Draft

STA 210 - Project

Ginger and Stats - Aimi Wen, Rakshita Ramakrishna, Nathan Nguyen

```r
library(tidyverse)
library(tidymodels)
library(tidytext)
library(patchwork)
library(stringr)
library(ggplot2)
library(sf)
library(rnaturalearth)
library(rnaturalearthdata)
library(countrycode)
library(kableExtra)
chocolate <- read_csv("../data/chocolate.csv")

world <- ne_countries(scale = "medium", returnclass = "sf")
```

Exploratory Data Analysis

Data description

- Description of the observations in the data set:
    - The observations in this data set represent a review of general characteristics for different chocolate bars. A single observation in this data set represents a single chocolate bar.
    - The general characteristics are as follows:
        - Company (Manufacturer) lists who made the chocolate bar reviewed; the dataset also lists where this company is located under Company Location.

        - The dataset characterizes the Country of Bean Origin, Specific Bean Origin or name of bar, Percentage of Cocoa within the bar for each chocolate bar.
        - The data also shows which ingredients are used using letters, where B = Beans, S = Sugar, S* = Sweeteners other than white can or beet sugar, C = Cocoa Butter, V = Vanilla, L = Lecithin, Sa = Salt.
        - Finally, the data shows the rating (which ranges from 1-5, incrementing by 0.25) given under their rating system, which is linked above, as well as the date it was reviewed on

- Description of how the data was originally collected (not how you found the data but how the original curator of the data collected it).
    - Data is being continuously collected and added to the dataset after reviewing chocolate bars - this can be seen as the first review years for chocolate bars began in 2006 and have continued until 2021.

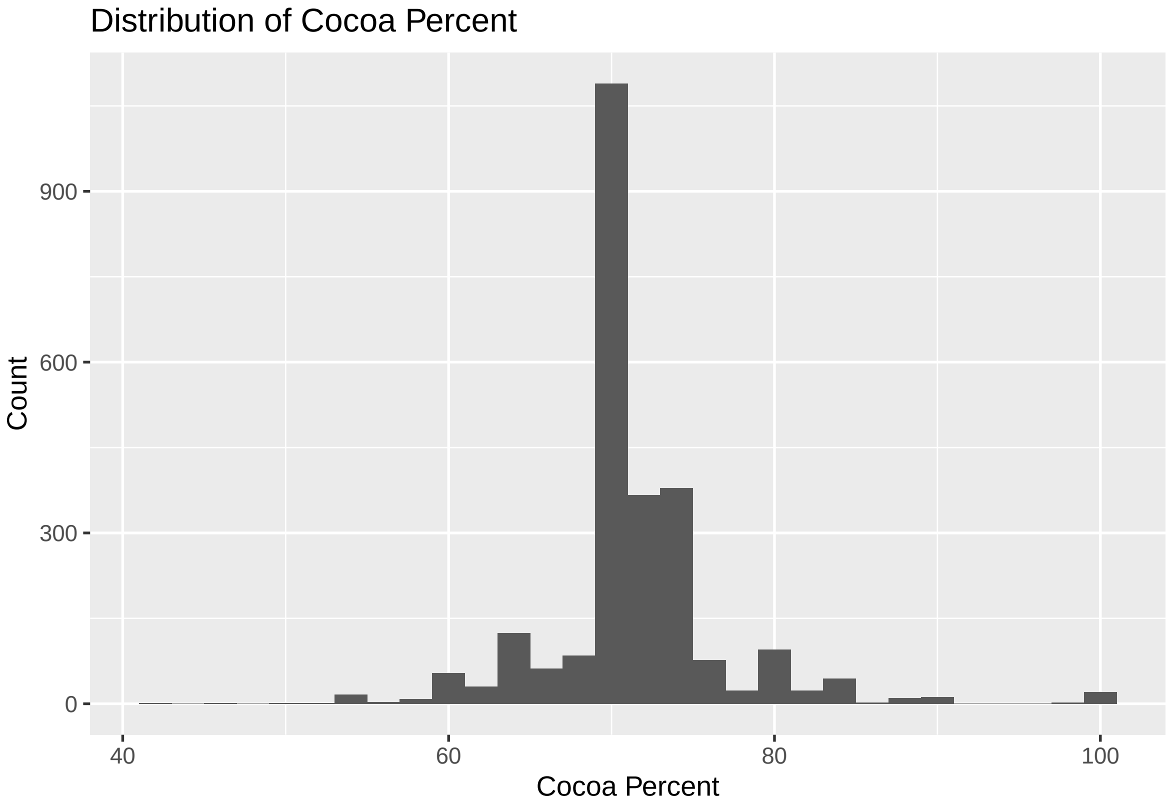
The data is collected by members of the Manhattan Chocolate Society reviewing chocolate bars using the rating system found at http://flavorsofcacao.com/review_guide.html and adding other characteristics about the bar itself.

Shape of Ratings (already done)

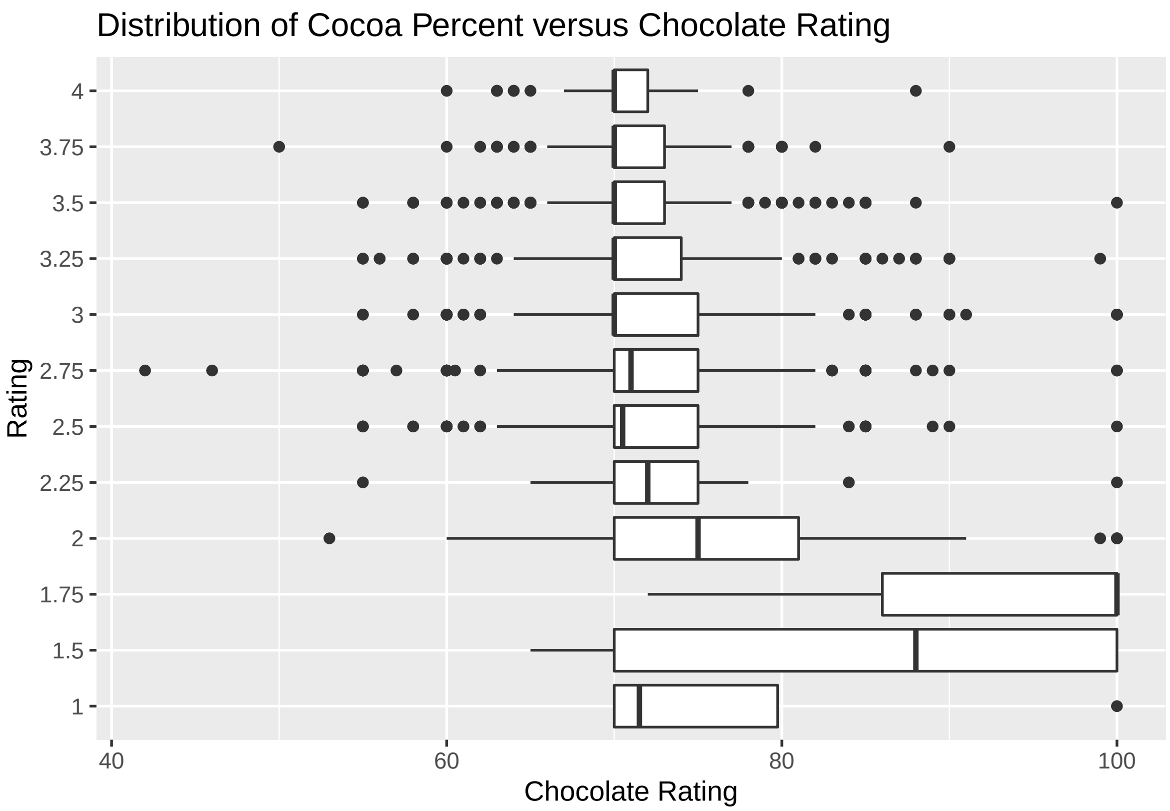
Cocoa Percent

```r
chocolate$cocoa_percent <- as.numeric(gsub('[,%]', '', chocolate$cocoa_percent))

chocolate$rating <- as.character(chocolate$rating)

ggplot(data= chocolate, aes(x= cocoa_percent)) + geom_histogram() +
  labs(title = "Distribution of Cocoa Percent",
       y = "Count",
       x = "Cocoa Percent")
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
ggplot(data= chocolate, aes(x= cocoa_percent, y= rating)) + geom_boxplot() +
  labs(title = "Distribution of Cocoa Percent versus Chocolate Rating",
       y = "Rating",
       x = "Chocolate Rating")
```


