HW4 MRN

Question 1

```r
#Loading needed libraries
library(tidyr)
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v purrr     1.0.2
v forcats   1.0.0     v readr     2.1.5
v ggplot2   3.5.1     v stringr   1.5.1
v lubridate 1.9.3     v tibble    3.2.1
-- 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(forcats)
library(ggplot2)
library(tidytuesdayR)
library(dplyr)
library(ggridges)
library(widyr)
library(ggraph)
library(igraph)
```

```
Attaching package: 'igraph'

The following objects are masked from 'package:lubridate':
```

```
    %--%, union

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

    as_data_frame, groups, union

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

    compose, simplify

The following object is masked from 'package:tibble':

    as_data_frame

The following object is masked from 'package:tidyr':

    crossing

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

    decompose, spectrum

The following object is masked from 'package:base':

    union
```

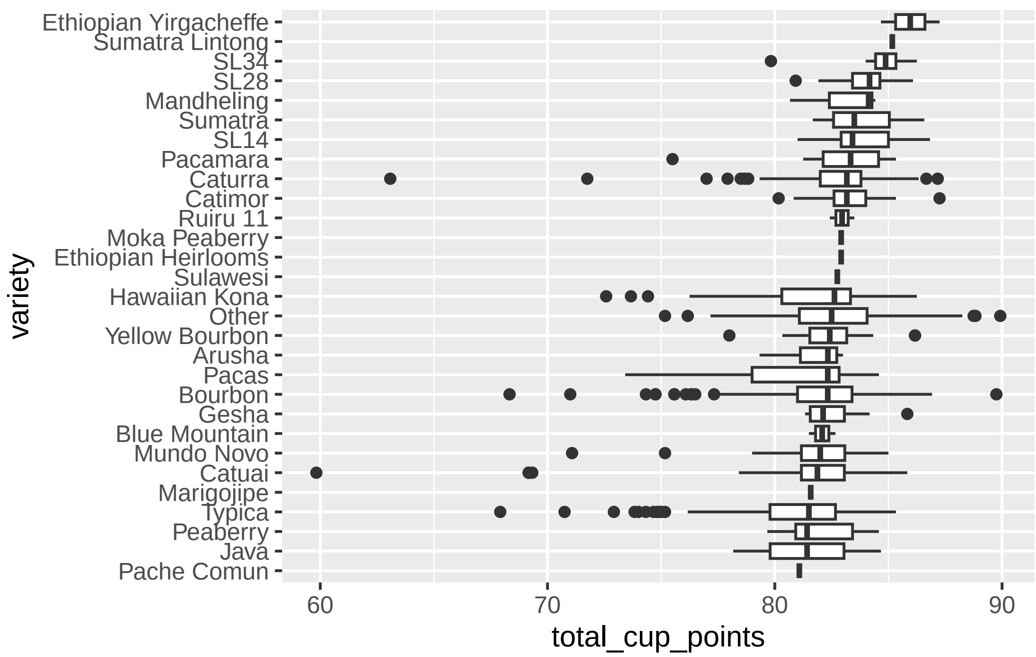
```
library(tidytext)
#Loading Dataset

coffee_ratings<- read.csv("/Users/marcusnogueira/Library/Mobile Documents/com~apple~CloudDocs
```

```
#Question 1 Part 1
#What does the variable total cup points represent? [10 points]

coffee_ratings <- coffee_ratings|>
  mutate(coffee_id = row_number()) |>
  filter(total_cup_points > 0)
```

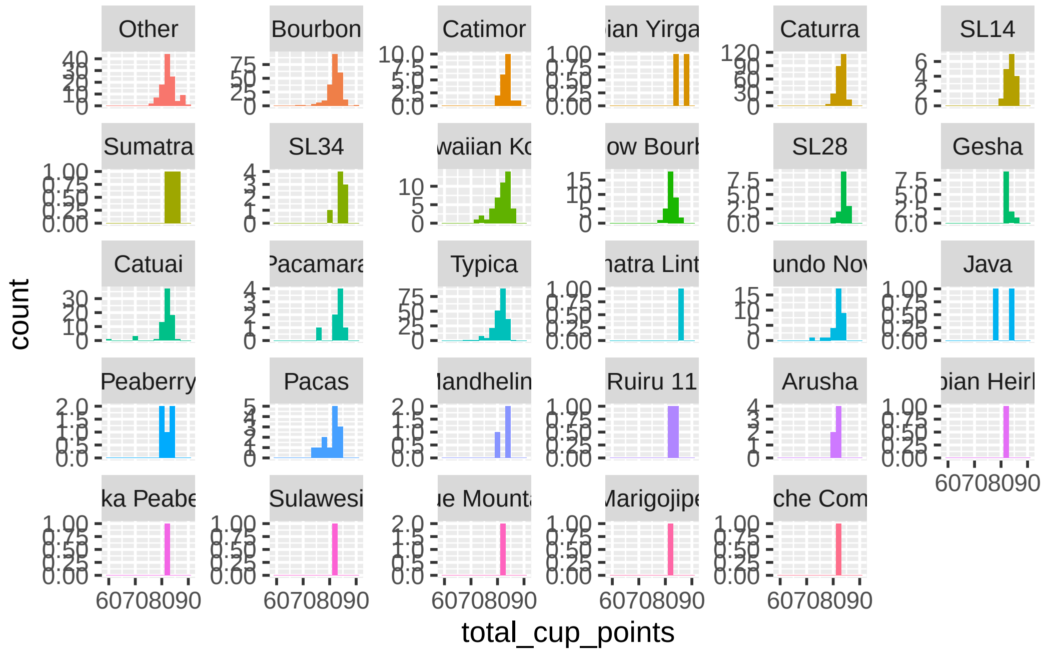
The variable "total_cup_points" represents the overall rating of the coffee, scaled from 0 to 100. This score is derived by dividing the total points accumulated, providing a measure of the coffee's quality.

```r
#Question 1 Part 2
#Based on the dataset, what factors could affect coffee quality? [10 points]
coffee_lumped <- coffee_ratings %>%
  filter(!is.na(variety)) %>%
  mutate(variety = fct_lump(variety, n = 12, w = NULL), .by = variety) %>%
  arrange(desc(variety))
```

```r
coffee_lumped|>
  mutate(variety = fct_reorder(variety, total_cup_points)) |>
  ggplot(aes(total_cup_points,variety)) +
  geom_boxplot()
```




```r
coffee_lumped|>
  ggplot(aes(total_cup_points,fill = variety)) +
  geom_histogram(binwidth = 2) +
  facet_wrap(~variety, scale = "free_y") +
  theme(legend.position = "none")
```




```r
library(dplyr)
library(tidyr)

# Summarize the percentage of non-missing values across all columns
non_missing_summary <- coffee_ratings |>
  summarize(across(everything(), ~ mean(!is.na(.)))) |>
  pivot_longer(cols = everything(), names_to = "variable", values_to = "non_missing_percenta

# Print the summary
print(non_missing_summary)
```

```
# A tibble: 44 x 2
   variable          non_missing_percentage
   <chr>                              <dbl>
 1 total_cup_points                   1
 2 species                            1
 3 owner                              0.995
 4 country_of_origin                  0.999
 5 farm_name                          0.732
 6 lot_number                         0.206
 7 mill                               0.765
 8 ico_number                         0.887
```

```
 9 company                            0.844
10 altitude                           0.831
# i 34 more rows
```

```r
# Count the top 10 producers by occurrences
top_10_producers <- coffee_ratings |>
  count(producer, sort = TRUE) |>
  slice_max(n, n = 10)

# Print the top 10 producers
print(top_10_producers)
```

```
                       producer   n
1                          <NA> 231
2                      La Plata  30
3           Ipanema Agrícola SA  22
4  Doi Tung Development Project  17
5              Ipanema Agricola  12
6                        VARIOS  12
7          Ipanema Agricola S.A  11
8            ROBERTO MONTERROSO  10
9                 AMILCAR LAPOLA   9
10                     LA PLATA   9
```

```r
library(dplyr)

top_10_companies <- coffee_ratings |>
  count(company, sort = TRUE) |>
  slice_max(n, n = 10)

print(top_10_companies)
```

