edX Data Science: Capstone Course - MovieLens Project

Elizabeth Plasse Dominguez

June 29, 2022

Overview

The goal of this Capstone Course MovieLens Project was to create a movie recommendation model able to predict movie ratings with a root mean squared error (RMSE) of less than 0.86490 when compared to actual movie ratings. The data used comes from the MovieLens 10M Dataset[1] and contains the following six variables:

- userId - unique number for each user,
- movieId - unique number for each movie,
- rating - numbers ranging from 0.5 to 5 in increments of 0.5,
- timestamp - number of seconds since midnight (UTC) of 1-1-1970,
- title - character string of movie title and year of movie release, and
- genres - character string of movie genre or multiple genres - when an observation has multiple genres each genre is separated by a "|" (pipe-separated) - there are 18 unique genres (Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, Western).

Data was downloaded from the MovieLens dataset and divided into the two following datasets, each containing these six variables - *userId, movieId, rating, timestamp, title, genres*:

- *edx*, with about 9 million observations, and
- *validation*, with about 1 million observations.

Data investigation, analysis and visualization of the *edx* data set, detailed in the Methods section, led to the creation of the following tidy datasets:

- *edx_tidy*, with about 23,370,000 observations, and
- *validation_tidy*, with about 2,597,000 observations.

edx_tidy and *validation_tidy* also each containing these six variables however note that two of the variables have changed - *userId, movieId, rating, date_rated, title, single_genres.*

edx_tidy was used to train a movie rating prediction model which incorporates the average rating of individual movies and an individual user bias calculated from the data. When this model was used to produce movie rating predictions from the data in *validation_tidy* and these predictions were compared to actual movie ratings, a RMSE of 0.85698 was obtained.

Methods

Accurate movie rating prediction was desired from the model so developing a sense for how rating varies in relation to the other variables in the data was important. Intuitively, rating information by movie and by user seems important. Popular movies may be rated more frequently and/or differently than small independent films and users probably differ in how critical or liberal they are with their ratings. Looking at the number of ratings movies have received and the number of ratings users have given was a way to get a sense for these differences.

Initial exploration of *edx* data revealed that many of the *genres* variable entries had more than one genre per observation and that the *timestamp* variable might be more useful in a year-month-day (y-m-d) format. Data wrangling to produce a tidy format with one genre per row and timestamp in y-m-d format resulted in the creation of the *edx_tidy* and *validation_tidy* datasets and the replacement of the *genres* and *timestamp* variables with the new variables *single_genres* and *date_rated, respectively.*

Working with *edx_tidy* data, plots of the number of ratings associated with the other five variables were produced, as were plots of the average ratings for movies, users and dates_rated. This information along with model building knowledge gained in the edX Data Science: Machine Learning course led to a rating prediction model using individual movie means and individual user effects.

Data Exploration/Cleaning/Analysis

Data from the Movielens dataset was downloaded and manipulated into two datasets, with object names *edx* and *validation*, which were to be used for model training and model prediction testing, respectively. The code provided in edX Data Science: Capstone course accomplished this.

The objects *edx* and *validation* are data tables with the same six variables (*userId, movieId, rating, timestamp, title, genres*) and one rating per observation. *edx* has 9,000,055 rows and *validation* has 999,999. Further analysis of the variables in *edx* showed the unique number of each variable which are presented in Table 1.

Table 1

Number of Unique Users	Number of Unique Movies	Number of Unique Ratings	Number of Unique Timestamps	Number of Unique Titles	Number of Unique Genres
69,878	10,677	10	6,519,590	10,676	797

797 was an unexpected number for the unique number of genres. The MovieLens variable description indicated 18 unique genres. This large number was the result of each single genre plus all of the unique multiple pipe_separated genres being recognized as unique.

Further investigation into the *genres* variable of *edx* showed that there were actually 20 unique genres (including the genre heading of (no genres listed)) and that 81% of the rows have multiple pipe-separated genres. A list of the genres in descending order of number of ratings is presented in Table 2.

Table 2

Rank by Number of Ratings	Genre Name	Number of Ratings	Rank by Number of Ratings	Genre Name	Number of Ratings
1	Drama	3,910,127	11	Horror	691,485
2	Comedy	3,540,930	12	Mystery	568,332
3	Action	2,560,545	13	War	511,147
4	Thriller	2,325,899	14	Animation	467,168
5	Adventure	1,908,892	15	Musical	433,080
6	Romance	1,712,100	16	Western	189,394
7	Sci-Fi	1,341,183	17	Film-Noir	118,541
8	Crime	1,327,715	18	Documentary	93,066
9	Fantasy	925,637	19	IMAX	8,181
10	Children	737,994	20	(no genres listed)	7

These finding in the *genres* variable of *edx* led to the conclusion that the data needed to be made tidy by creating a new data frame, *edx_tidy*, with separate observations for each genre in a new variable called *single_genres.* Here is an example: if a row of *edx* had three genres listed, say Children|Comedy|Fantasy, this row needed to be adjusted in *edx_tidy* to have just Children as the *single_genres* variable and two new additional rows needed to be added - one with Comedy and the other with Fantasy as the *single_genres* variable in *edx_tidy.* Both new rows have identical information in the other 5 variables (*userId, movieId, rating, date_rated and title*).

