MovieLens Project

Harvard PH125.9x Data Science: Capstone

Frankie Inguanez

2022-09-26

Contents

List of Figures

1 Introduction

This report documents the research undertaken for the MovieLens project submission in part fulfillment of the Harvard PH125.9x Data Science: Capstone module by the author, Frankie Inguanez. The MovieLens dataset is prepared by GroupLens Research at the University of Minnesota which contains user reviews on movies. For this project a subset of the entire dataset has been utilized.

The aim of this research is to propose a movie recommender system. The fitness of the proposed model is evaluated using the Root Mean Square Error (RMSE) and the objective of this research is to have a RMSE lower than 0.86490.

This document proceeds with an overview of the data analysis undertaken, highlighting the data cleaning decisions and illustrating the data visualisations which justify the methodology undertaken in model creation. All findings are presented in the Results section, with final remarks found in conclusion.

2 Analysis

The analysis process is made up of the following stages: Exploratory Analysis; Data cleaning; Data Visualisation; Data Modelling.

The initial setup code provided downloaded the 10 million dataset and split it in a 90:10 ratio for training and testing respectively, resulting in 9,000,055 observations (rows) for training, with 999,999 observations (rows). The dataset has 9 variables(columns).

2.1 Exploratory Analysis

The next step was to explore the data by checking the datatypes, counts and presence of null (NA) values. No missing values were found, the identification variables for users and movies are integer and numeric respectively. Rating is numeric whilst title and genres are character variables. It was noted that genres is a multi-valued using the pipe (|) as a separator. The timestamp variable represents the review date in milliseconds since the 1st January 1970 GMT (epoch time).

Checking the count of each variable resulted in a discrepency with 10,677 unique movie ID values but 10,676 unique movie title values. The movie in question is War of the Worlds (2005). Upon further investigation this was found to be a genuine case where in the year 2005 two movies were released with the same name, one directed by Steven Spielberg, whilst the other directed by David Michael Latt. So it was determined that no data cleaning is needed, yet it is important to use the movieID variable rather than title variable when aggregating by movie.

2.2 Data Cleaning

The data cleaning consisted in three main tasks. First the timestamp variable was converted to a human readable date without the time information. This variable was rounded to the nearest day and named reviewDate.

The next task was to separate the release year from the movie title. A regular expression pattern was created to search at the end of the title variable for a space followed by digits in parenthesis. The year variable was also converted to integer.

The last task involved the use of the prior two tasks. In this research we are assuming that there is no real intrinsic value in the review date but rather the difference in years from the movie release year to the actual review date. So a variable called reviewDelay was created to measure this difference.

2.3 Data Visualisation

2.3.1 Rating

The rating variable ranges from 0 till 5 with 0.5 increments. The whole numbers occur more frequently then the half point ratings.

2.3.2 Movies

Looking at the mean rating by movie we see that it is at around 3.5.

