# MovieLens Capstone Project

Ujjawal Madan

14/02/2020

## Contents

# 1 Introduction

Inspired by the 2006 Netflix Challenge which "sought to substantially improve the accuracy of predictions about how much someone is going to enjoy a movie based on their movie preferences", in this project I build a movie recommendation system using the movielens dataset. The first prize winner of the Netflix Challenge, BellKor's Pragmatic Chaos team won the million dollar-prize in late 2009 after improving Netflix's own algorithm by approximately 10.06 percent (RMSE) and achieving an RMSE of 0.8567 - the primary component of their solution being matrix factorization. While matrix factorization can certainly be utilized in R (e.g. using the recommenderlab package or other similar packages) and could be used in this context, this project is rather based upon previous coursework in the Harvard Data Science Certificate Program. As the Netflix data is not publicly available, the GroupLens research lab114 generated movielens dataset (10M version) is utilized, which contains approximately 10 million ratings of over 27,000 movies by more than 138,000 users. The aim of this recommendation system is to try to predict the rating based primarily on individual movie average, individual user average, genre average and time elapsed between year of review and year of release. The final metric used to judge the recommendation system is the Root Mean Squared Error (RMSE) value, the same metric used in the Netflix Challenge.

10M version of the movieLens dataset available here https://grouplens.org/datasets/movielens/10m/

## 1.1 Split Raw Data: Training and Validation Sets

The raw data, which is the movielens dataset, is split into a training set and a validation set (90% training: 10% validation). Within the training set there will be another split for internal training and testing (90% training: 10% testing) and the validation set will not be utilized until the very end when the final model is tested on the validation set.

## 1.2 Metrics Used to Evaluate Model

While there are variety of methods one could use to evaluate a regression analysis, such as Mean Average Error or Mean Absolute Error, the metric utilized in the Netflix Challenge is the Root Mean Squared Error. The RMSE is the square root of the averaged squared difference between the target value and the value predicted by the model and is preferred in this context as it poses a higher penalty on large errors. It is therefore the metric used to evaluate the models.

# 2 Data Visualization

## 2.1 General

| | userId | movieId | rating | timestamp | title | genres |
|---|---|---|---|---|---|---|
| 1 | 1 | 122 | 5 | 838985046 | Boomerang (1992) | Comedy\|Romance |
| 2 | 1 | 185 | 5 | 838983525 | Net, The (1995) | Action\|Crime\|Thriller |
| 4 | 1 | 292 | 5 | 838983421 | Outbreak (1995) | Action\|Drama\|Sci-Fi\|Thriller |
| 5 | 1 | 316 | 5 | 838983392 | Stargate (1994) | Action\|Adventure\|Sci-Fi |
| 6 | 1 | 329 | 5 | 838983392 | Star Trek: Generations (1994) | Action\|Adventure\|Drama\|Sci-Fi |
| 7 | 1 | 355 | 5 | 838984474 | Flintstones, The (1994) | Children\|Comedy\|Fantasy |
| 8 | 1 | 356 | 5 | 838983653 | Forrest Gump (1994) | Comedy\|Drama\|Romance\|War |
| 9 | 1 | 362 | 5 | 838984885 | Jungle Book, The (1994) | Adventure\|Children\|Romance |
| 10 | 1 | 364 | 5 | 838983707 | Lion King, The (1994) | Adventure\|Animation\|Children\|Drama\|Musical |
| 11 | 1 | 370 | 5 | 838984596 | Naked Gun 33 1/3: The Final Insult (1994) | Action\|Comedy |

Above are the top 10 rows of the edx dataset. The userId is a unique user identification number specific to each unique user and the movieId, similar to the userId, is also an identification number specific to each unique movie. The movie rating is on a scale between 0 and 5 in half point increments. The timestamp is the time of review and ranges from _ to asd. There is also a corresponsing title for each movieId and there are approximately 800 genres.

Table 1: Number of Distinct Movies, Distinct Users, and Distinct Genres

| num_users | num_movies | num_genres |
|---|---|---|
| 69878 | 10677 | 797 |

The above table specifies the number of distinct users, number of distinct movies and number of distinct genres in the edx set.

Table 2: Movies that Have Received the Most Ratings

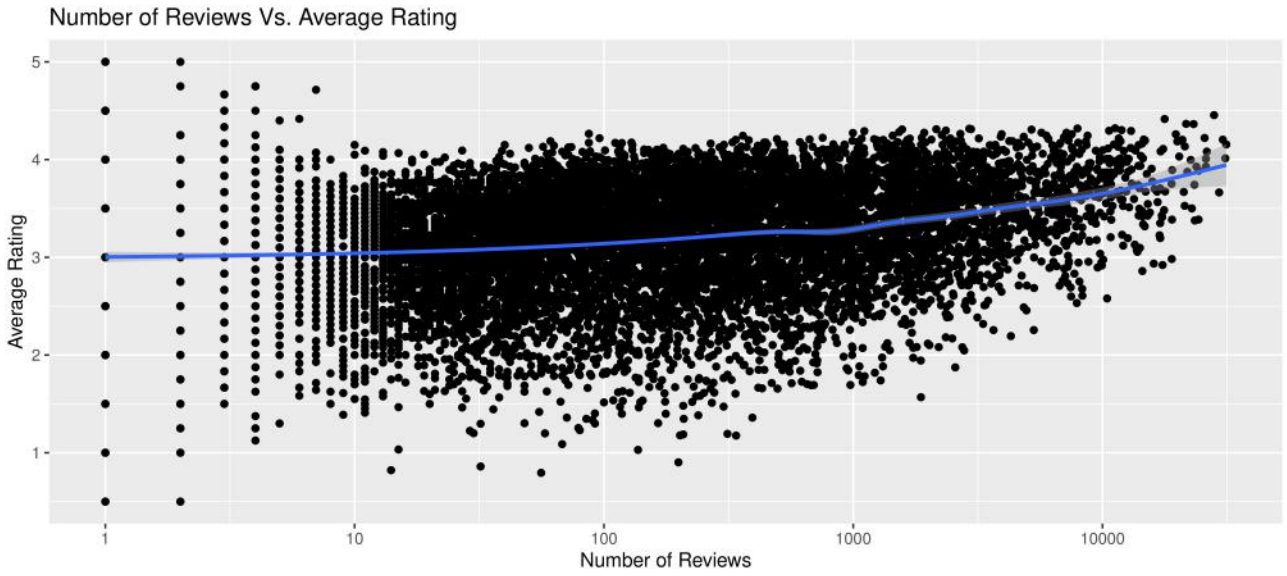
| title | num_reviews | avg_rating |
|---|---|---|
| Pulp Fiction (1994) | 31362 | 4.154789 |
| Forrest Gump (1994) | 31079 | 4.012822 |
| Silence of the Lambs, The (1991) | 30382 | 4.204101 |
| Jurassic Park (1993) | 29360 | 3.663522 |
| Shawshank Redemption, The (1994) | 28015 | 4.455131 |
| Braveheart (1995) | 26212 | 4.081852 |
| Fugitive, The (1993) | 25998 | 4.009155 |
| Terminator 2: Judgment Day (1991) | 25984 | 3.927859 |
| Star Wars: Episode IV - A New Hope (a.k.a. Star Wars) (1977) | 25672 | 4.221311 |
| Apollo 13 (1995) | 24284 | 3.885789 |

The most reviewed movies seem to be quite popular, likely to be familiar to many - receiving approximately 30,000 reviews. Curiously, they are also highly rated and are from the early 1990s. We will later explore the links between number of reviews versus average rating as well as the year of release versus average rating.

