D607 - Assignment 2

Marco Castro

2024-09-15

Overview

I conducted a simple survey asking six participants to rate six movies on a scale from 1 (low) to 5 (hi). Survey submissions were stored in a MySQL database and retrieved in R using the RMySQL package.

```
## Warning: package 'RMySQL' was built under R version 4.3.3
```

```
## Loading required package: DBI
```

```
## Warning: package 'digest' was built under R version 4.3.3
```

```
## Welcome to clipr. See ?write_clip for advisories on writing to the clipboard in R.
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.2     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 become errors
##
## Attaching package: 'scales'
##
##
## The following object is masked from 'package:purrr':
##
##     discard
##
##
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
# Read data from db
mydb <-  dbConnect(MySQL(), user = db_user, password = db_password,
                   dbname = db_name, host = db_host, port = db_port)
```

```r
query <- "SELECT r.response_id, p.FirstName, m.title, r.rating FROM survey_movie_ratings AS r LEFT JOIN
rs <- dbSendQuery(mydb, query)
df <-  fetch(rs, n = -1)
dbDisconnect(mydb)
```

```
## Warning: Closing open result sets
```

```
## [1] TRUE
```

```r
head(df, 10)
```

```
##    response_id FirstName                      title rating
## 1            1     Nadia                         Up      5
## 2            2     Nadia                      Moana     NA
## 3            3     Nadia                 Inside Out      5
## 4            4     Nadia Nightmare Before Christmas      4
## 5            5     Nadia                Beetlejuice      3
## 6            6     Nadia                 Home Alone      2
## 7            7      Luna                         Up      5
## 8            8      Luna                      Moana      5
## 9            9      Luna                 Inside Out      5
## 10          10      Luna Nightmare Before Christmas     NA
```

Handling Missing Data

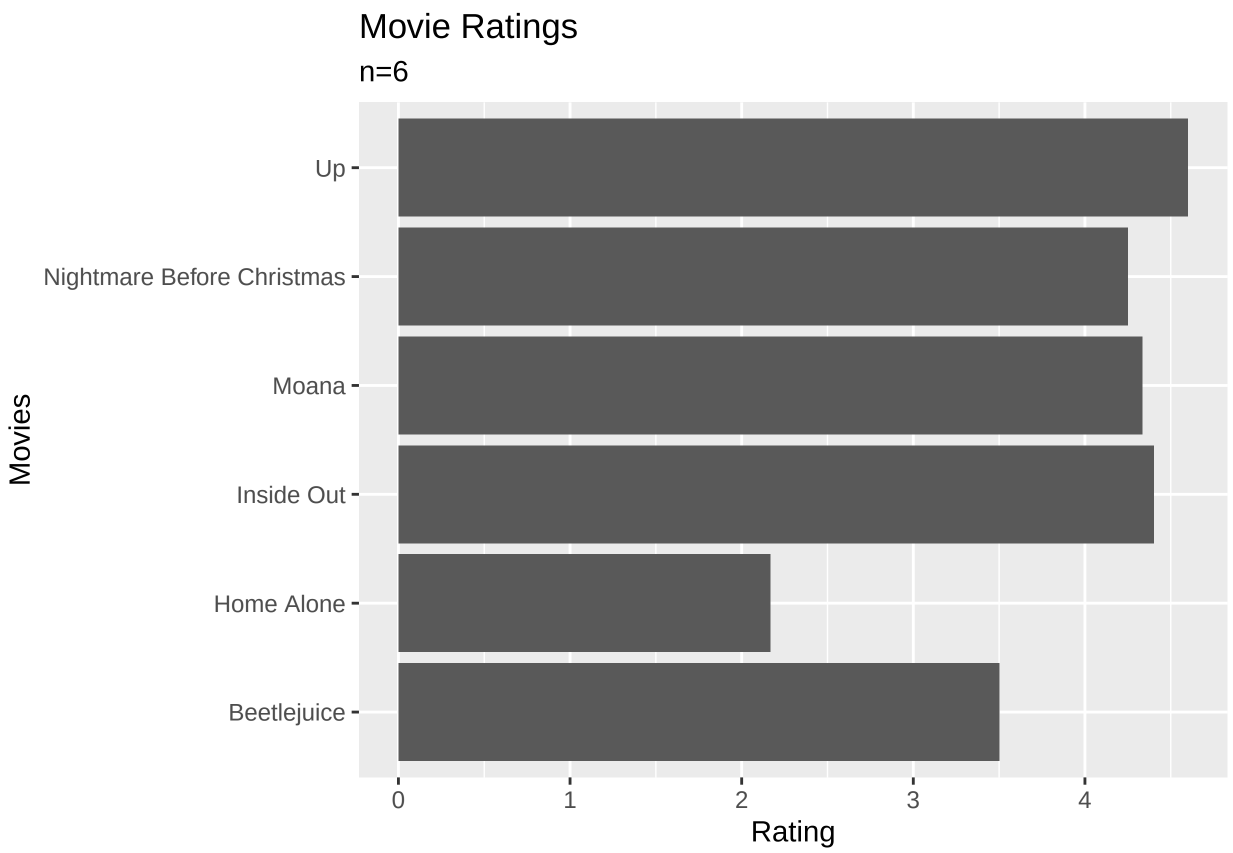
As some participants did not provide ratings for all six movies, I omitted missing values using na.omit(). I chose this strategy over other strategies such as imputing by replacing values with the mean or median since the sample size was relatively small and I did not want to artificially inflate the average ratings for each movie.

```r
movie_ratings <- na.omit(df)
```

Visualizing the ratings

I plotted a bar graph to visualize the average rating for each movie.

```r
ggplot(data=movie_ratings, aes(x=title, y=rating)) +
  geom_bar(stat="summary", fun="mean") +
  coord_flip() +
  labs(
    title = "Movie Ratings",
    subtitle = "n=6",
    x = "Movies",
    y = "Rating"
  )
```



```
avg_movie_ratings <- movie_ratings |>
    group_by(title) |>
    summarize(
      avg_rating = mean(rating),
      n = n()
    )
avg_movie_ratings
```

```
## # A tibble: 6 x 3
##   title                      avg_rating     n
##   <chr>                           <dbl> <int>
## 1 Beetlejuice                      3.5      4
## 2 Home Alone                       2.17     6
## 3 Inside Out                       4.4      5
## 4 Moana                            4.33     3
## 5 Nightmare Before Christmas       4.25     4
## 6 Up                               4.6      5
```

Conclusion

Based on this short survey, the top three rated movies were “Up”, “Inside Out”, and “Moana” with an average rating 4.6, 4.4, 4.3 respectively. However, it would be worth noting that all three movies had missing responses for at least one movie. It is possible that rankings would change if the participants rated movies they didn’t provide answers for previously.