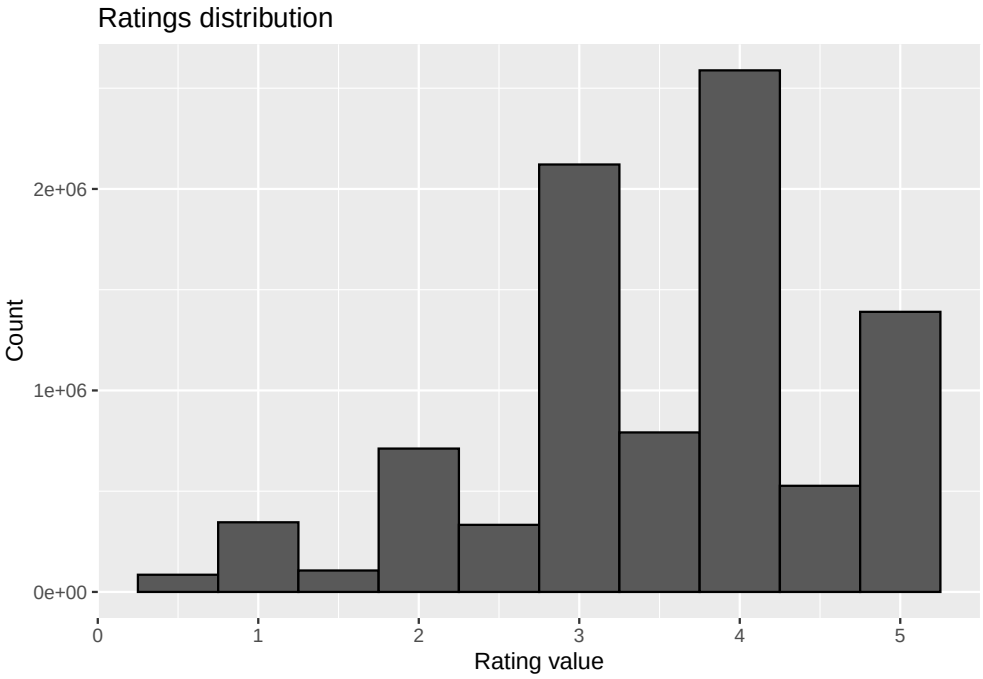

Figure 1: Ratings distribution

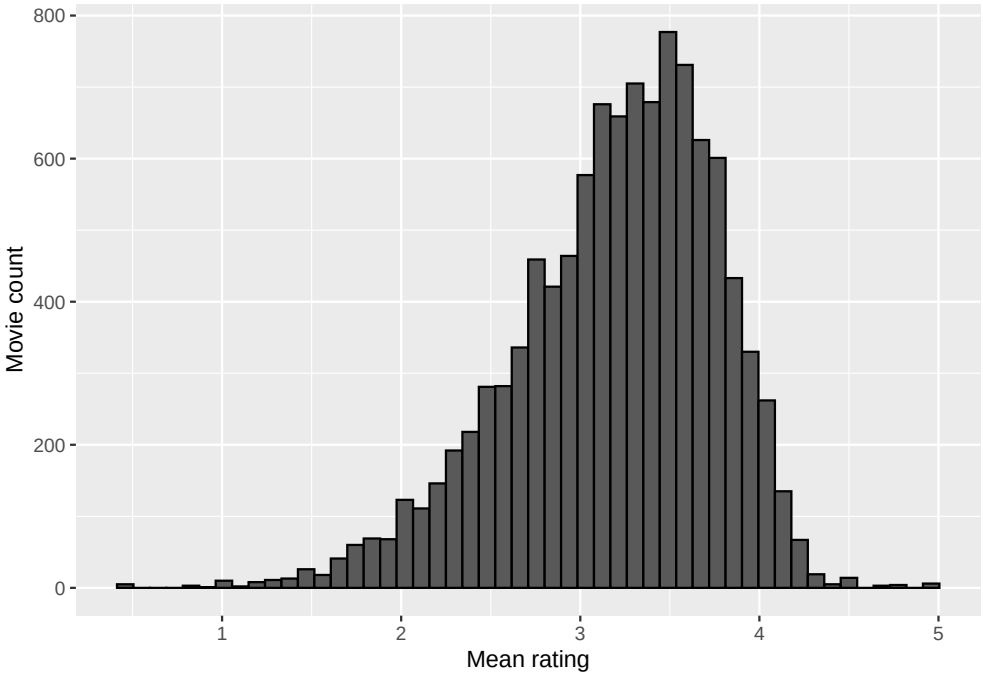

Figure 2: Movie count by mean rating

2.3.3 Users

A similar yet more condensed observation is made when looking at the mean rating by user in Figure 3. It is also noted from Figure 4 that certain users rate much more than other users.