Table 3: Movies that Have Received the Least Ratings

| title | num_reviews | avg_rating |
|---|---|---|
| 1, 2, 3, Sun (Un, deuz, trois, soleil) (1993) | 1 | 2.0 |
| 100 Feet (2008) | 1 | 2.0 |
| 4 (2005) | 1 | 2.5 |
| Accused (Anklaget) (2005) | 1 | 0.5 |
| Ace of Hearts (2008) | 1 | 2.0 |
| Ace of Hearts, The (1921) | 1 | 3.5 |
| Adios, Sabata (Indio Black, sai che ti dico: Sei un gran figlio di...) (1971) | 1 | 1.5 |
| Africa addio (1966) | 1 | 3.0 |
| Aleksandra (2007) | 1 | 3.0 |
| Bad Blood (Mauvais sang) (1986) | 1 | 4.5 |

The movies that received less reviews seem to have lower ratings. This is in line with the previous table which hinted at a positive correlation between number of reviews and average rating.




Do more ratings mean higher average rating? The trend in the above plot clearly indicates that a higher number of ratings means that is it more likely the average rating of that movie will also be higher.

Table 4: Top 10 Best Performing Movies

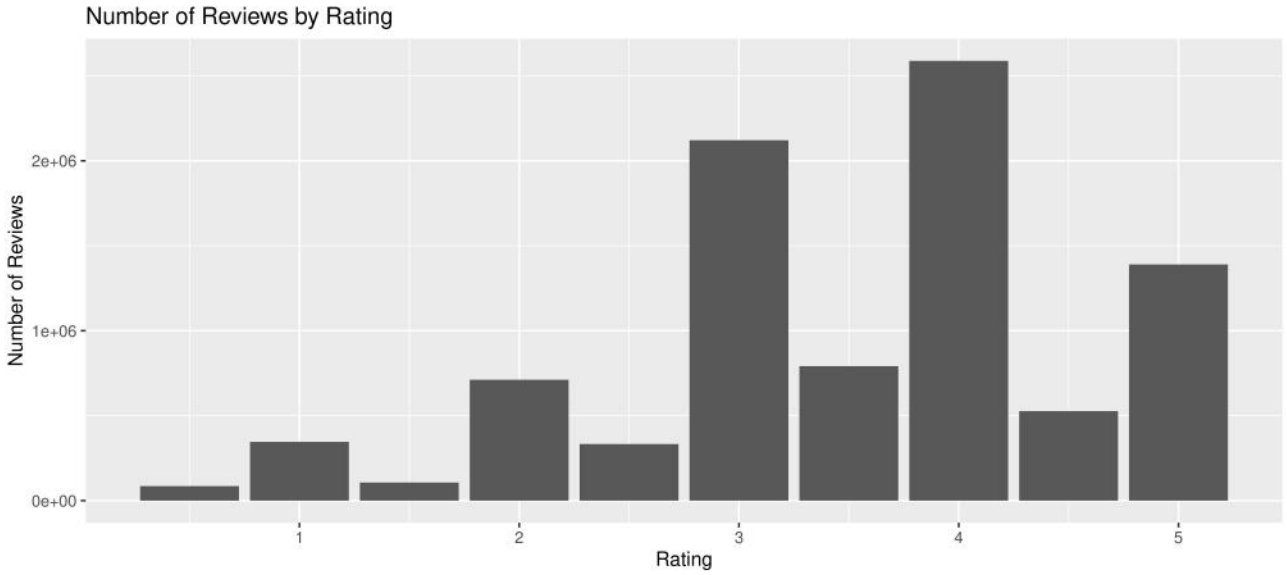
| title | avg_rating | num_reviews |
|---|---|---|
| Shawshank Redemption, The (1994) | 4.455131 | 28015 |
| Godfather, The (1972) | 4.415366 | 17747 |
| Usual Suspects, The (1995) | 4.365854 | 21648 |
| Schindler's List (1993) | 4.363493 | 23193 |
| Casablanca (1942) | 4.320424 | 11232 |
| Rear Window (1954) | 4.318651 | 7935 |
| Sunset Blvd. (a.k.a. Sunset Boulevard) (1950) | 4.315880 | 2922 |
| Third Man, The (1949) | 4.311426 | 2967 |
| Double Indemnity (1944) | 4.310817 | 2154 |
| Paths of Glory (1957) | 4.308721 | 1571 |

As we can see above, three of the movies that are in this list are also in the 'Movies that have Received the Most Ratings' table as well.

