Coffee Preferences

Brilliant Cassowary - Appendix to report

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

Data cleaning

Please have tidyverse, tidymodels, usethis, and probably installed for our packages and libraries to work.

```
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   3.5.1     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 beco
```

```
library(tidyverse)
library(janitor)
```

```
Attaching package: 'janitor'

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

    chisq.test, fisher.test
```

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

```
-- Attaching packages ------------------------------------- tidymodels 1.1.1 --
v broom        1.0.5     v rsample      1.2.0
v dials        1.2.0     v tune         1.1.2
v infer        1.0.5     v workflows    1.1.4
v modeldata    1.2.0     v workflowsets 1.0.1
v parsnip      1.2.1     v yardstick    1.3.1
v recipes      1.0.9
-- Conflicts ---------------------------------------- tidymodels_conflicts() --
x scales::discard() masks purrr::discard()
x dplyr::filter()   masks stats::filter()
x recipes::fixed()  masks stringr::fixed()
x dplyr::lag()      masks stats::lag()
x yardstick::spec() masks readr::spec()
x recipes::step()   masks stats::step()
* Dig deeper into tidy modeling with R at https://www.tmwr.org
```

```
library(probably)
```

```
Attaching package: 'probably'

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

    as.factor, as.ordered
```

```
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.
```

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

#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)

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

#renaming
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_t
    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)

#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")))

coffee_clean_factors |>
  write_rds(file = "data/coffee_clean_factor.rds")

# coffee_remove_unused <- coffee_clean_factors |>
#   select(age, cups_of_coffee_per_day, add_to_none, add_to_milk, contains("sugar_sweetener_a
```

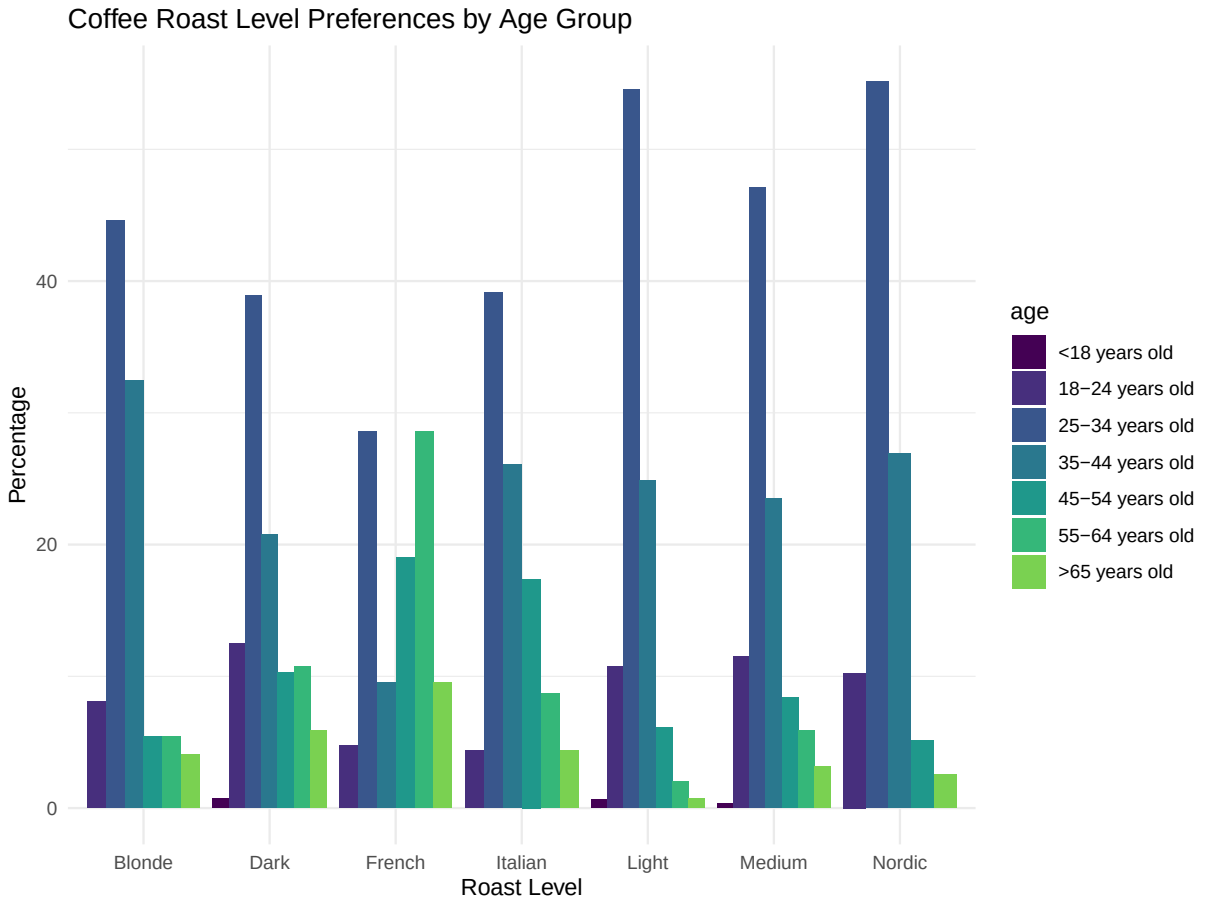
Extra Data Analysis figures

