Final Project CS597

Code

Sagar Jain

4 May 2020

1 Executive summary

I wanted to do a kernel in which I would make a recommendation engine, and chose the TMDB 5000 Movie Dataset (https://www.kaggle.com/tmdb/tmdb-movie-metadata/home) dataset. Kaggle has removed the original IMDB version of this dataset per a DMCA takedown request from IMDB, and replaced it with a similar set of movies from The Movie Database (TMDb).

2 Preparing the data

In this dataset, two csv-files are provided: tmdb_5000_movies.csv and tmdb_5000_credits.csv.

Code

2.1 The movies file

The movies file contains 20 variables on 4,803 movies.

Code

```
## Observations: 4,803
## Variables: 20
## $ budget               <dbl> 2.37e+08, 3.00e+08, 2.45e+08, 2.50e+08, 2.60e+08…
## $ genres               <chr> "[{\"id\": 28, \"name\": \"Action\"}, {\"id\": 1…
## $ homepage             <chr> "http://www.avatarmovie.com/", "http://disney.go…
## $ id                   <dbl> 19995, 285, 206647, 49026, 49529, 559, 38757, 99…
## $ keywords             <chr> "[{\"id\": 1463, \"name\": \"culture clash\"}, {…
## $ original_language    <chr> "en", "en", "en", "en", "en", "en", "en", "en", …
## $ original_title       <chr> "Avatar", "Pirates of the Caribbean: At World's …
## $ overview             <chr> "In the 22nd century, a paraplegic Marine is dis…
## $ popularity           <dbl> 150.43758, 139.08262, 107.37679, 112.31295, 43.9…
## $ production_companies <chr> "[{\"name\": \"Ingenious Film Partners\", \"id\"…
## $ production_countries <chr> "[{\"iso_3166_1\": \"US\", \"name\": \"United St…
## $ release_date         <date> 2009-12-10, 2007-05-19, 2015-10-26, 2012-07-16,…
## $ revenue              <dbl> 2787965087, 961000000, 880674609, 1084939099, 28…
## $ runtime              <dbl> 162, 169, 148, 165, 132, 139, 100, 141, 153, 151…
## $ spoken_languages     <chr> "[{\"iso_639_1\": \"en\", \"name\": \"English\"}…
## $ status               <chr> "Released", "Released", "Released", "Released", …
## $ tagline              <chr> "Enter the World of Pandora.", "At the end of th…
## $ title                <chr> "Avatar", "Pirates of the Caribbean: At World's …
## $ vote_average         <dbl> 7.2, 6.9, 6.3, 7.6, 6.1, 5.9, 7.4, 7.3, 7.4, 5.7…
## $ vote_count           <dbl> 11800, 4500, 4466, 9106, 2124, 3576, 3330, 6767,…
```

There are three occurences where two movies have the exact same title. When looking at the release date, we see that these are in fact different movies. This means that I have to be a little bit carefull and use the id as the unique identifier instead of the title.

Five columns are in the JSON format (genres, keywords, production_companies, production_countries, and spoken_languages).

As a start, I created dataframes for keywords and genres that can be joined with movies. Altogether, these 4803 movies have 12,160 movie-genre combinations, and 36,194 movie-keyword combinations.

Code

Code

2.1.1 Joining the Top3 genres

For genres, I assume that they are ranked in the order of importance. I only want to keep the three most important genres for each movie.

Code

idtitle		genres
19995	Avatar	Action
19995	Avatar	Adventure
19995	Avatar	Fantasy
19995	Avatar	Science Fiction
285	Pirates of the Caribbean: At World's End	Adventure
285	Pirates of the Caribbean: At World's End	Fantasy
285	Pirates of the Caribbean: At World's End	Action
206647	Spectre	Action
206647	Spectre	Adventure
206647	Spectre	Crime
49026	The Dark Knight Rises	Action
49026	The Dark Knight Rises	Crime
49026	The Dark Knight Rises	Drama
49026	The Dark Knight Rises	Thriller

So if you look at the 4 movies above, this means that I don't want Science Fiction as a Top3 genre for Avatar, and also not keep Thriller a a genre for The Dark Night Rises. Below, you can see the result of this Top3 selection. This dataframe with a maximum of 3 genres per movie is joined with the movies dataframe.

Code

idtitle		genre_1	genre_2	genre_3
285	Pirates of the Caribbean: At World's End	Adventure	Fantasy	Action
19995	Avatar	Action	Adventure	Fantasy
49026	The Dark Knight Rises	Action	Crime	Drama
206647	Spectre	Action	Adventure	Crime

2.2 The credits file (cast and crew members)

This file only contains the movie_id, title, and 2 JSON columns. In the remainder of this section, I am creating two dataframes from these JSON columns (cast and crew), which can easily be joined with the movies dataframe.

Code

```
## Observations: 4,803
## Variables: 4
## $ movie_id <dbl> 19995, 285, 206647, 49026, 49529, 559, 38757, 99861, 767, 20…
## $ title    <chr> "Avatar", "Pirates of the Caribbean: At World's End", "Spect…
## $ cast     <chr> "[{\"cast_id\": 242, \"character\": \"Jake Sully\", \"credit…
## $ crew     <chr> "[{\"credit_id\": \"52fe48009251416c750aca23\", \"department…
```

2.2.1 Joining the Top3 actors with movies

Below you can see a glimpse of the new cast dataframe, and also the roles that Leonardo di Caprio played as an example.

Code

```
## Observations: 106,257
## Variables: 8
## $ movie_id      <dbl> 19995, 19995, 19995, 19995, 19995, 19995, 19995, 19995,…
## $ title         <fct> Avatar, Avatar, Avatar, Avatar, Avatar, Avatar, Avatar,…
## $ movie_cast_id <int> 242, 3, 25, 4, 5, 8, 7, 9, 11, 10, 12, 13, 32, 33, 34, …
## $ character     <fct> Jake Sully, Neytiri, Dr. Grace Augustine, Col. Quaritch…
## $ gender        <int> 2, 1, 1, 2, 1, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 0, 0, 2…
## $ actor_id      <int> 65731, 8691, 10205, 32747, 17647, 1771, 59231, 30485, 1…
## $ actor         <fct> Sam Worthington, Zoe Saldana, Sigourney Weaver, Stephen…
## $ order         <int> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 1…
```

Code

movie_id	title	movie_cast_id	character	gender	actor_id	actor	order
597	Titanic	21	Jack Dawson	2	6193	Leonardo DiCaprio	1
64682	The Great Gatsby	2	Jay Gatsby	2	6193	Leonardo DiCaprio	0
27205	Inception	1	Dom Cobb	2	6193	Leonardo DiCaprio	0
281957	The Revenant	1	Hugh Glass	2	6193	Leonardo DiCaprio	0
2567	The Aviator	1	Howard Hughes	2	6193	Leonardo DiCaprio	0
68718	Django Unchained	3	Calvin Candie	2	6193	Leonardo DiCaprio	2
1372	Blood Diamond	27	Danny Archer	2	6193	Leonardo DiCaprio	0
106646	The Wolf of Wall Street	8	Jordan Belfort	2	6193	Leonardo DiCaprio	0
3131	Gangs of New York	2	Amsterdam Vallon	2	6193	Leonardo DiCaprio	0

movie_id	title	movie_cast_id	character	gender	actor_id	actor	order
1422	The Departed	2	William “Billy” Costigan, Jr.	2	6193	Leonardo DiCaprio	0
11324	Shutter Island	2	Teddy Daniels	2	6193	Leonardo DiCaprio	0
12113	Body of Lies	4	Roger Ferris	2	6193	Leonardo DiCaprio	0
640	Catch Me If You Can	23	Frank Abagnale Jr.	2	6193	Leonardo DiCaprio	0
1907	The Beach	2	Richard	2	6193	Leonardo DiCaprio	0
4148	Revolutionary Road	2	Frank Wheeler	2	6193	Leonardo DiCaprio	0
9313	The Man in the Iron Mask	1	King Louis XIV / Philippe	2	6193	Leonardo DiCaprio	0
88794	J. Edgar	1	J. Edgar Hoover	2	6193	Leonardo DiCaprio	0
12106	The Quick and the Dead	14	Fee Herod "The Kid“	2	6193	Leonardo DiCaprio	3
9819	Marvin’s Room	3	Hank	2	6193	Leonardo DiCaprio	2
454	Romeo + Juliet	1	Romeo	2	6193	Leonardo DiCaprio	0
9466	Celebrity	10	Brandon	2	6193	Leonardo DiCaprio	3
1587	What’s Eating Gilbert Grape	2	Arnie Grape	2	6193	Leonardo DiCaprio	1

I noticed that the “order” for DiCaprio is mostly 0, 1, or 2. I suspect that this indicated that he had a leading role in the those movies. Let see what the order looks like when I look at the Titanic.