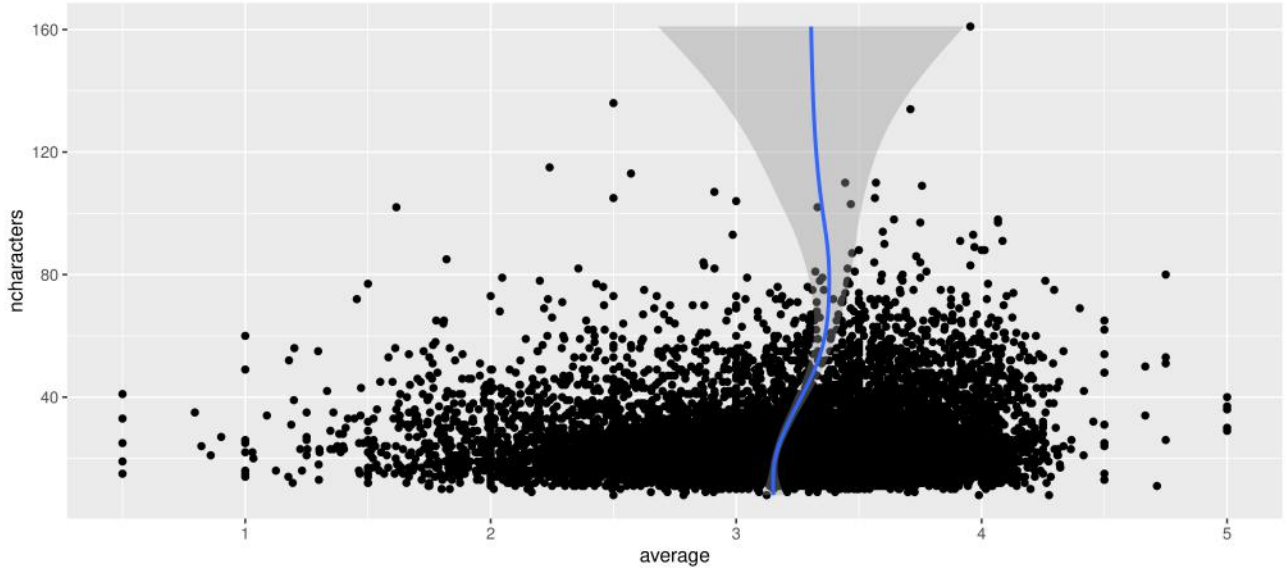
Table 5: Top 10 Worst Performing Movies

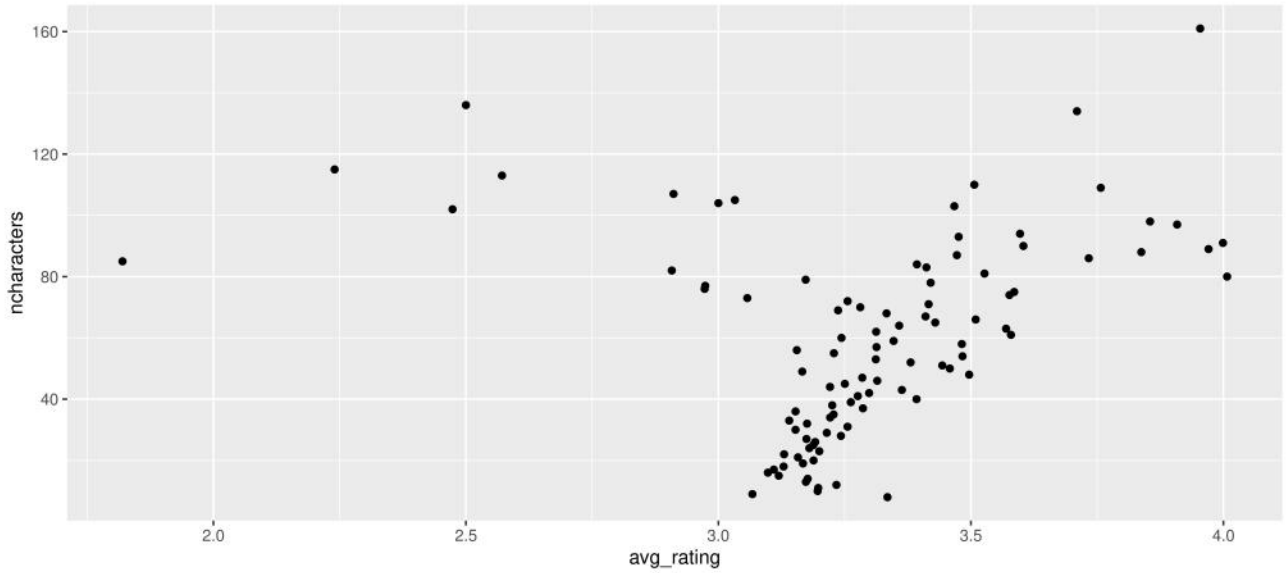
| title | avg_rating | num_reviews |
|---|---|---|
| Battlefield Earth (2000) | 1.568218 | 1869 |
| Police Academy 6: City Under Siege (1989) | 1.723443 | 1092 |
| Spice World (1997) | 1.739907 | 1288 |
| Police Academy 5: Assignment: Miami Beach (1988) | 1.783078 | 1111 |
| Jaws 3-D (1983) | 1.794202 | 1052 |
| Home Alone 3 (1997) | 1.818591 | 1221 |
| Speed 2: Cruise Control (1997) | 1.873733 | 2566 |
| Mighty Morphin Power Rangers: The Movie (1995) | 1.883739 | 1316 |
| Look Who's Talking Now (1993) | 1.883853 | 1059 |
| Grease 2 (1982) | 1.943462 | 1866 |

Above are the top 10 worst performing movies.



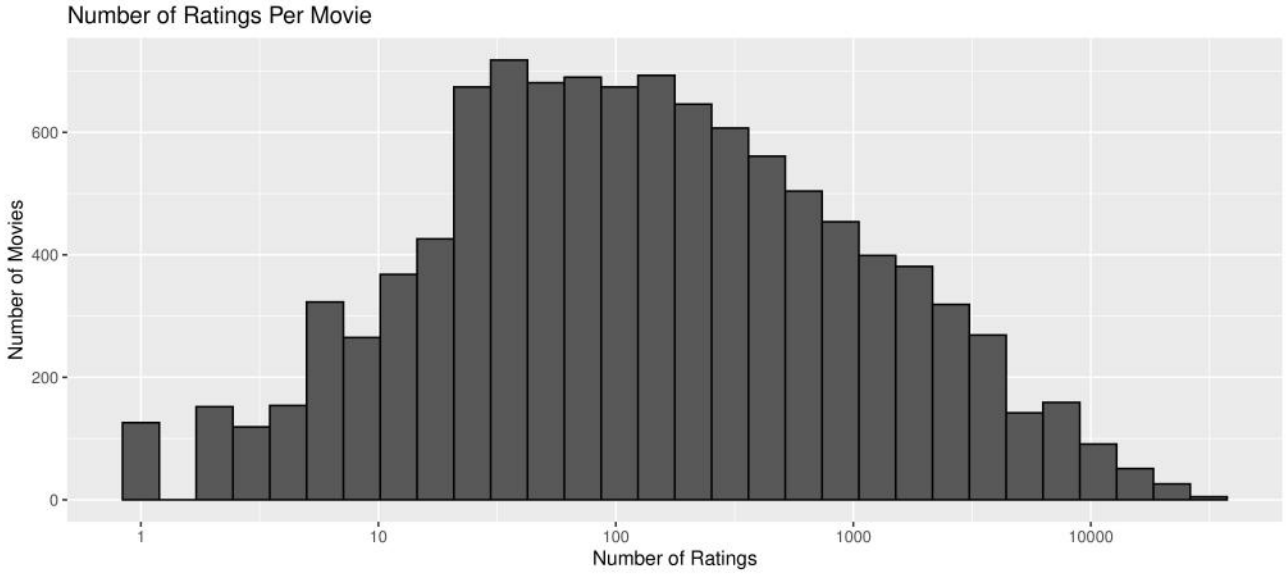

It seems clear that most users tends to give whole number reviews (e.g. 2,3,4) rather than in half increments (e.g. 2.5, 3.5, 4.5), the most common rating being 4/5.






Although this will not factor into our models, I was curious to see if the length of the title in any way corresponded with the average movie rating. It does not seem that there is a strong link although one can see that the average movie rating for movies with titles less than 40 characters is indeed less than the average movie rating.

## 2.2 Movie Reviews based on Individual Movies

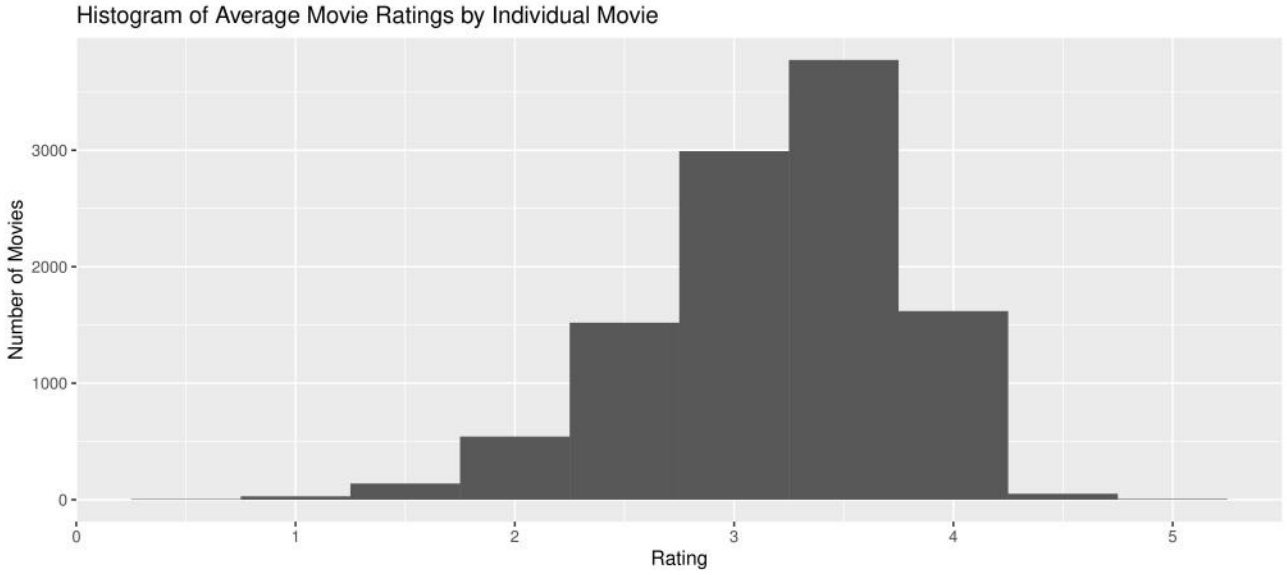



The above plot shows that the median movie receive approximately 100-150 reviews although some movies receive up to 30,000 reviews as well.

