DATA WRANGLING

FINAL PROJECT REPORT

Github- https://github.com/Varsha306/Data-Wrangling-Project

INTRODUCTION

In this project, I have cleaned and explored the MovieLens latest dataset available on the website of the GroupLens Research Lab at the University of Minnesota. The dataset consists of 100,000 ratings and 3,600 tag applications applied to 9,000 movies by 600 users. The movieLens dataset is most often used for building recommender systems which recommends movies to users and also predicts user movie ratings based on other users' ratings. But in this project, I will try to gain some insights on the movies data.

WEB SCRAPING

The other data source I used was a website, https://www.comparitech.com/tv-streaming/netflix-subscribers/ from where I scraped a table containing the following columns:

a) Country

b) # of Subscribers (2018)

c) Average Monthly Revenue per Paying Membership

d) Total Yearly Revenue from Paid Memberships (2018)

e) # of Subscribers (June 19)

f) Total from Paid Memberships (First Half of 2019)

g) # of Subscribers (Dec 19)

h) Total from Paid Memberships (Second Half of 2019)

i) Total from Paid Memberships (Second Half of 2019)

Country	# of Subscribers (2018)	Average Monthly Revenue per Paying Membership	Total Yearly Revenue from Paid Memberships (2018)	# of Subscribers (June 19)	Total from Paid Memberships (First Half of 2019)	# of Subscribers (Dec 19)	Total from Paid Memberships (Second Half of 2019)	Total Estimated Revenue by End of 2019
United States	58,486,000	$11.40	$7,646,647,000.00	60,103,000	$4,372,744,000.00	61,761,843	$4,224,510,047.52	$8,597,254,047.52
Australia	11,262,000	$9.43	$1,274,407,920.00	12,751,963	$721,506,043.91	14,439,047	$816,961,293.52	$1,538,467,337.43
United Kingdom	9,780,000	$9.43	$1,106,704,800.00	11,073,894	$626,560,922.52	12,538,970	$709,454,932.57	$1,336,015,855.09
Brazil	8,500,000	$9.43	$961,860,000.00	9,624,550	$544,557,039.00	10,897,878	$616,601,935.26	$1,161,158,974.26
Canada	6,300,000	$9.43	$712,908,000.00	7,133,490	$403,612,864.20	8,077,251	$457,010,846.13	$860,623,710.33
Germany	5,100,000	$9.43	$577,116,000.00	5,774,730	$326,734,223.40	6,538,727	$369,961,161.16	$696,695,384.56
France	5,000,000	$9.43	$565,800,000.00	5,661,500	$320,327,670.00	6,410,516	$362,707,020.74	$683,034,690.74
Spain	4,000,000	$9.43	$452,640,000.00	4,529,200	$256,262,136.00	5,128,413	$290,165,616.59	$546,427,752.59
Japan	3,274,276	$9.43	$370,517,072.16	3,707,463	$209,768,240.40	4,197,960	$237,520,578.61	$447,288,819.01
Netherlands	2,940,000	$9.43	$332,690,400.00	3,328,962	$188,352,669.96	3,769,384	$213,271,728.20	$401,624,398.16

Figure 1. Original Table

Country <chr>	Subscriber_count_18 <chr>	Avg_Monthly_Revenue_18 <chr>	Total_Yearly_Revenue_18 <chr>	Subscriber_count_june19 <chr>	Total_Revenue_june19 <chr>	Subscriber_count_dec19 <chr>
United States	58,486,000	$11.40	$7,646,647,000.00	60,103,000	$4,372,744,000.00	61,761,843
Australia	11,262,000	$9.43	$1,274,407,920.00	12,751,963	$721,506,043.91	14,439,047
United Kingdom	9,780,000	$9.43	$1,106,704,800.00	11,073,894	$626,560,922.52	12,538,970
Brazil	8,500,000	$9.43	$961,860,000.00	9,624,550	$544,557,039.00	10,897,878
Canada	6,300,000	$9.43	$712,908,000.00	7,133,490	$403,612,864.20	8,077,251
Germany	5,100,000	$9.43	$577,116,000.00	5,774,730	$326,734,223.40	6,538,727
France	5,000,000	$9.43	$565,800,000.00	5,661,500	$320,327,670.00	6,410,516
Spain	4,000,000	$9.43	$452,640,000.00	4,529,200	$256,262,136.00	5,128,413
Japan	3,274,276	$9.43	$370,517,072.16	3,707,463	$209,768,240.40	4,197,960
Netherlands	2,940,000	$9.43	$332,690,400.00	3,328,962	$188,352,669.96	3,769,384