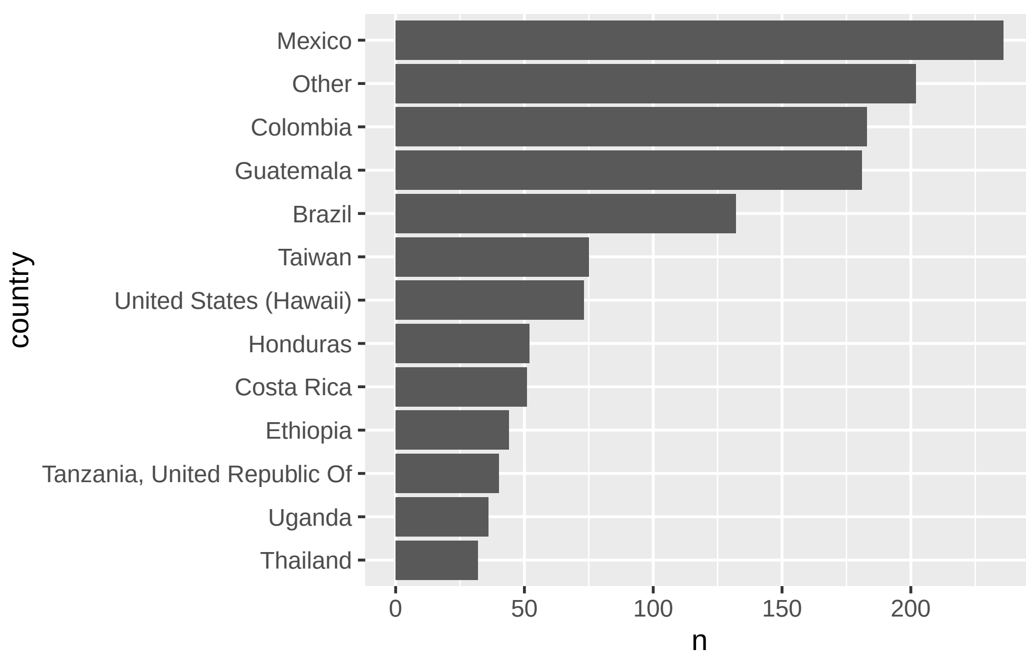
```
                            company   n
1                              <NA> 209
2              unex guatemala, s.a.  86
3                   ipanema coffees  50
4     exportadora de cafe condor s.a  40
5  kona pacific farmers cooperative  40
6                 racafe & cia s.c.a  40
7           blossom valley    25
8                      carcafe ltda  25
```

```
9                           nucoffee   24
10          taiwan coffee laboratory   20
```

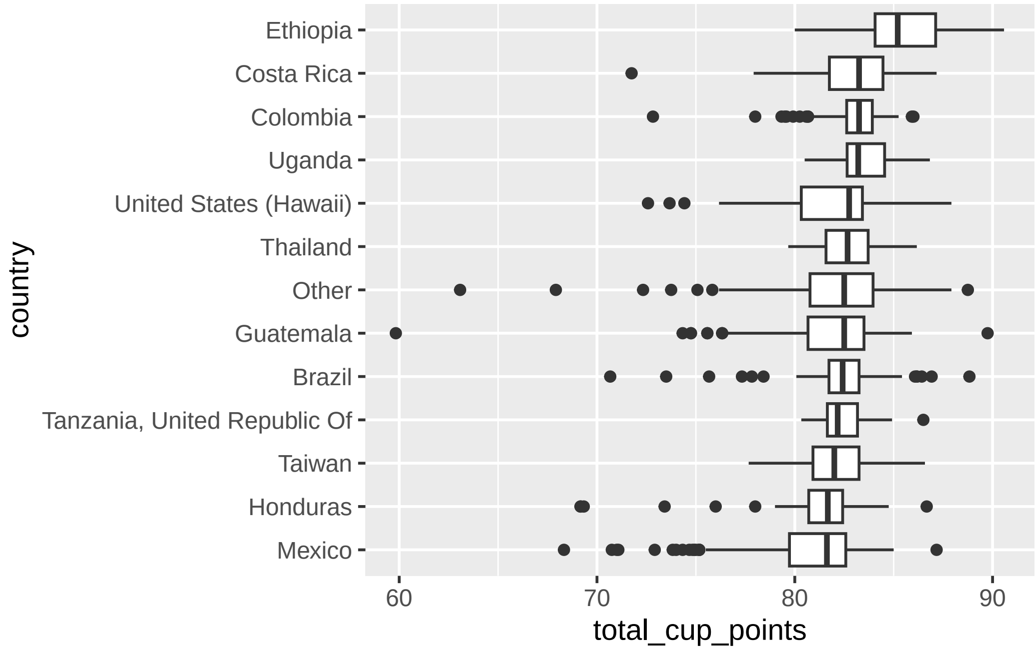
```
coffee_ratings |>
  count(color, sort = TRUE)
```

```
          color   n
1         Green 869
2          <NA> 218
3 Bluish-Green 114
4   Blue-Green  85
5          None  52
```

```
coffee_ratings |>
  count(country= fct_lump(country_of_origin, 12), sort = TRUE) |>
  filter(!is.na(country)) |>
  mutate(country = fct_reorder(country,n)) |>
  ggplot(aes(n,country)) +
  geom_col()
```

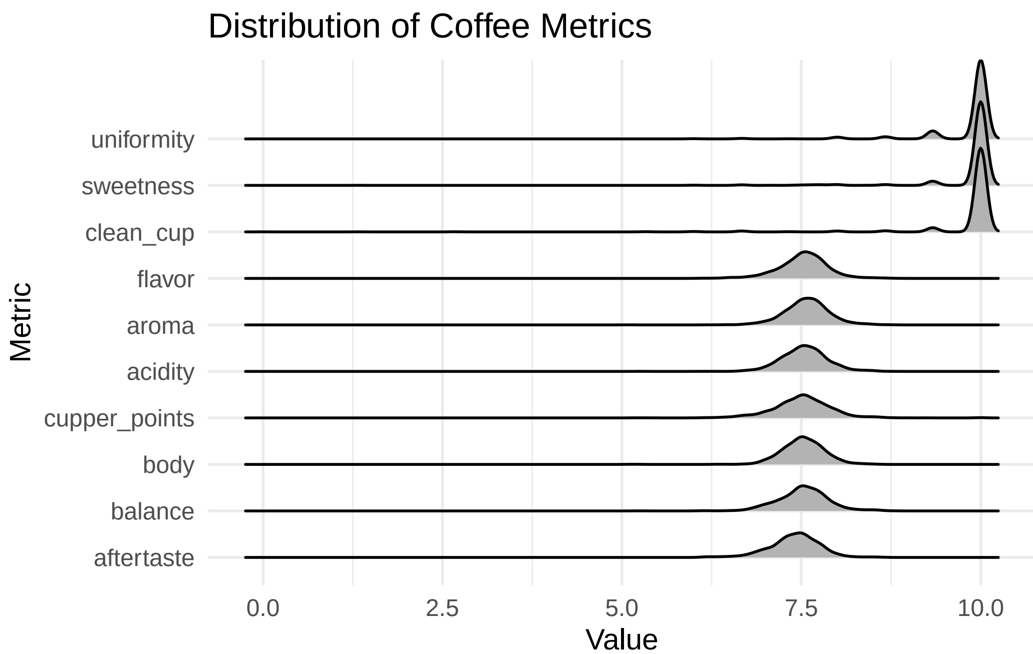


```
coffee_ratings |>
  filter(!is.na(country_of_origin)) |>
  mutate(country = fct_lump(country_of_origin, 12),
         country = fct_reorder(country, total_cup_points)) |>
  ggplot(aes(total_cup_points, country)) +
  geom_boxplot()
```




```
# Assuming coffee_ratings is your dataset
coffee_metrics <- coffee_ratings |>
  select(coffee_id, total_cup_points, variety, company, country_of_origin, altitude_mean_mete
         uniformity, sweetness, clean_cup, flavor, aroma, acidity, cupper_points, body, balan
  pivot_longer(cols = uniformity:aftertaste, names_to = "metric", values_to = "value")

coffee_metrics |>
  mutate(metric = fct_reorder(metric, value)) |>
  ggplot(aes(x = value, y = metric)) +
  geom_density_ridges() +
  labs(title = "Distribution of Coffee Metrics",
       x = "Value",
       y = "Metric") +
  theme_minimal()
```

```
Picking joint bandwidth of 0.0814
```