| avg_num_ratings | median_num_ratings |
|---|---|
| 842.9386 | 122 |

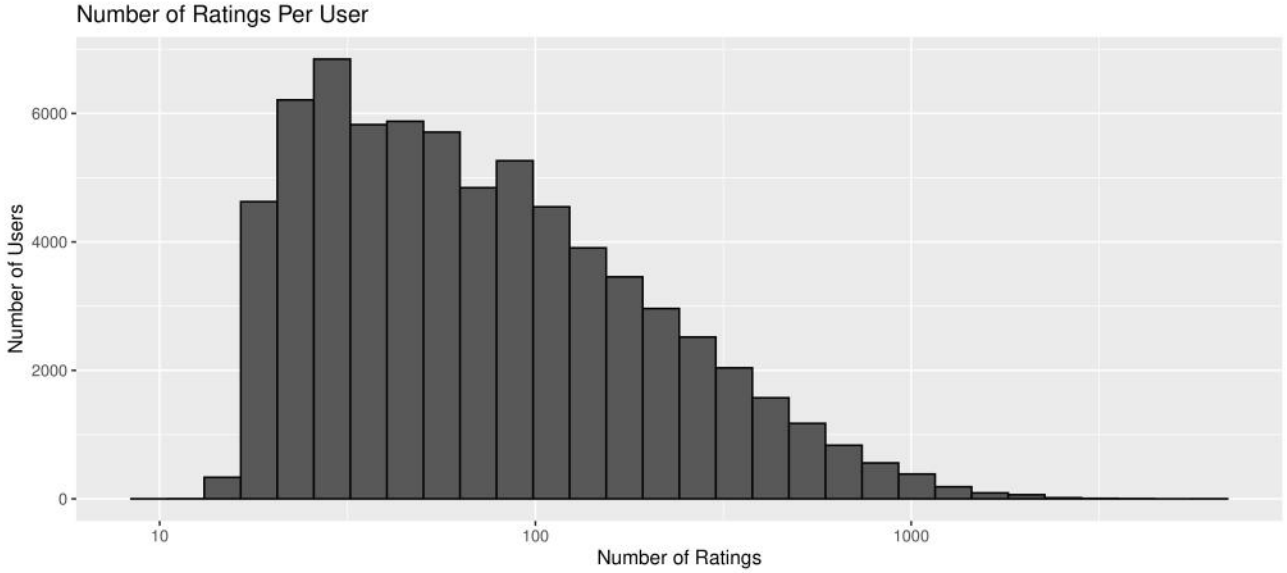
A movie receives on average 937 reviews and the median movie receives only 135 reviews. This tells us that the Number of Ratings per Movie plot is heavily skewed to the right and that there are some movies that are receving a disproportionate amount of attention.

| avg_rating | median_rating | sd_rating |
|---|---|---|
| 3.191736 | 3.267857 | 0.5713265 |




The most common movie rating seems to be between 3.25 and 3.75 and that the average movie ratings plot forms a normal distribution that is slighly skewed to the left.

## 2.3 Movie Reviews based on Individual Users

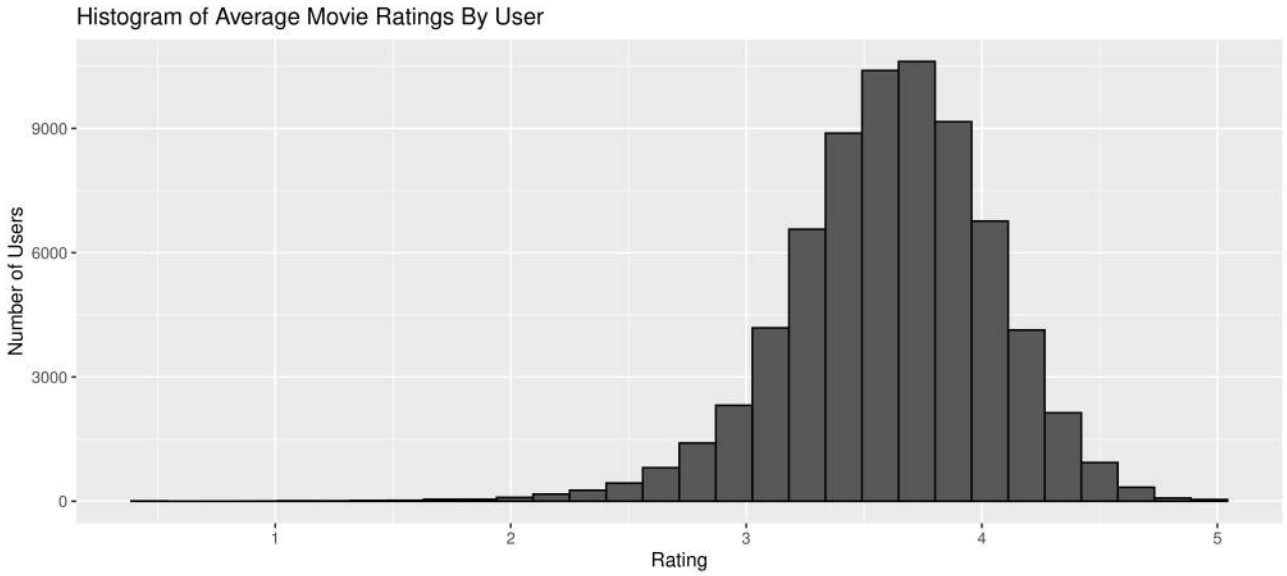



This plot, similar to Plot 1 of 2.2, is a histogram of the number of ratings made by each user. Once again, it seems heavily skewed to the right, with some users making very few reviews and some users reviewing up to thousands of movies.

| avg_num_ratings | med_num_ratings |
|---|---|
| 128.7967 | 62 |

A user on average makes 143 reviews, although the median user only makes 69 reviews.

| avg_rating | median_rating | sd_rating |
|---|---|---|
| 3.613602 | 3.635135 | 0.4306889 |




As shown above, the average user rating is approximately 3.6 and the above plot is a normal distribution. There seem to be some users that are more lenient than the average and some users that are much more critical. It is possible that the seeemingly lenient reviewers are only watching critically acclaimed movies and the more critical reviewers are watching only movies that have low ratings. However, this seems like an unlikely possibility.