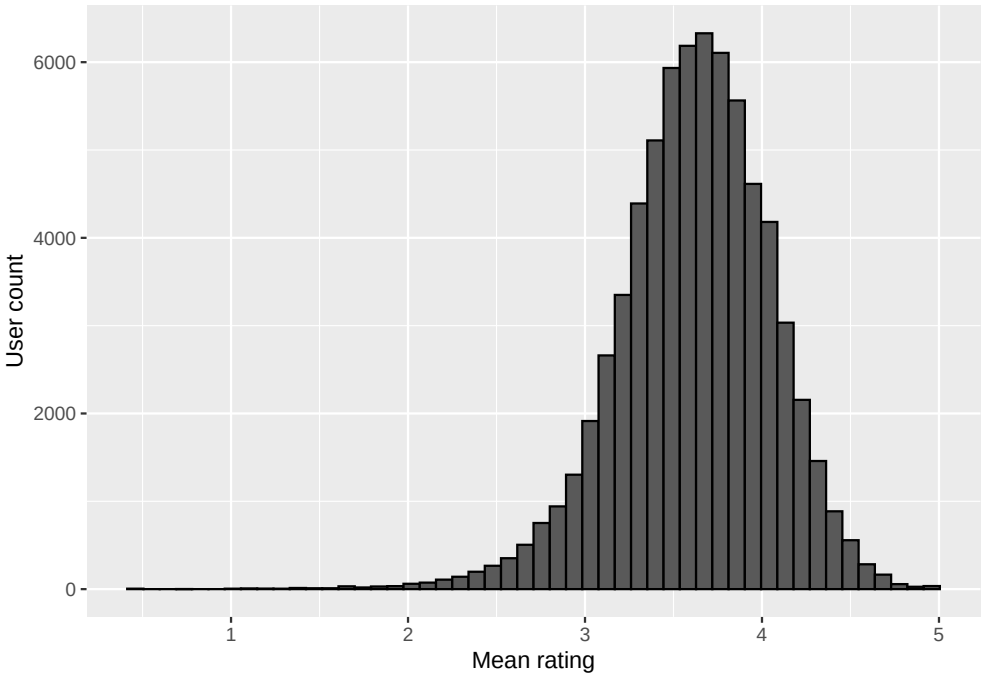

Figure 3: User count by mean rating

2.3.4 Genre

In observing the mean rating per genre some data manipulation was needed to extract the average rating per genre rather than per genre combination. The generation of such figure was very computationally intensive and the benefit of averaging by genre rather than genre combination is to be determined. Depending on whether the project objective is achieved a decision shall be made on which variable to use. From Figure 5 we can observe that the average rating does fluctuate a lot per genre, namely for Horror and Film-Noir. The list of genres with ratings count and mean rating is also provided in Table 1.

2.3.5 Release Year

From Figure 6 we can observe a fluctuation in the mean average rating for the release year of the movie. This could be due to the disproportionate number of movies available per release year as shown in Figure 7.

2.3.6 Review Date & Delay

When observing the mean rating per review date as shown in Figure 8 no real fluctuation could be observed. It was noted that there was a strange range of values in the review delay, which is the number of years of the review date from the movie release year as shown in Figure 9. Some reviews had a negative value, which follow some research was established to be a pre-release screening of a

