") +
  labs(title = "Preferred Brewing Methods at Home by Employment Status",
       x = "Brewing Method",
       y = "Count",
       fill= "employment status") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        legend.position = "right")+
  geom_text(aes(label = n, y = n), position = position_dodge(width = 0.9), vjust = -0.5, colo
```

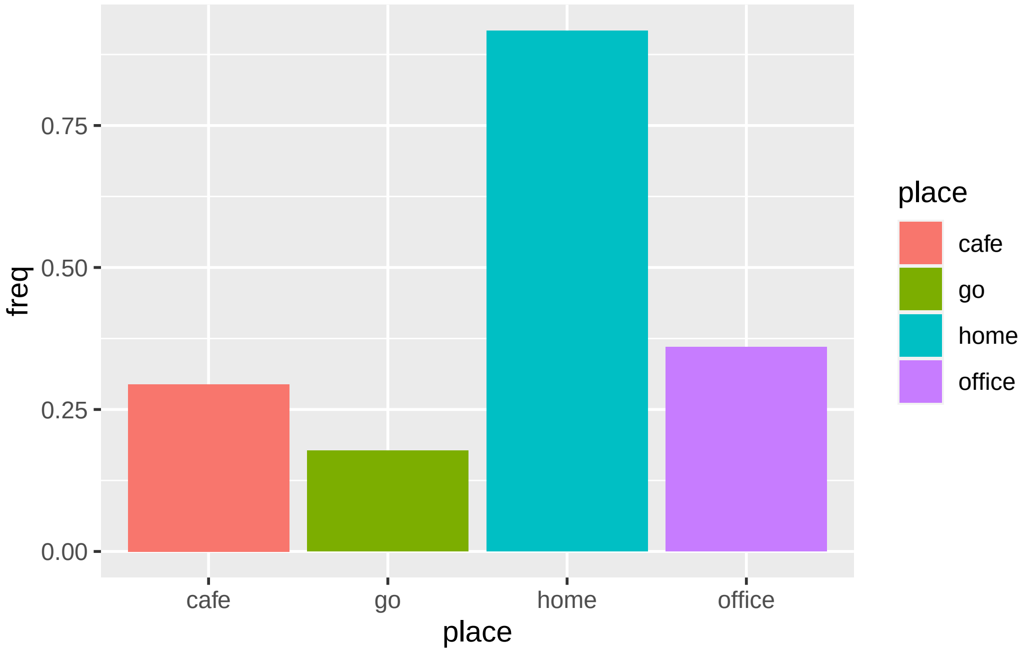
```
Warning: Use of `t10_brews$how_else_at_home` is discouraged.
i Use `how_else_at_home` instead.

Warning: Use of `t10_brews$employment_status` is discouraged.
i Use `employment_status` instead.

Warning: Use of `t10_brews$how_else_at_home` is discouraged.
i Use `how_else_at_home` instead.

