Assignment 3.2

Mateus Parolin Gomes

01/30/2025

Task 1: Reflection

After reading Wilke's Chapter 2 and understanding the importance about aesthetics in data visualization highlight that poor design can mislead the audience. An example of how aesthetics influence perception was seen in our first session, when the professor made it seem like he ate a candle, but it was actually an apple shaped like a candle with an almond wick. This emphasizes how misleading or well-designed visual elements can shape our interpretation of data. In addition, in Hans Rosling's video he demonstrated an example of an aesthetic visualization, displaying lifespan and income of 200 countries over time, having population size as bubble sizes and giving context to trends and changes in the chart over time. His approach aligns with our discussion in the first lecture regarding truth, facts, and context, showing that data alone does not tell the full story. By combining visuals with context, Rosling effectively relates trends to their eventual causes.

Task 2: Lord of the Rings

Load and clean data

First we load, restructure, and clean the data.

```
# libraries
library(tidyverse)

# Loading datasets
fellowship <- read_csv("data/The_Fellowship_Of_The_Ring.csv")
tt <- read_csv("data/The_Two_Towers.csv")
rotk <- read_csv("data/The_Return_Of_The_King.csv")

# combine all datasets
lotr_wide <- bind_rows(fellowship, tt, rotk) %>%
  # Make the Film column a categorical variable (factor), and put it in the
  # order the categories appear (so the films are in the correct order)
  mutate(Film = fct_inorder(Film))

# Make this wide data tidy
lotr <- lotr_wide %>%
  # This is the new way to make data long
  pivot_longer(cols = c(Female, Male),
               names_to = "Gender", values_to = "Words")
```

Race

Does a certain race dominate the entire trilogy? (hint: group by `Race`)

```
# Group by Race and sum the words spoken
race_dominance <- lotr %>%
  group_by(Race) %>%
  summarise(Total_Words = sum(Words, na.rm = TRUE)) %>%
  arrange(desc(Total_Words))

# Extract the dominant race and its word count
dominant_race <- race_dominance$Race[1]
dominant_words <- race_dominance$Total_Words[1]

# Print the result
print(race_dominance)
```

```
## # A tibble: 3 × 2
##   Race   Total_Words
##   <chr>        <dbl>
## 1 Hobbit        8796
## 2 Man           8712
## 3 Elf           3737
```

```
# Print a conclusion message with dominant race and number of words
cat("\nThe race that dominates the trilogy is", dominant_race,
    "with a total of", dominant_words, "words spoken.\n")
```

```
##
## The race that dominates the trilogy is Hobbit with a total of 8796 words
spoken.
```