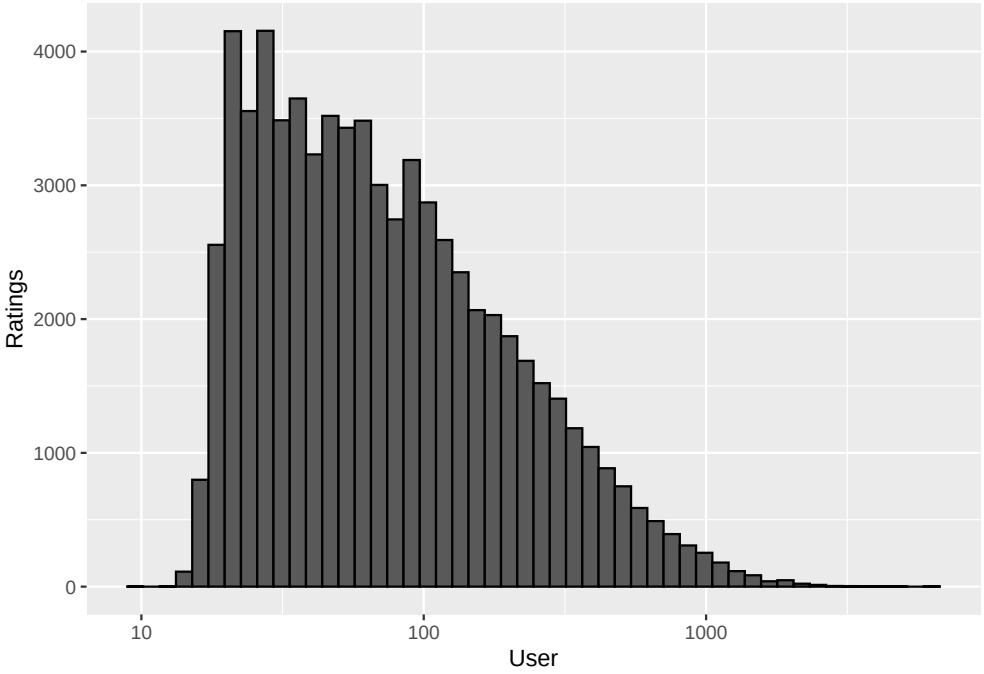

Figure 4: User rating count

Table 1: Ranked genres by ratings count

Genre	Ratings Count	Mean Rating
Drama	3910127	3.6731
Comedy	3540930	3.4369
Action	2560545	3.4214
Thriller	2325899	3.5077
Adventure	1908892	3.4935
Romance	1712100	3.5538
Sci-Fi	1341183	3.3957
Crime	1327715	3.6659
Fantasy	925637	3.5019
Children	737994	3.4187
Horror	691485	3.2698
Mystery	568332	3.6770
War	511147	3.7808
Animation	467168	3.6006
Musical	433080	3.5633
Western	189394	3.5559
Film-Noir	118541	4.0116
Documentary	93066	3.7835
IMAX	8181	3.7677
(no genres listed)	7	3.6429