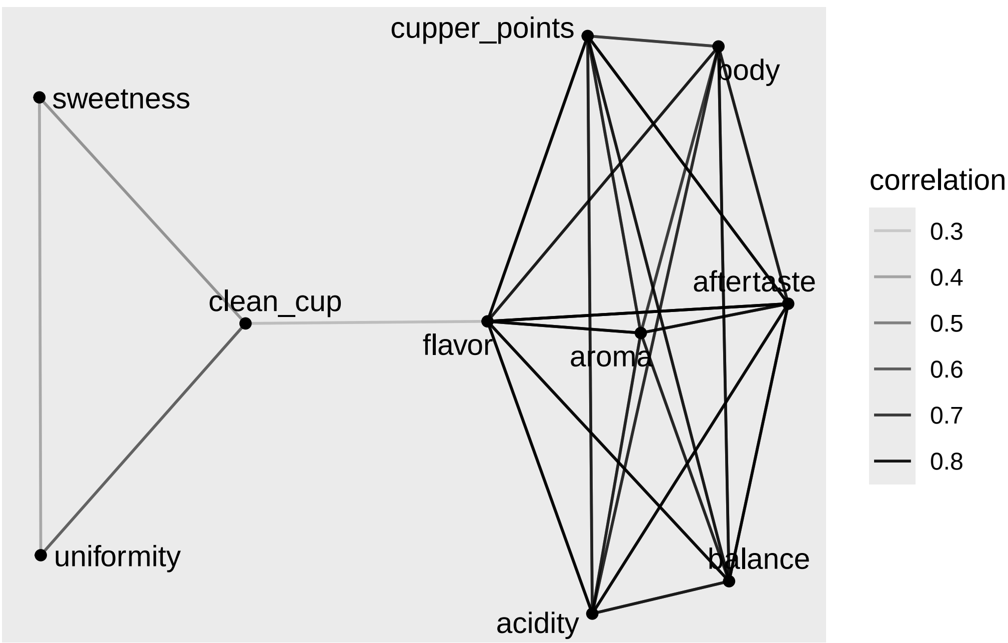



```r
coffee_metrics |>
  group_by(metric) |>
  summarize(average = mean(value), sd = sd(value)) |>
  arrange(desc(average))
```

```
# A tibble: 10 x 3
   metric        average    sd
   <chr>           <dbl> <dbl>
 1 sweetness        9.86 0.554
 2 clean_cup        9.84 0.715
 3 uniformity       9.84 0.485
 4 aroma            7.57 0.316
 5 acidity          7.54 0.319
 6 flavor           7.53 0.341
 7 balance          7.52 0.354
 8 body             7.52 0.308
 9 cupper_points    7.51 0.427
10 aftertaste       7.41 0.350
```

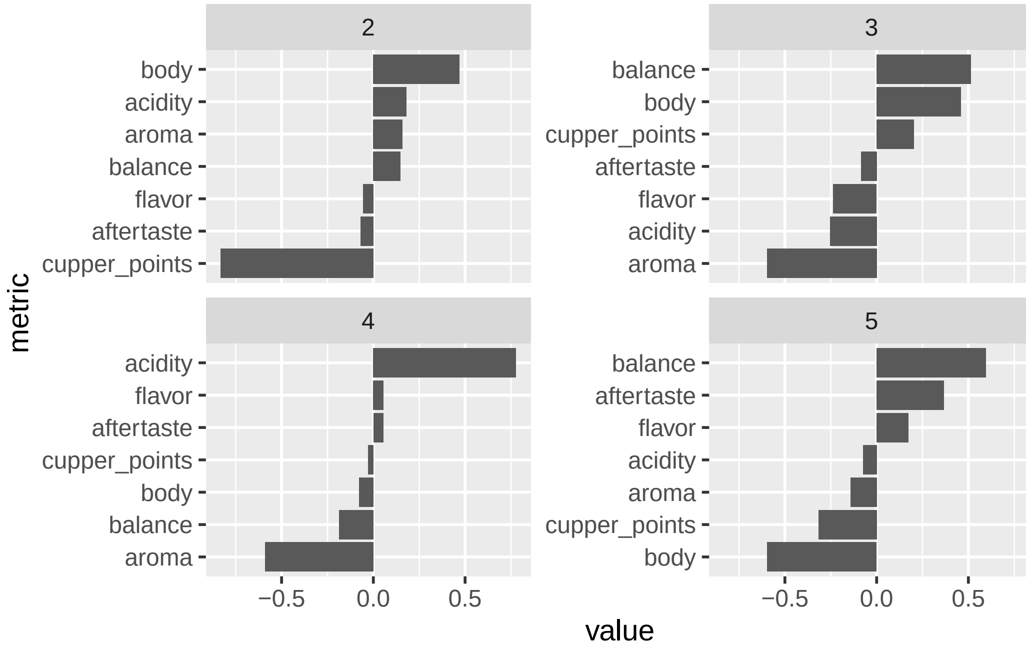
```
correlations <- coffee_metrics |>
  pairwise_cor(metric, coffee_id, value, sort = TRUE)
correlations |>
  head(50) |>
  graph_from_data_frame() |>
  ggraph() +
  geom_edge_link(aes(edge_alpha = correlation)) +
  geom_node_point() +
  geom_node_text(aes(label = name), repel = TRUE)
```

```
Using "stress" as default layout
```



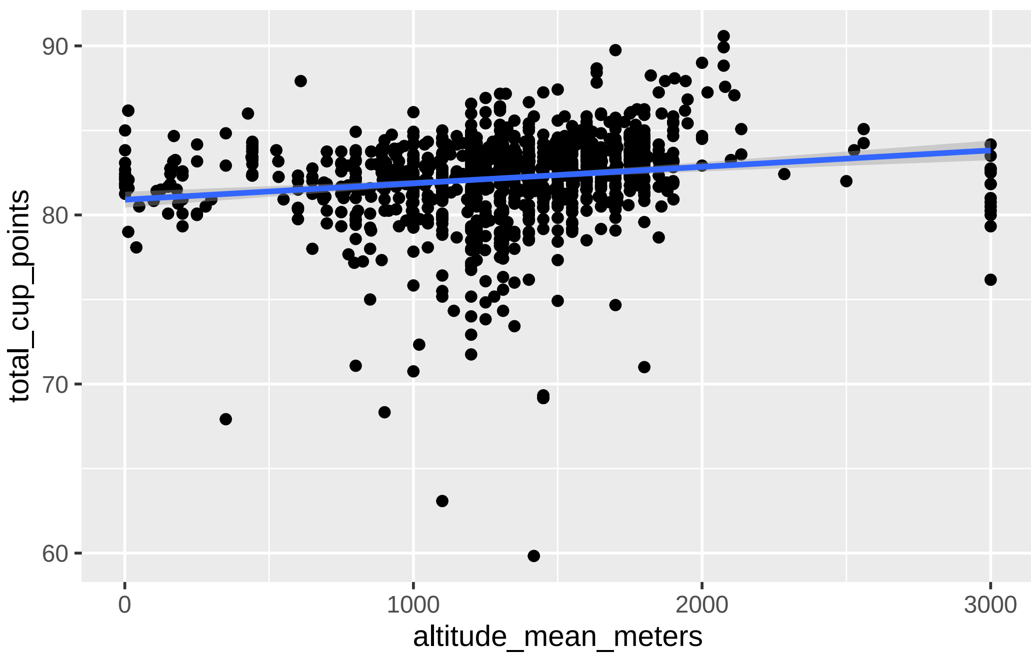

```
coffee_metrics |>
  filter(!metric %in% c("sweetness", "clean_cup", "uniformity")) |>
  group_by(metric) |>
  mutate(centered = value - mean(value)) |>
  ungroup() |>
  widely_svd(metric, coffee_id, value) |>
  filter(between(dimension, 2,5)) |>
  mutate(metric = reorder_within(metric, value, dimension)) |>
  ggplot(aes(value, metric)) +
```

```
  geom_col() +
  scale_y_reordered() +
  facet_wrap(~dimension, scales = "free_y")
```