Code

title	order	actor	character
Titanic	0	Kate Winslet	Rose DeWitt Bukater
Titanic	1	Leonardo DiCaprio	Jack Dawson
Titanic	2	Frances Fisher	Ruth Dewitt Bukater
Titanic	3	Billy Zane	Caledon ‘Cal’ Hockley
Titanic	4	Kathy Bates	Molly Brown
Titanic	5	Gloria Stuart	Old Rose
Titanic	6	Bill Paxton	Brock Lovett
Titanic	7	Bernard Hill	Captain Edward James Smith
Titanic	8	David Warner	Spicer Lovejoy
Titanic	9	Victor Garber	Thomas Andrews

This confirms that the order indicates the leading roles indeed. The Titanic cast consisted of 136 roles. As I am sure that people are mostly only interested in the actors who played the leading roles in a movie, I am only keeping the Top3 actors of each movie.

Before I do so, I also want to spread them into a wide format. However, I have to solve an issue because some movies have multiple values for some orders.

Code

movie_idtitle	orderactor
453A Beautiful Mind	0Russell Crowe
453A Beautiful Mind	0Ed Harris
453A Beautiful Mind	1Jennifer Connelly
453A Beautiful Mind	2Christopher Plummer

Below, you can see the result of my efforts. Besides two well-known movies, I am also displaying the one above as this shows that the issue has been solved too.

Code

movie_idtitle	actor_1	actor_2	actor_3
453A Beautiful Mind	Russell Crowe	Ed Harris	Jennifer Connelly
597Titanic	Kate Winslet	Leonardo DiCaprio	Frances Fisher
19995Avatar	Sam Worthington	Zoe Saldana	Sigourney Weaver

Finally, I am joining this dataframe with the Top3 actors for each movie to the movies dataframe.

Code

2.2.2 Joining the Director with movies

Below you can see a glimpse of the new crew dataframe, and also the crew-roles that James Cameron had as an example. As you can see, James Cameron had multiple roles in most movies in which he was involved.

Code

```
## Warning: unnest() has a new interface. See ?unnest for details.
## Try `df %>% unnest(c(js, .name_repair))`, with `mutate()` if needed
```

Code

```
## Observations: 129,581
## Variables: 8
## $ movie_id     <dbl> 19995, 19995, 19995, 19995, 19995, 19995, 19995, 19995, …
## $ title        <fct> Avatar, Avatar, Avatar, Avatar, Avatar, Avatar, Avatar, …
## $ department   <fct> Editing, Art, Sound, Sound, Production, Sound, Directing…
## $ gender       <int> 0, 2, 0, 0, 1, 2, 2, 2, 2, 2, 2, 2, 0, 1, 2, 0, 0, 0, 0,…
## $ crew_id      <int> 1721, 496, 900, 900, 1262, 1729, 2710, 2710, 2710, 2710,…
## $ job          <fct> Editor, Production Design, Sound Designer, Supervising S…
## $ crew         <fct> Stephen E. Rivkin, Rick Carter, Christopher Boyes, Chris…
## $ .name_repair <fct> unique, unique, unique, unique, unique, unique, unique, …
```

Code

movie_id	title	department	gender	crew_id	job	crew	.name_repair
19995	Avatar	Directing	2	2710	Director	James Cameron	unique
19995	Avatar	Writing	2	2710	Writer	James Cameron	unique
19995	Avatar	Editing	2	2710	Editor	James Cameron	unique
19995	Avatar	Production	2	2710	Producer	James Cameron	unique
19995	Avatar	Writing	2	2710	Screenplay	James Cameron	unique
597	Titanic	Writing	2	2710	Screenplay	James Cameron	unique
597	Titanic	Directing	2	2710	Director	James Cameron	unique
597	Titanic	Editing	2	2710	Editor	James Cameron	unique
597	Titanic	Production	2	2710	Producer	James Cameron	unique
296	Terminator 3: Rise of the Machines	Writing	2	2710	Characters	James Cameron	unique
87101	Terminator Genisys	Writing	2	2710	Characters	James Cameron	unique
280	Terminator 2: Judgment Day	Directing	2	2710	Director	James Cameron	unique
280	Terminator 2: Judgment Day	Production	2	2710	Producer	James Cameron	unique
280	Terminator 2: Judgment Day	Writing	2	2710	Writer	James Cameron	unique
36955	True Lies	Writing	2	2710	Screenplay	James Cameron	unique
36955	True Lies	Directing	2	2710	Director	James Cameron	unique

movie_id	title	department	gender	crew_id	job	crew	.name_repair
36955	True Lies	Production	2	2710	Producer	James Cameron	unique
2756	The Abyss	Writing	2	2710	Screenplay	James Cameron	unique
2756	The Abyss	Directing	2	2710	Director	James Cameron	unique
1369	Rambo: First Blood Part II	Writing	2	2710	Screenplay	James Cameron	unique
94352	Cirque du Soleil: Worlds Away	Production	2	2710	Producer	James Cameron	unique
679	Aliens	Writing	2	2710	Screenplay	James Cameron	unique
679	Aliens	Directing	2	2710	Director	James Cameron	unique
218	The Terminator	Directing	2	2710	Director	James Cameron	unique
218	The Terminator	Writing	2	2710	Writer	James Cameron	unique

Similar to only selecting the Top-cast, I think that people are mostly interested in the director and not so much in the rest of the crew. However, a bunch of movies have multiple people labeled as “Director” in the crew file (and 30 movies have no Director). Altogether, 4465 movies have one, unique director. I am joining only those with the movies dataframe.