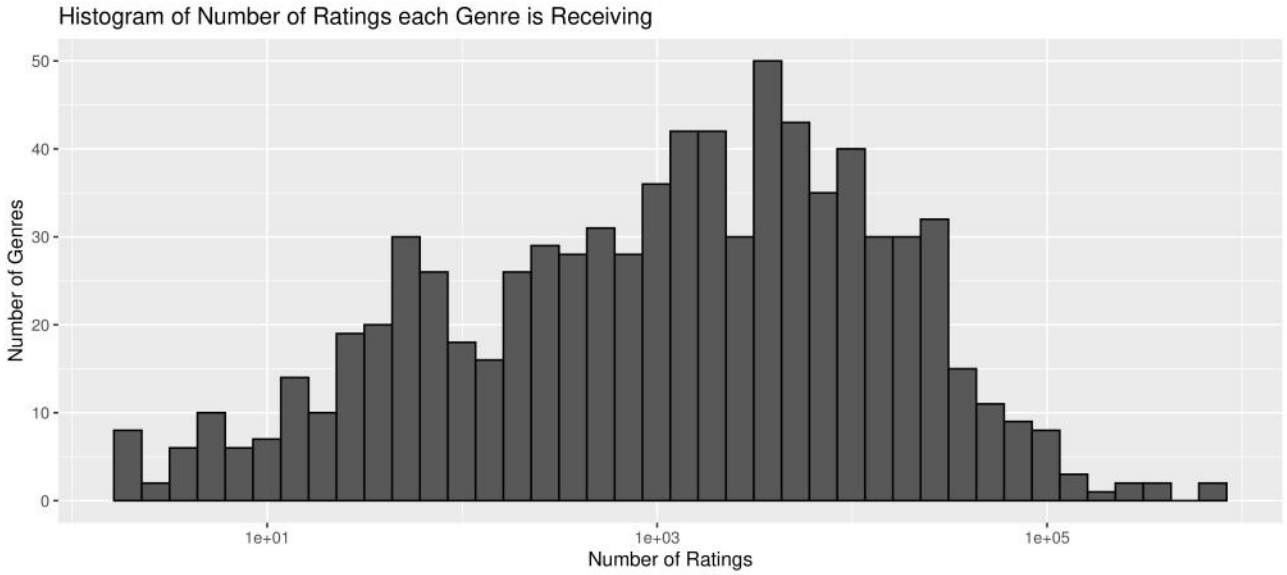
## 2.4 Movie Reviews Based on Genres

| genres | avg_rating | num_reviews |
|---|---|---|
| Animation\|IMAX\|Sci-Fi | 4.714286 | 7 |
| Drama\|Film-Noir\|Romance | 4.304115 | 2989 |
| Action\|Crime\|Drama\|IMAX | 4.297068 | 2353 |
| Animation\|Children\|Comedy\|Crime | 4.275429 | 7167 |
| Film-Noir\|Mystery | 4.239479 | 5988 |
| Crime\|Film-Noir\|Mystery | 4.216803 | 4029 |
| Film-Noir\|Romance\|Thriller | 4.216470 | 2453 |
| Crime\|Film-Noir\|Thriller | 4.210157 | 4844 |
| Crime\|Mystery\|Thriller | 4.198981 | 26892 |
| Action\|Adventure\|Comedy\|Fantasy\|Romance | 4.195557 | 14809 |

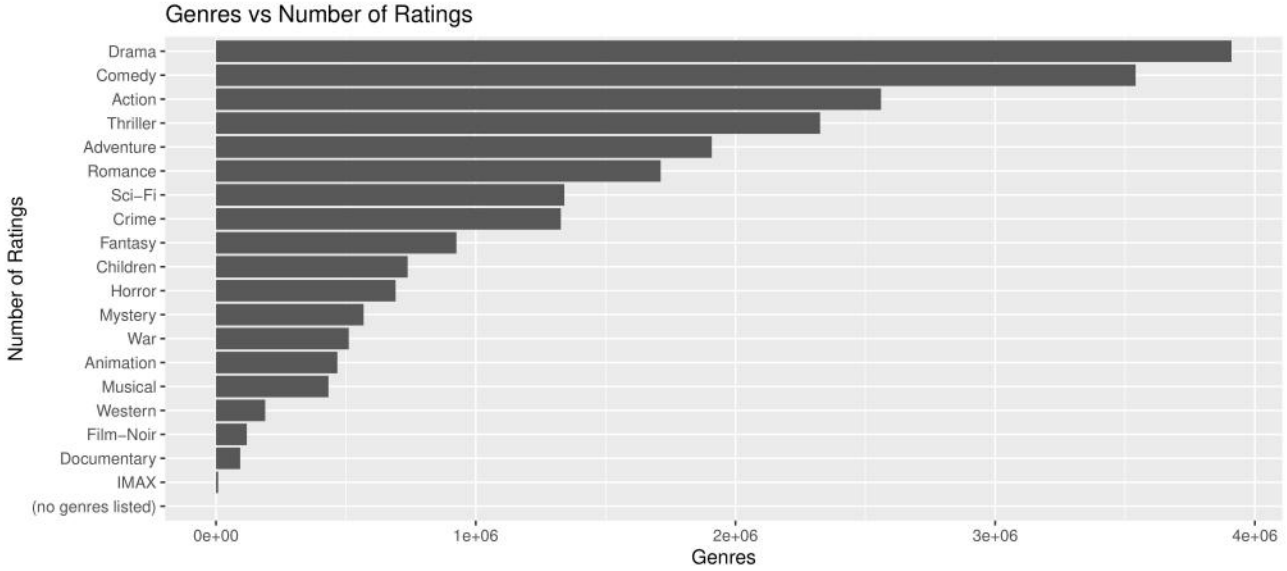
In the current edx format, there are 797 genres, the most popular genres being listed above.

| genres | avg_rating | num_reviews |
|---|---|---|
| Documentary\|Horror | 1.449112 | 619 |
| Action\|Animation\|Comedy\|Horror | 1.500000 | 2 |
| Action\|Horror\|Mystery\|Thriller | 1.607034 | 327 |
| Comedy\|Film-Noir\|Thriller | 1.642857 | 21 |
| Action\|Drama\|Horror\|Sci-Fi | 1.750000 | 4 |
| Adventure\|Drama\|Horror\|Sci-Fi\|Thriller | 1.751152 | 217 |
| Action\|Adventure\|Drama\|Fantasy\|Sci-Fi | 1.903509 | 57 |
| Action\|Children\|Comedy | 1.910232 | 518 |
| Action\|Adventure\|Children | 1.915048 | 824 |
| Adventure\|Animation\|Children\|Fantasy\|Sci-Fi | 1.924747 | 691 |

The lowest rated genres are indicated above.




It is clear that some genres are getting a lot more ratings than others. The genre variable in its current format may not suitable to incorporate as a feature variable as it may lead to overfitting. A possible solution would be to seperate the genres.