Figure 2. Scraped Table

DATA CLEANING

Here we load the ratings, movies and tags files and perform necessary cleaning and some transformations to make the data suit our needs.

Ratings:

In the ratings data we have 100,836 ratings and 4 variables: userId: :movieId: :rating: :timestamp. After converting the timestamp variable to datetime format I checked for NA's and didn't find any. The summary of the data shows that the ratings range between 0.5 and 5.

userId <dbl>	movieId <dbl>	rating <dbl>	timestamp <S3: POSIXct>
1	1	4.0	2000-07-30 18:45:03
1	3	4.0	2000-07-30 18:20:47
1	6	4.0	2000-07-30 18:37:04
1	47	5.0	2000-07-30 19:03:35
1	50	5.0	2000-07-30 18:48:51
1	70	3.0	2000-07-30 18:40:00
1	101	5.0	2000-07-30 18:14:28
1	110	4.0	2000-07-30 18:36:16
1	151	5.0	2000-07-30 19:07:21
1	157	5.0	2000-07-30 19:08:20

Figure 3. Ratings Data

Movies:

There are 9742 movies in the movies data and 3 variables: movieId: :title: :genre. The movie titles were of the form Title(year), so i split it into title and year to increase the scope of our analysis.

movieId <dbl>	titles <chr>	year <int>	genres <chr>
1	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
2	Jumanji	1995	Adventure|Children|Fantasy
3	Grumpier Old Men	1995	Comedy|Romance
4	Waiting to Exhale	1995	Comedy|Drama|Romance
5	Father of the Bride Part II	1995	Comedy
6	Heat	1995	Action|Crime|Thriller
7	Sabrina	1995	Comedy|Romance
8	Tom and Huck	1995	Adventure|Children
9	Sudden Death	1995	Action
10	GoldenEye	1995	Action|Adventure|Thriller

Figure 4. Movies Data

Tags:

In the tags data, we have about 3683 observations and 4 variables: userId: :movieId: :tags: :timestamp. Here too, I converted timestamp to datetime format.

userId <dbl>	movieId <dbl>	tag <chr>	timestamp <S3: POSIXct>
2	60756	funny	2015-10-24 19:29:54
2	60756	Highly quotable	2015-10-24 19:29:56
2	60756	will ferrell	2015-10-24 19:29:52
2	89774	Boxing story	2015-10-24 19:33:27
2	89774	MMA	2015-10-24 19:33:20
2	89774	Tom Hardy	2015-10-24 19:33:25
2	106782	drugs	2015-10-24 19:30:54
2	106782	Leonardo DiCaprio	2015-10-24 19:30:51
2	106782	Martin Scorsese	2015-10-24 19:30:56
7	48516	way too long	2007-01-25 01:08:45

Figure 5. Tags Data

Movielens:

The ratings and movieId data were merged to give the movielens data frame and the observations are ordered by movieId with 610 distinct users and 9724 distinct movies.