Distribution of Cocoa Percent versus Chocolate Rating

```r
chocolate$rating <- as.numeric(chocolate$rating)
```

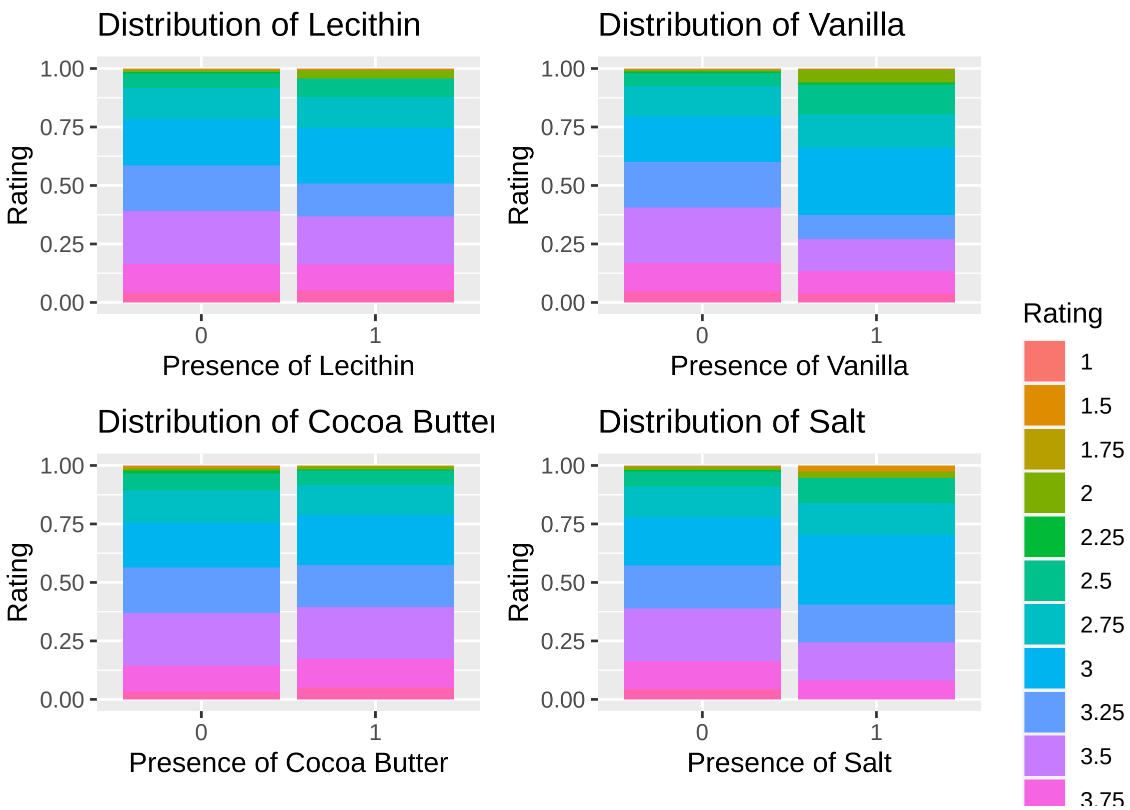
From the distribution of cocoa_percent, we see that the distribution is roughly symmetric and unimodal, and centered around 72 percent, and has apparent outliers around 55 percent and 100 percent.

From the boxplot, we can see a general rough trend that as the median cocoa percent is lower, the rating of the chocolate bar is higher. Furthermore, there appear to be a lot of outliers in the middle ratings (2.25 - 3.75), which might be due to the fact that that is the rating for the bulk of the chocolates tested.

Ingredients

```r
chocolate <- chocolate %>%
  mutate(lecithin = case_when(
    grepl("L", ingredients) ~ 1,
    T ~ 0
  ),
  vanilla = case_when(
    grepl("V", ingredients) ~ 1,
    T ~ 0
  ),
  cocoa = case_when(
    grepl("C", ingredients) ~ 1,
    T ~ 0
  ),
  salt = case_when(
    grepl("Sa", ingredients) ~ 1,
    T ~ 0
  ),

  lecithin = as.factor(lecithin),
  vanilla = as.factor(vanilla),
  cocoa = as.factor(cocoa),
  salt = as.factor(salt)
  )
```

```
pL <- ggplot(chocolate, aes(lecithin, fill = as.factor(rating))) +
  geom_bar(position = "fill") +
  labs(title = "Distribution of Lecithin",
       y = "Rating",
       x = "Presence of Lecithin") +
  theme(legend.position = "none")
pV <- ggplot(chocolate, aes(vanilla, fill = as.factor(rating))) +
    labs(title = "Distribution of Vanilla",
       y = "Rating",
       x = "Presence of Vanilla") +
  geom_bar(position = "fill") +
  theme(legend.position = "none")
pC <- ggplot(chocolate, aes(cocoa, fill = as.factor(rating))) +
  geom_bar(position = "fill") +
    labs(title = "Distribution of Cocoa Butter",
       y = "Rating",
       x = "Presence of Cocoa Butter") +
  theme(legend.position = "none")
pSa <- ggplot(chocolate, aes(salt, fill = as.factor(rating))) +
    labs(title = "Distribution of Salt",
       y = "Rating",
       x = "Presence of Salt",
      fill = "Rating") +
  geom_bar(position = "fill")

(pL + pV)/(pC + pSa)
```