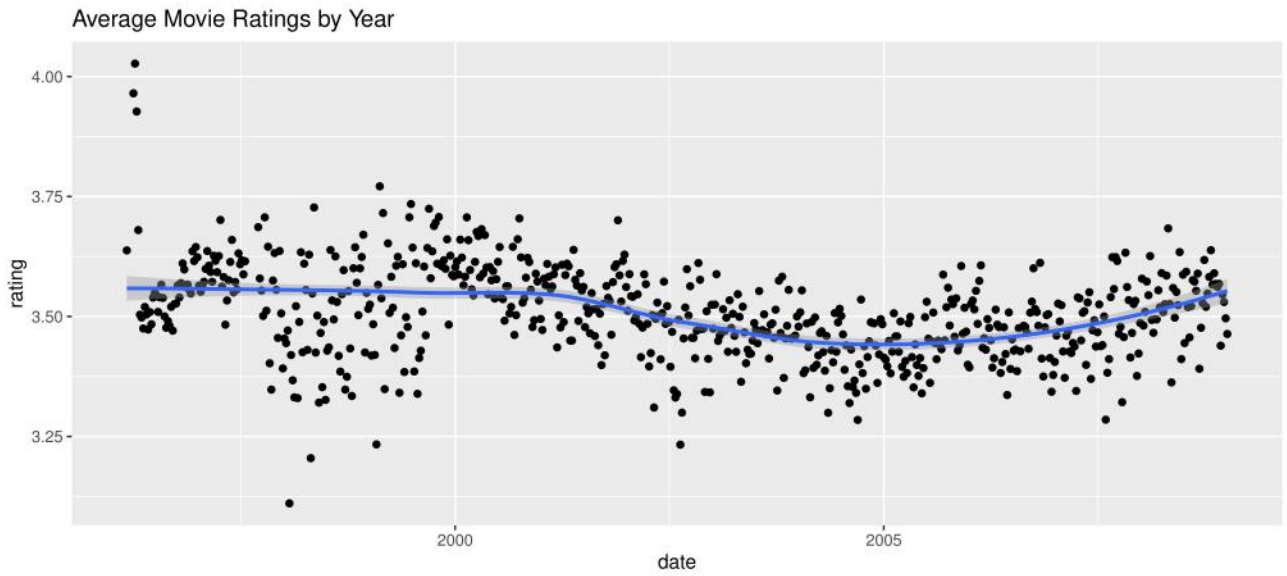

Much better! Now we can better visualize the number of ratings by genre and, in this format, it will likely be more appropriate for our final model.

Table 6: Highest Rated Genres

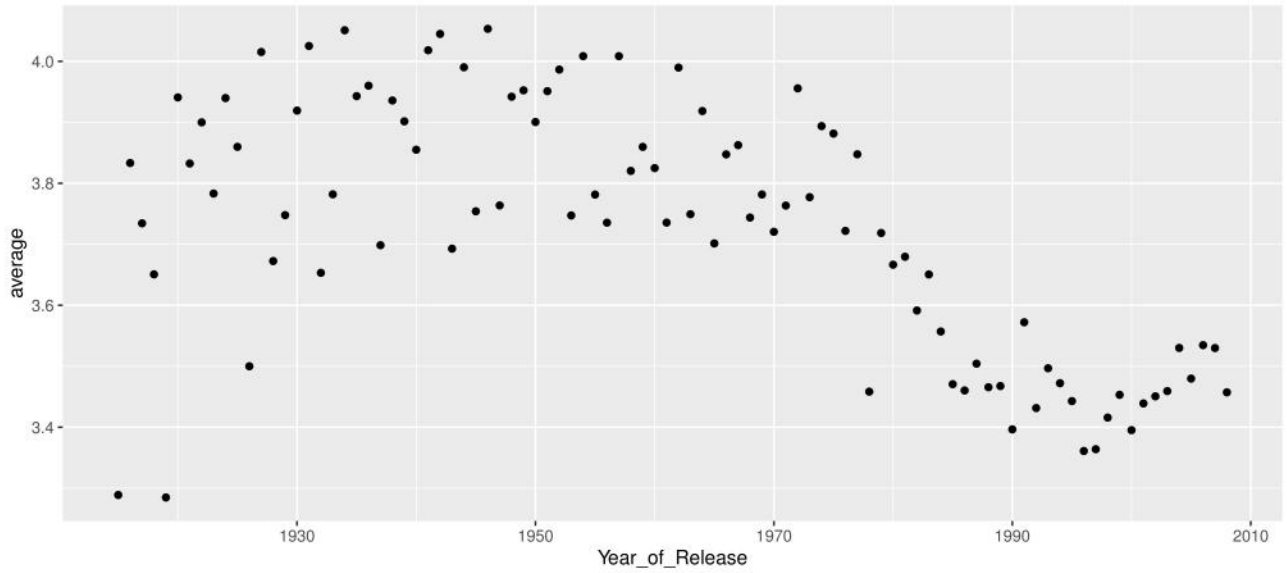
| genres | count | avg_rating |
|---|---|---|
| Film-Noir | 118541 | 4.011625 |
| Documentary | 93066 | 3.783487 |
| War | 511147 | 3.780813 |
| IMAX | 8181 | 3.767693 |
| Mystery | 568332 | 3.677001 |
| Drama | 3910127 | 3.673131 |
| Crime | 1327715 | 3.665925 |
| (no genres listed) | 7 | 3.642857 |
| Animation | 467168 | 3.600644 |
| Musical | 433080 | 3.563305 |

Above are the genres with the highest average rating in descending order of average rating.

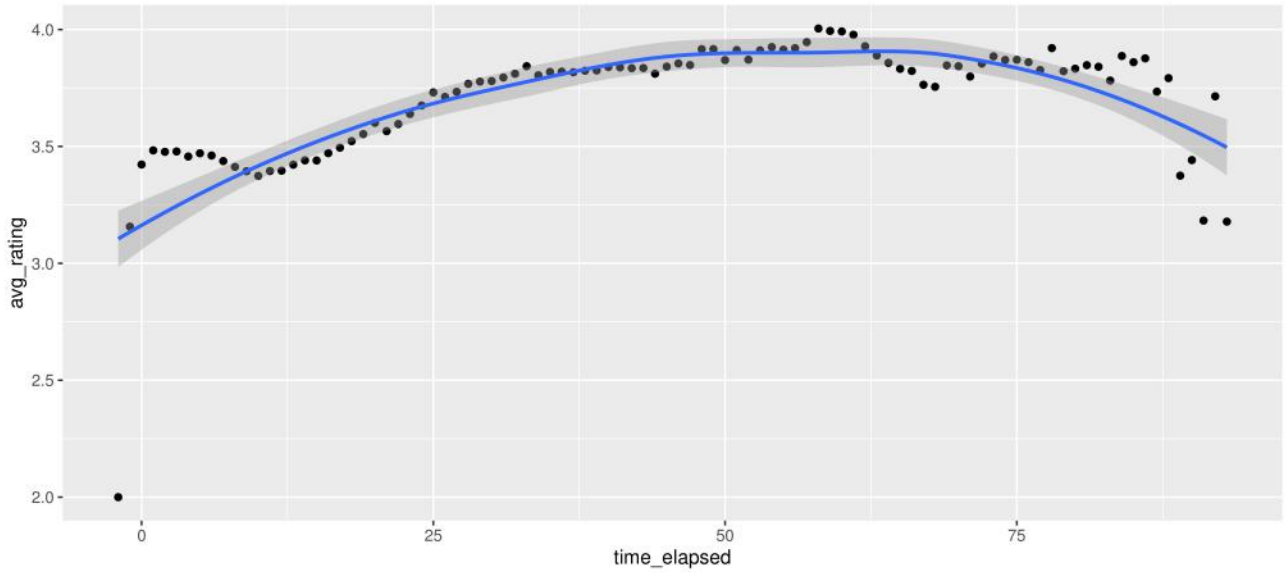
## 2.5 Movie Ratings by Date




As one can see, the average ratings for movies that were rated before 2000 is slightly higher than the average, with ratings of movies in 2005 being the lowest. Perhaps it is the movies that were being reviewed at the time which were not of the highest quality, or perhaps reviewers were simply raising their standards.