	movieId <dbl>	userId <dbl>	rating <dbl>	timestamp <S3: POSIXct>	titles <chr>	year <int>	genres <chr>
1	1	1	4.0	2000-07-30 18:45:03	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
2	1	555	4.0	2001-01-06 01:55:59	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
3	1	232	3.5	2004-02-16 18:20:21	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
4	1	590	4.0	2009-11-17 01:13:28	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
5	1	601	4.0	2018-03-19 13:56:41	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
6	1	179	4.0	1997-01-01 10:20:51	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
7	1	606	2.5	2012-10-01 09:15:50	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
8	1	328	5.0	2017-05-08 02:31:05	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
9	1	206	5.0	1996-12-16 19:07:47	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy
10	1	468	4.0	1996-05-06 16:34:04	Toy Story	1995	Adventure|Animation|Children|Comedy|Fantasy

DATA EXPLORATION

PART A : GENRES

- I created a data frame by separating each genre, grouping the movies data by these genres, and counting the number of movies in each group of genres. The output data frame consists of top 10 genres based on the number of movies. We can see that Drama genre has most number of movies while Fantasy genre has least number of movies.

genres	movies_count
Drama	4361
Comedy	3756
Thriller	1894
Action	1828
Romance	1596
Adventure	1263
Crime	1199
Sci-Fi	980
Horror	978
Fantasy	779

Figure 7. Count for each genre by movies

- I have arranged the genres by year and displayed the number of movies in a particular genre year wise.

year	genres	count
1902	Action	1
1902	Adventure	1
1902	Fantasy	1
1902	Sci-Fi	1
1903	Crime	1
1903	Western	1
1908	Animation	1
1908	Comedy	1
1908	Sci-Fi	1

Figure 8. Genre count by year

- Here, I created a dataframe in the same way as above except here, number of ratings were counted in each group of genre.The output data frame consists of top 10 genres based on the number of ratings. We can see that again Drama has the most total number of ratings. However, Children genre has the least total number of ratings.

genres	ratings_count
Drama	41928
Comedy	39053
Action	30635
Thriller	26452
Adventure	24161
Romance	18124
Sci-Fi	17243
Crime	16681
Fantasy	11834
Children	9208

Figure 9. Count for each genre by ratings

- To visualize the above result, I made a wordcloud. We notice that the “Drama” genre has the top number of movies ratings, followed by the “Comedy” and the “Action” genres.

Figure 10. Wordcloud for popular genres by ratings

- We can also visualize using a bar graph.