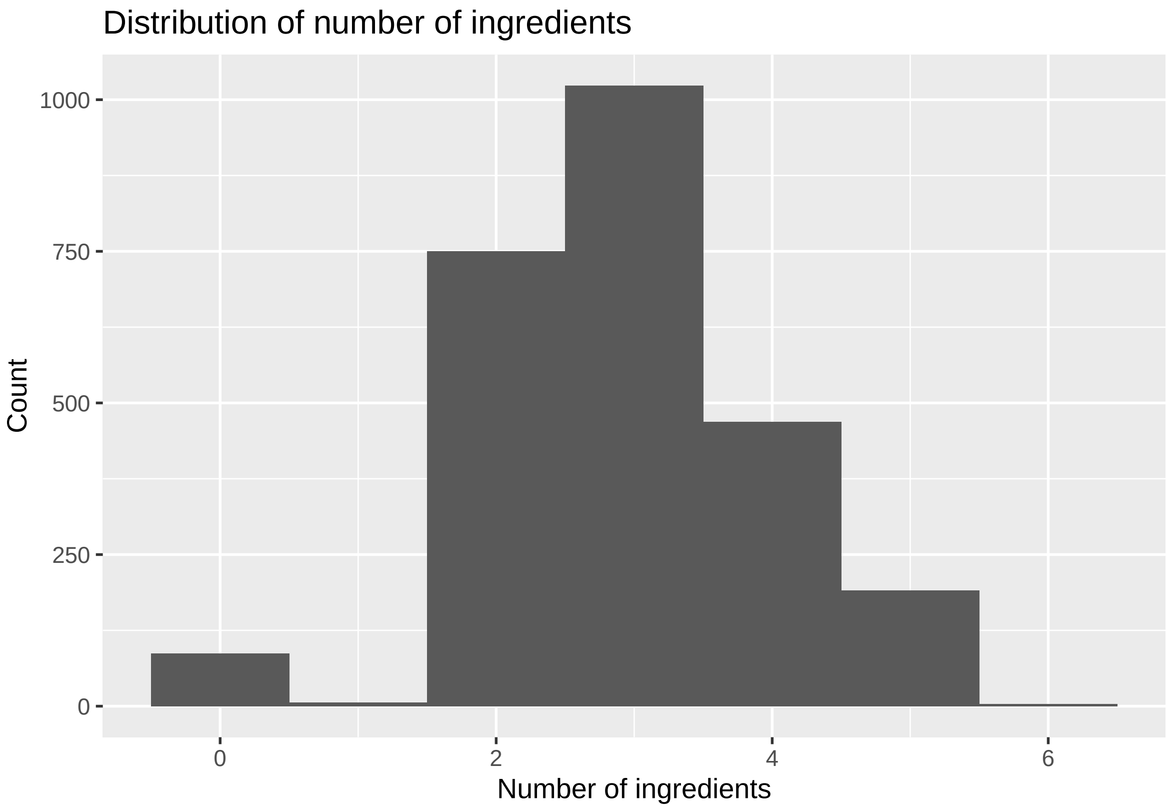


From this visualization, we can see that the presence of salt and vanilla seem to affect the rating the most out of all the predicters. The presence of salt and vanilla results in more lower ratings, while the amount of high and low ratings remains roughly the same with/without the presence of cocoa butter and lecithin.

```
chocolate <- chocolate %>%
  mutate(
    num_ingres = if_else(is.na(ingredients), "0", str_sub(ingredients, 1, 1)),
    num_ingres = as.numeric(num_ingres)
  )

chocolate %>%
  drop_na(
    ingredients
  ) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1  2443
```

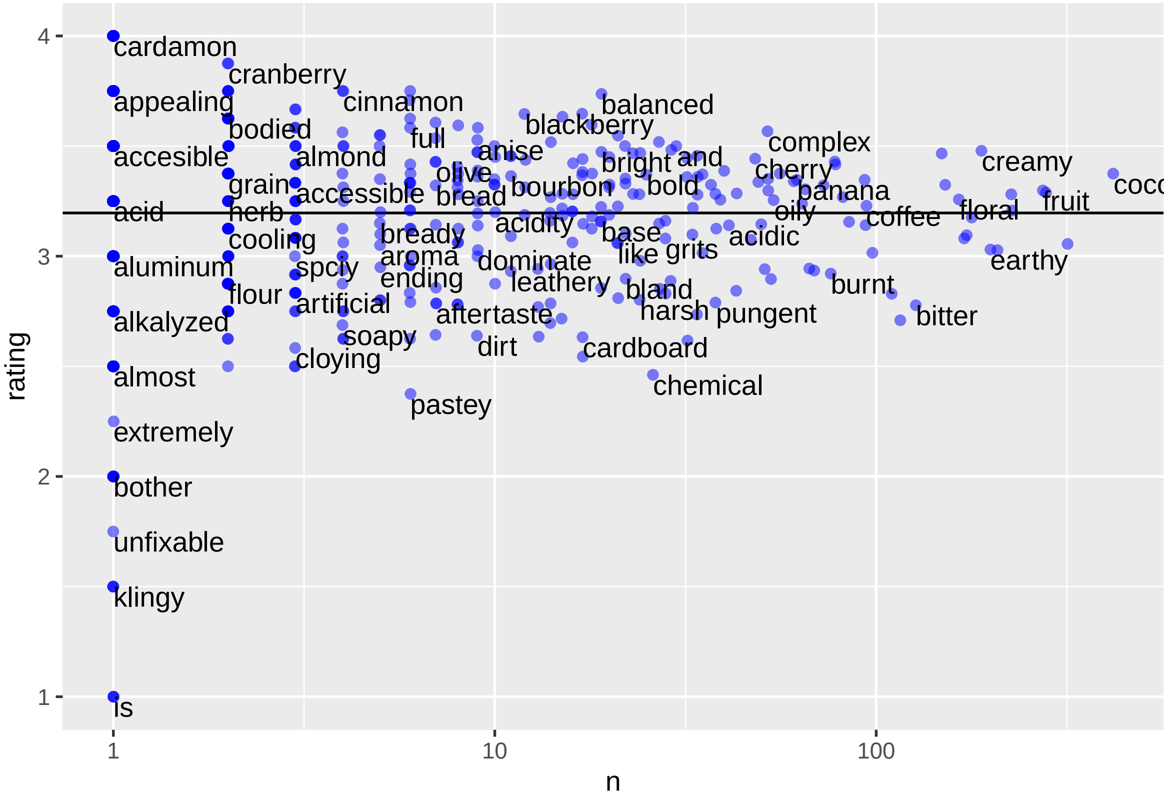
```
ggplot(chocolate, aes(num_ingres))+
  geom_histogram(binwidth = 1)+
  labs(
    title = "Distribution of number of ingredients",
    x = "Number of ingredients",
    y = "Count"
  )
```



This visualization showcases a right skewed distribution for the number of ingredients. The median is somewhere around 3 ingredients, and there appears to be an outlier centered around 0. This could be as many chocolate bars use at least one of the common ingredients, and it is quite rare for a chocolate bar not to have any of those ingredients.