Preliminary plots of *edx* variables (not included here) also indicated that the *timestamp* variable might be more useful if converted to year-month-day (y-m-d) format. So the *timestamp* variable of *edx* was converted to the y-m-d format in a new variable called *date_rated* in *edx_tidy* This facilitated averaging ratings by *date_rated* and looking for possible seasonality or cyclicality in the data.

Thus the MovieLens data was manipulated into two new data frames, *edx_tidy* and *validation_tidy.* Both with the following six variables: *userId, movieId, rating, date_rated, title and single_genres.*

edx_tidy, and *validation_tidy* are tibble data frames with 23,370,479 and 2,596,715 rows, respectively. As *edx_tidy* data would be used to train the model in this project, further analysis of the variables in this tibble data frame was conducted to show the unique number of each variable and is presented in Table 3.

Table 3

Number of Unique Users	Number of Unique Movies	Number of Unique Ratings	Number of Unique Dates_Rated	Number of Unique Titles	Number of Unique Single_Genres
69,878	10,677	10	4,640	10,676	20

The numbers of unique users, movies, ratings, and titles were the same in *edx_tidy* as in *edx.* The number of unique dates_rated was 4,640 due to the y-m-d format change and the number of unique single_genres was 20 which agrees with the 18 listed in the MovieLens data description plus the addition of IMAX and (no genres listed).

Visualization

The following visualizations of *edx_tidy* data demonstrated the many aspects of variability in the data.

Figures 1 and 2 plot the number of ratings for each movie and each user, respectively. Both plots exhibit exponential decay.

Figure 1 shows that some movies are rated less often than others with many movies receiving a small number of ratings and a small number of movies receiving many ratings.