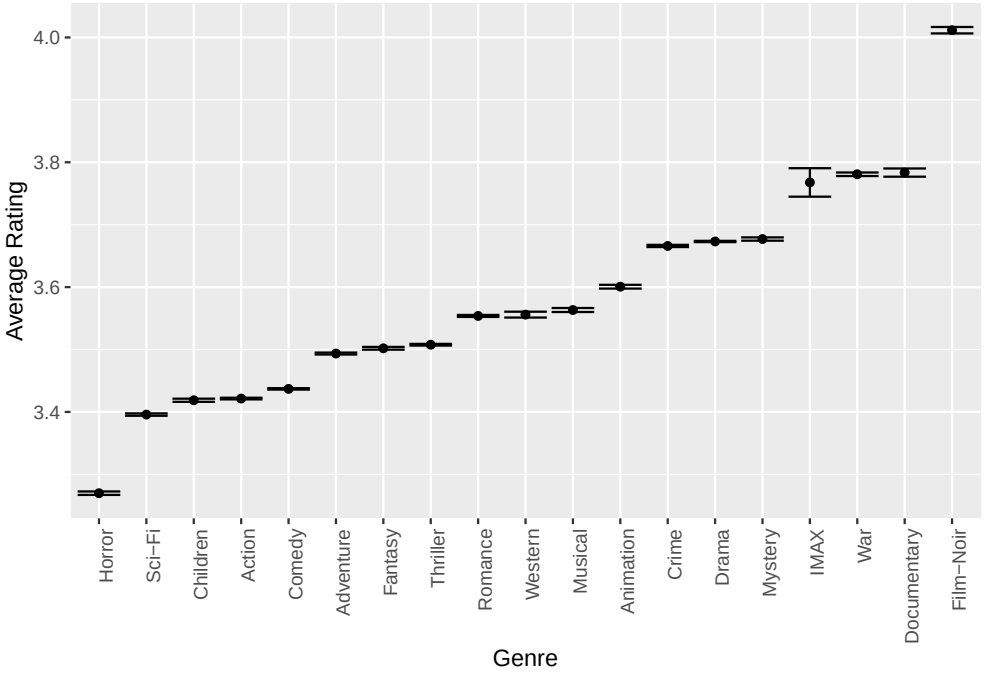

Figure 5: Mean rating by genre

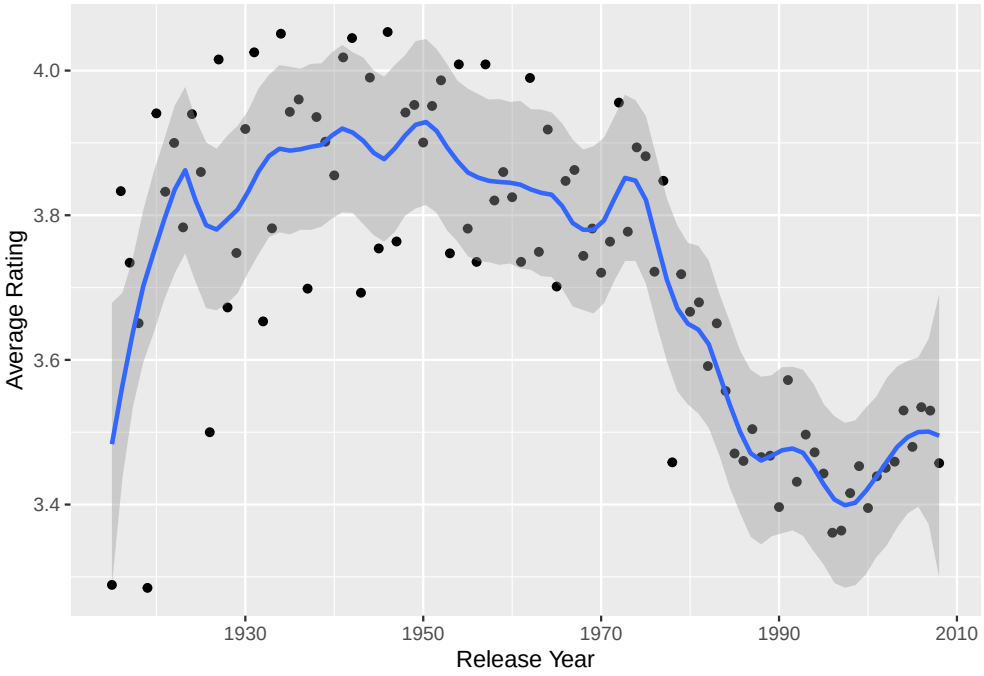

Figure 6: Year mean rating

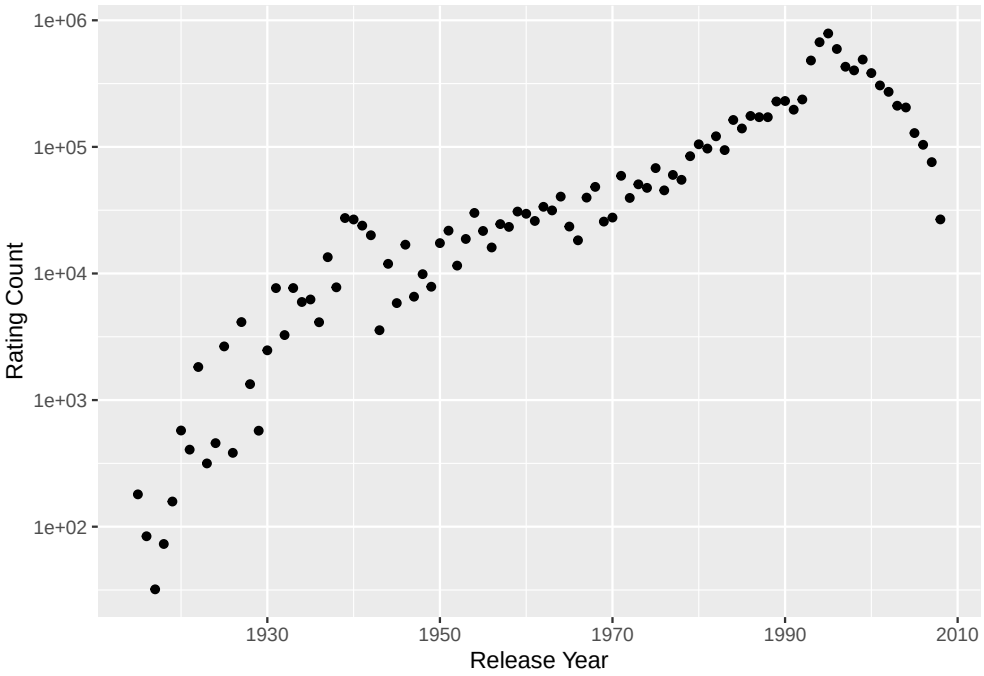

Figure 7: Year rating count

movie for critics which is common practice. It was noted that the number of reviews on a movie is inversely proportional to the review delay. From Figure 10 it can be observed that the mean rating does fluctuate by review delay, more specifically the mean rating in the first few years is slightly higher than the subsequent 5-10 years. The mean rating then rises drastically following 10 years of release. Also the mean rating prior to release thens to be considerably lower. With these observations a decision has been made to use review delay instead of review date in the data modelling phase.