Code

3 Exploring the data

3.1 Numbers of votes

3.1.1 Histogram of number of votes

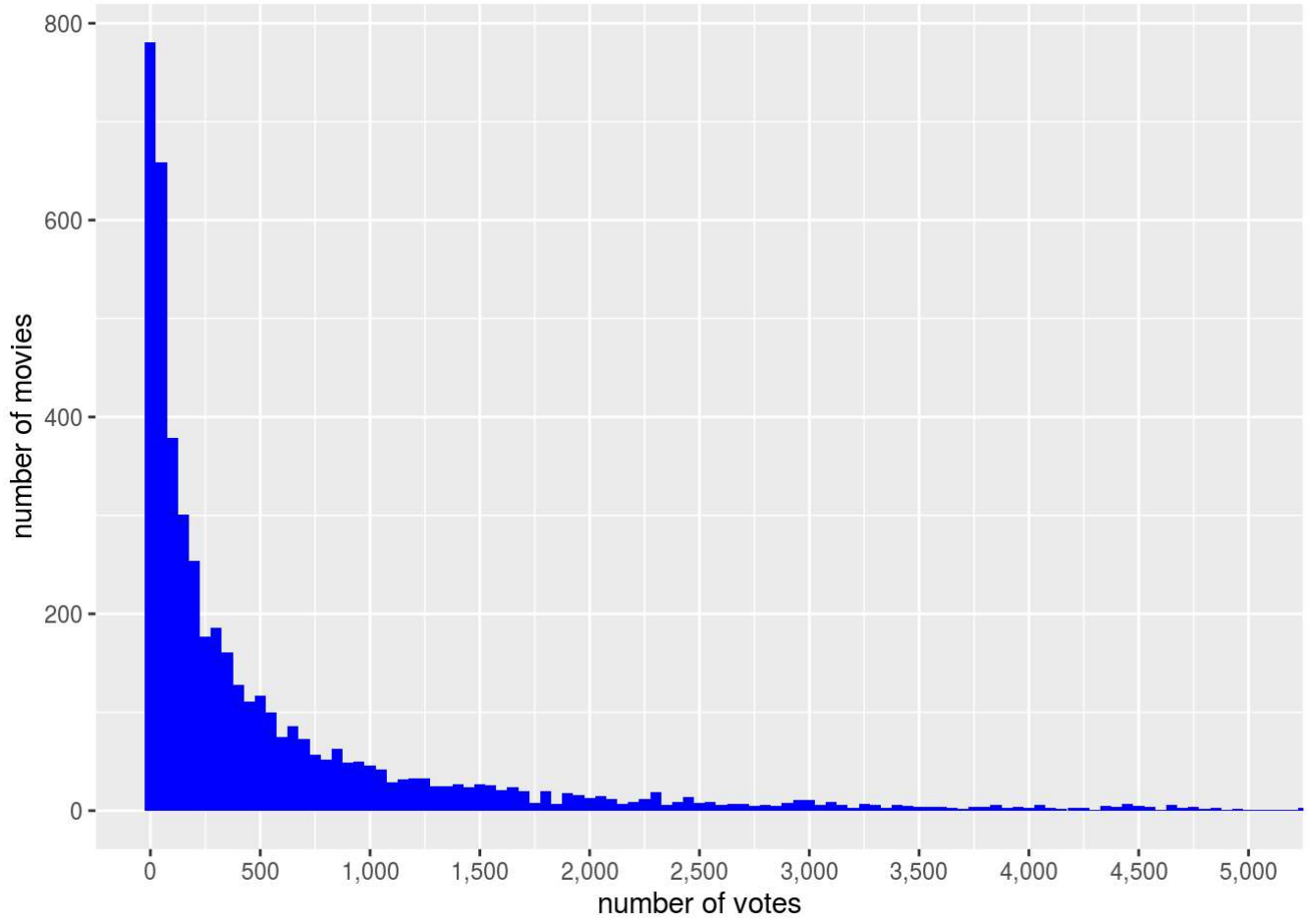
The number of votes per movie ranges from 0 to 13,752

Code

```
## [1]     0 13752
```

As there the tail of movies with more than 5,000 votes is hardly visible I am only displaying the histogram for movies with less than 5,000 votes.

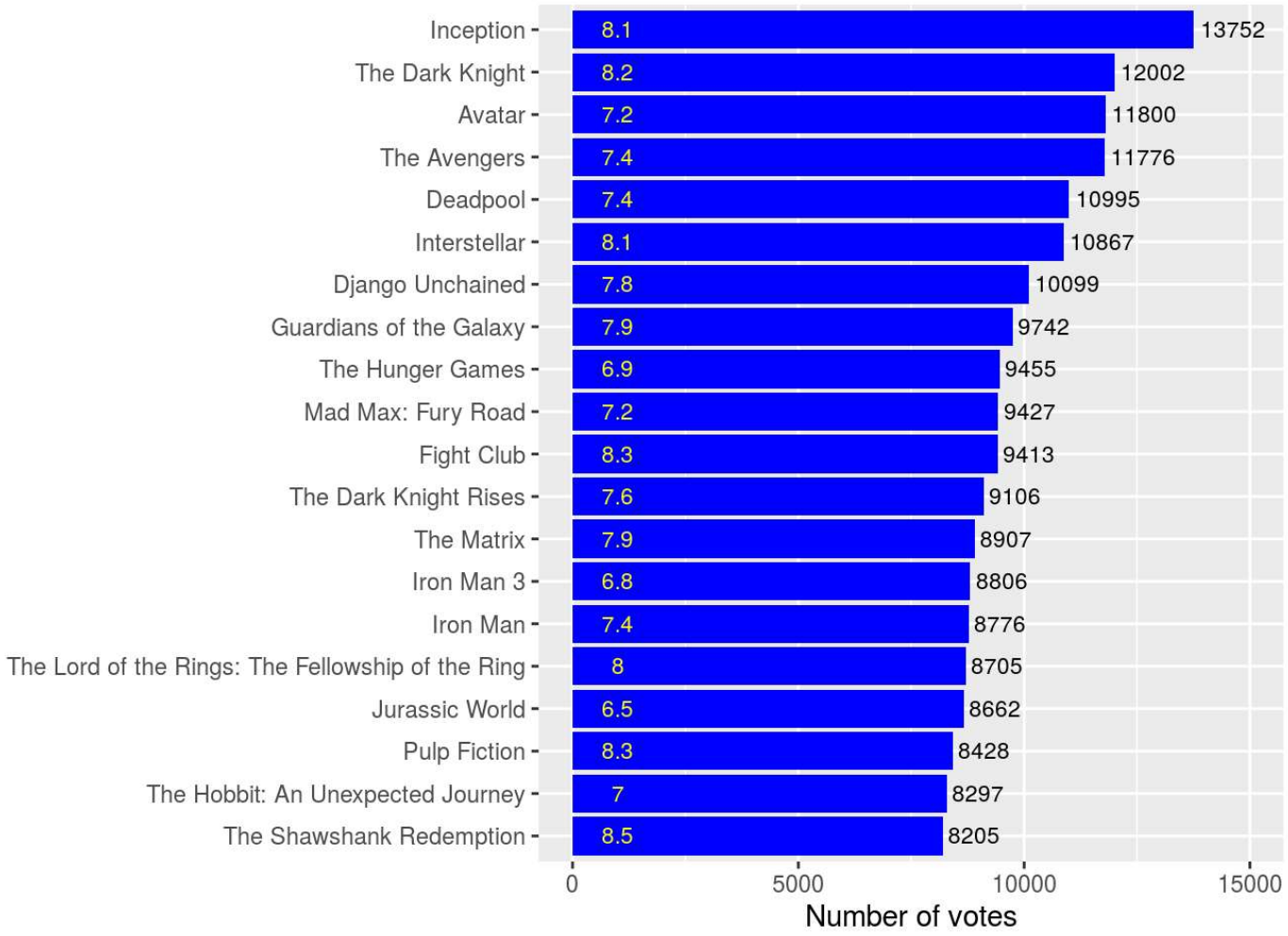
Code



3.1.2 Movies with the highest number of votes

When looking at the movies with most votes, we see that most of them had pretty good voting average too. The lowest vote_average within the Top20 movies with most votes is 6.8.