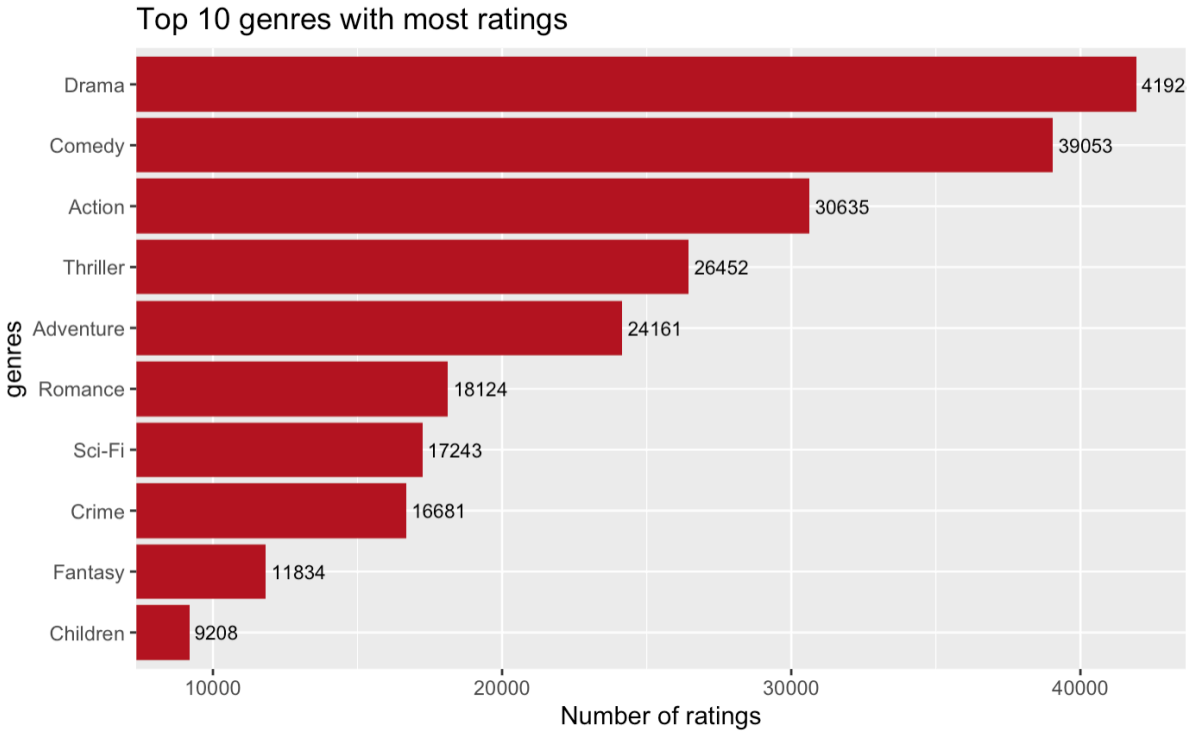

Figure 11. Top 10 genres with most ratings

- Here we have a function that returns the genre of the movie inputted in the function.

```{r}
get_genre <- function(titlex){
  movies %>%
    select(titles,genres) %>%
    group_by(genres,titles) %>%
    filter(titles==titlex)
}
get_genre('Jumanji')
```

titles	genres
<chr>	<chr>
Jumanji	Adventure|Children|Fantasy

Figure 12. Get genre

- In the next case, we look at how each movie genre is tagged by users. It is a great way to know how users describe the movies and what they think about it. Here we first join the movies and tags data by movieId after separating the genres, then group the data by genres and output each genre and their corresponding tags. Then we create a data frame which gives the group of tags for a particular genre which in this case is Comedy.

Figure 13. Wordcloud of tags that represent the genre Comedy

Part B: TITLES

- We get the count of the ratings for each movie and get the top 10 movies. Forrest Gump has the most number of ratings at 329, followed by Shawshank Redemption and Pulp Fiction.

title	count
Forrest Gump (1994)	329
Shawshank Redemption, The (1994)	317
Pulp Fiction (1994)	307
Silence of the Lambs, The (1991)	279
Matrix, The (1999)	278
Star Wars: Episode IV - A New Hope (1977)	251
Jurassic Park (1993)	238
Braveheart (1995)	237
Terminator 2: Judgment Day (1991)	224
Schindler's List (1993)	220

Figure 14. Top 20 movies with most ratings

- We visualize the above output in the bar graph below.