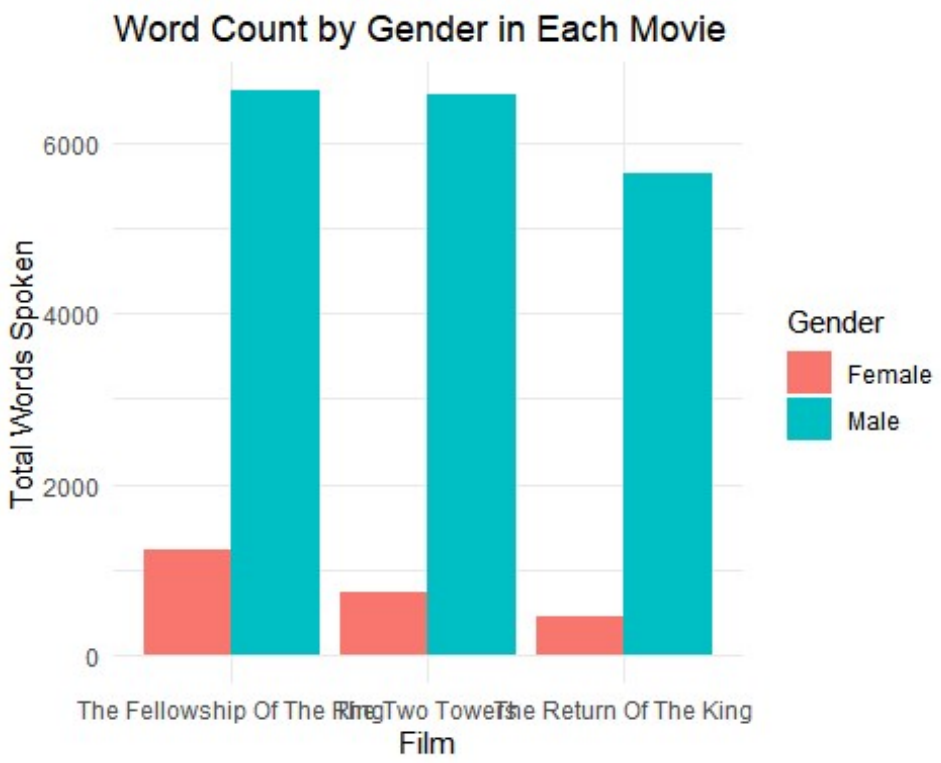
Gender and film

Does a certain gender dominate a movie? (lolz of course it does, but still, graph it) (Hint: group by both `Gender` and `Film`.) Experiment with filling by `Gender` or `Film` and faceting by `Gender` or `Film`.

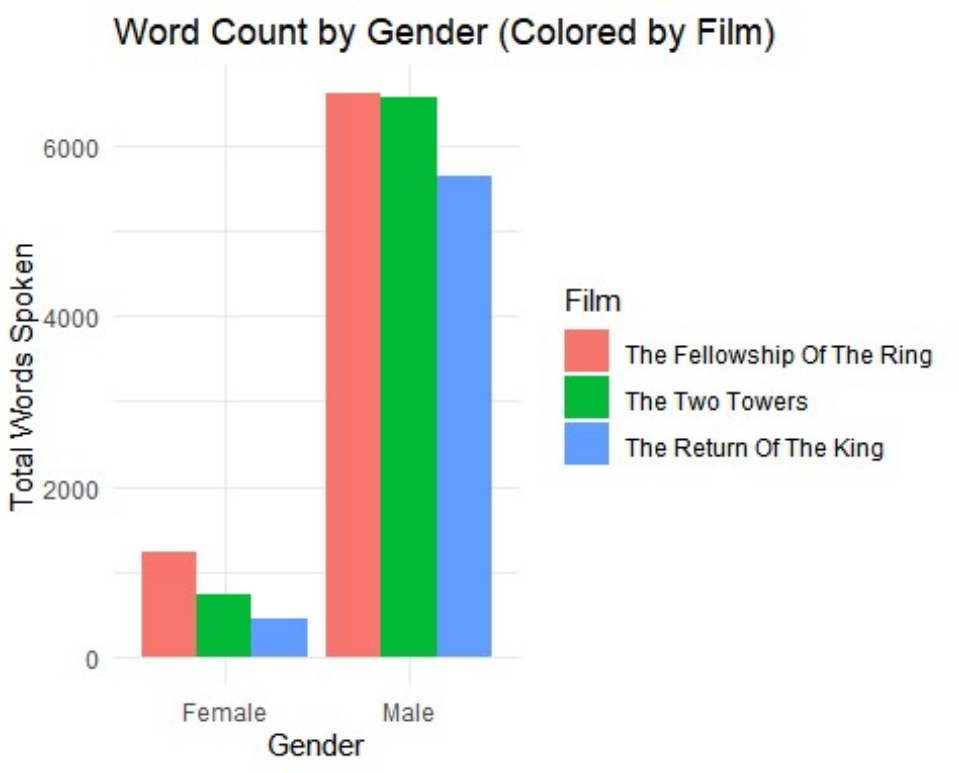
```
# Load library
library(ggplot2)

# Group by Gender and Film, then sum the words spoken
gender_film_summary <- lotr %>%
  group_by(Gender, Film) %>%
  summarise(Total_Words = sum(Words, na.rm = TRUE))
```