Warning: Use of `t10_brews$employment_status` is discouraged.
i Use `employment_status` instead.
```



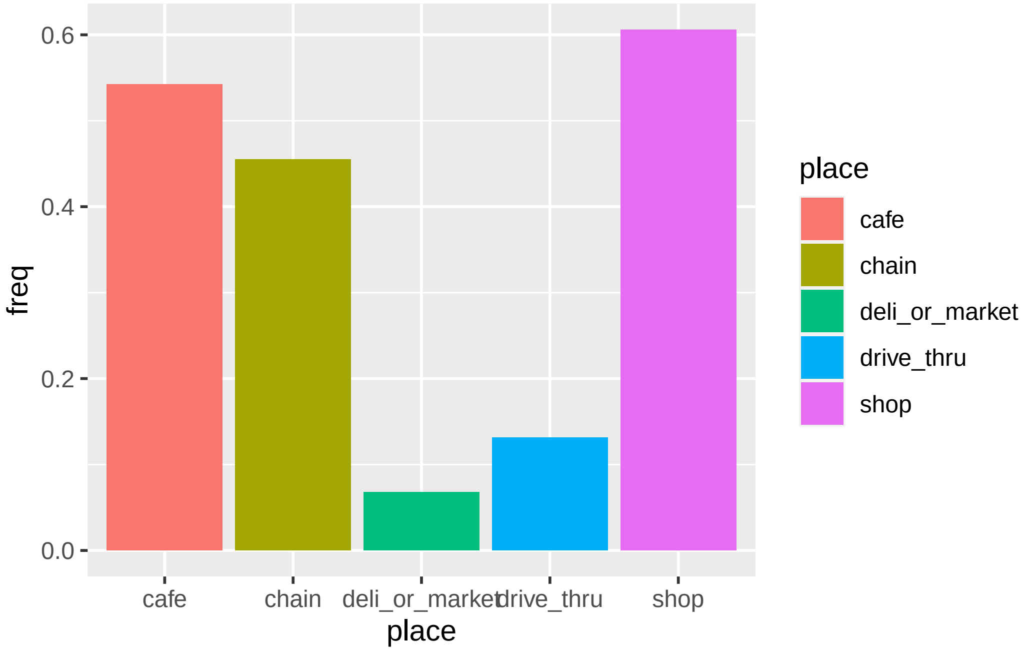
```
# Coffees per day and Employment
coffee_clean_employ= coffee_clean|>
  mutate(employment_status = ifelse(is.na(employment_status), "No disclosure", employment_sta
ggplot(coffee_clean_employ, aes(x = cups_of_coffee_per_day, y = employment_status)) +
  geom_count() +
  labs(title = "Correlation between Coffees per Day and Employment Status",
       x = "Cups of Coffee per Day",
       y = "Employment Status")
```



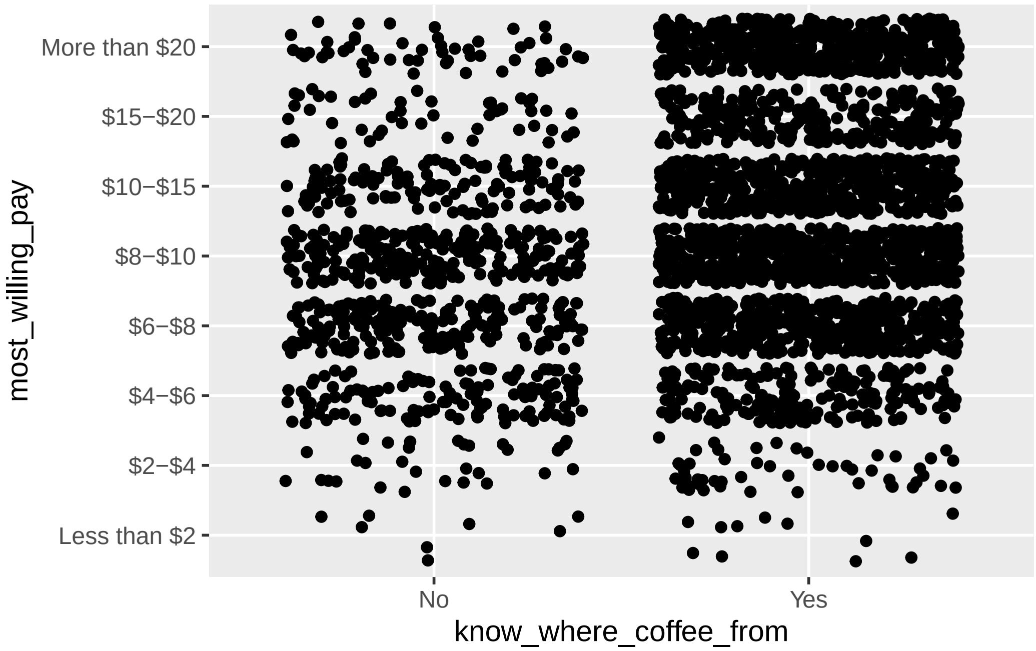
```
#Nidhi graphs