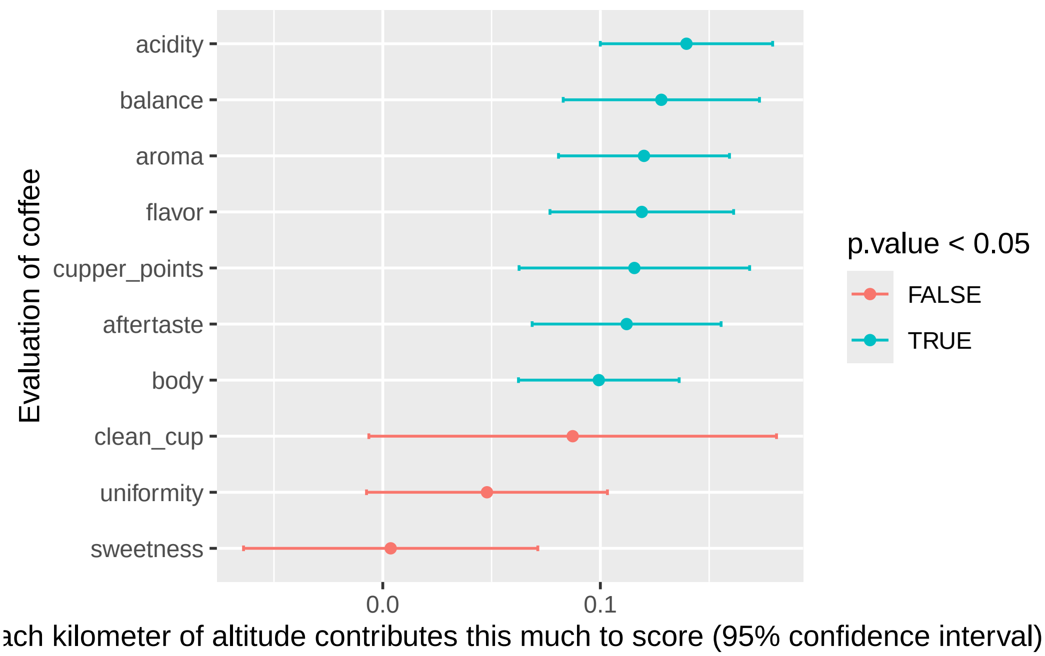



```
coffee_ratings |>
  filter(altitude_mean_meters < 10000) |>
  mutate(altitude_mean_meters = pmin(altitude_mean_meters, 3000)) |>
  ggplot(aes(altitude_mean_meters, total_cup_points)) +
  geom_point() +
  geom_smooth(method = "lm")
```

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




```
coffee_metrics <- coffee_metrics %>%
  filter(altitude_mean_meters < 10000) %>%
  mutate(altitude_mean_meters = pmin(altitude_mean_meters, 3000)) %>%
  mutate(km = altitude_mean_meters / 1000) %>%
  group_by(metric) %>%
  summarize(correlation = cor(altitude_mean_meters, value),
            model = list(lm(value ~ km))) %>%
  mutate(tidied = map(model, broom::tidy, conf.int = TRUE)) %>%
  unnest(tidied) %>%
  filter(term == "km") %>%
  ungroup() %>%
  mutate(metric = fct_reorder(metric, estimate)) %>%
  ggplot(aes(estimate, metric, color = p.value < .05)) +
  geom_point() +
  geom_errorbarh(aes(xmin = conf.low, xmax = conf.high), height = .1) +
  labs(y = "Evaluation of coffee",
       x = "Each kilometer of altitude contributes this much to score (95% confidence interva

print(coffee_metrics)
```



Part 3 [10 points]

Based on the dataset, it appears that variables such as species, country of origin, altitude, region, and producer could influence coffee quality.

Part 4 [10 points]

```
library(dplyr)
median_coffee_quality <- coffee_ratings |>
  group_by(country_of_origin) |>
  summarize(median_coffee_quality = median(total_cup_points, na.rm = TRUE)) |>
  arrange(desc(median_coffee_quality))
median_coffee_quality
```

```
# A tibble: 37 x 2
   country_of_origin median_coffee_quality
   <chr>                             <dbl>
 1 United States                      86.6
 2 Papua New Guinea                   85.8
 3 Ethiopia                           85.2
 4 Japan                              84.7
 5 Kenya                              84.6
 6 Panama                             84.1
 7 Colombia                           83.2
```

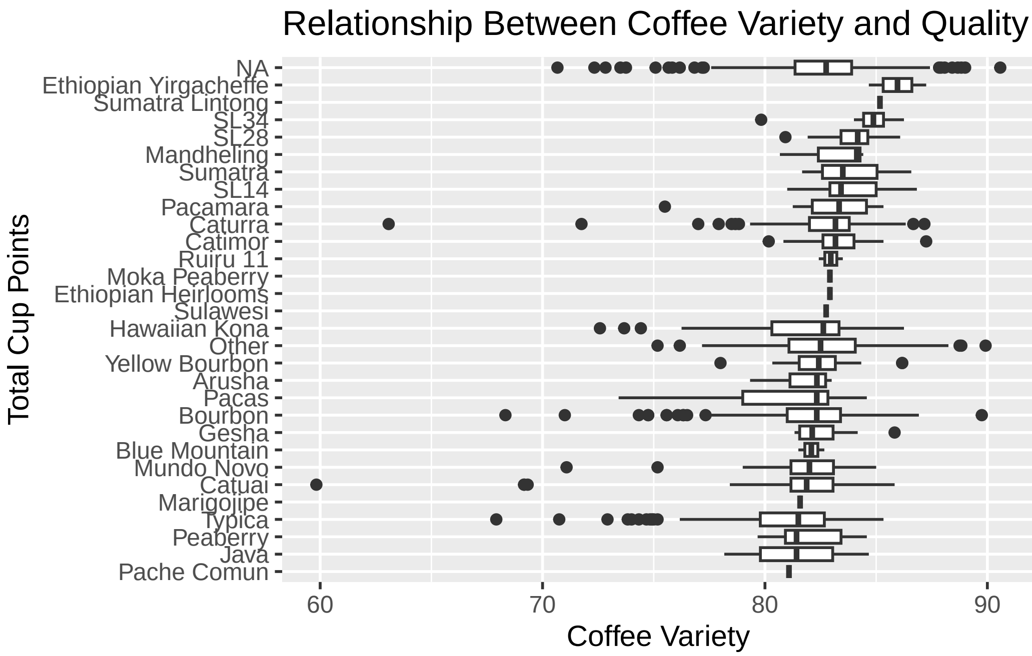
```
 8 Costa Rica                        83.2
 9 Uganda                            83.2
10 China                             83.2
# i 27 more rows
```

```r
head(median_coffee_quality, 3)
```

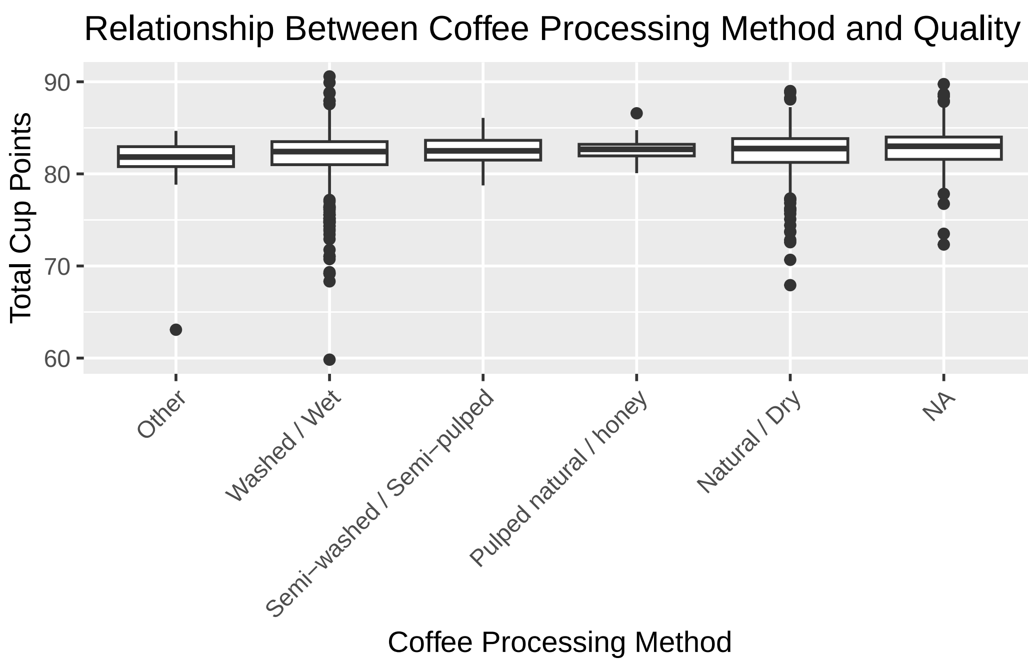
```
# A tibble: 3 x 2
  country_of_origin median_coffee_quality
  <chr>                             <dbl>
1 United States                      86.6
2 Papua New Guinea                   85.8
3 Ethiopia                           85.2
```