Code

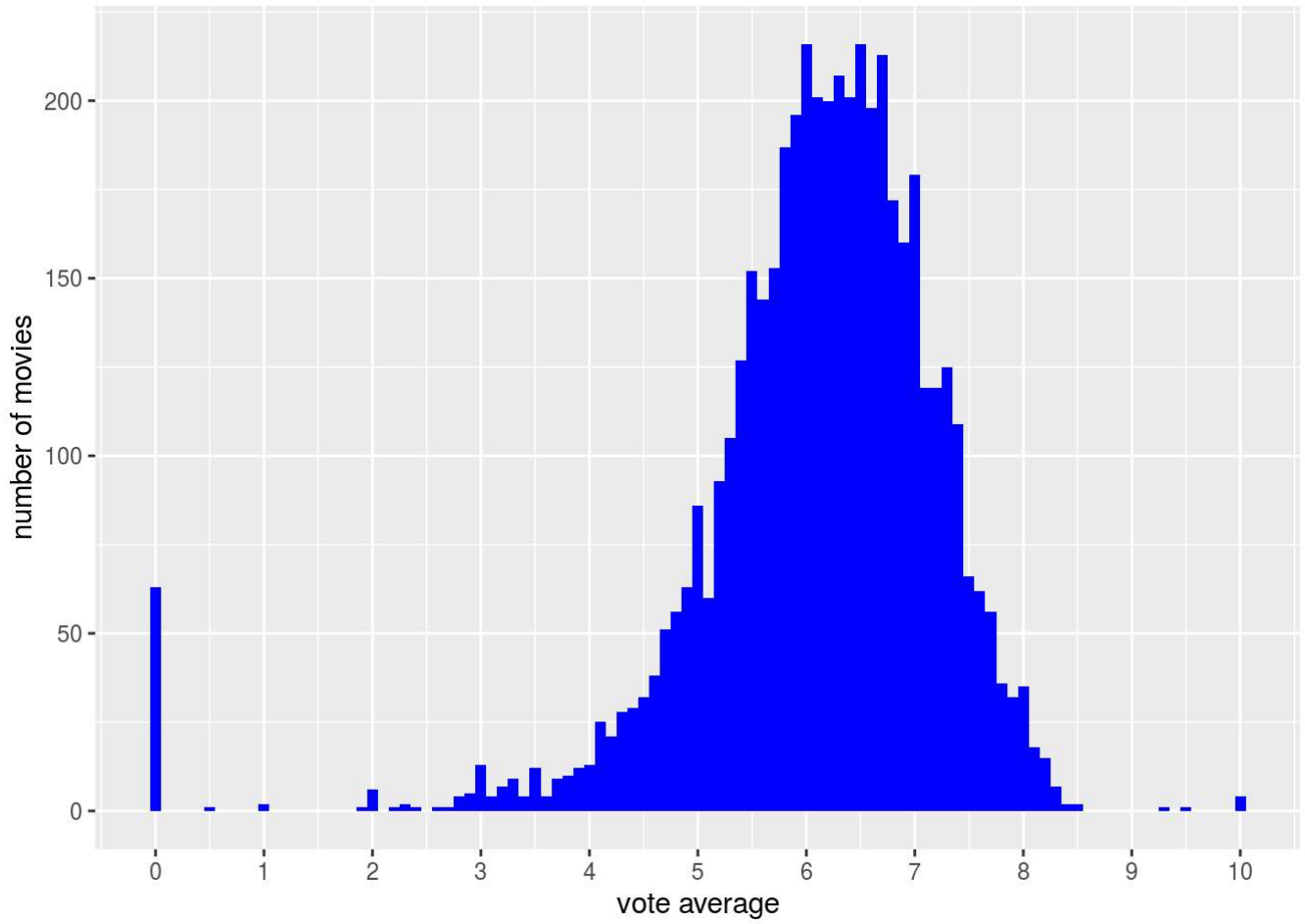


3.2 Vote average

3.2.1 Histogram of vote average

As you can see, the vote_average is left skewed, and the skew is -1.96.

Code



However, we should realize that very high or very low vote_averages are generally based on small numbers of votes per movie. There are 62 movies with 0 votes and a vote_average of 0.0.

Below, I am filtering the movies on vote_average below 4.5 or above 8.5. As you can see, there are only 11 movies that “achieved” their vote_average in this range with at vote_count of at least 400.

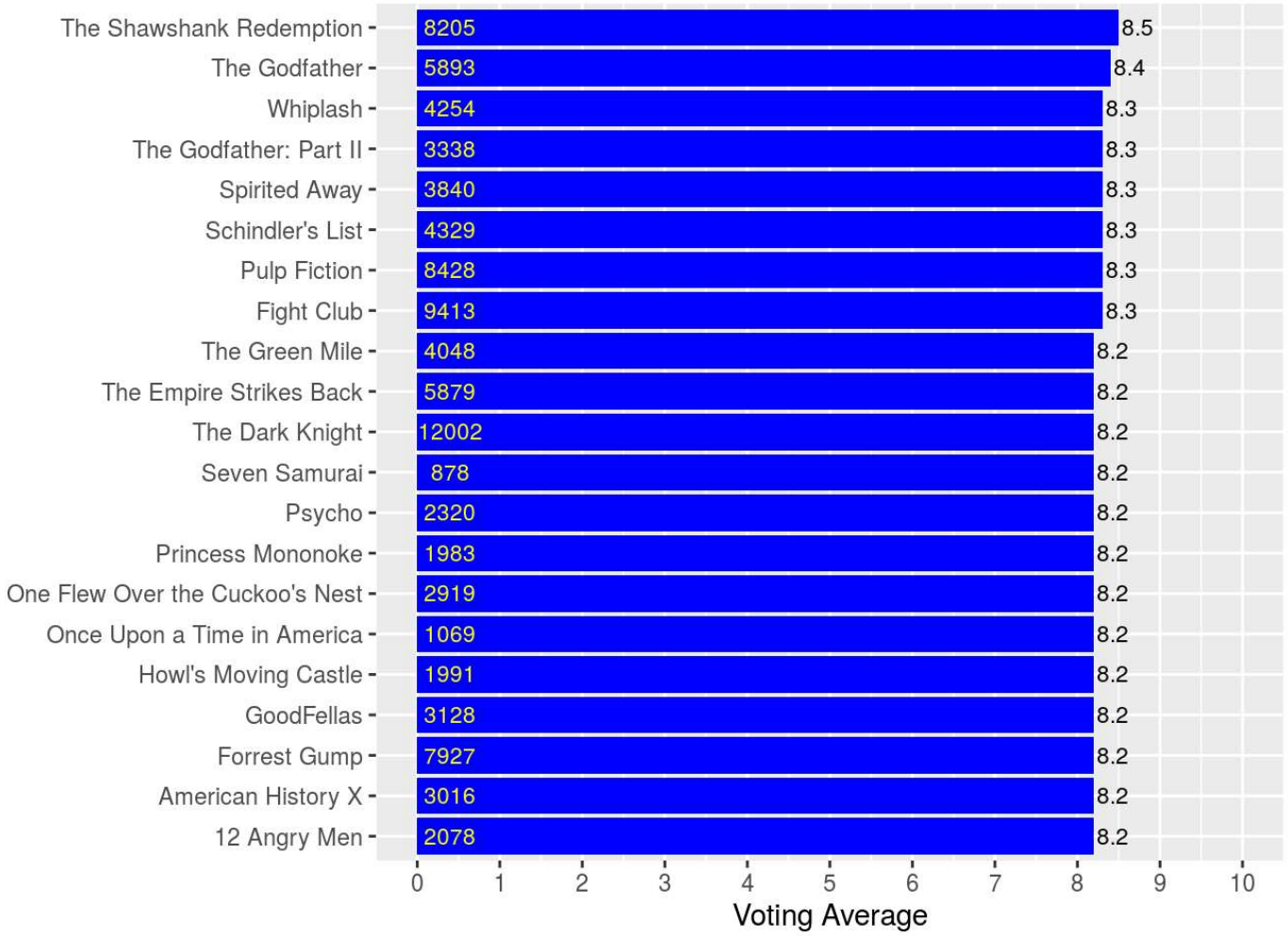
Code

	idtitle	vote_average	vote_count
14164	Dragonball Evolution	2.9	462
205321	Sharknado	3.8	476
1639	Speed 2: Cruise Control	4.1	434
71880	Jack and Jill	4.1	604
241251	The Boy Next Door	4.1	1022
415	Batman & Robin	4.2	1418
314	Catwoman	4.2	808
166424	Fantastic Four	4.4	2278
188207	The Legend of Hercules	4.4	533
20533	Jonah Hex	4.4	420
87818	Movie 43	4.4	797

3.2.2 Movies with the highest vote average

In this secton, I am displaying the top20 of movies with the highest vote_average that received at least 250 votes (first column, yellow numbers).