#where people typically drink
coffee_clean_factors |>
  select(starts_with("drink")) |>
  summarise(home = mean(drink_at_home, na.rm = TRUE), office = mean(drink_at_the_office, na.r
            go = mean(drink_on_the_go, na.rm = TRUE), cafe = mean(drink_at_a_cafe, na.rm = T
  pivot_longer(
    cols = everything(),
    names_to = "place",
    values_to = "freq"
  ) |>
  ggplot(aes(x = place, y = freq, fill = place)) +
  geom_col()
```



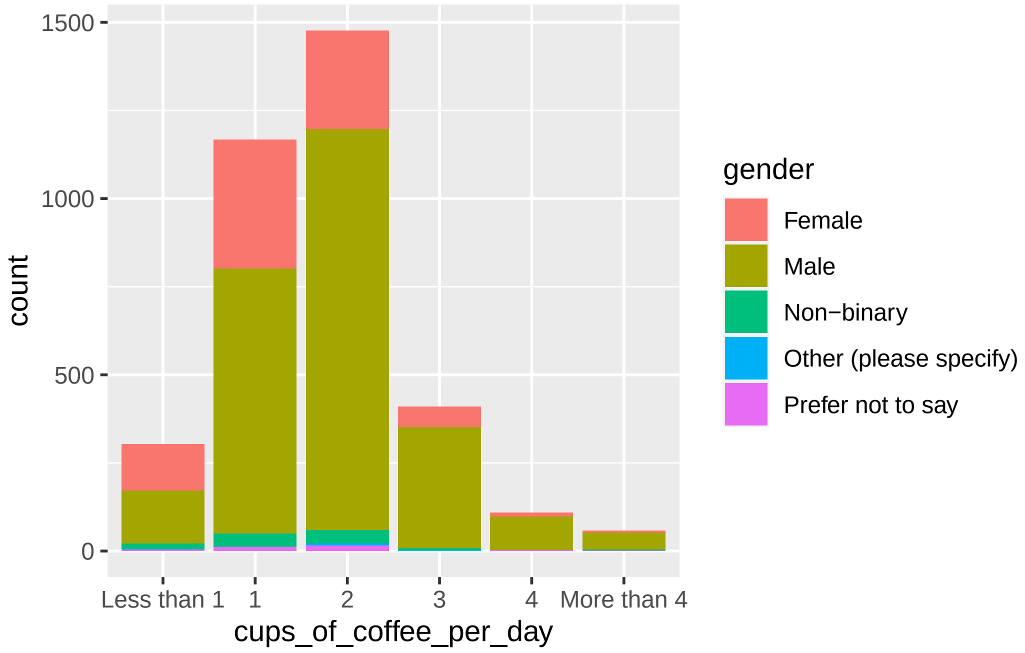
```
#where people typically purchase
purchase_online <- coffee_clean_factors |>
  select(contains("purchase")) |>
  select(where_else_purchase_coffee) |>
  na.omit() |>
  filter(grepl("online", where_else_purchase_coffee, ignore.case = TRUE))
purchase_gas_station <- coffee_clean_factors |>
  select(contains("purchase")) |>
  select(where_else_purchase_coffee) |>
  na.omit() |>
  filter(grepl("gas station", where_else_purchase_coffee, ignore.case = TRUE))

#currently not including gas station and online in other category

coffee_clean_factors |>
  select(contains("purchase")) |>
  summarise(chain = mean(purchase_national_chain, na.rm = TRUE),
            cafe = mean(purchase_local_cafe, na.rm = TRUE),
            drive_thru = mean(`purchase_drive-thru`, na.rm = TRUE),
            shop = mean(purchase_specialty_coffee_shop, na.rm = TRUE),
            deli_or_market = mean(purchase_deli_or_supermarket, na.rm = TRUE)) |>
   pivot_longer(
    cols = everything(),
    names_to = "place",
    values_to = "freq"
  ) |>
  ggplot(aes(x = place, y = freq, fill = place)) +
  geom_col()
```



```
#does knowing where coffee comes from affect how much one is willing to pay
coffee_clean_factors |>
  select(know_where_coffee_from, most_willing_pay) |>
  drop_na() |>
  ggplot(aes(x = know_where_coffee_from, y = most_willing_pay)) +
  geom_jitter()
```



