Data_exploration

2025-10-03

INTRODUCTION

This document provides a detailed report of our data exploration where we focus on exploring the raw data before any cleaning or modeling occurs, including: - How the datasets were loaded - Data quality observations: missing values and outliers - An overview of the structure and the relevant variables

RESEARCH QUESTION

How does runtime influence audience ratings for movies compared to TV episodes, controlling for the release year?

RESEARCH METHOD

By conducting a multiple linear regression, we will estimate the independent effect of runtime on ratings while controlling for release year and content type (movie/TV episode).

DEPENDENCIES

For the data exploration step, we used the following dependencies: - R - Make - Installed packages in R:

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.1.0
## -- 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
```

```
library(ggplot2)
library(data.table)
```

```
##
## Attaching package: 'data.table'
##
## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year
##
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
##
## The following object is masked from 'package:purrr':
##
##     transpose
```

LOADING THE DATASETS

Two separate datasets were acquired from IMDb (https://datasets.imdbws.com/): - The first dataset contains information on the contents' duration and release year, saved as "title_basics_raw" - The second dataset contains information on the contents' ratings, saved as "ratings_raw"

The following chunk of code will first create the folder "raw" to store the raw datasets, and second, load the datasets into R:

```
# Ensure the "raw" folder exists
dir.create("raw", recursive = TRUE, showWarnings = FALSE)

# Retrieve dataset for Movie and TV Episode Duration and Release Year
title_url <- "https://datasets.imdbws.com/title.basics.tsv.gz"
title_basics_raw <- read_tsv(title_url, na = c("\\N", ""))
```

```
## Warning: One or more parsing issues, call `problems()` on your data frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)

## Rows: 11946100 Columns: 9
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (5): tconst, titleType, primaryTitle, originalTitle, genres
## dbl (4): isAdult, startYear, endYear, runtimeMinutes
##
## 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.
```

```
# Retrieve dataset for rating
rating_url <- "https://datasets.imdbws.com/title.ratings.tsv.gz"
ratings_raw <- read_tsv(rating_url, na = c("\\N", ""))
```

```
## Rows: 1621426 Columns: 3
## -- Column specification --------------------------------------------------------
```

```
## Delimiter: "\t"
## chr (1): tconst
## dbl (2): averageRating, numVotes
##
## 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.
```

```
# Save the downloaded datasets to folder "raw"
write_tsv(title_basics_raw, "raw/title_basics_raw.tsv")
write_tsv(ratings_raw, "raw/ratings_raw.tsv")
```

DATA QUALITY OBSERVATIONS

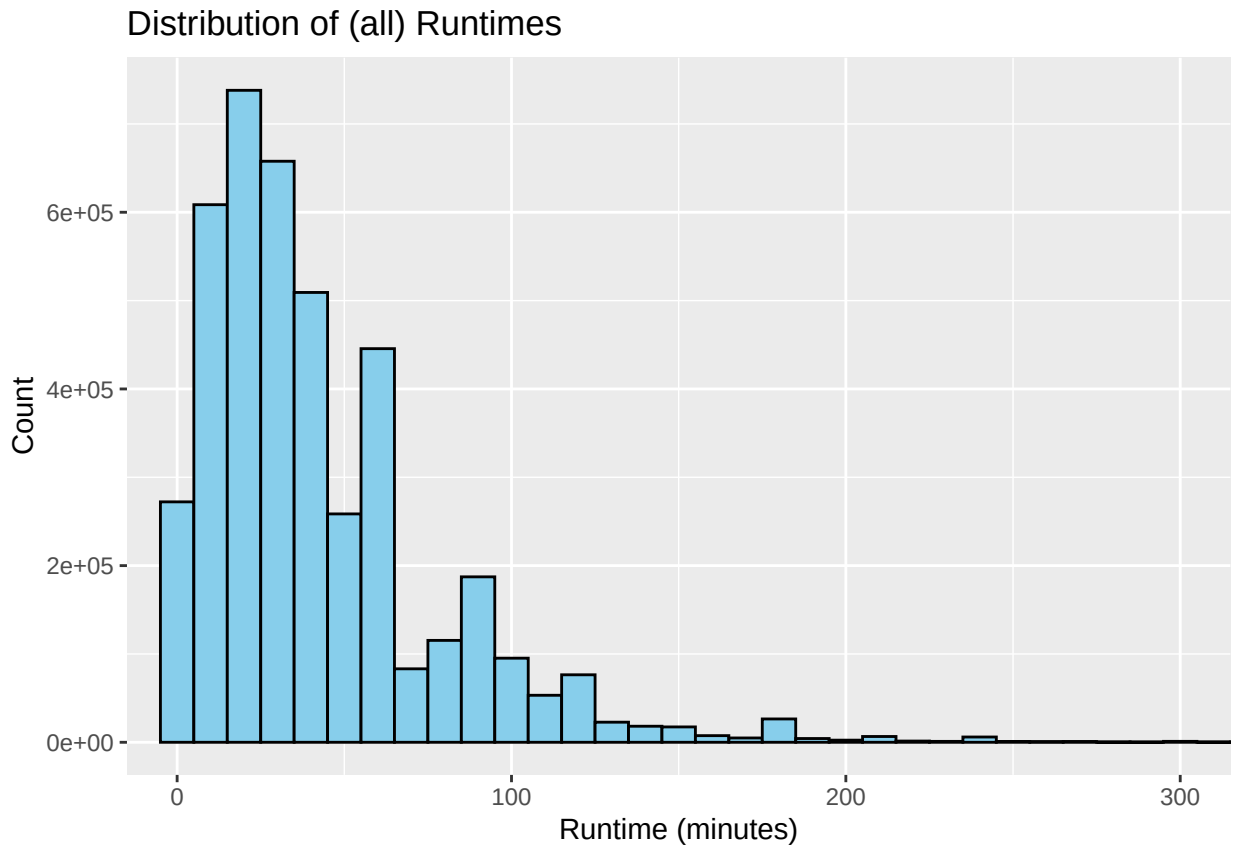
The following chunk of code provides a data quality check to identify missing values and outliers:

```
# Data quality check for title_basics_raw
  # Missing values
colSums(is.na(title_basics_raw[c("tconst", "primaryTitle", "startYear", "runtimeMinutes")]))
```

```
##         tconst   primaryTitle      startYear runtimeMinutes
##              0              0        1441712        7715865
```

```
  # Outliers
setDT(title_basics_raw)
ggplot(title_basics_raw, aes(x = runtimeMinutes)) +
  geom_histogram(binwidth = 10, fill = "skyblue", color = "black") +
  coord_cartesian(xlim = c(0, 300)) +  # focus on typical runtimes
  labs(x = "Runtime (minutes)", y = "Count", title = "Distribution of (all) Runtimes")
```

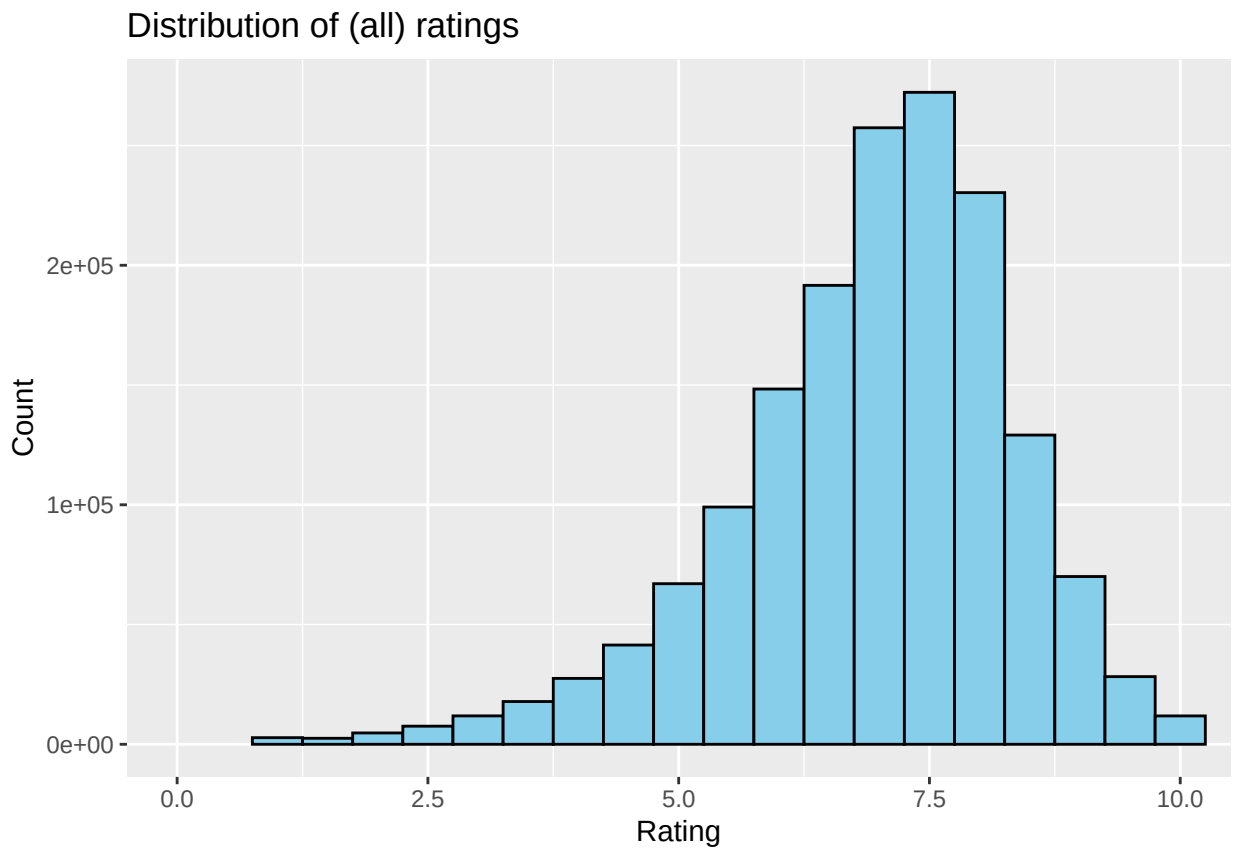
```
## Warning: Removed 7715865 rows containing non-finite outside the scale range
## (`stat_bin()`).
```




```
# Data quality check for title_ratings
  # Missing values
colSums(is.na(ratings_raw[c("tconst","averageRating")]))
```

```
##        tconst averageRating
##             0             0
```

```
  # Outliers
setDT(ratings_raw)
ggplot(ratings_raw, aes(x = averageRating)) +
  geom_histogram(binwidth = 0.5, fill = "skyblue", color = "black") +
  coord_cartesian(xlim = c(0, 10)) +
  labs(x = "Rating", y = "Count", title = "Distribution of (all) ratings")
```



OVERVIEW OF THE STRUCTURE AND RELEVANT VARIABLES

Below you will find the variable names and descriptions that are relevant to this study. 1. tconst(string): alphanumeric unique identifier of the title 2. primaryTitle (string):the more popular title/the title used by the filmmakers on promotional materials at the point of release 3. startYear: represents the release year of a title 4. runtimeMinutes: primary runtime of the title, in minutes 5. averageRating: weighted average of all the individual user ratings 6. is_tvepisode (boolean): 0: content type is Movie, 1: content type is TV Episode