2.3.7 Data Modelling

The evaluation metric utilized in the data modelling of this research is the Root Mean Square Error. The objective of this project was to achieve a RMSE lower than 0.86490.

$$RMSE = \sqrt{\frac{1}{N}\sum_{u,i}(\hat{y}_{u,i} - y_{u,i})^2}$$

The modelling was made on the edx dataset and not the validation (hold-out) dataset. This was split into training and testing datasets at 80%:20% ratios respectively. A total of seven models were created in an incremental fashion as shown in Table 2. The second model (Movie) uses the Naive Prediction based on mean rating average and adds compensates for the movie effect. The subsequent, builds on model 2 and compensates for the user effect, until we have model 7 were regularization is used to compensate for the disparity in movie ratings, considering the mean rating and compensating for movie, user, genre combination, movie release year and review delay effect.

When regularlization was applied to the algorithm a number of lambdas were considered and the

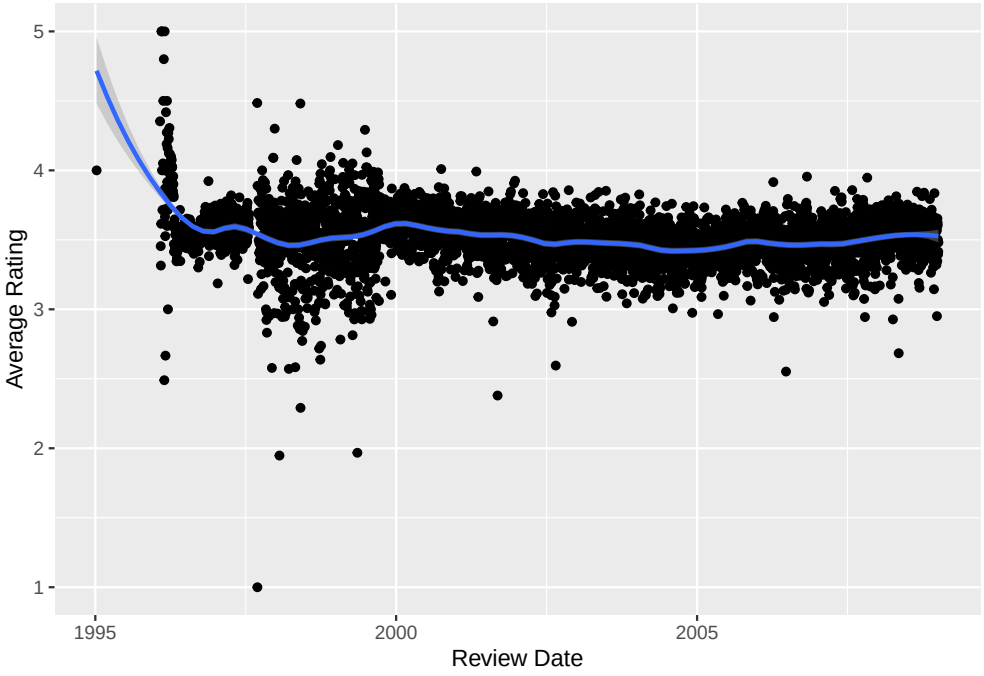

Figure 8: Review date mean rating

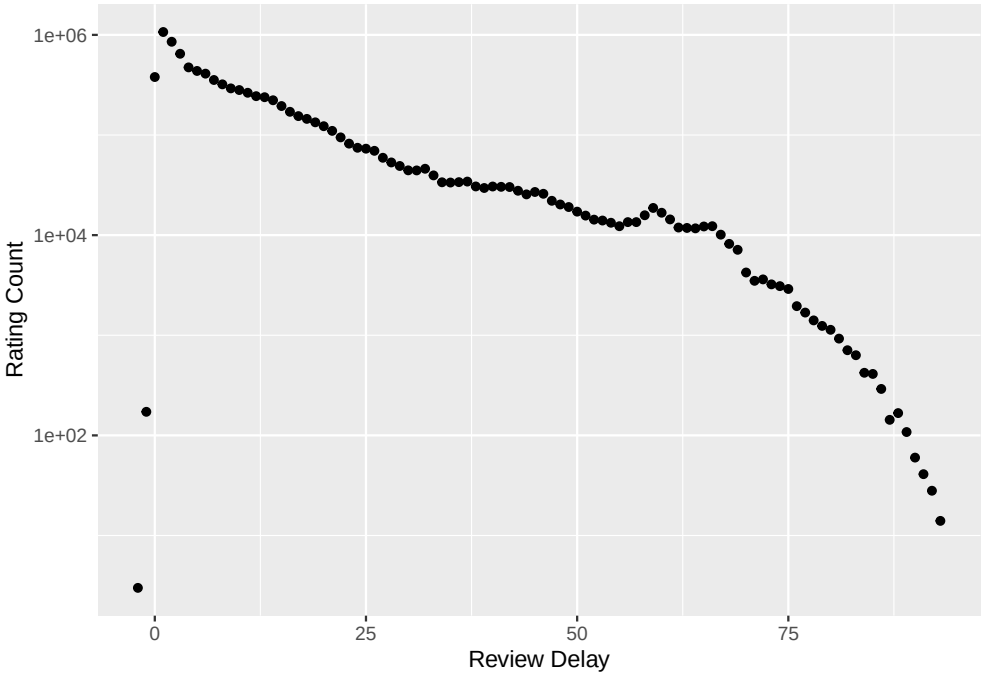

Figure 9: Review delay rating count

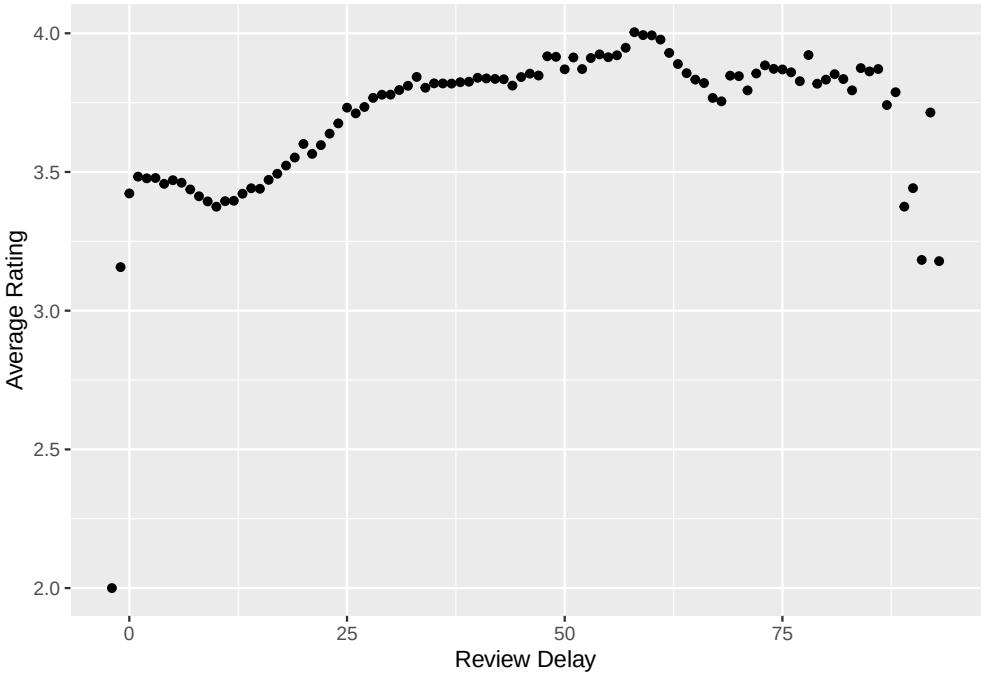

Figure 10: Review delay mean rating

Table 2: Models

Model
Naive Prediction (Average)
+ Movie Effect
+ User Effect
+ Genre Combination Effect
+ Release Year Effect
+ Review Delay Effect
+ Regularized