Most Memorable Characteristic (Aimi)

```r
tidy_chocolate<- chocolate %>%
  unnest_tokens(word, most_memorable_characteristics)

tidy_chocolate %>%
  group_by(word) %>%
  summarize( n= n(),
             rating= mean(rating) ) %>%
  ggplot(aes(n, rating)) +
  geom_hline(yintercept= mean(chocolate$rating)) +
  geom_jitter(color= "blue", alpha= 0.5) +
  geom_text(aes(label= word),
            check_overlap= TRUE,
            vjust= "top",
            hjust= "left") +
  scale_x_log10()
```

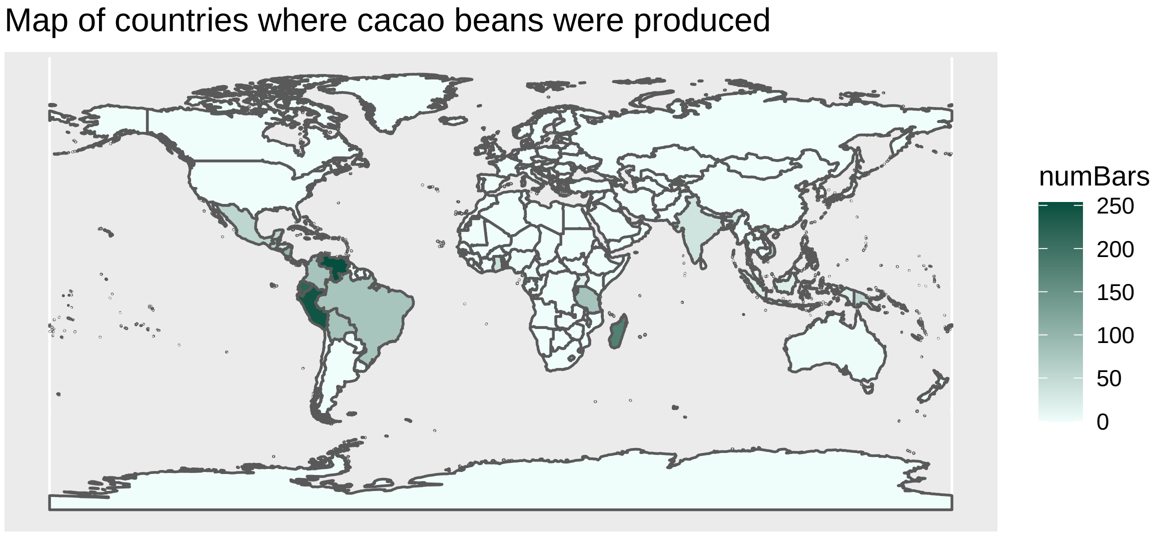


From this visualization, we can see that the phrases and most memorable charactersists that were often associated with a higher rating were “balanced” and “complex”, as well as fruity chocolate like “fruit”, “Cardamon”, “floral”.

Country Bean of Origin (Rakshita)

```
chocolate_modified <- chocolate %>%
  mutate(name_long = country_of_bean_origin) %>%
  group_by(name_long) %>%
  count(name_long)

chocworld_data <- world %>%
  full_join(y = chocolate_modified,
  by = "name_long") %>%
  mutate(numBars = ifelse(is.na(n), 0, n))
```

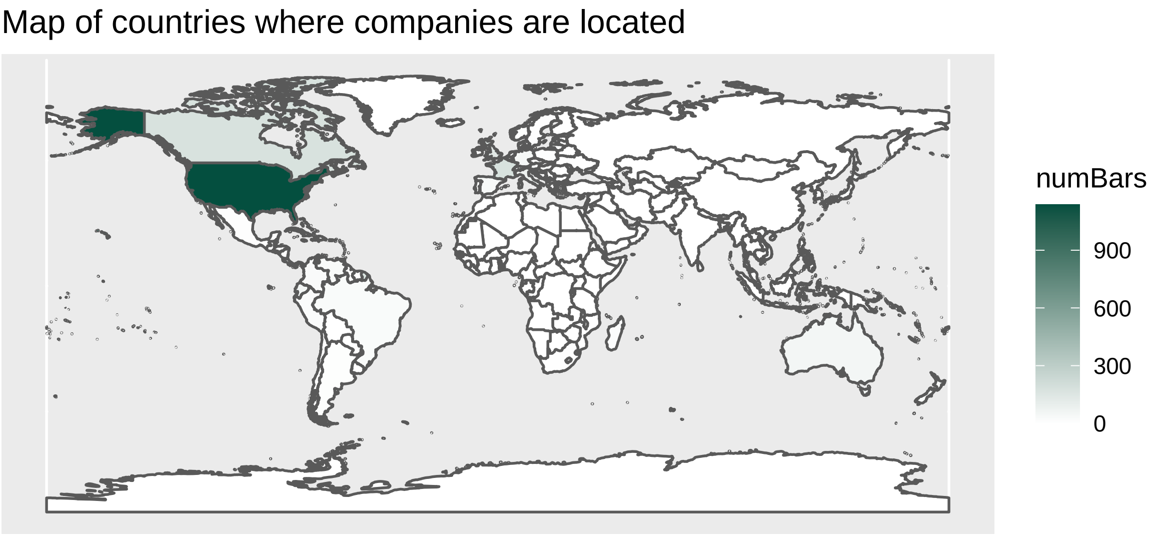
```
ggplot(data = chocworld_data) +
  scale_fill_gradient(low = "#F0FEFB", high = "#044F3F") +
  geom_sf(aes(fill = numBars, geometry = geometry)) +
  labs(title = "Map of countries where cacao beans were produced")
```



This map shows that the majority of cacao beans are produced in central America, South America, Asia, and Africa.