```
#age vs  roast
#coffee like or not by age
roast_totals <- coffee_clean_factors |>
  group_by(roast_level) |>
  summarise(total = n())
```

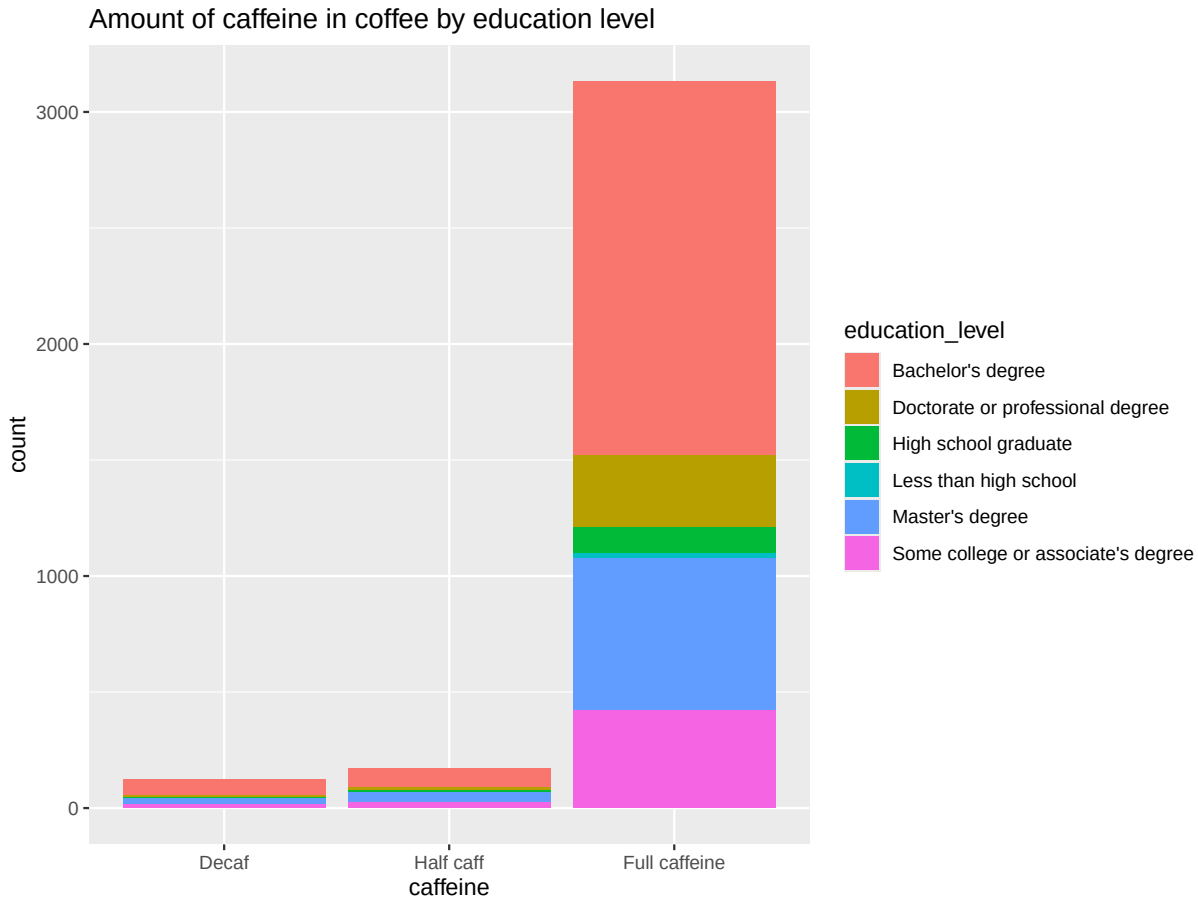
```r
age_roast_counts <- coffee_clean_factors |>
  filter(!is.na(roast_level) & !is.na(age)) |>
  group_by(age, roast_level) |>
  summarise(count = n()) |>
  left_join(roast_totals, by = "roast_level") |>
  mutate(percentage = count / total * 100)
```