Code

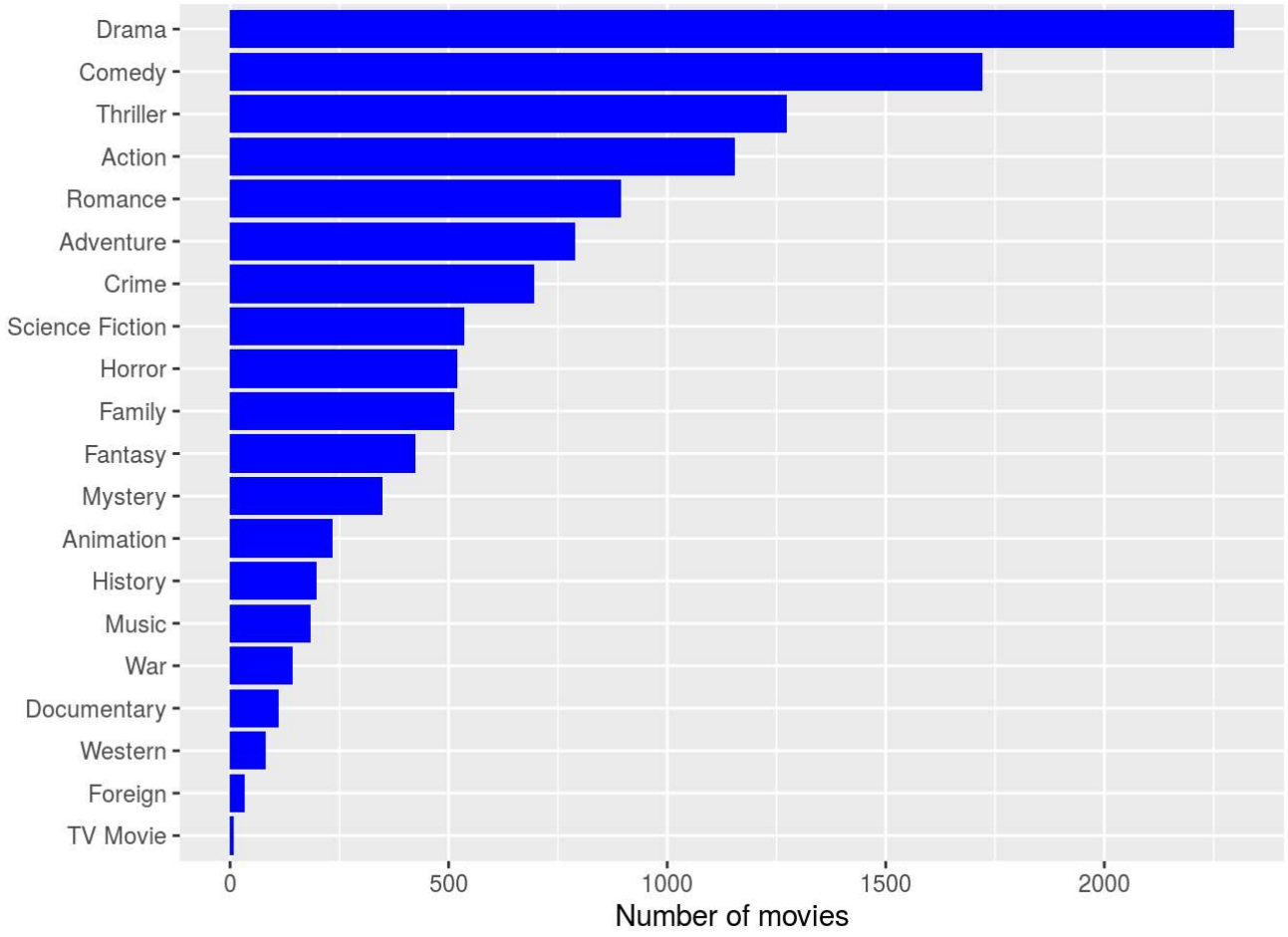


3.3 Genres

3.3.1 Numbers of movies by genre

Please be aware that a movie can be labeled with multiple genres.