```
#amount of coffee by gender (may need to standardize by frequencies of gender cause what if r
coffee_clean_factors |>
  select(gender, cups_of_coffee_per_day) |>
  drop_na() |>
  ggplot(aes(x = cups_of_coffee_per_day, fill = gender)) +
  geom_bar(position= "stack")
```

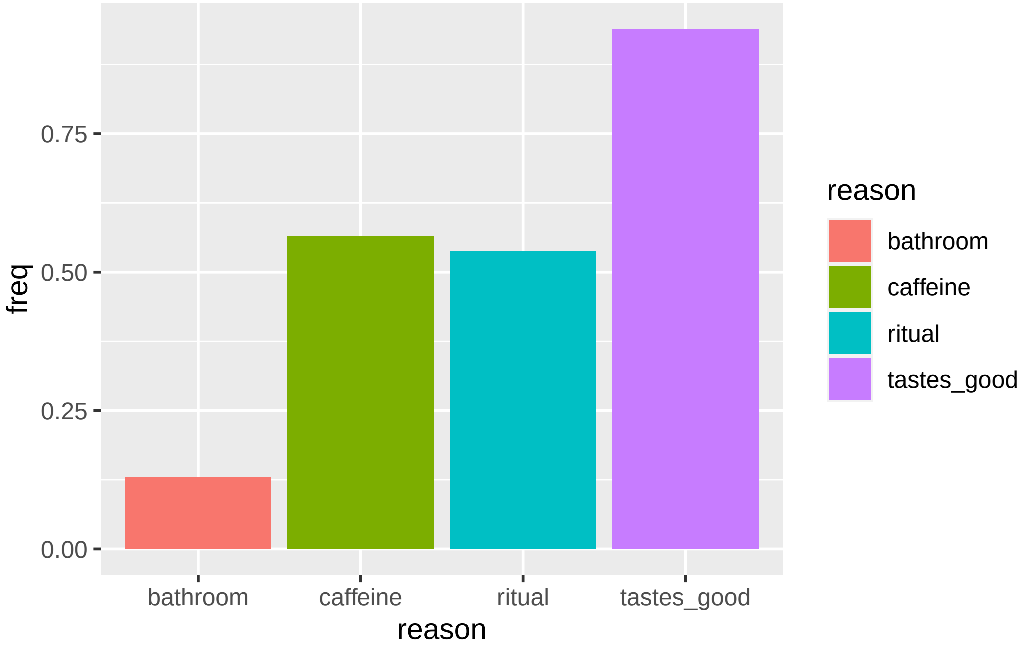


```
coffee_clean_factors |>
  select(contains("reason")) |>
  select(other_reason_for_drinking_coffee) |>
  na.omit()
```

```
# A tibble: 167 x 1
   other_reason_for_drinking_coffee
   <chr>
 1 I don't
 2 Comforting, warmth
 3 Fun and devirse
 4 Support local business
 5 My wife and I are both coffee professionals
 6 I like hanging out in coffee shops
 7 It sparks joy
 8 It smells nice
 9 Nostalgia, comfort
10 interesting to explore
# i 157 more rows
```

```
#looking at these words and their associations would be interesting. currently not looking a

coffee_clean_factors |>
```

```r
select(contains("reason")) |>
 summarise(tastes_good = mean(reason_it_tastes_good, na.rm = TRUE),
          caffeine = mean(reason_i_need_the_caffeine, na.rm = TRUE),
          ritual = mean(reason_i_need_the_ritual, na.rm = TRUE),
          bathroom = mean(reason_it_makes_me_go_to_the_bathroom, na.rm = TRUE)) |>
 pivot_longer(
  cols = everything(),
  names_to = "reason",
  values_to = "freq"
) |>
ggplot(aes(x = reason, y = freq, fill = reason)) +
geom_col()
```



Questions for reviewers

List specific questions for your peer reviewers and project mentor to answer in giving you feedback on this phase.

Do we need to close the issue you opened for the first phase?