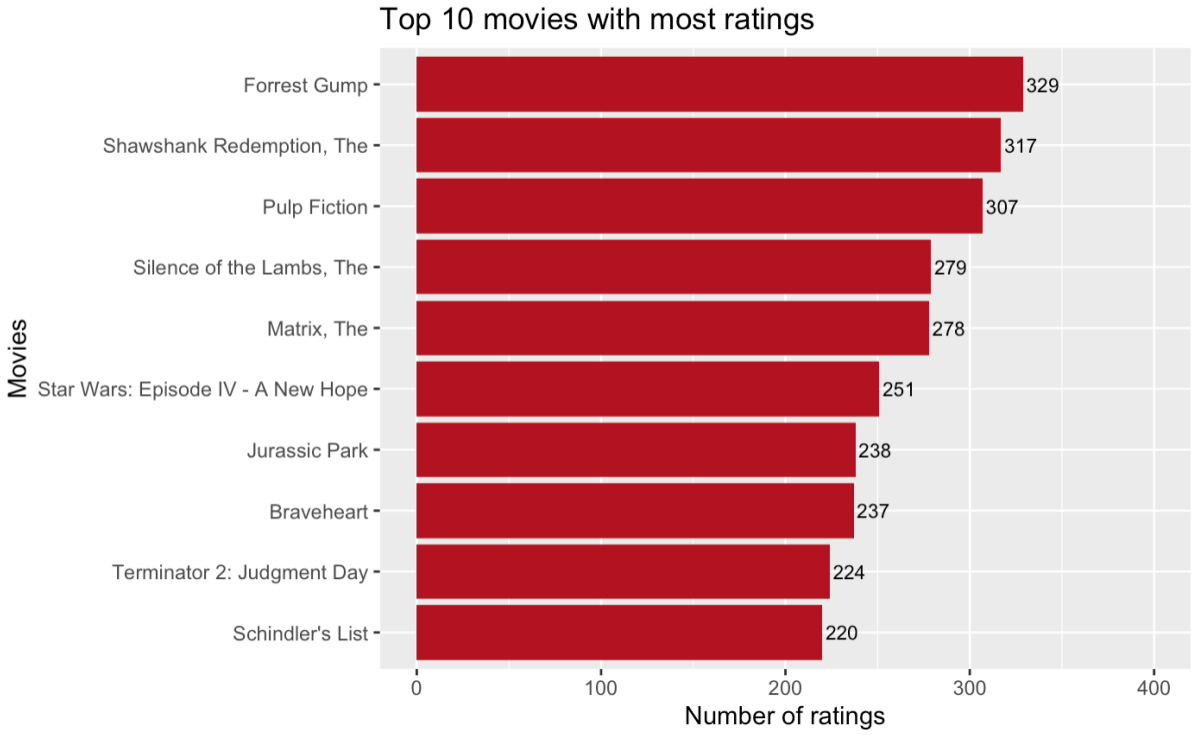

Figure 15. Top 20 movies with most ratings

- We can also visualize the same output using wordcloud.

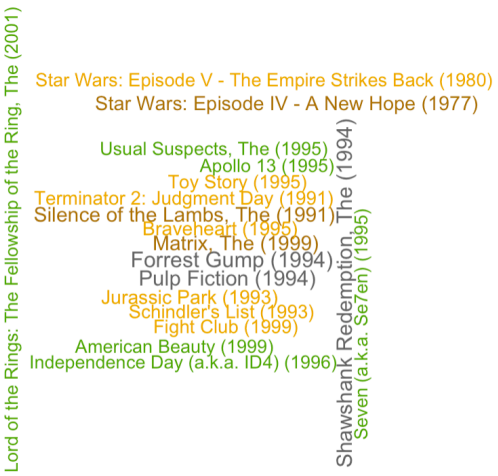

Figure 16. Wordcloud for top 20 titles with most ratings

- Here we have a bar graph which shows the distribution of ratings for the top 5 movies with the most number of ratings. The categories here are Very bad (0-1], bad (1-2], Average (2,3], Good (3,4] and Very good (4,5]. Forrest gump and Silence of the Lambs have most ratings in the Good category while Matrix , Pulp Fiction and Shawshank Redemption have mostly 'Very good' ratings.