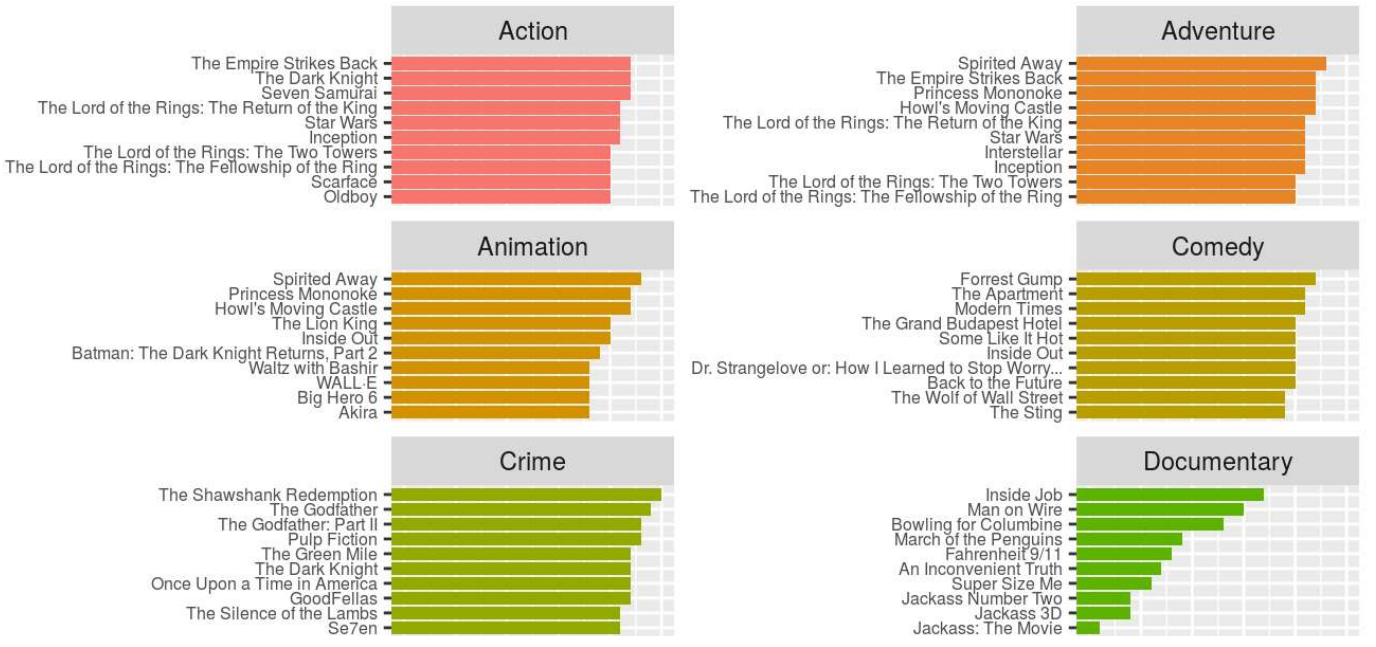
Code

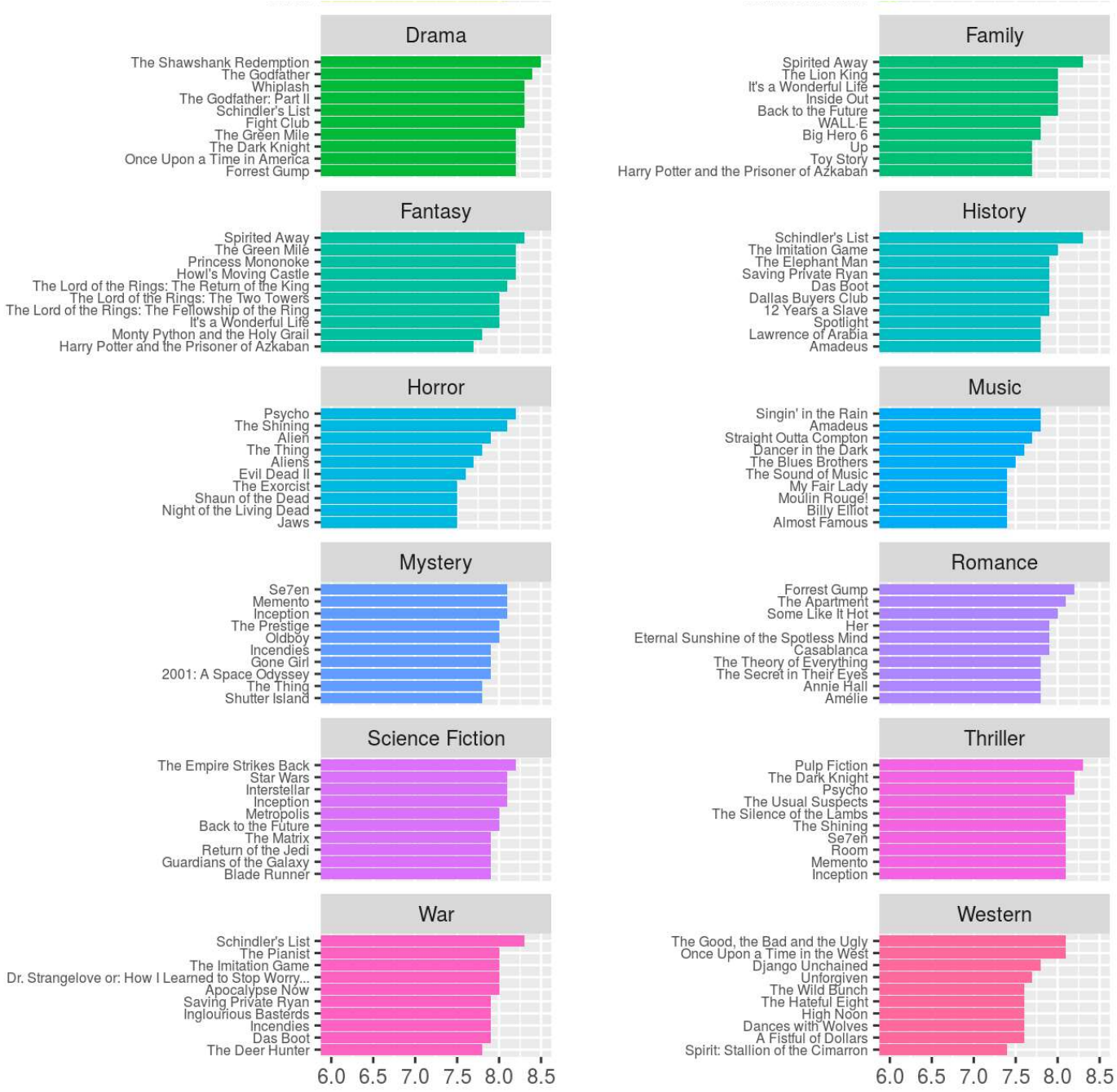


3.3.2 Highest rated movies by genre

In this section, I am displaying the Top10 of highest rated moves by genre. Again, I have only taken movies with at least 250 votes into consideration, and I am displaying all genres except the small genres “Foreign” and “TV movie”.

Code



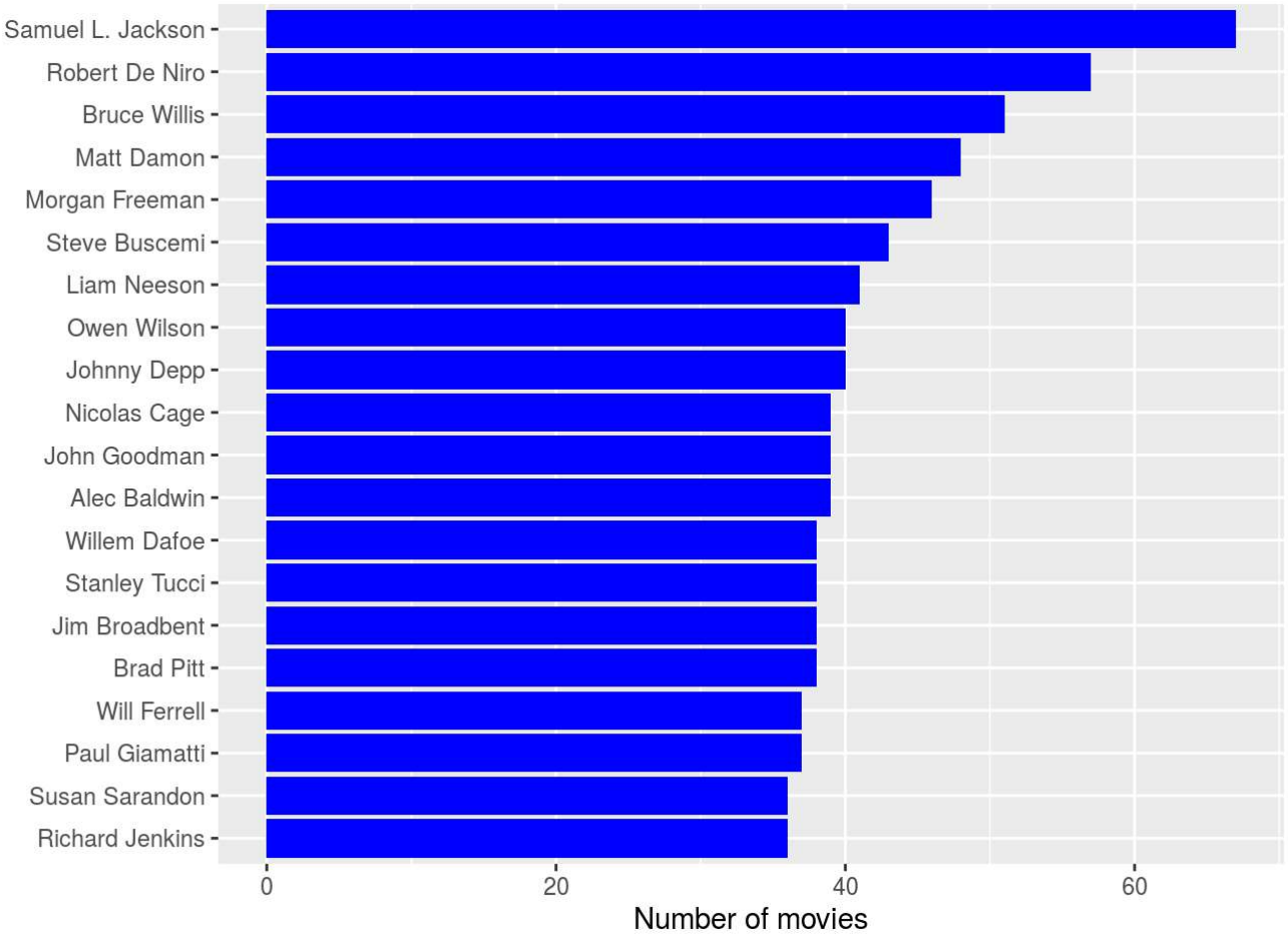


3.4 Popular actors and directors

3.4.1 Actors with most appearances

Below, you can find the 20 actors with most appearances.