Part 5 [10 points]

```r
coffee_ratings|>
  mutate(variety = fct_reorder(variety, total_cup_points)) |>
  ggplot(aes(total_cup_points,variety)) +
  geom_boxplot() +
    labs(
    title = "Relationship Between Coffee Variety and Quality",
    x = "Coffee Variety",
    y = "Total Cup Points"
)
```



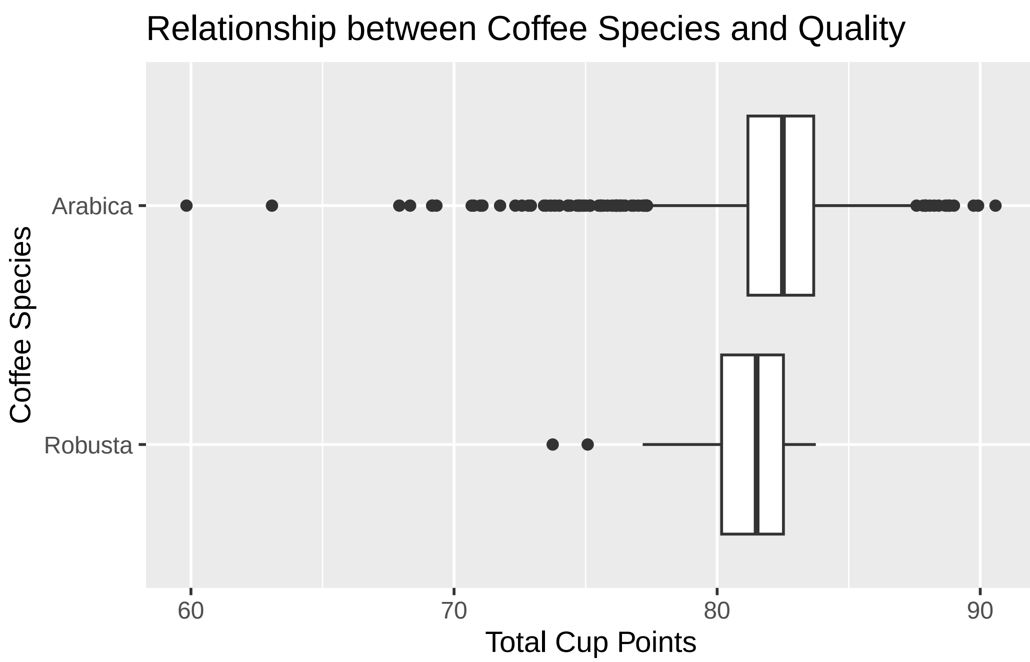

```r
library(dplyr)
library(ggplot2)
library(forcats)

coffee_ratings |>
  mutate(processing_method = fct_reorder(processing_method, total_cup_points)) |>
  ggplot(aes(x = processing_method, y = total_cup_points)) +
  geom_boxplot() +
  labs(
    title = "Relationship Between Coffee Processing Method and Quality",
    x = "Coffee Processing Method",
    y = "Total Cup Points"
  ) +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
```



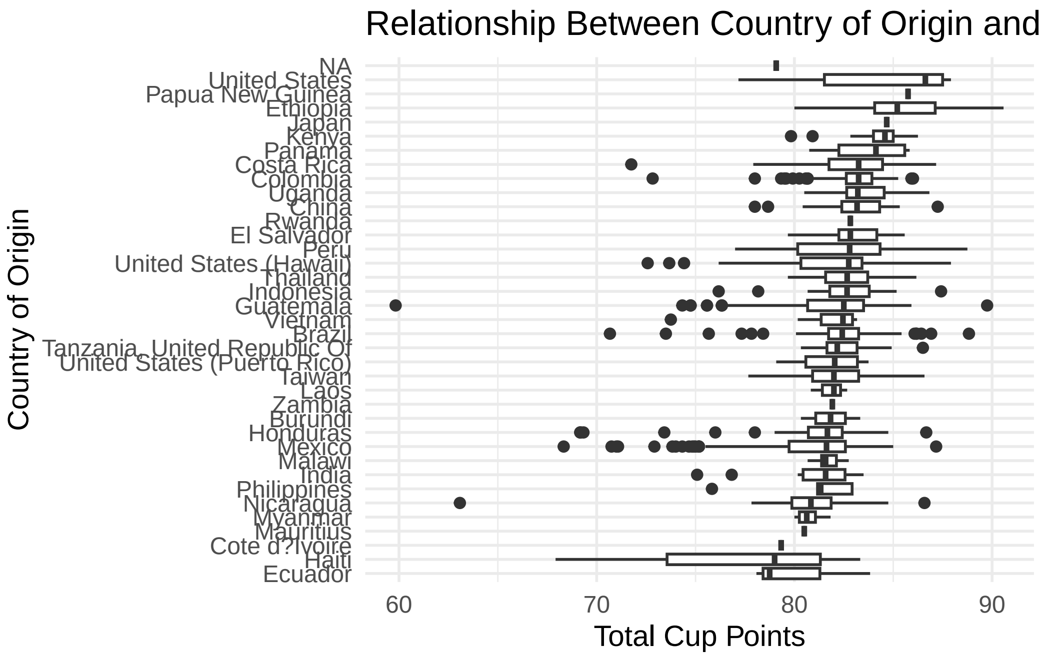

```
unique(coffee_ratings$species)
```

```
[1] "Arabica" "Robusta"
```

```
# graph 1
coffee_ratings|>
  mutate(species = fct_reorder(species, total_cup_points)) |>
  ggplot(aes(total_cup_points,species)) +
  geom_boxplot() +
  labs(title = "Relationship between Coffee Species and Quality",
       y = "Coffee Species",
       x = "Total Cup Points")
```




```r
library(dplyr)
library(ggplot2)
library(forcats)

coffee_ratings |>
  mutate(country_of_origin = fct_reorder(country_of_origin, total_cup_points)) |>
  ggplot(aes(x = total_cup_points, y = country_of_origin)) +
  geom_boxplot() +
  labs(
    title = "Relationship Between Country of Origin and Coffee Quality",
    x = "Total Cup Points",
    y = "Country of Origin"
  ) +
  theme_minimal()
```




```
3
```

```
[1] 3
```

Graph 1: Relationship between Coffee Species and Quality

Importance:

Understanding the relationship between coffee species (Arabica and Robusta) and their quality scores provides valuable insights into the differences in quality between these two major coffee types. This information can guide coffee producers, marketers, and consumers in making informed decisions about the coffee they grow, sell, or purchase.

Graph Description:

The graph depicts the distribution of total cup points for Arabica and Robusta coffee species. The box plots show the range of scores, including the median and interquartile range, for each species. - **Arabica**: The distribution shows a higher concentration of total cup points around the 75 mark, indicating generally higher quality scores. - **Robusta**: The distribution shows a lower range of total cup points, with many scores clustered below 75, indicating generally lower quality scores compared to Arabica.

Graph 2: Distribution of Coffee Quality by Country of Origin

Importance:

Analyzing the distribution of coffee quality scores by country of origin helps to identify regions that produce high-quality coffee. This information can be used by coffee importers, exporters, and consumers to source and appreciate coffee from different regions known for their quality.

Graph Description:

The graph shows the distribution of total cup points for coffee from various countries of origin, with box plots illustrating the range of scores for each country. - **Countries with High Scores**: Countries like Ethiopia, Kenya, and Colombia show high-quality scores, with most of their scores clustered around the upper range (close to 75). - **Countries with Varied Scores**: Some countries, such as Brazil and Mexico, show a wider range of scores, indicating variability in coffee quality. - **Outliers**: There are notable outliers with extremely high or low scores, highlighting exceptional cases of quality or lack thereof within certain countries.