Table 3: Model evaluations

Model	RMSE
Naive Prediction (Average)	1.05991
+ Movie Effect	0.94374
+ User Effect	0.86593
+ Genre Combination Effect	0.86559
+ Release Year Effect	0.86542
+ Review Delay Effect	0.86512
+ Regularized	0.86446

one yielding the lowest RMSE was chosen. With the final formula identified the model was applied to the hold-out dataset.

3 Results

For regularization a number of lambdas were considered as shown in Figure 11. The best lambda was 4.85 and a minimum RMSE of 0.86446 was obtained on the edx dataset. The RMSE values of each model are provided in Table 3.

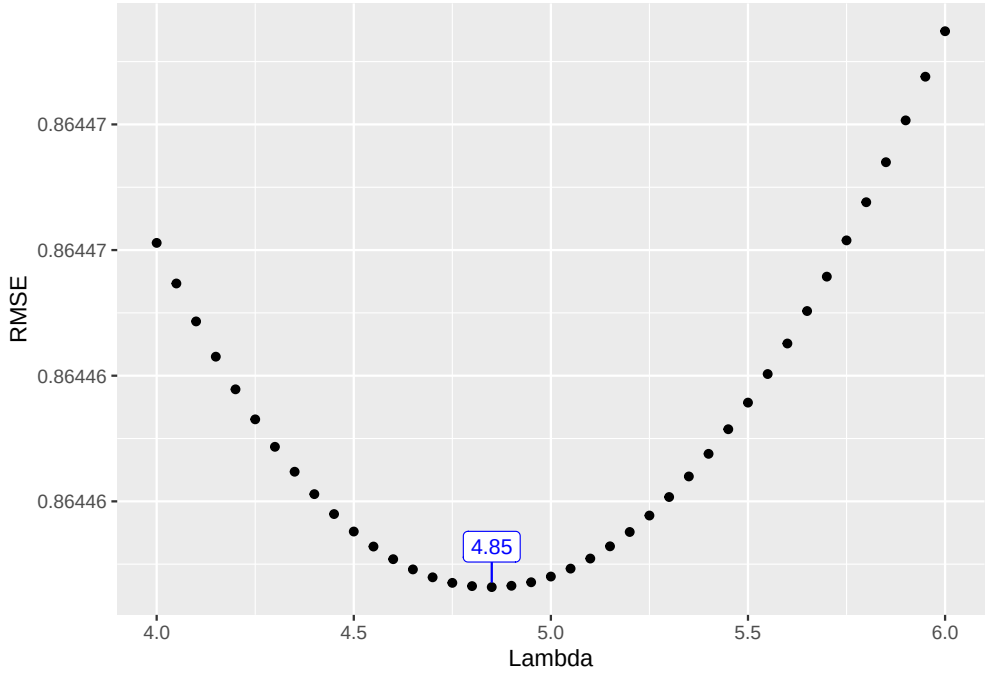

Figure 11: Lambdas vs RMSE evaluation

Prior to running the model on the hold-out dataset, the validation data needed to be transformed to match the changes done to the edx dataset. Namely:

1. Converting the review date
2. Extracting the movie release year

3. Calculating the review delay.

After applying the regularized model with all predictors a RMSE of 0.86396 was achieved, meeting the goal of this project.

4 Conclusion

In this research a recommender system was created for the 10 million movielens dataset with a RMSE of 0.86396 when tested on a hold-out dataset, also known as the validation dataset which is 10% of the original dataset. A total of seven models were considered on the main dataset with the final and best model using regularization of the mean rating whilst compensating for the movie, user, genre combination, movie release year and review delay effects. The best RMSE obtained during training was of 0.86446.

It is possible to further improve on this model by considering the individual genre bias rather than the combination and other recommender models such as collaborative filtering. It is also recommended to explore different training approaches rather than just a train:test data split as applied here, such as cross validation.