We can see above that movies that are slightly older were rater better. The average rating of a movie between the 1930s and 1970s seems to be approximately 3.9 compared to movies from approximately 2000 which is approximately 3.5 (a substantial difference). This can be for many reasons. Perhaps the reviewers are only watching only those older movies that are critically acclaimed and are considered classics. It may not necessarily mean that movies of an older generation were of higher quality than they are now. Another plasuible reason could be that older movies are more appreciated with time, and as such, are reviewed more favorably as time passes. This is what we will investigate next.




There is a slight link between the time elapsed (number of years between the year of release and year of review) and the rating - in line with previous thinking.‘

# 3 Methods and Analysis

We will now test a series of models designed to predict the movie ratings of our validation set. As the validation set is to be used only with the final model, we will create our own training and testing sets with the edx data. To improve accuracy, the following models will be run on ten different randomly chosen training and testing sets and the average RMSE from the results will be taken. As the primary purpose of this is to simply ascertain which models perform better than others, cross validation does not prove necessary.

## 3.1 Running the Models

### 3.1.1 Model 1 - Naive Model

Model 1 - The first model we will try, the Naive-Model, will simply take the average of the training set and predict that average for all ratings in the test set. While this may seem absurdly simple, it establishes a basline model with which we can compare subsuequent models.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.060632 |

This is not a great RMSE, but is a starting point for further improvement.

### 3.1.2 Model 2 - Movie Effect Model

This model builds on the previous one by also taking into account the average ratings of each individual movie. Shawshank Redemption for example (personally one of my favourite movies), which achieved an average rating of 4.455 would be the predicted value if it was the movieId in the the test_set.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |

This model definitely improves upon the naive model, although there is still room for further improvement.

#### 3.1.3 Model 3 - Movie and User Effect

Building upon the last model, this model factors in the average user rating. Given that some critics are more stringent than others, it only makes sense to take into account the user in this context to aid our prediction. For example, if the movie in question is again Shawshank Redemption (4.46 average rating) and the user in question is 67385 (user bias - -0.109), then the prediction model will predict (mean of training set + movie bias + user bias) equalling approximately 4.35.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |

This greatly improves upon Model 2, decreasing our RMSE by 0.08.

#### 3.1.4 Model 4a - Movie and User Model with Regularization

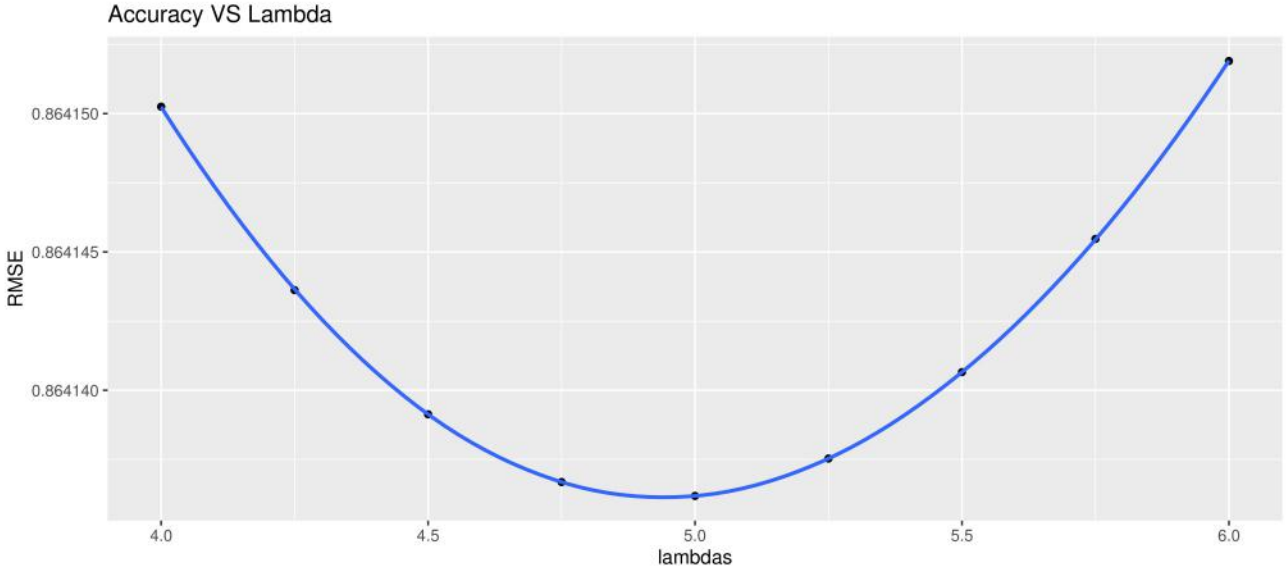
To further improve results, this model will implement the concept of regularization onto the previous model. The visualization section demonstrated that some movies were rated only once and that some users only rated few movies. Hence this can strongly influence the prediction. Regularization will be used to reduce the effect of overfitting.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |

As we can see, while this did not greatly reduce the RMSE, it certainly made a difference.

#### 3.1.5 Model 4b - Movie and User Model with Regularization (Optimized Lambda)

This model builds upon 4a in that the lambda value is optimized. Rather than choosing the default lambda value (3), the lambda that most reduces the RMSE is chosen.




| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |

The above plot clearly shows that the ideal lambda is around 5. While the optimized lambda does not make a huge difference, it does clearly reduce the RMSE.

#### 3.1.6 Model 5a - Movie, User and Genre Effects (Current Format)

As noted earlier during the visualization section, the genre variable may help us predict our rating as some genres have higher average ratings than others. In its current format, there are approximately 800 genres. As mentioned, this may lead to overfitting. Nevertheless we will try it.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |
| Model 5a - Movie, User and Genre Effect (Current Format) | 0.8646583 |

Incorporating genres in our model did certainly make a difference, albeit a relatively small one. Let us try a model now where we separate the genres.

#### 3.1.7 Model 5b - Movie, User and Genre Effects (Separated Format)

Using the separated format, let us see if that makes a bigger difference to the RMSE.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |
| Model 5a - Movie, User and Genre Effect (Current Format) | 0.8646583 |
| Model 5b - Movie, User and Genre Effect (Separated) | 0.8569999 |

Wow! This made much more of a difference. The RMSE is much less than if we were to incorporate genres in the current format.

#### 3.1.8 Model 5c - Movie, User and Genre Effects (Separated and averaged)

Given that the format of the validation set does not have the genres separated and it may go against the requirements of this project to test the model on an altered validation set, the average rating of the separated rows by genre is taken.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |
| Model 5a - Movie, User and Genre Effect (Current Format) | 0.8646583 |
| Model 5b - Movie, User and Genre Effect (Separated) | 0.8569999 |
| Model 5c - Movie, User and Genre Effect (Separated and Averaged) | 0.8646583 |

It seems that this idea not work in practice given that we ended up achieving the same result as 5a.