These graphs provide a comprehensive overview of how coffee species and country of origin influence coffee quality, aiding stakeholders in the coffee industry in their decision-making processes.

Question 2 UN Voting: all three datasets

First Research Question: What type of policy does the UN vote most on?

```r
# Load Required Libraries
library(lubridate)

# Loading Dataset
votes <- read_csv("/Users/marcusnogueira/Library/Mobile Documents/com~apple~CloudDocs/812 S24
```

```
Rows: 869937 Columns: 4
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (3): country, country_code, vote
dbl (1): rcid

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
issues <- read_csv("/Users/marcusnogueira/Library/Mobile Documents/com~apple~CloudDocs/812 S
```

```
Rows: 5745 Columns: 3
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (2): short_name, issue
dbl (1): rcid

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
rollcall <- read_csv("/Users/marcusnogueira/Library/Mobile Documents/com~apple~CloudDocs/812
```

```
Rows: 6202 Columns: 9
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (3): unres, short, descr
dbl  (5): rcid, session, importantvote, amend, para
date (1): date

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 duplicates
votes <- votes %>%
  distinct(rcid, country, .keep_all = TRUE)

issues <- issues %>%
  distinct(rcid, .keep_all = TRUE)

rollcall <- rollcall %>%
  distinct(rcid, .keep_all = TRUE)

# Combine all three datasets, ensuring that the `country` column is preserved
main_data <- votes %>%
  left_join(issues, by = "rcid") %>%
  left_join(rollcall, by = "rcid")

# Check the structure of the combined data
str(main_data)
```

```
tibble [869,937 x 14] (S3: tbl_df/tbl/data.frame)
 $ rcid         : num [1:869937] 3 3 3 3 3 3 3 3 3 3 ...
 $ country      : chr [1:869937] "United States" "Canada" "Cuba" "Haiti" ...
 $ country_code : chr [1:869937] "US" "CA" "CU" "HT" ...
 $ vote         : chr [1:869937] "yes" "no" "yes" "yes" ...
 $ short_name   : chr [1:869937] NA NA NA NA ...
 $ issue        : chr [1:869937] NA NA NA NA ...
 $ session      : num [1:869937] 1 1 1 1 1 1 1 1 1 1 ...
 $ importantvote: num [1:869937] 0 0 0 0 0 0 0 0 0 0 ...
 $ date         : Date[1:869937], format: "1946-01-01" "1946-01-01" ...
 $ unres        : chr [1:869937] "R/1/66" "R/1/66" "R/1/66" "R/1/66" ...
 $ amend        : num [1:869937] 1 1 1 1 1 1 1 1 1 1 ...
 $ para         : num [1:869937] 0 0 0 0 0 0 0 0 0 0 ...
 $ short        : chr [1:869937] "AMENDMENTS, RULES OF PROCEDURE" "AMENDMENTS, RULES OF PROCE
 $ descr        : chr [1:869937] "TO ADOPT A CUBAN AMENDMENT TO THE UK PROPOSAL REFERRING THE
```

```
# List unique countries in the dataset
unique_countries <- unique(main_data$country)

# Display the unique countries
print(unique_countries)
```

```
  [1] "United States"                    "Canada"
  [3] "Cuba"                             "Haiti"
  [5] "Dominican Republic"               "Mexico"
  [7] "Guatemala"                        "Honduras"
  [9] "El Salvador"                      "Nicaragua"
 [11] "Costa Rica"                       "Panama"
 [13] "Colombia"                         "Venezuela"
 [15] "Ecuador"                          "Peru"
 [17] "Brazil"                           "Bolivia"
 [19] "Paraguay"                         "Chile"
 [21] "Argentina"                        "Uruguay"
 [23] "United Kingdom"                   "Netherlands"
 [25] "Belgium"                          "Luxembourg"
 [27] "France"                           "Poland"
 [29] "Czechoslovakia"                   "Yugoslavia"
 [31] "Greece"                           "Russia"
 [33] "Ukraine"                          "Belarus"
 [35] "Norway"                           "Denmark"
 [37] "Liberia"                          "Ethiopia"
 [39] "South Africa"                     "Iran"
```

```
 [41] "Turkey"                           "Iraq"
 [43] "Egypt"                            "Syria"
 [45] "Lebanon"                          "Saudi Arabia"
 [47] "Taiwan"                           "India"
 [49] "Philippines"                      "Australia"
 [51] "New Zealand"                      "Sweden"
 [53] "Iceland"                          "Afghanistan"
 [55] "Thailand"                         "Yemen Arab Republic"
 [57] "Pakistan"                         "Myanmar (Burma)"
 [59] "Israel"                           "Indonesia"
 [61] "Hungary"                          "Jordan"
 [63] "Sri Lanka"                        "Spain"
 [65] "Romania"                          "Ireland"
 [67] "Portugal"                         "Austria"
 [69] "Italy"                            "Albania"
 [71] "Bulgaria"                         "Finland"
 [73] "Libya"                            "Nepal"
 [75] "Cambodia"                         "Laos"
 [77] "Sudan"                            "Morocco"
 [79] "Tunisia"                          "Japan"
 [81] "Ghana"                            "Malaysia"
 [83] "Guinea"                           "Cyprus"
 [85] "Mali"                             "Senegal"
 [87] "Benin"                            "Niger"
 [89] "Côte d’Ivoire"                    "Burkina Faso"
 [91] "Togo"                             "Cameroon"
 [93] "Gabon"                            "Central African Republic"
 [95] "Chad"                             "Congo - Brazzaville"
 [97] "Madagascar"                       "Somalia"
 [99] "Nigeria"                          "Congo - Kinshasa"
[101] "Sierra Leone"                     "Mongolia"
[103] "Mauritania"                       "Tanzania"
[105] "Jamaica"                          "Burundi"
[107] "Rwanda"                           "Trinidad & Tobago"
[109] "Algeria"                          "Uganda"
[111] "Kuwait"                           "Kenya"
[113] "Zanzibar"                         "Malta"
[115] "Zambia"                           "Malawi"
[117] "Maldives"                         "Singapore"
[119] "Gambia"                           "Guyana"
[121] "Lesotho"                          "Botswana"
[123] "Barbados"                         "Yemen People's Republic"
[125] "Mauritius"                        "Equatorial Guinea"
```

```
[127] "Eswatini"                               "Fiji"
[129] "Bhutan"                                 "Bahrain"
[131] "Qatar"                                  "Oman"
[133] "China"                                  "United Arab Emirates"
[135] "Federal Republic of Germany"            "German Democratic Republic"
[137] "Bahamas"                                "Bangladesh"
[139] "Grenada"                                "Guinea-Bissau"
[141] "Cape Verde"                             "São Tomé & Príncipe"
[143] "Mozambique"                             "Comoros"
[145] "Papua New Guinea"                       "Suriname"
[147] "Angola"                                 "Djibouti"
[149] "Vietnam"                                "Samoa"
[151] "Seychelles"                             "Solomon Islands"
[153] "St. Lucia"                              "Zimbabwe"
[155] "St. Vincent & Grenadines"               "Vanuatu"
[157] "Belize"                                 "Antigua & Barbuda"
[159] "Dominica"                               "St. Kitts & Nevis"
[161] "Brunei"                                 "Liechtenstein"
[163] "Namibia"                                "Germany"
[165] "Estonia"                                "Latvia"
[167] "Lithuania"                              "Yemen"
[169] "North Korea"                            "South Korea"
[171] "Micronesia (Federated States of)" "Marshall Islands"
[173] "San Marino"                             "Bosnia & Herzegovina"
[175] "Armenia"                                "Azerbaijan"
[177] "Croatia"                                "Slovenia"
[179] "Moldova"                                "Turkmenistan"
[181] "Kyrgyzstan"                             "Kazakhstan"
[183] "Tajikistan"                             "Monaco"
[185] "Andorra"                                "Czechia"
[187] "Slovakia"                               "North Macedonia"
[189] "Eritrea"                                "Georgia"
[191] "Uzbekistan"                             "Palau"
[193] "Tonga"                                  "Nauru"
[195] "Kiribati"                               "Tuvalu"
[197] "Switzerland"                            "Timor-Leste"
[199] "Montenegro"                             "South Sudan"
```

```r
# Count the total number of unique countries in the dataset
total_unique_countries <- length(unique(main_data$country))

# Display the total number of unique countries
```

```r
print(total_unique_countries)
```

```
[1] 200
```

```r
# Question 1: What type of policy does the UN vote most on?

# Count the number of votes by issue category
policy_counts <- main_data %>%
  group_by(issue) %>%
  summarize(count = n()) %>%
  arrange(desc(count))

# Display the policy counts
print(policy_counts)
```

```
# A tibble: 7 x 2
  issue                                 count
  <chr>                                 <int>
1 <NA>                                 265456
2 Palestinian conflict                 158656
3 Nuclear weapons and nuclear material 126849
4 Human rights                         111480
5 Arms control and disarmament          79464
6 Economic development                  66925
7 Colonialism                           61107
```

```r
# Extract year from date and count policies by year
main_data <- main_data %>%
  mutate(year = year(ymd(date)))

# Filter out rows with NA in the issue column
main_data_filtered <- main_data %>%
  filter(!is.na(issue))

# Count the number of votes by year and issue category
policy_counts_by_year <- main_data_filtered %>%
  group_by(year, issue) %>%
  summarize(count = n(), .groups = 'drop') %>%
  arrange(desc(count))
```

```r
# Display the policy counts by year
print(policy_counts_by_year)
```

```
# A tibble: 394 x 3
    year issue                              count
   <dbl> <chr>                              <int>
 1  1983 Palestinian conflict                5146
 2  1982 Palestinian conflict                4986
 3  1984 Palestinian conflict                4825
 4  1981 Palestinian conflict                4773
 5  2017 Nuclear weapons and nuclear material  4612
 6  1985 Palestinian conflict                4594
 7  1987 Palestinian conflict                4575
 8  1989 Palestinian conflict                4494
 9  1988 Palestinian conflict                4356
10  1986 Palestinian conflict                4354
# i 384 more rows
```

