Task-04

Importing necessary library

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(knitr)
```

Loading data

```
users_file <- "../data/users.csv"
reviews_file <- "../data/reviews.csv"

users <- read.csv(users_file, stringsAsFactors = FALSE)
reviews <- read.csv(reviews_file, stringsAsFactors = FALSE)
```

Grouping

First, clean the reviews data by replacing missing text values (NA) with empty strings, then calculate and store the length of each review in a new column called review_length. Next, the reviews data is merged with the users data based on user_id using an inner join, resulting in *merged_data*. Finally, merged_data is filtered to create analysis_data_stars, which contains only the rows where the stars column does not have missing values (NA).

```
users$member_since <- ymd_hms(users$member_since, truncated = 3, quiet = TRUE) # Convert the 'member_si

users <- users[!is.na(users$member_since), ] # Remove rows where the member_since column has NA values

# Create user groups: "Before 2020" and "2020 and After"
users$user_group <- ifelse(users$member_since < as.Date("2020-01-01"),
                           "Before 2020",
                           "2020 and After")
users$user_group <- as.factor(users$user_group) # Convert the data type of the newly created user_group
```

Next, calculate the length of the text

```
reviews$text[is.na(reviews$text)] <- "" # eplace all NA (missing) values in the 'text' column of the 'r
reviews$review_length <- nchar(reviews$text) # creating new column for length of the text
merged_data <- inner_join(reviews, users, by = "user_id") # merge data using inner join
analysis_data_stars <- merged_data[!is.na(merged_data$stars), ]
```

Calculate summary statistics for star ratings *star* based on user groups *user_group* from the *analysis_data_stars* dataset. First, group the data by user_group. Then, for each group, compute the mean *mean_stars*, median *median_stars*, standard deviation *sd_stars* of the star ratings, and the number of reviews *count* in that group. Then, visualize the table using kable.

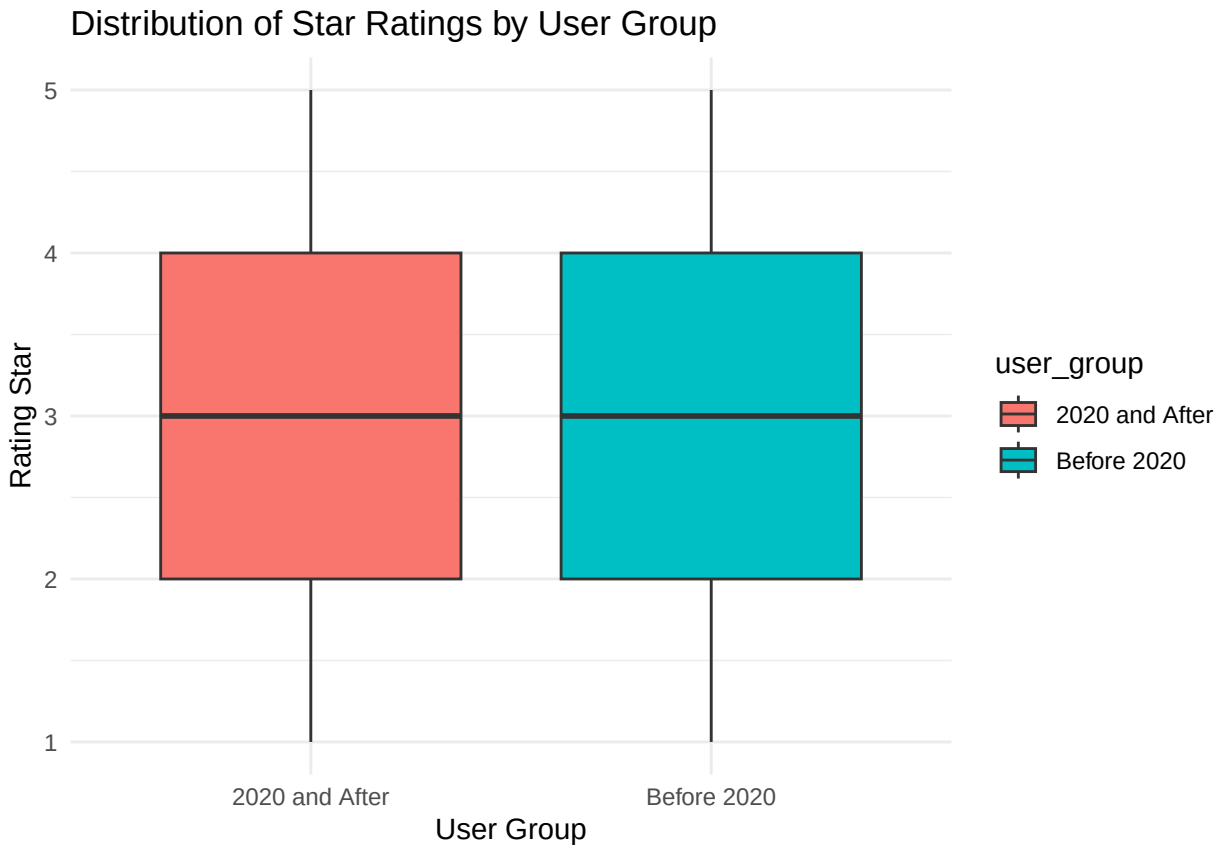
```
# Calculate average star rating by user group
avg_stars_by_group <- analysis_data_stars %>%
  group_by(user_group) %>%
  summarise(
    mean_stars = mean(stars),
    median_stars = median(stars),
    sd_stars = sd(stars),
    count = n()
  )

print(kable(avg_stars_by_group))
```

```
##
##
## |user_group     | mean_stars| median_stars| sd_stars| count|
## |:--------------|----------:|------------:|--------:|-----:|
## |2020 and After |   3.002351|            3| 1.414329| 99115|
## |Before 2020    |   2.997296|            3| 1.412799| 83937|
```

Visualization

Display the distribution of star ratings *stars* for each user group *user_group* from the *analysis_data_stars* dataset. This boxplot will show how the star ratings are distributed (median, quartiles, and outliers) for each *user_group*, with each group given a different fill color for easier identification.