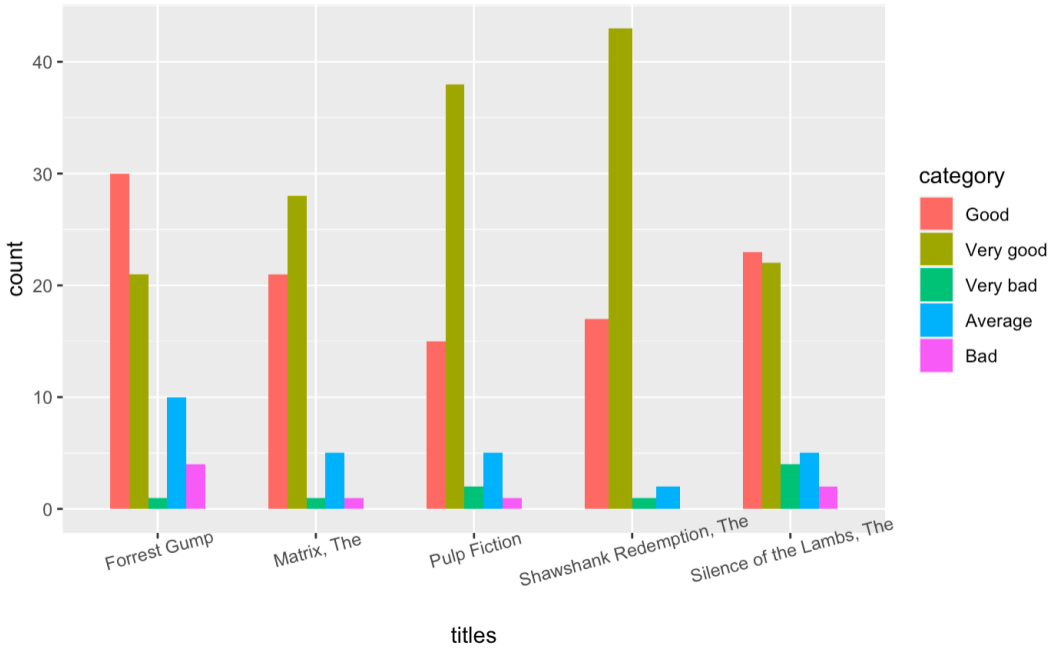

Figure 17. Bar plot for top 5 Movies and their rating distribution

- I have also created a function that returns all the movies in the data that were released in the year inputted in the function

```
get_movies(1995)
```

titles <chr>	year <int>
Toy Story	1995
Jumanji	1995
Grumpier Old Men	1995
Waiting to Exhale	1995
Father of the Bride Part II	1995
Heat	1995
Sabrina	1995
Tom and Huck	1995
Sudden Death	1995
GoldenEye	1995

Figure 18. Get movies

Part C: RATINGS

- Here, we explore the ratings. Since our ratings are on a scale of 1-5, we divide the ratings into five groups- Very bad(0-1], bad(1-2],Average(2,3],Good(3,4]and Very good(4,5]. We have plotted the rating against its count. We can see that most people have given a rating of 4, followed by rating of 3. Not a single user has given a zero rating.

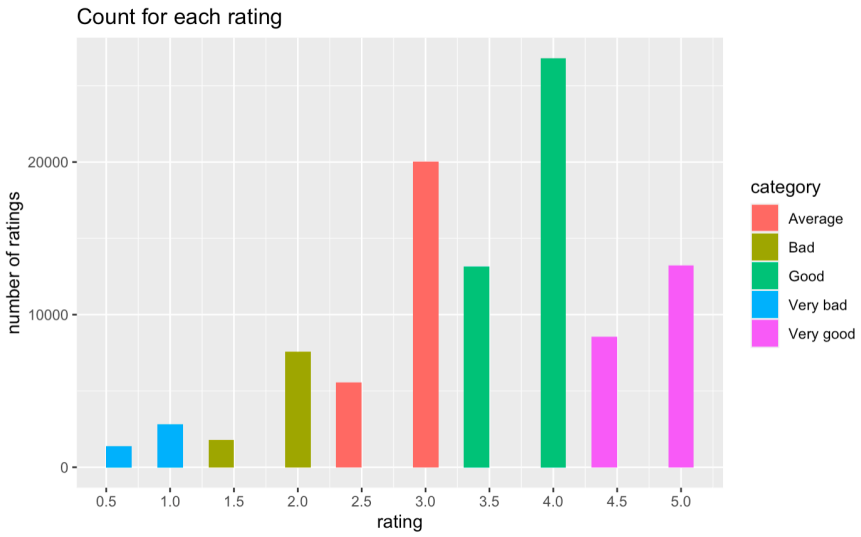

Figure 19. Count for each rating

- Here, we have a boxplot for ratings of different genre. There is no distribution for the ratings of Adventure genre since all the ratings for this genre are 4. For the Drama genre, the median is the same as the highest rating recorded i.e., 4. Also there are 2 outliers for this genre at 1 and 0.5. The thriller genre too has an outlier at 1.