#### 3.1.9 Model 6a - Movie, User and Date Effects

The date of review is another variable that we can further explore. Perhaps world events (such as 9/11) affected the overall mood of the world and movie reviewers were not rating movies as highly.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |
| Model 5a - Movie, User and Genre Effect (Current Format) | 0.8646583 |
| Model 5b - Movie, User and Genre Effect (Separated) | 0.8569999 |
| Model 5c - Movie, User and Genre Effect (Separated and Averaged) | 0.8646583 |
| Model 6a - Movie, User and Date Effect | 0.8648623 |

While it did make a difference, it is almost negligible and may not be worth incorporating into the final model.

#### 3.1.10 Model 6b - Movie, User and Time In Between Effects

Explored as well is the time in between the release of the movie and the review. As explored earlier in the visualization section the longer the interval was, the higher the average movie rating.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |
| Model 5a - Movie, User and Genre Effect (Current Format) | 0.8646583 |
| Model 5b - Movie, User and Genre Effect (Separated) | 0.8569999 |
| Model 5c - Movie, User and Genre Effect (Separated and Averaged) | 0.8646583 |
| Model 6a - Movie, User and Date Effect | 0.8648623 |
| Model 6b - Movie, User and Time (in Between Release and Rating) | 0.8645106 |

While this does not reduce RMSE drastically, it does seem to make a difference, certainly more than the date of review variable. As such, it will be incorporated into our final model.

#### 3.1.11 Model 7a - Movie, User, Genre (separated) and Time Elapsed Effects

The final model is tested with the variables believed to be most predictive of the rating. This includes the average movie rating, the average user rating, the average genre rating (separated) and the time elapsed variable. As such it is not clear at this time whether the genre variable can be separated, this model is based on the assumption that it can and the validation set that is used is separated by genre.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |
| Model 5a - Movie, User and Genre Effect (Current Format) | 0.8646583 |
| Model 5b - Movie, User and Genre Effect (Separated) | 0.8569999 |
| Model 5c - Movie, User and Genre Effect (Separated and Averaged) | 0.8646583 |
| Model 6a - Movie, User and Date Effect | 0.8648623 |
| Model 6b - Movie, User and Time (in Between Release and Rating) | 0.8645106 |
| Model 7a - Movie, User, Genre (Separated) and Time Elapsed Effects | 0.8564530 |

This model thus far has produced the best results and will therefore be tested with the validation set.

#### 3.1.12 Model 7b - Movie, User, Genre (Unseparated) and Time Elapsed Effects

This model is with the genre unseparated. As we cannot assume that the validation set can be altered, this model is designed to meet that requirement. This is tested with the movie average, user average, time_elapsed average and genre average.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |
| Model 5a - Movie, User and Genre Effect (Current Format) | 0.8646583 |
| Model 5b - Movie, User and Genre Effect (Separated) | 0.8569999 |
| Model 5c - Movie, User and Genre Effect (Separated and Averaged) | 0.8646583 |
| Model 6a - Movie, User and Date Effect | 0.8648623 |
| Model 6b - Movie, User and Time (in Between Release and Rating) | 0.8645106 |
| Model 7a - Movie, User, Genre (Separated) and Time Elapsed Effects | 0.8564530 |
| Model 7b - Movie, User, Genre (Unseparated), and Time Elapsed effects | 0.8641935 |

As this is the best model with the genre variable unseparated, this model will also be used on the final validation set as well.

# 4 Results (Testing with Validation Set)

## 4.1 Seperated Genre Validation Set

If we are allowed to alter the validation set in a way that allows us to separate the rows by genre, then mode 7a should perform the best. Let us test it.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |
| Model 5a - Movie, User and Genre Effect (Current Format) | 0.8646583 |
| Model 5b - Movie, User and Genre Effect (Separated) | 0.8569999 |
| Model 5c - Movie, User and Genre Effect (Separated and Averaged) | 0.8646583 |
| Model 6a - Movie, User and Date Effect | 0.8648623 |
| Model 6b - Movie, User and Time (in Between Release and Rating) | 0.8645106 |
| Model 7a - Movie, User, Genre (Separated) and Time Elapsed Effects | 0.8564530 |
| Model 7b - Movie, User, Genre (Unseparated), and Time Elapsed effects | 0.8641935 |
| Final Test with Validation Set (Separated) | 0.8623618 |

As we can see, the results are below the threhsold of 0.86490 which meets the requirements of this capstone project. Let us test it now without the genre row split up.

## 4.2 Unseparated Genres Validation Set

The model that we are using now is the 7b model, which is based on movie average, user average, time elapsed average and genre average. As we cannot assume that validation set can be altered, this is the best performing model that meets that requirement.

| Method | RMSE |
|---|---|
| Model 1 - Just the Average | 1.0606318 |
| Model 2 - Movie Effect Model | 0.9437963 |
| Model 3 - Movie and User Model | 0.8655154 |
| Model 4a - Movie and User Effect (Regularization) | 0.8650320 |
| Model 4b - Movie and User Effect (Regularization with Optimized Lambda) | 0.8649748 |
| Model 5a - Movie, User and Genre Effect (Current Format) | 0.8646583 |
| Model 5b - Movie, User and Genre Effect (Separated) | 0.8569999 |
| Model 5c - Movie, User and Genre Effect (Separated and Averaged) | 0.8646583 |
| Model 6a - Movie, User and Date Effect | 0.8648623 |
| Model 6b - Movie, User and Time (in Between Release and Rating) | 0.8645106 |
| Model 7a - Movie, User, Genre (Separated) and Time Elapsed Effects | 0.8564530 |
| Model 7b - Movie, User, Genre (Unseparated), and Time Elapsed effects | 0.8641935 |
| Final Test with Validation Set (Separated) | 0.8623618 |
| Final Test with Validation Set (Unseparated) | 0.8639807 |

While this does not perform as well as the previous model, it is nevertheless lower than 0.8649 and hence, fulfills the RMSE requirement for this project.

# 5 Conclusion

The techniques utilized in this project are based upon the coursework of the Harvard Data Science Certificate Program. I particularly enjoyed the visualization section personally as it made me think of creative ways the data can be showcased and also required me to examine unexplored relationships between different variables which could potentially be utilized to create better models. The variables "genre" and "time elapsed" are examples of the results of such exploration and ultimately both variables helped to create better models. While both final models tested on the validation set in this project achieve the required RMSE, there is no doubt that there is room to substantially improve the models created here. Some projects that have utilized other machine learning algorithms (primarily matrix factorization algorithms) achieved an RMSE of approximately 0.70, which are considerably less than the lowest RMSE achieved here. Utilization of algorithms such as Random Forest, SVD and other such algorithms may also yield better results. Using other programming languages that are more efficient for this type of project would also be an improvement. If one is attempting to create a model using such algorithms, one should keep in mind RAM and processing constraints as the movielens dataset is quite large. Using other programming languages that are more efficient for this type of project would help alleviate such concerns.

## 5.1 Resources:


- Irizarry, R. A. (2020, March 2). Introduction to Data Science. Retrieved from https://rafalab.github.io/dsbook/introduction-to-machine-learning.html#notation-1
- Ricci, F., Rokach, L., & Shapira, B. (2015). Recommender systems: introduction and challenges. In Recommender systems handbook. Springer, Boston, MA.
- Koren, Y. (2009). The bellkor solution to the netflix grand prize. Netflix prize documentation, 81, 1-10.
- Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, No. 10). New York: Springer series in statistics.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017). An introduction to statistical learning: with applications in R. New York: Springer.