```
`summarise()` has grouped output by 'age'. You can override using the `.groups`
argument.
```

```r
ggplot(age_roast_counts, aes(x = roast_level, y = percentage, fill = age)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Coffee Roast Level Preferences by Age Group",
       x = "Roast Level",
       y = "Percentage") +
  theme_minimal() +
  scale_fill_viridis_d(end = 0.8)
```



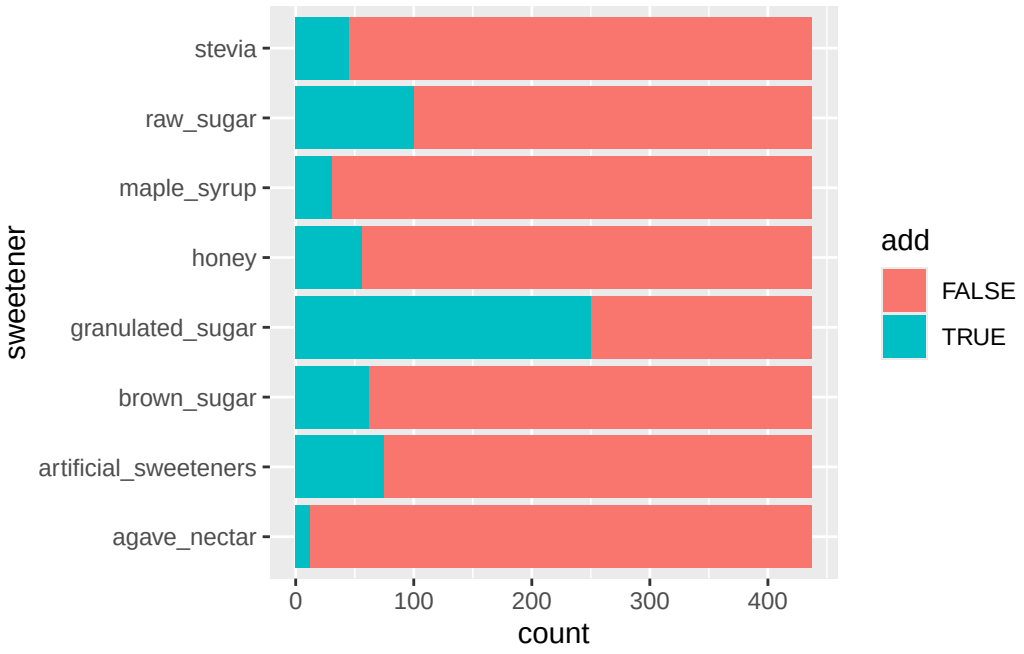