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

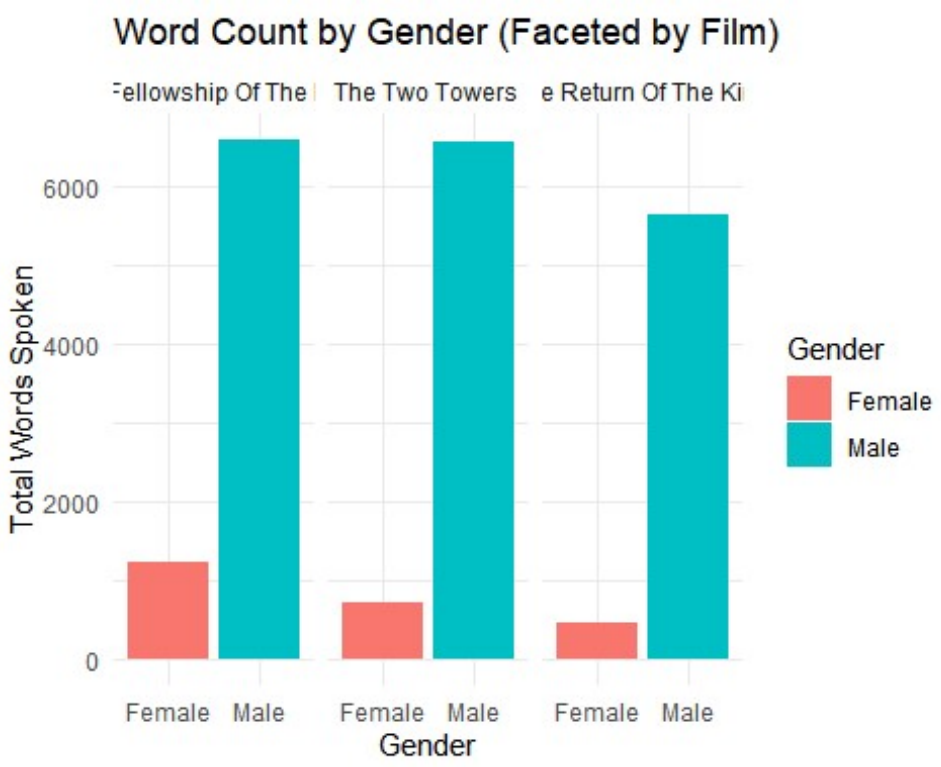
```r
# Create a bar plot
ggplot(gender_film_summary, aes(x = Film, y = Total_Words, fill = Gender)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Word Count by Gender in Each Movie",
       x = "Film",
       y = "Total Words Spoken") +
  theme_minimal()
```




```r
# Fill by Film instead of Gender

ggplot(gender_film_summary, aes(x = Gender, y = Total_Words, fill = Film)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Word Count by Gender (Colored by Film)",
       x = "Gender",
       y = "Total Words Spoken") +
  theme_minimal()
```




```r
# Facet by Film

ggplot(gender_film_summary, aes(x = Gender, y = Total_Words, fill = Gender))
+
  geom_bar(stat = "identity") +
  facet_wrap(~Film) +
  labs(title = "Word Count by Gender (Faceted by Film)",
       x = "Gender",
       y = "Total Words Spoken") +
  theme_minimal()
```