```
star_rating_plot <- ggplot(analysis_data_stars, aes(x = user_group, y = stars, fill = user_group)) +
  geom_boxplot(na.rm = TRUE) +
  labs(
    title = "Distribution of Star Ratings by User Group",
    x = "User Group",
    y = "Rating Star"
  ) +
  theme_minimal()
print(star_rating_plot)
```

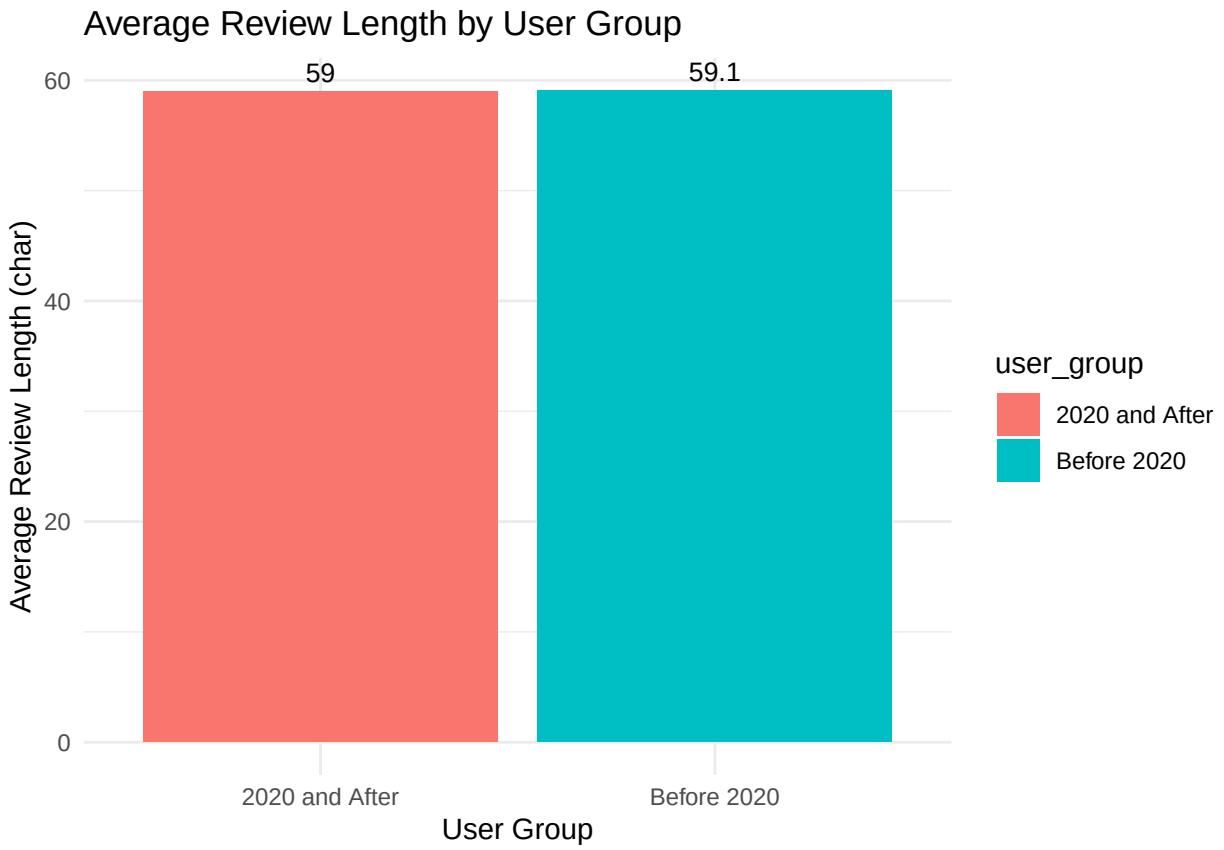



```r
# Calculate average review length by user group
# 'merged_data' can be used here as review_length=0 for NA text is fine.
avg_length_by_group <- merged_data %>%
  group_by(user_group) %>%
  summarise(
    mean_length = mean(review_length),
    median_length = median(review_length),
    sd_length = sd(review_length),
    count = n()
  )

print(kable(avg_length_by_group))
```

```
##
##
## |user_group     | mean_length| median_length| sd_length| count|
## |:--------------|-----------:|-------------:|---------:|-----:|
## |2020 and After |    58.98301|            50|  34.25794| 99115|
## |Before 2020    |    59.09321|            49|  34.64324| 83937|
```

```r
# Visualize average review length (as requested)
avg_length_plot <- ggplot(avg_length_by_group, aes(x = user_group, y = mean_length, fill = user_group)) +
  geom_bar(stat = "identity", position = position_dodge()) +
  geom_text(aes(label = round(mean_length, 1)), vjust = -0.5, position = position_dodge(width = 0.9), si
  labs(
    title = "Average Review Length by User Group",
    x = "User Group",
    y = "Average Review Length (char)"
```

```
  ) +
  theme_minimal()
print(avg_length_plot)
```



There is no statistically significant difference between the two groups.