```
# #Preferences by education level
coffee_clean_factors |>
  select(caffeine, education_level) |>
  drop_na() |>
  ggplot(aes(x = caffeine, fill = education_level)) +
  geom_bar() +
  labs(title = "Amount of caffeine in coffee by education level")
```



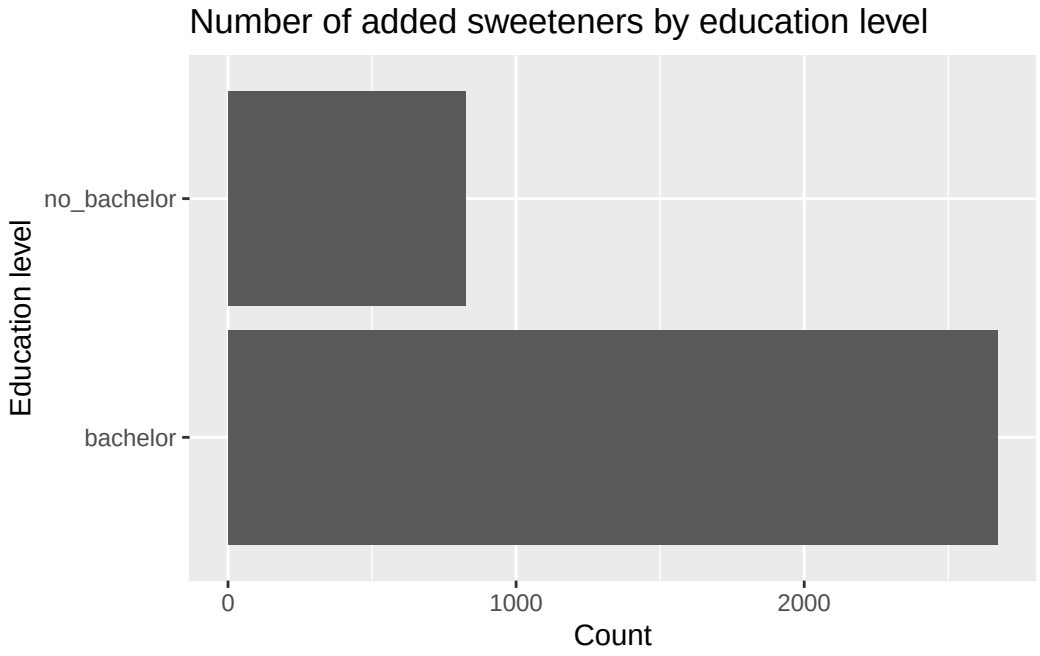

```
# Preference by Age
roast_totals <- coffee_clean_factors |>
  group_by(roast_level) |>
  summarise(total = n())
```

```
#clean original dataset
sweet_edu_df <- coffee_clean_factors |>
  select(education_level, contains("sugar_sweetener_add")) |>
  drop_na()
names(sweet_edu_df) <- gsub("sugar_sweetener_add_", "", names(sweet_edu_df))
sweet_edu_df <- sweet_edu_df |>
  mutate(education_level = case_when(
    education_level %in% c("Bachelor's degree", "Master's degree", "Doctorate or professional
    TRUE ~ "no_bachelor"
  )) |>
  mutate(education_level = factor(education_level,
                                  c("bachelor",
```

```
                                    "no_bachelor")))

#data set for analysis, with number of sugar types
sweet_edu_df_analysis <- sweet_edu_df |>
  rowwise() |>
  mutate(sweet_count = sum(across(granulated_sugar:raw_sugar), na.rm = TRUE))