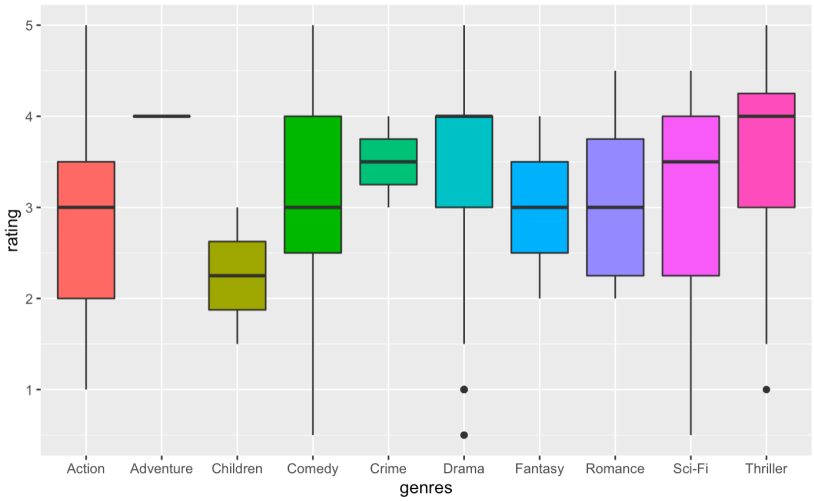

Figure 20. Boxplot for top 10 genres and their ratings distribution

- Next, we calculate the average, minimum and maximum rating for each for movie by first selecting movieId, title and rating from movielens data, grouping them by movieId and title and computing the mean, min and max of ratings in each group which is outputted in descending order of number of ratings for each movie. We can see that Forrest Gump has maximum number of rating with mean rating of 4.16, minimum rating of 0.5 and maximum rating of 5. Since mean is closer to the maximum rating, we can say that most people liked the movie

movieId <dbl>	title <chr>	count <int>	mean <dbl>	min <dbl>	max <dbl>
356	Forrest Gump (1994)	329	4.164134	0.5	5.0
318	Shawshank Redemption, The (1994)	317	4.429022	1.0	5.0
296	Pulp Fiction (1994)	307	4.197068	0.5	5.0
593	Silence of the Lambs, The (1991)	279	4.161290	0.5	5.0
2571	Matrix, The (1999)	278	4.192446	0.5	5.0
260	Star Wars: Episode IV - A New Hope (1977)	251	4.231076	0.5	5.0
480	Jurassic Park (1993)	238	3.750000	0.5	5.0
110	Braveheart (1995)	237	4.031646	0.5	5.0
589	Terminator 2: Judgment Day (1991)	224	3.970982	0.5	5.0
527	Schindler's List (1993)	220	4.225000	0.5	5.0

Figure 21. Average rating for each movie

- We can see the discrepancy her as the movie with movieId has the most number of ratings and the difference in ratings between this movie and the next most rated movie is huge.