Company Location (Rakshita)

```
chocolate_modified2 <- chocolate %>%
  mutate(name_long = case_when(
    company_location == "U.S.A." ~ "United States",
    company_location == "U.K." ~ "United Kingdom",
    company_location == company_location ~ company_location)) %>%
  group_by(name_long) %>%
  count(name_long)

chocworld_data1 <- world %>%
  full_join(y = chocolate_modified2,
  by = "name_long") %>%
  mutate(numBars = ifelse(is.na(n), 0, n))

ggplot(data = chocworld_data1) +
  scale_fill_gradient(low = "#ffffff", high = "#044F3F") +
  geom_sf(aes(fill = numBars, geometry = geometry)) +
  labs(title = "Map of countries where companies are located")
```



```
chocolate %>%
    count(company_location, sort = TRUE)
```

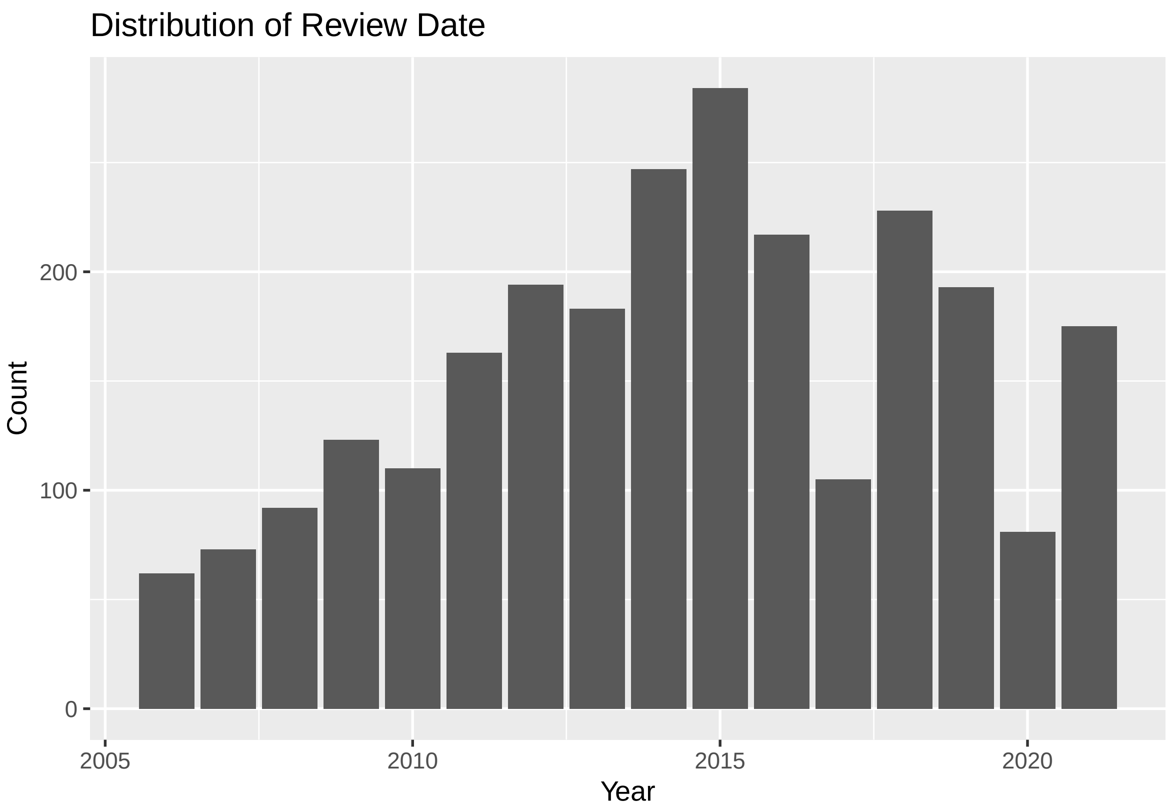
```
# A tibble: 67 x 2
```

```
   company_location     n
   <chr>            <int>
 1 U.S.A.            1136
 2 Canada             177
 3 France             176
 4 U.K.               133
 5 Italy               78
 6 Belgium             63
 7 Ecuador             58
 8 Australia           53
 9 Switzerland         44
10 Germany             42
# ... with 57 more rows
```

This map shows that the majority of countries that chocolate companies are located in are concentrated in North America and Europe, and that the US is host to the largest amount of chocolate companies.

Review Date (Nathan)

```
ggplot(chocolate, aes(review_date))+
  geom_bar() +
  labs(
    title = "Distribution of Review Date",
    x = "Year",
    y = "Count"
  )
```

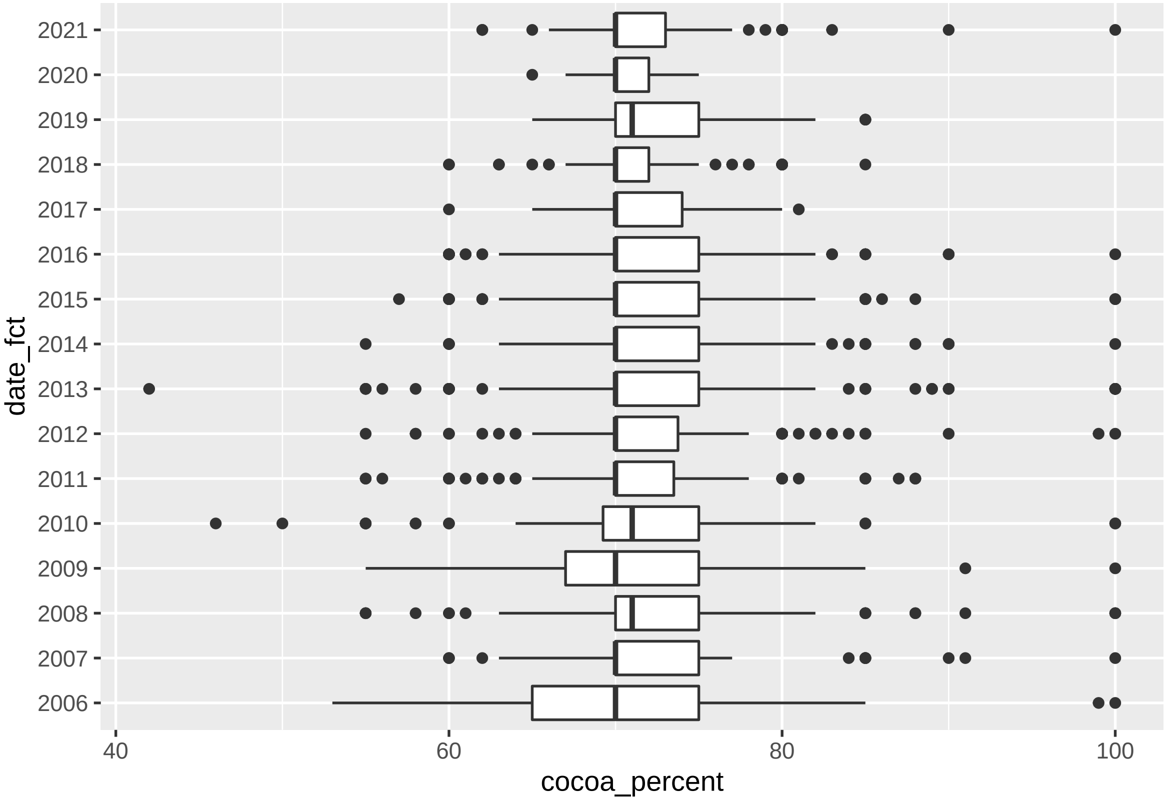


Here, we can see that the distribution of chocolate bars reviewed over time has a roughly unimodal distribution with a peak around 2015. Furthermore there was a signficant dip in 2020, probably due to the COVID-19 Pandemic, as well as a dip in 2017, due to unknown reasons. The distribution is centered around 2014 and is roughly symmetric.