Code



3.4.1.1 Top 10 profitable movies in TMDB

```
##
## Attaching package: 'formattable'
```

```
## The following objects are masked from 'package:scales':
##
##     comma, percent, scientific
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   budget = col_double(),
##   id = col_double(),
##   popularity = col_double(),
##   release_date = col_date(format = ""),
##   revenue = col_double(),
##   runtime = col_double(),
##   vote_average = col_double(),
##   vote_count = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

```
## Warning in prettyNum(.Internal(format(x, trim, digits, nsmall, width, 3L, :
## 'big.mark' and 'decimal.mark' are both ',', which could be confusing
```

```
## Selecting by profit
```

title	profit
Avatar	2,550,965,087
Titanic	1,645,034,188
Jurassic World	1,363,528,810
Furious 7	1,316,249,360
The Avengers	1,299,557,910
Avengers: Age of Ultron	1,125,403,694
Frozen	1,124,219,009
Minions	1,082,730,962
The Lord of the Rings: The Return of the King	1,024,888,979
Iron Man 3	1,015,439,994

3.5 Keywords

3.5.1 Number of distinct keywords in the database

Altogether, TMBD uses 9,813 keywords to describe its movies.

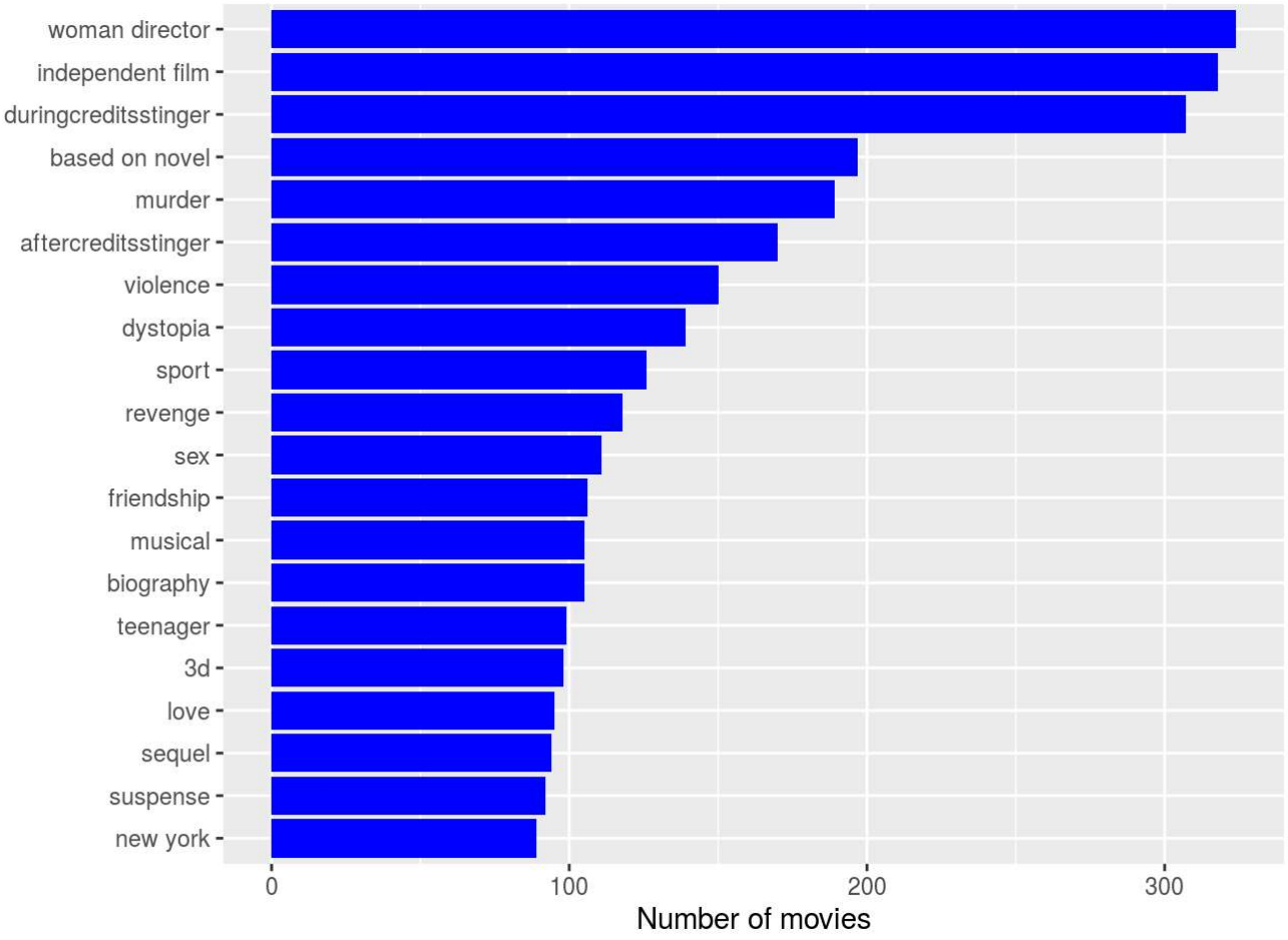
Code