#other dataset for visualization purposes
sweet_edu_df2 <- sweet_edu_df |>
  pivot_longer(
    cols = 2:9,
    names_to = "sweetener",
    values_to = "add"
  )

sweet_edu_df3 <- sweet_edu_df2 |>
  select(education_level, sweetener, add)|>
  filter(add == TRUE)

sweet_edu_df <- sweet_edu_df

#Popularity of sweeteners
sweet_edu_df2 |>
  ggplot(aes(y = sweetener, fill = add) ) +
  geom_bar()
```

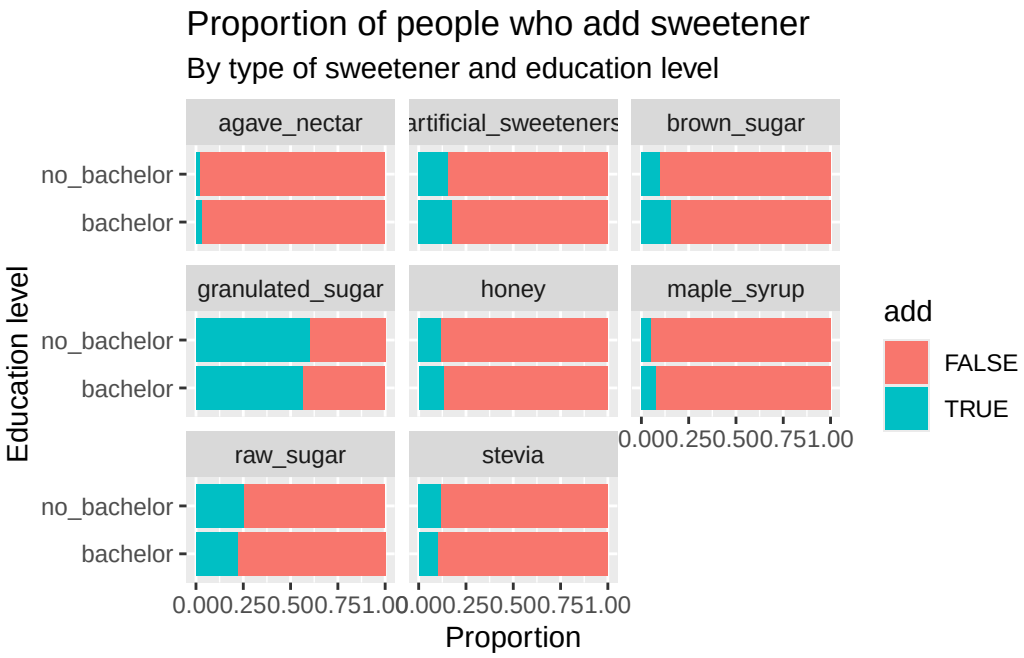



```
#number of added sweeteners by education level
sweet_edu_df2 |>
  ggplot(aes(y = education_level), fill = "blue") +
  geom_bar() +
  labs(
    title = "Number of added sweeteners by education level",
    x = "Count",
    y = "Education level"
  )
```



These graphs show the distributions for each variable, so we can get an idea of the different groups we are working with. We see that granulated_sugar is the most popular sweetener, and people with higher than a Bachelor's degree are more common in this sample, which could potentially skew results. Next, we see how these two variables are related to each other through proportions, where we can more directly compare the visual differences.

```
# #sweetener preference proportions by education level
# sweet_edu_df2 |>
#   ggplot(aes(y = sweetener, fill = add) ) +
#   geom_bar(position = "fill") +
#   facet_wrap(.~education_level)

#proportion preferring sweetener by education level
sweet_edu_df2 |>
  ggplot(aes(y = education_level, fill = add) ) +
  geom_bar(position = "fill") +
  facet_wrap(.~sweetener) +
  labs(
    title = "Proportion of people who add sweetener",
    subtitle= "By type of sweetener and education level",
    x = "Proportion",
    y= "Education level"
  )
```


Proportion of people who add sweetener
By type of sweetener and education level
agave_nectar
artificial_sweeteners
brown_sugar
granulated_sugar
honey
maple_syrup
raw_sugar
stevia
no_bachelor
bachelor
Education level
0.00
0.25
0.50
0.75
1.00
Proportion
add
FALSE
TRUE