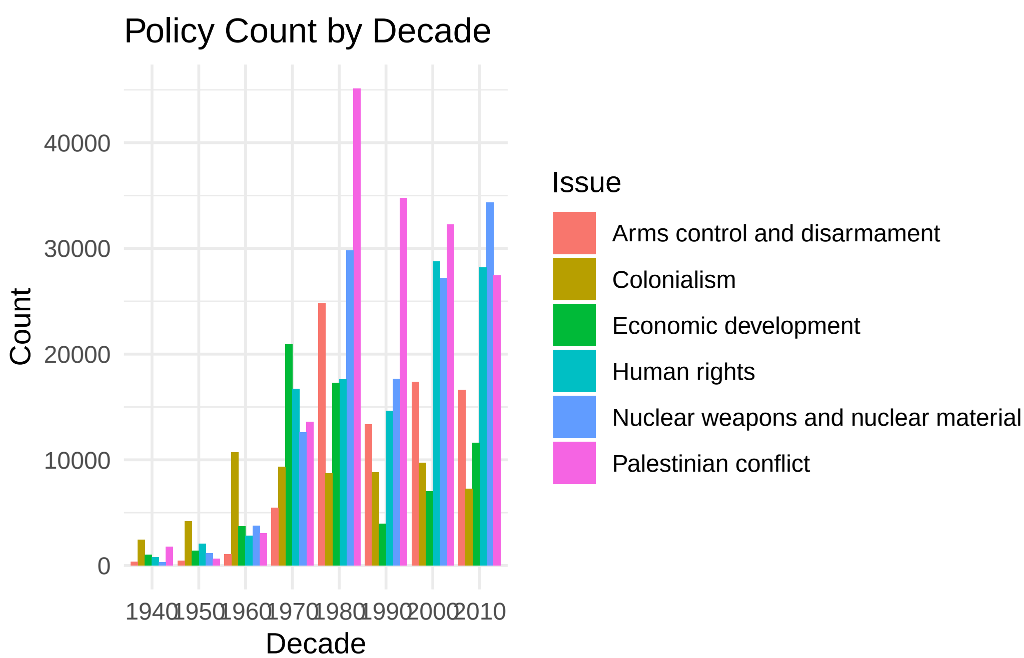
```r
#I dropped NA's for issue column
```

```r
# At first I plotted by year, it was too messy so now I'm doing it by decade
main_data <- main_data %>%
  mutate(year = year(ymd(date)),
         decade = floor(year / 10) * 10)

# Filter out rows with NA in the issue column
main_data_filtered <- main_data %>%
  filter(!is.na(issue))

# Count the number of votes by decade and issue category
policy_counts_by_decade <- main_data_filtered %>%
  group_by(decade, issue) %>%
  summarize(count = n(), .groups = 'drop') %>%
  arrange(decade, desc(count))

# Display the policy counts by decade
print(policy_counts_by_decade)
```

```
# A tibble: 48 x 3
   decade issue                              count
    <dbl> <chr>                              <int>
```

```
 1   1940 Colonialism                             2456
 2   1940 Palestinian conflict                    1793
 3   1940 Economic development                    1016
 4   1940 Human rights                             756
 5   1940 Arms control and disarmament             357
 6   1940 Nuclear weapons and nuclear material     322
 7   1950 Colonialism                             4167
 8   1950 Human rights                            2037
 9   1950 Economic development                    1406
10   1950 Nuclear weapons and nuclear material    1135
# i 38 more rows
```

```r
# Visualization of policy count by decade
ggplot(policy_counts_by_decade, aes(x = factor(decade), y = count, fill = issue)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Policy Count by Decade",
       x = "Decade",
       y = "Count",
       fill = "Issue") +
  theme_minimal()
```



Question 1: What type of policy does the UN vote most on?

Importance:

Understanding the types of policies the UN votes on most frequently is crucial for identifying global priorities and areas of focus. This analysis provides insights into the collective concerns and issues that the international community addresses through the UN. It helps policymakers, researchers, and the public to track trends in international relations and understand the efforts made towards global governance.

Graph Description:

The graph displays the count of votes on different policy issues by decade. The issues include Arms control and disarmament, Colonialism, Economic development, Human rights, Nuclear weapons and nuclear material, and Palestinian conflict. Each bar represents the number of votes on a particular issue in a given decade.

Key Observations:

1. **Arms Control and Disarmament**:
   - This issue has seen a consistent number of votes over the decades, with notable peaks in the 1980s and early 2000s.
   - **Observation**: Reflects ongoing international efforts to regulate and control arms, particularly during periods of heightened global tension.
2. **Colonialism**:
   - Voting on colonialism peaked significantly in the 1960s, coinciding with the decolonization movements across Africa and Asia.
   - **Observation**: Indicates the global focus on ending colonial rule and supporting newly independent nations during that period.
3. **Economic Development**:
   - Votes on economic development have been consistently high, especially from the 1970s onwards.
   - **Observation**: Highlights the importance of economic issues and the UN's role in promoting global economic stability and development.
4. **Human Rights**:
   - Human rights issues have seen a steady increase in votes, peaking in the 1990s and 2000s.
   - **Observation**: Reflects the growing global emphasis on protecting and promoting human rights across different regions and contexts.