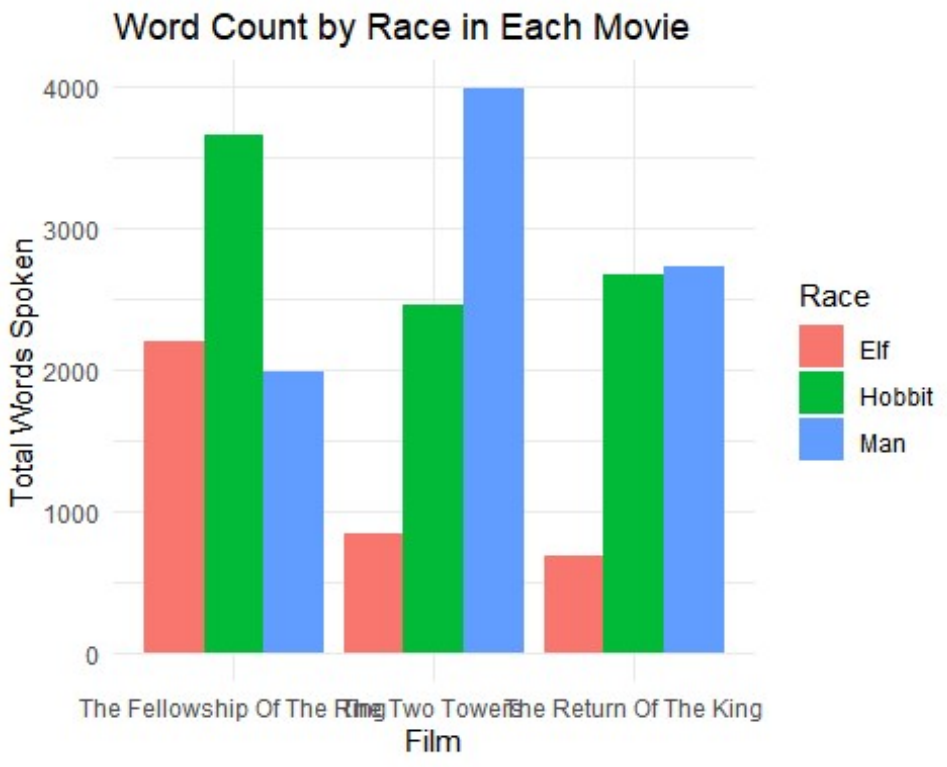
Race and film

Does the dominant race differ across the three movies? (Hint: group by both `Race` and `Film`.) Experiment with filling by `Race` or `Film` and faceting by `Race` or `Film`.

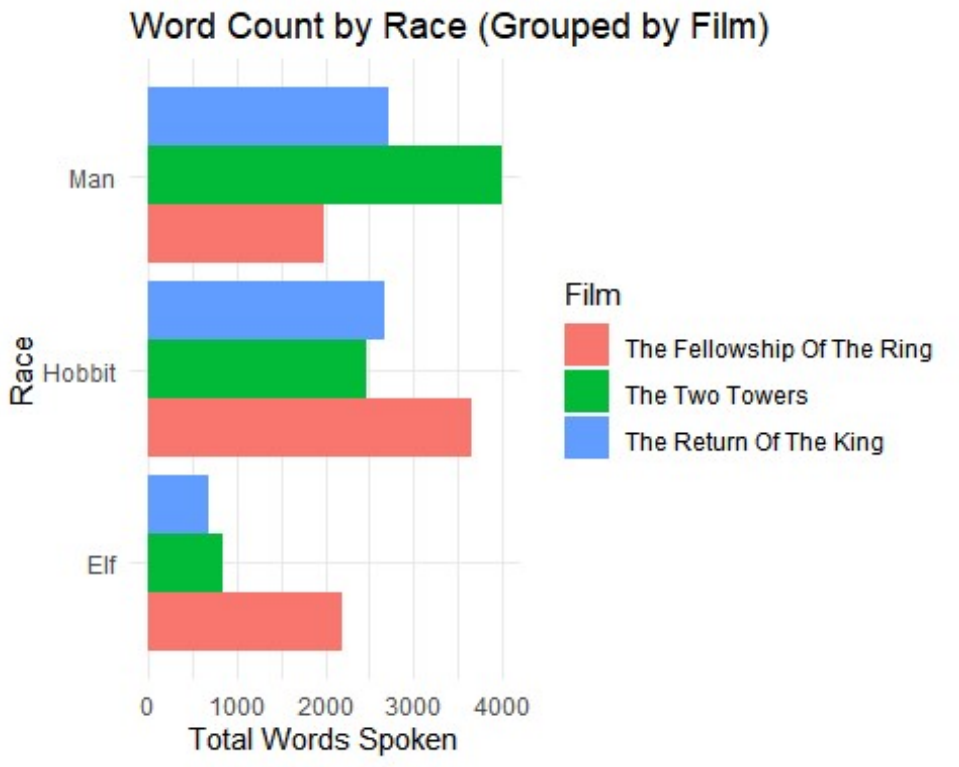
```
# Group by Race and Film, then sum the words spoken
race_film_summary <- lotr %>%
  group_by(Race, Film) %>%
  summarise(Total_Words = sum(Words, na.rm = TRUE)) %>%
  arrange(Film, desc(Total_Words))

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

# Print the summarized data
print(race_film_summary)

## # A tibble: 9 × 3
## # Groups:   Race [3]
##   Race   Film                        Total_Words
##   <chr>  <fct>                             <dbl>
## 1 Hobbit The Fellowship Of The Ring         3658
## 2 Elf    The Fellowship Of The Ring         2200
## 3 Man    The Fellowship Of The Ring         1995
## 4 Man    The Two Towers                     3990
## 5 Hobbit The Two Towers                     2463
## 6 Elf    The Two Towers                      844
```

```
## 7 Man    The Return Of The King            2727
## 8 Hobbit The Return Of The King            2675
## 9 Elf    The Return Of The King             693
```