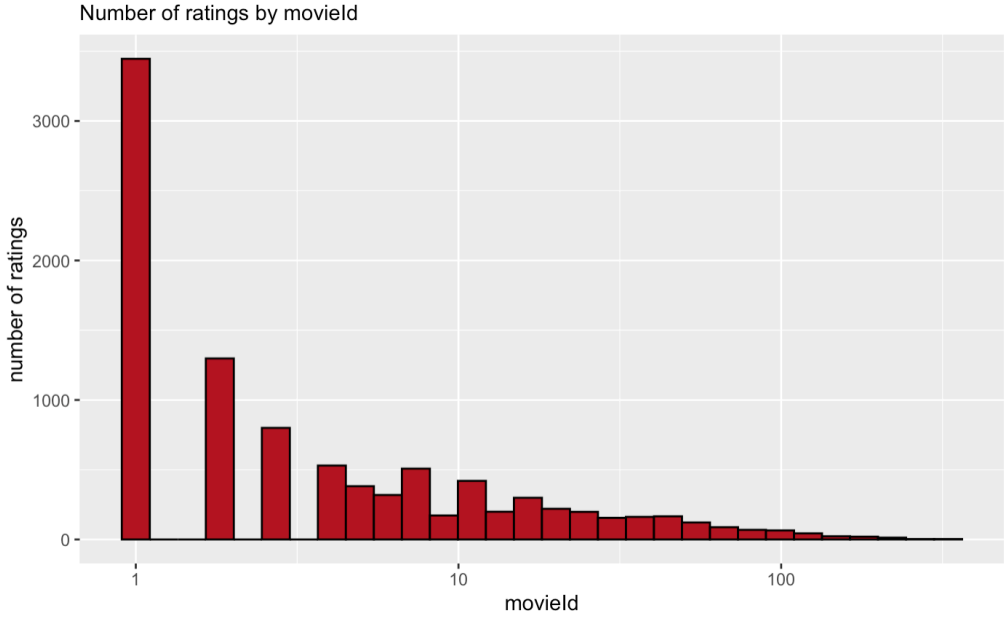

Figure 22. Number of ratings vs movieId

- We can see that there some users who are more active than the others.

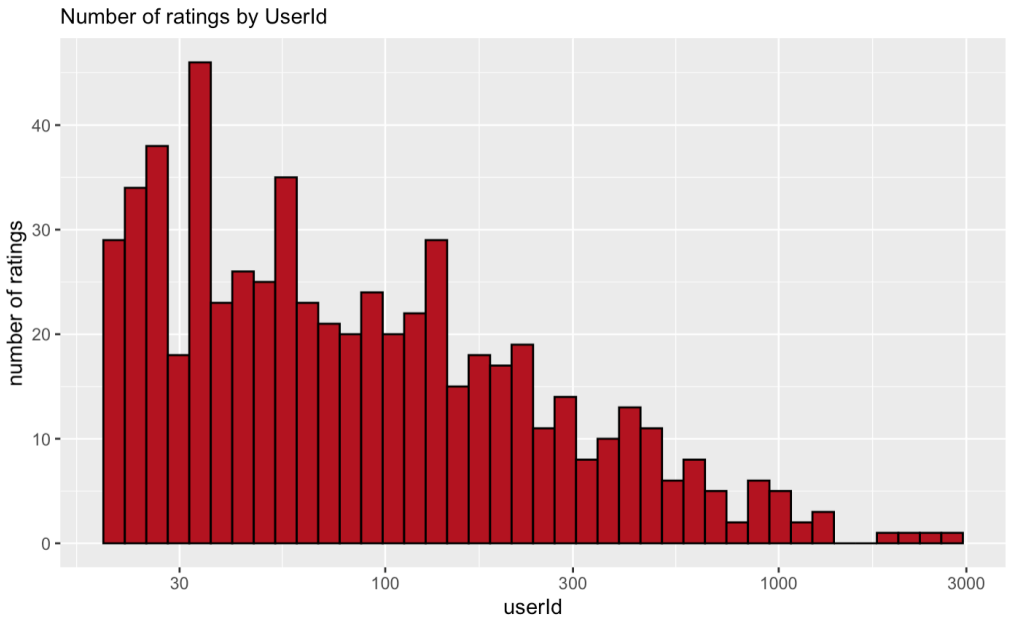

Figure 22. Number of ratings vs userId

CONCLUSION

We have seen that the movielens data can be explored in many ways and analyzing the dataset gave many interesting insights into the movie business. Although it is mainly used for recommendation systems, we were still able to extract some trends in the data.