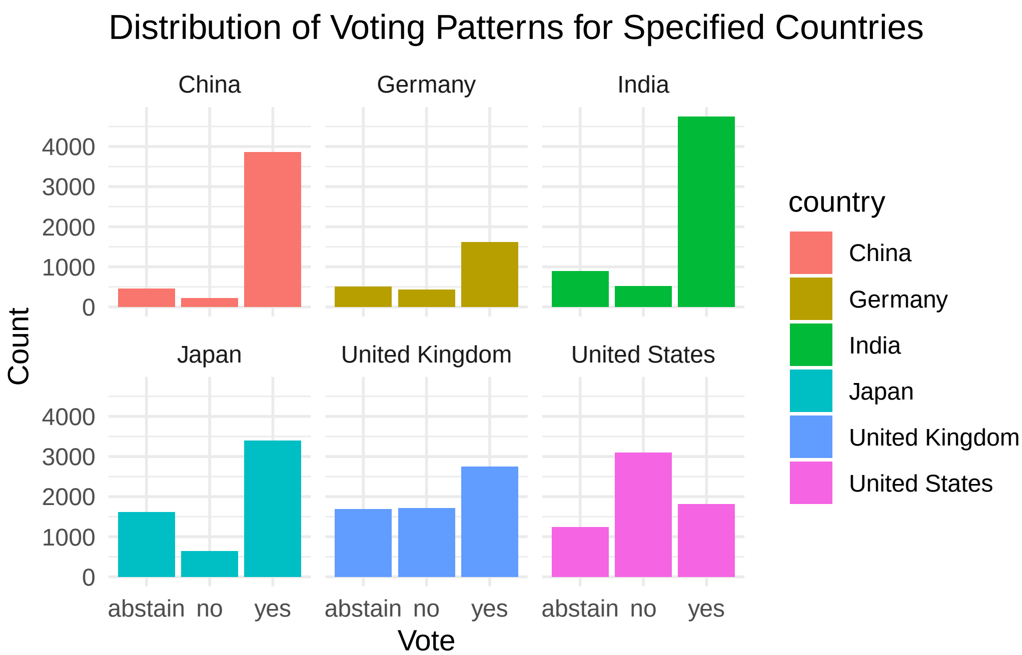
5. **Nuclear Weapons and Nuclear Material**:
   - This issue has consistently been a focus, with significant voting activity in the 1980s and early 2000s.
   - **Observation**: Indicates the persistent concern about nuclear proliferation and the efforts to regulate nuclear materials.
6. **Palestinian Conflict**:
   - Voting on the Palestinian conflict has seen significant activity, especially from the 1970s to the present.
   - **Observation**: Reflects the ongoing and complex nature of the Israeli-Palestinian conflict and the international community's involvement in seeking resolutions.

```r
# Question 2: Distribution of Voting Patterns for Specified Countries
 #Filter data for the specified countries
countries <- c("United States", "United Kingdom", "Germany", "China", "India", "Japan")

country_votes <- main_data %>%
  filter(country %in% countries)

# Create a distribution of voting patterns for specified countries
voting_patterns <- country_votes %>%
  group_by(country, vote) %>%
  summarize(count = n(), .groups = 'drop') %>%
  spread(vote, count, fill = 0)

# Display the voting patterns
print(voting_patterns)
```

```
# A tibble: 6 x 4
  country        abstain    no   yes
  <chr>            <dbl> <dbl> <dbl>
1 China              452   222  3857
2 Germany            504   428  1619
3 India              891   516  4747
4 Japan             1613   644  3397
5 United Kingdom    1685  1709  2746
6 United States     1245  3100  1810
```

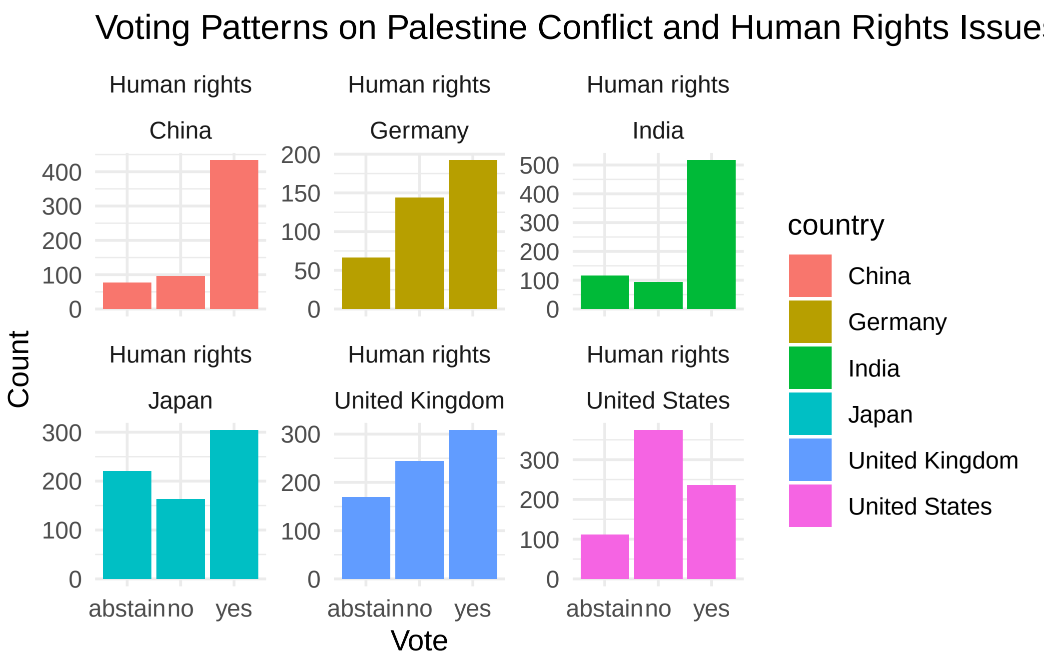
```r
# Visualization of voting patterns
ggplot(country_votes, aes(x = vote, fill = country)) +
  geom_bar(position = "dodge") +
```

```
  facet_wrap(~ country) +
  labs(title = "Distribution of Voting Patterns for Specified Countries",
       x = "Vote",
       y = "Count") +
  theme_minimal()
```




```
# Filter data for the specified countries and specific issues
countries <- c("United States", "United Kingdom", "Germany", "China", "India", "Japan")
specific_issues <- c("Palestine conflict", "Human rights")

issue_specific_votes <- main_data %>%
  filter(country %in% countries, issue %in% specific_issues)

# Group by country, issue, and vote, then summarize the counts
issue_voting_patterns <- issue_specific_votes %>%
  group_by(country, issue, vote) %>%
  summarize(count = n(), .groups = 'drop') %>%
  arrange(country, issue, vote)

# Visualization of voting patterns for specific issues
ggplot(issue_voting_patterns, aes(x = vote, y = count, fill = country)) +
  geom_bar(stat = "identity", position = "dodge") +
```

```
  facet_wrap(~ issue + country, scales = "free_y") +
  labs(title = "Voting Patterns on Palestine Conflict and Human Rights Issues",
       x = "Vote",
       y = "Count") +
  theme_minimal()
```



Question 2: Distribution of Voting Patterns for Specified Countries

Analysis of Voting Patterns on Palestine Conflict and Human Rights Issues

The visualization displays the voting patterns for six major countries—China, Germany, India, Japan, the United Kingdom, and the United States—on two specific issues: "Palestine conflict" and "Human rights." The bars represent the counts of each type of vote: "yes," "no," and "abstain."

Key Observations:

1. **China**:

   - **Human Rights**: China predominantly votes "yes" with a significant number of abstentions and fewer "no" votes.

- **Observation**: This suggests strong support for human rights issues, although a notable portion of abstentions indicates some reservations or neutrality in certain cases.

2. **Germany**:
   - **Human Rights**: Germany shows a balanced pattern with a high number of "yes" votes, moderate "abstain" votes, and fewer "no" votes.
   - **Observation**: This reflects Germany's overall support for human rights issues with a relatively low level of opposition.

3. **India**:
   - **Human Rights**: India predominantly votes "yes," with a considerable number of abstentions and "no" votes.
   - **Observation**: This indicates a strong inclination towards supporting human rights issues while also maintaining a significant number of neutral or opposing stances.

4. **Japan**:
   - **Human Rights**: Japan has a high number of "yes" votes, followed by abstentions and fewer "no" votes.
   - **Observation**: This shows Japan's strong support for human rights issues with a balanced level of abstentions and opposition.

5. **United Kingdom**:
   - **Human Rights**: The UK exhibits a balanced distribution with a high number of "yes" votes, significant "abstain" votes, and a considerable number of "no" votes.
   - **Observation**: The UK's voting pattern indicates a strong support for human rights issues while also reflecting significant neutrality and opposition.

6. **United States**:
   - **Human Rights**: The US stands out with a high number of "no" votes, followed by "yes" and "abstain" votes.
   - **Observation**: This suggests a more critical stance towards certain human rights issues compared to the other countries, with a notable amount of opposition.

Importance:

Analyzing the voting patterns of these influential countries on crucial issues like the Palestine conflict and human rights provides valuable insights into their international stances and diplomatic strategies. It highlights their priorities, alliances, and areas of contention, contributing to a better understanding of global governance and international relations. This information is crucial for policymakers, researchers, and the public to gauge the global political landscape and the dynamics at play within the United Nations.