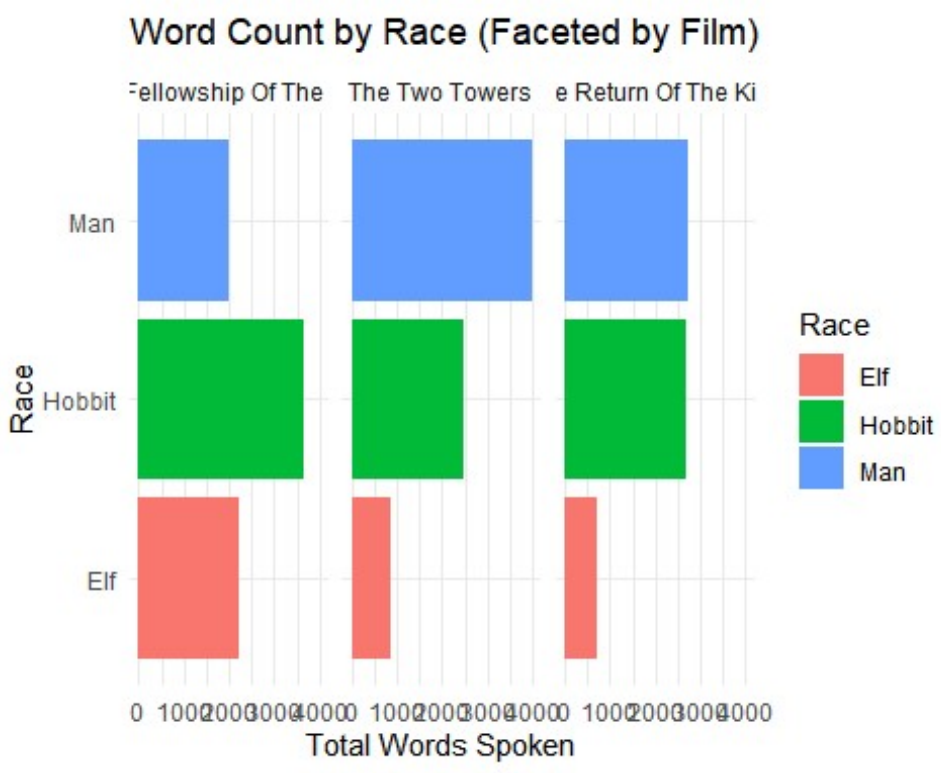
```r
# Bar Plot with Race Fill

ggplot(race_film_summary, aes(x = Film, y = Total_Words, fill = Race)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Word Count by Race in Each Movie",
       x = "Film",
       y = "Total Words Spoken") +
  theme_minimal()
```




```r
# Bar Plot with Film Fill

ggplot(race_film_summary, aes(x = Race, y = Total_Words, fill = Film)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Word Count by Race (Grouped by Film)",
       x = "Race",
       y = "Total Words Spoken") +
  theme_minimal() +
  coord_flip()  # Flip for better readability if many races
```




```r
# Faceted by Film

ggplot(race_film_summary, aes(x = Race, y = Total_Words, fill = Race)) +
  geom_bar(stat = "identity") +
  facet_wrap(~Film) +
  labs(title = "Word Count by Race (Faceted by Film)",
       x = "Race",
       y = "Total Words Spoken") +
  theme_minimal() +
  coord_flip()
```



Race and gender and film

Create a plot that visualizes the number of words spoken by race, gender, and film simultaneously. Use the complete tidy `lotr` data frame. You don't need to create a new summarized dataset (with `group_by(Race, Gender, Film)`) because the original data already has a row for each of those (you could make a summarized dataset, but it would be identical to the full version).

You need to show `Race`, `Gender`, and `Film` at the same time, but you only have two possible aesthetics (`x` and `fill`), so you'll also need to facet by the third. Play around with different combinations (e.g. try `x = Race`, then `x = Film`) until you find one that tells the clearest story. For fun, add a `labs()` layer to add a title and subtitle and caption.

```
# Plot Breakdown by Race, Gender, and Film

ggplot(lotr, aes(x = Gender, y = Words, fill = Race)) +
  geom_bar(stat = "identity", position = "dodge") +
  facet_wrap(~Film) +
  labs(
    title = "The Lord of the Rings Trilogy Analysis",
    subtitle = "Breakdown by Race, Gender, and Film",
    x = "Gender",
    y = "Total Words Spoken",
    fill = "Race",
    caption = "Data Source: LOTR Dataset"
```

```
  ) +
theme_minimal()
```