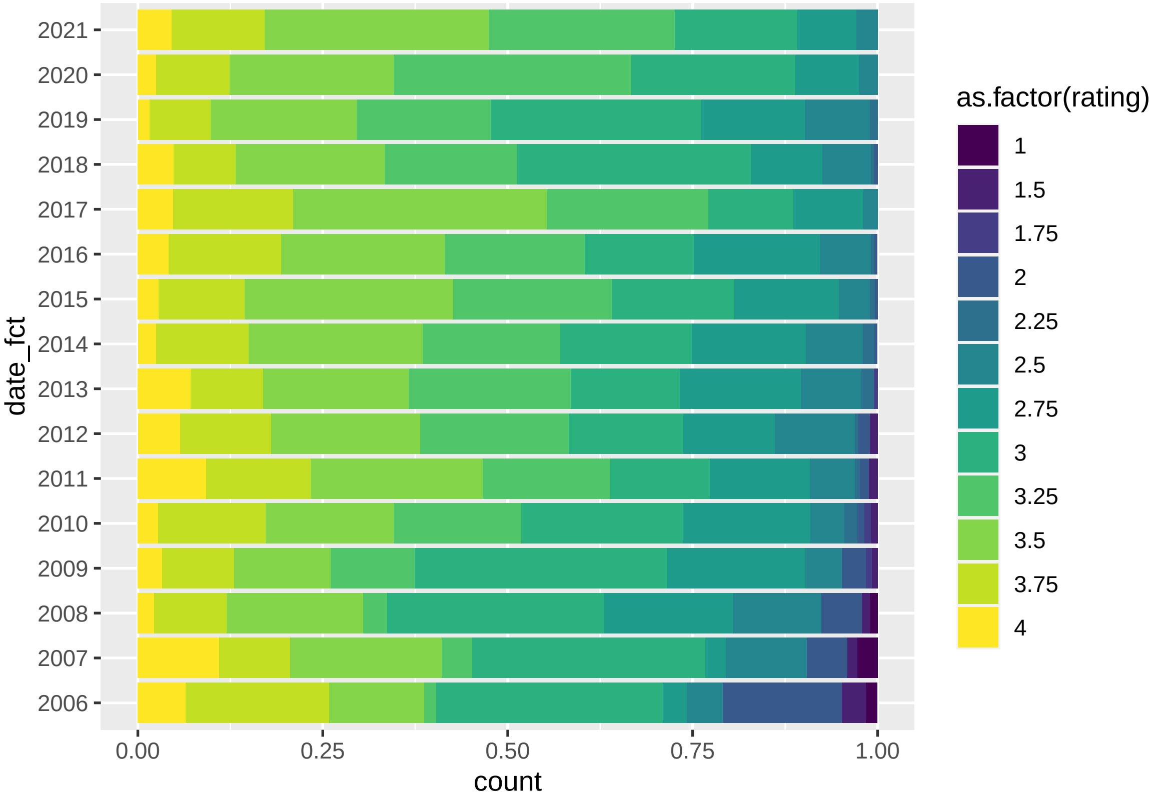
```
# statistics of review dates

chocolate %>%
  summarise(mean = mean(review_date),
            median = median(review_date),
            sd = sd(review_date))
```

```
# A tibble: 1 x 3
   mean median    sd
  <dbl>  <dbl> <dbl>
1 2014.   2015  3.97
```

```r
#review date vs cocoa_percent and ratings

chocolate <- chocolate %>%
  mutate(
    date_fct = as.factor(review_date)
  )

ggplot(chocolate, aes(date_fct, cocoa_percent))+
  geom_boxplot()+
  coord_flip()
```



```r
ggplot(chocolate, aes(date_fct, fill = as.factor(rating)))+
  geom_bar(position = "fill")+
  coord_flip()+
  scale_fill_viridis_d()
```



This visualization showcases the distribution of ratings for each review year. There is no apparent change or pattern to the change in ratings of years, and it appears that ratings from 2.5 - 3.25 compose the bulk of the ratings each year.

```
chocolate_clean <- chocolate %>%
  separate(most_memorable_characteristics, sep= ",", into= c("most_memorable", "other_memoral
  select(-other_memorable)
```

```
Warning: Expected 2 pieces. Missing pieces filled with `NA` in 95 rows [14, 34,
39, 41, 99, 145, 168, 228, 240, 264, 281, 290, 357, 365, 368, 405, 426, 433,
442, 477, ...].
```

```
#|label: cleaning-dataset
chocolate_clean <- chocolate_clean %>%
  mutate(
    top_memorable= case_when(
      str_detect(most_memorable, "cream") ~ "fatty_smooth",
      str_detect(most_memorable, "fatty") ~ "fatty_smooth",
      str_detect(most_memorable, "smooth") ~ "fatty_smooth",
      str_detect(most_memorable, "dairy") ~ "fatty_smooth",
      str_detect(most_memorable, "roast") ~ "roast",
      str_detect(most_memorable, "earth") ~ "roast",
      str_detect(most_memorable, "smoke") ~ "roast",
      str_detect(most_memorable, "wood") ~ "roast",
      str_detect(most_memorable, "bitter") ~ "roast",
      str_detect(most_memorable, "intense") ~ "strong_sweet",
      str_detect(most_memorable, "sweet") ~ "strong_sweet",
      str_detect(most_memorable, "cocoa") ~ "strong_sweet",
      str_detect(most_memorable, "caramel") ~ "strong_sweet",
      str_detect(most_memorable, "brownie")~ "strong_sweet",
      str_detect(most_memorable, "sandy") ~ "rough_texture",
      str_detect(most_memorable, "dry") ~ "rough_texture",
      str_detect(most_memorable, "gritty") ~ "rough_texture",
      str_detect(most_memorable, "coarse") ~ "rough_texture",
      str_detect(most_memorable, "chalky") ~ "rough_texture",
      str_detect(most_memorable, "powdery") ~ "rough_texture",
      str_detect(most_memorable, "nut") ~ "nutty",
      str_detect(most_memorable, "sticky") ~ "greasy",
      str_detect(most_memorable, "oily") ~ "greasy",
      str_detect(most_memorable, "spic") ~ "spiced",
      str_detect(most_memorable, "molasses") ~ "spiced",
      str_detect(most_memorable, "floral") ~ "floral",
      str_detect(most_memorable, "grassy") ~ "floral",
      str_detect(most_memorable, "vanilla") ~ "floral",
      str_detect(most_memorable, "fruit") ~ "fruity",
      str_detect(most_memorable, "tart") ~ "fruity",
      str_detect(most_memorable, "banana") ~ "fruity",
      str_detect(most_memorable, "berry") ~ "fruity",
      str_detect(most_memorable, "berries") ~ "fruity",
      str_detect(most_memorable, "citrus") ~ "fruity",
      str_detect(most_memorable, "lemon") ~ "fruity",
      str_detect(most_memorable, "complex") ~ "complex",
      TRUE ~ "other"
    )
  )
```

```
chocolate_clean$continent_bean <- countrycode(sourcevar= chocolate_clean[["country_of_bean_o
                                              destination= "continent")
```

```
Warning in countrycode_convert(sourcevar = sourcevar, origin = origin, destination = dest, :
```

```
chocolate_clean <- chocolate_clean %>%
  mutate(continent_bean= ifelse(
    country_of_bean_origin== "U.S.A.", "North America", continent_bean
  ))

chocolate_clean <- chocolate_clean %>%
  mutate(continent_bean= ifelse(
    continent_bean== "Americas", "South America", continent_bean
  ))
```

```
chocolate_clean <- chocolate_clean %>%
  mutate(continent_bean= case_when(
    continent_bean== "South America" ~ "South America",
    continent_bean== "Africa" ~ "Africa",
    continent_bean== "Asia" ~ "Asia",
    TRUE ~ "Other"
  ))
```

```
chocolate_clean$continent_company <- countrycode(sourcevar= chocolate_clean[["company_locati
                                              destination= "continent")
```

```
Warning in countrycode_convert(sourcevar = sourcevar, origin = origin, destination = dest, :
```

```
chocolate_clean <- chocolate_clean %>%
  mutate(continent_company= ifelse(
    company_location== "U.S.A.", "North America", continent_company
  )) %>%
  mutate(continent_company=ifelse(
    company_location== "Canada", "North America", continent_company
  )) %>%
  mutate(continent_company= ifelse(
    continent_company== "Americas", "South America", continent_company
    )
  )
```

```
chocolate_clean <- chocolate_clean %>%
  mutate(continent_company= case_when(
    continent_company== "North America" ~ "North America",
    continent_company== "Europe" ~ "Europe",
    TRUE ~ "Other"
  ))
```