```
## [1] 9813
```

3.5.2 Most-used Keywords

You can find the Top-20 most-used keywords below. As I did not know what “during the credits stinger” was, I looked it up. Apparently, such a stinger (after or during the credits) is an extra/special scene.

Code



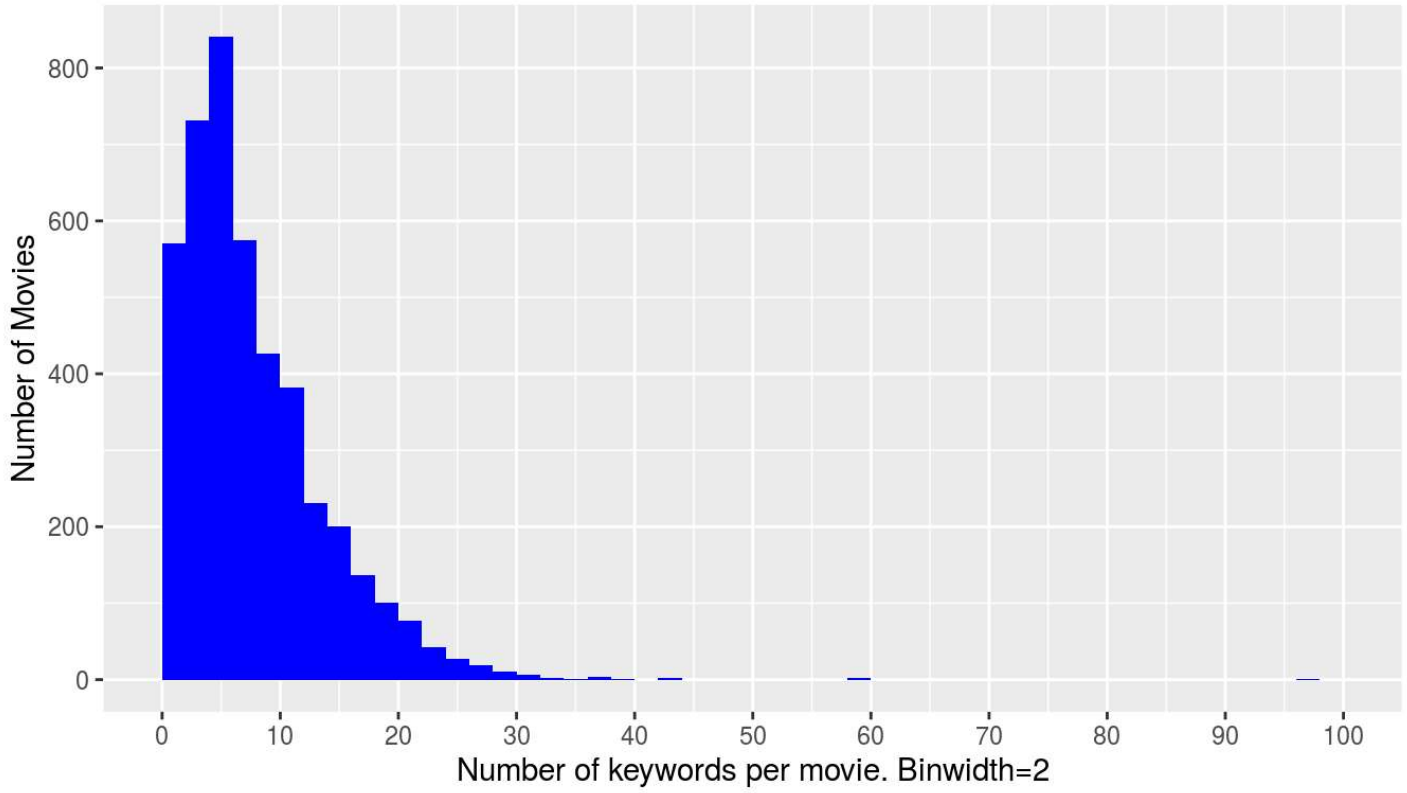
Below, you can find some more keywords in a wordcloud.

Code

3.5.3 Histogram of the number of keywords per movie

There is a small number of outliers with lots of keywords.

Code



4 Text analysis on tagline

Have you heard about these taglines? “Man is the warmest place to hide”. “Family isn’t a word. It’s a sentence”. These tagline are so memorable and precisely capture the spirit of the movie.

Let’s explore what are high frequency words the movie marketers would like to use. Tidy the “tagline” column in TMDB dataset.

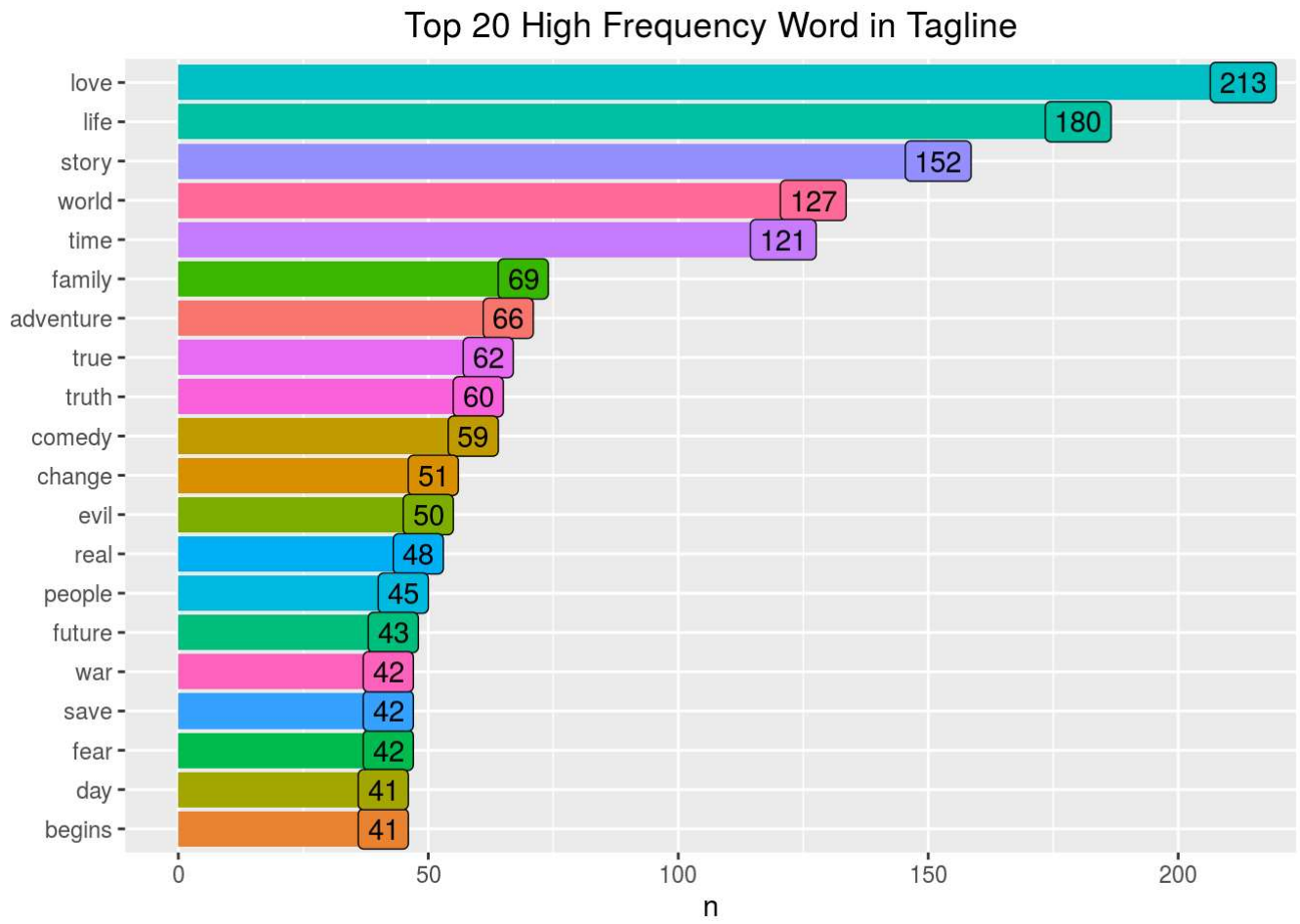
The following two barplots clearly shows that: * “love”, “life”, “story”, “world” and “family” are the most 5 frequent words in tagline. * “love story”, “motion picture” and “romantic comedy” are the top 3 most frequent pair words in tagline.

This supports that the most 2 popular movie genre are “Drama” and “Comedy” .

4.1 High frequency words | 4.2 High frequency bi-grams

```
## Joining, by = "word"
```

```
## Selecting by n
```



5 Explore production companies

Covert “production_companies” into data.frame similar to genre. Use the formattable package to achieve a better visualization.

From the following three tables, I found an interesting phenomenon:

“Warner Bros.”, “Universal Pictures”, “Paramount Pictures” and “Twentieth Century Fox Film Corporation 222” have the most release and revenue. However, they are not listed in the top 10 profitable companies.

I could provide a reasonable explanation for this: as independent films that become more popular, you don’t necessarily need big budgets to get big returns in the movie world. There are many good examples such as “Little Miss Sunshine”, “The King’s Speech”“,”Lost in Thailand".

```
## Warning: unnest() has a new interface. See ?unnest for details.
## Try `df %>% unnest(c(js, .name_repair))`, with `mutate()` if needed
```

5.0.1 Top 10 companies based on release

```
## Selecting by count
```

company	count

company	count
Warner Bros.	319
Universal Pictures	311
Paramount Pictures	285
Twentieth Century Fox Film Corporation	222
Columbia Pictures	201
New Line Cinema	165
Metro-Goldwyn-Mayer (MGM)	122
Touchstone Pictures	118
Walt Disney Pictures	114
Relativity Media	102

5.0.2 Top 10 companies based on revenue

```
## Selecting by revenue (billion USD)
```

company	revenue (billion USD)
Pixar Animation Studios	9.81
Columbia Pictures Corporation	9.35
Revolution Sun Studios	8.29
TSG Entertainment	8.19
Ingenious Film Partners	7.91
Regency Enterprises	7.58
Jerry Bruckheimer Films	7.21
WingNut Films	7.08
Summit Entertainment	6.93
Lucasfilm	6.61

5.0.3 Top 10 companies based on profit

```
## Selecting by profit (billion USD)
```

company	profit (billion USD)

company	profit (billion USD)
DreamWorks SKG	9.97
Dune Entertainment	9.78
DreamWorks Animation	9.09
Village Roadshow Pictures	7.93
Marvel Studios	7.77
Legendary Pictures	7.71
Pixar Animation Studios	7.51
Relativity Media	7.39
TSG Entertainment	6.08
Revolution Sun Studios	6.06