Analysis approach

Ratings vs cocoa percent, ingredients, most memorable characteristics

```
set.seed(2100)
choco_split <- initial_split(chocolate_clean)
choco_training <- training(choco_split)
choco_testing <- testing(choco_split)

choco_spec <- linear_reg() %>%
  set_engine("lm")

choco_rec1 <- recipe(rating ~ cocoa_percent + vanilla + salt + num_ingres + top_memorable, da
  step_center(num_ingres, cocoa_percent) %>%
  step_dummy(all_nominal_predictors()) %>%
  step_zv(all_predictors())

choco_wflow1 <- workflow() %>%
  add_model(choco_spec) %>%
  add_recipe(choco_rec1)

set.seed(2500)
choco_folds <- vfold_cv(choco_training, v = 5)
choco_fit_rs1 <- choco_wflow1 %>%
  fit_resamples(choco_folds)

cv_metrics1 <- collect_metrics(choco_fit_rs1, summarize = FALSE)

cv_metrics1 %>%
  mutate(.estimate = round(.estimate, 3)) %>%
  pivot_wider(id_cols = id, names_from = .metric, values_from = .estimate) %>%
  kable(col.names = c("Fold", "RMSE", "R-squared"), caption = "Model 1")
```

```
#choco_fit <- linear_reg() %>%
  #set_engine("lm") %>%
  #fit(rating ~ cocoa_percent + vanilla + salt +
```

Table 1: Model 1

Fold	RMSE	R-squared
Fold1	0.425	0.117
Fold2	0.386	0.111
Fold3	0.415	0.139
Fold4	0.418	0.119
Fold5	0.385	0.101

```
        #num_ingres + top_memorable, data = chocolate_clean)

#tidy(choco_fit)

#glance(choco_fit) %>%
  #select(adj.r.squared, AIC, BIC)
```

All predictors

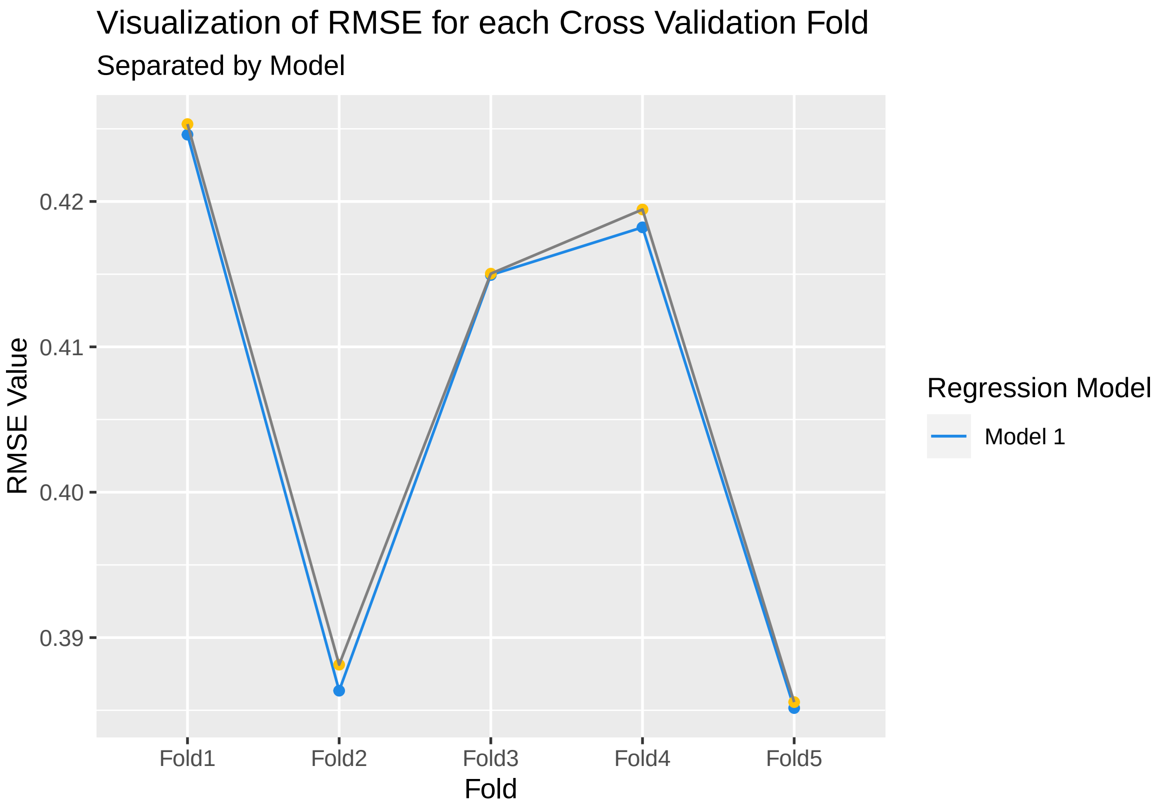
```
choco_rec2 <- recipe(rating ~ cocoa_percent + vanilla + salt + num_ingres +
                       top_memorable + continent_company + continent_bean,
                     data = choco_training) %>%
  step_center(num_ingres, cocoa_percent) %>%
  step_dummy(all_nominal_predictors()) %>%
  step_zv(all_predictors())

choco_wflow2 <- workflow() %>%
  add_model(choco_spec) %>%
  add_recipe(choco_rec2)

set.seed(2500)
choco_folds <- vfold_cv(choco_training, v = 5)
choco_fit_rs2 <- choco_wflow2 %>%
  fit_resamples(choco_folds)

cv_metrics2 <- collect_metrics(choco_fit_rs2, summarize = FALSE)

cv_metrics2 %>%
  mutate(.estimate = round(.estimate, 3)) %>%
  pivot_wider(id_cols = id, names_from = .metric, values_from = .estimate) %>%
  kable(col.names = c("Fold", "RMSE", "R-squared"), caption = "Model 2")
```

Table 2: Model 2

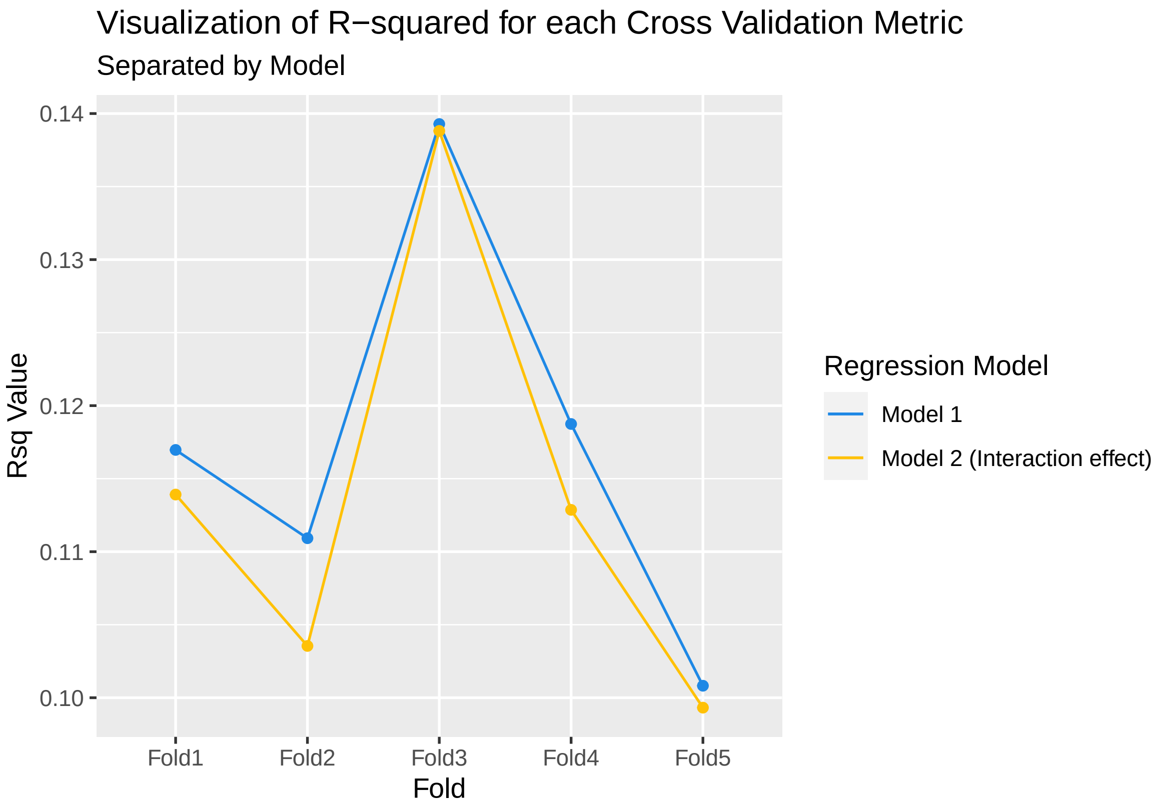
Fold	RMSE	R-squared
Fold1	0.425	0.114
Fold2	0.388	0.104
Fold3	0.415	0.139
Fold4	0.419	0.113
Fold5	0.386	0.099

```
#choco_fit_full <- linear_reg() %>%
  #set_engine("lm") %>%
  #fit(rating ~ cocoa_percent + vanilla + salt +
        #num_ingres + top_memorable + continent_company,
      #data = chocolate_clean)

#tidy(choco_fit_full)

#glance(choco_fit_full) %>%
  #select(adj.r.squared, AIC, BIC)
```

```
#RMSE Visualization
ggplot() +
  geom_point(data = cv_metrics1 %>% filter(.metric == "rmse"),
    mapping = aes(x = id, y = .estimate),
    color = "#1E88E5") +
  geom_line(data = cv_metrics1 %>% filter(.metric == "rmse"),
    mapping = aes(x = id,y = .estimate,
    color = "Model 1", group = 1)) +
  geom_point(data = cv_metrics2 %>% filter(.metric == "rmse"),
    mapping = aes(x = id,y = .estimate),
    color = "#FFC107") +
  geom_line(data = cv_metrics2 %>% filter(.metric == "rmse"),
    mapping = aes(x = id,y = .estimate,
    color = "Model 2 (Full model)", group = 1)) +
  scale_color_manual(name = "Regression Model",
    breaks=c("Model 1", "Model 2 (Full Model)"),
    values=c("Model 1" = "#1E88E5",
    "Model 2 (Interaction effect)" = "#FFC107")) +
  labs(
    title = "Visualization of RMSE for each Cross Validation Fold",
    subtitle = "Separated by Model",
    y = "RMSE Value",
    x = "Fold"
  )
```



```
#RSQ Visualization

ggplot() +
  geom_point(data = cv_metrics1 %>% filter(.metric == "rsq"),
    mapping = aes(x = id, y = .estimate),
    color = "#1E88E5") +
  geom_line(data = cv_metrics1 %>% filter(.metric == "rsq"),
    mapping = aes(x = id,y = .estimate,
    color = "Model 1", group = 1)) +
  geom_point(data = cv_metrics2 %>% filter(.metric == "rsq"),
    mapping = aes(x = id,y = .estimate),
    color = "#FFC107") +
  geom_line(data = cv_metrics2 %>% filter(.metric == "rsq"),
    mapping = aes(x = id,y = .estimate,
    color = "Model 2 (Interaction effect)", group = 1)) +
    scale_color_manual(name = "Regression Model",
    breaks=c("Model 1", "Model 2 (Interaction effect)"),
    values=c("Model 1" = "#1E88E5",
    "Model 2 (Interaction effect)" = "#FFC107")) +
  labs(
    title = "Visualization of R-squared for each Cross Validation Metric",
    subtitle = "Separated by Model",
    y = "Rsq Value",
    x = "Fold"
  )
```



As both models have similar RMSE and R-squared values for each fold in cross-validation we will choose the first model as it has fewer predictor variables and aligns with the goals of parsimony.

Data

The data dictionary can be found here.