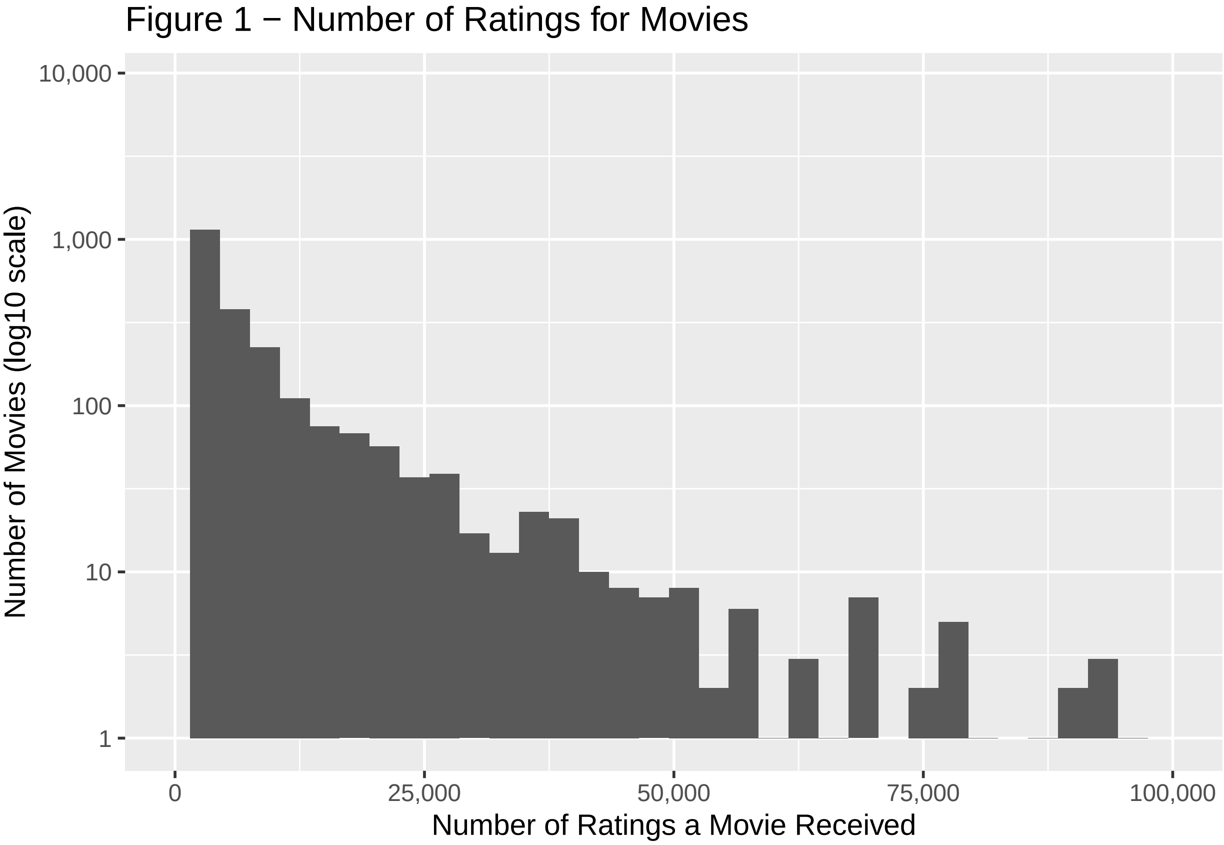


Figure 2 shows that some users rate more movies than other users do, with many users rating a small number of movies and a small number of users rating many movies.

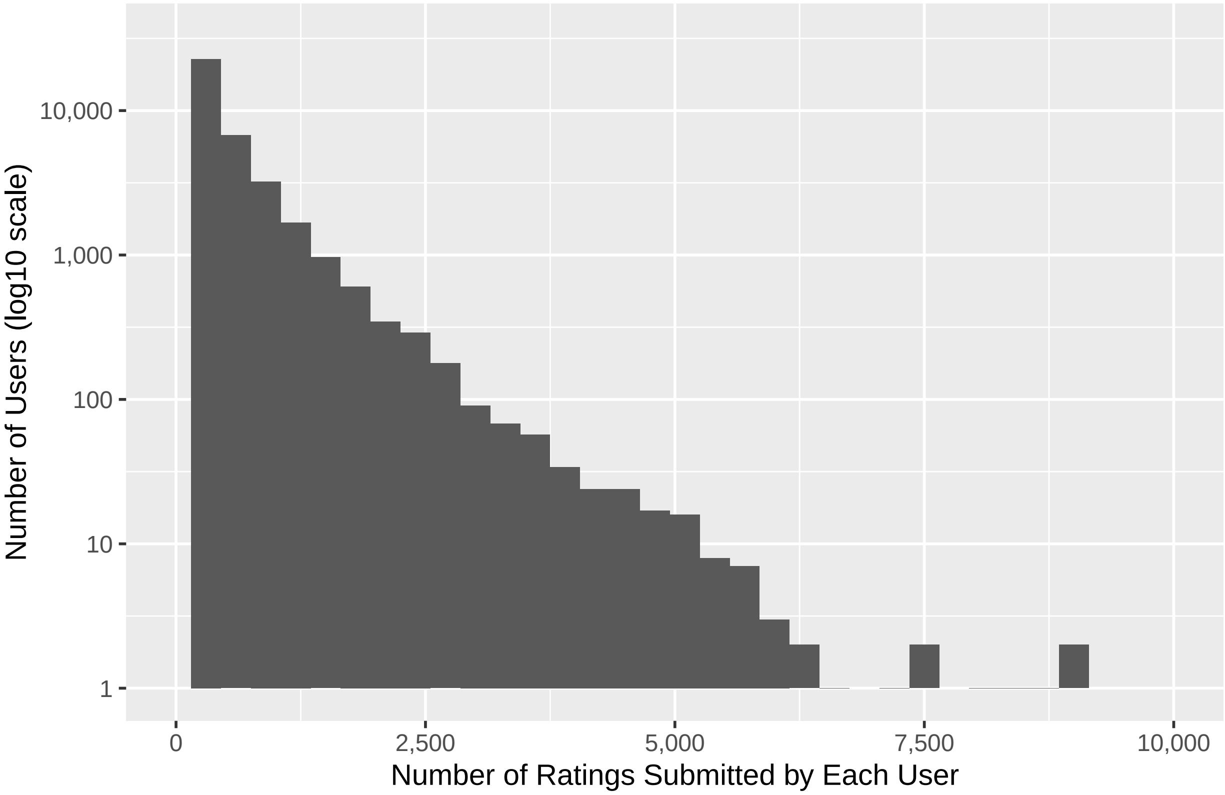


Figure 2 − Number of Ratings Sumitted by Users

Figure 3 plots the number of ratings submitted on a given day and shows how this number varies. There are many days with low numbers of ratings and fewer days with high numbers of ratings.

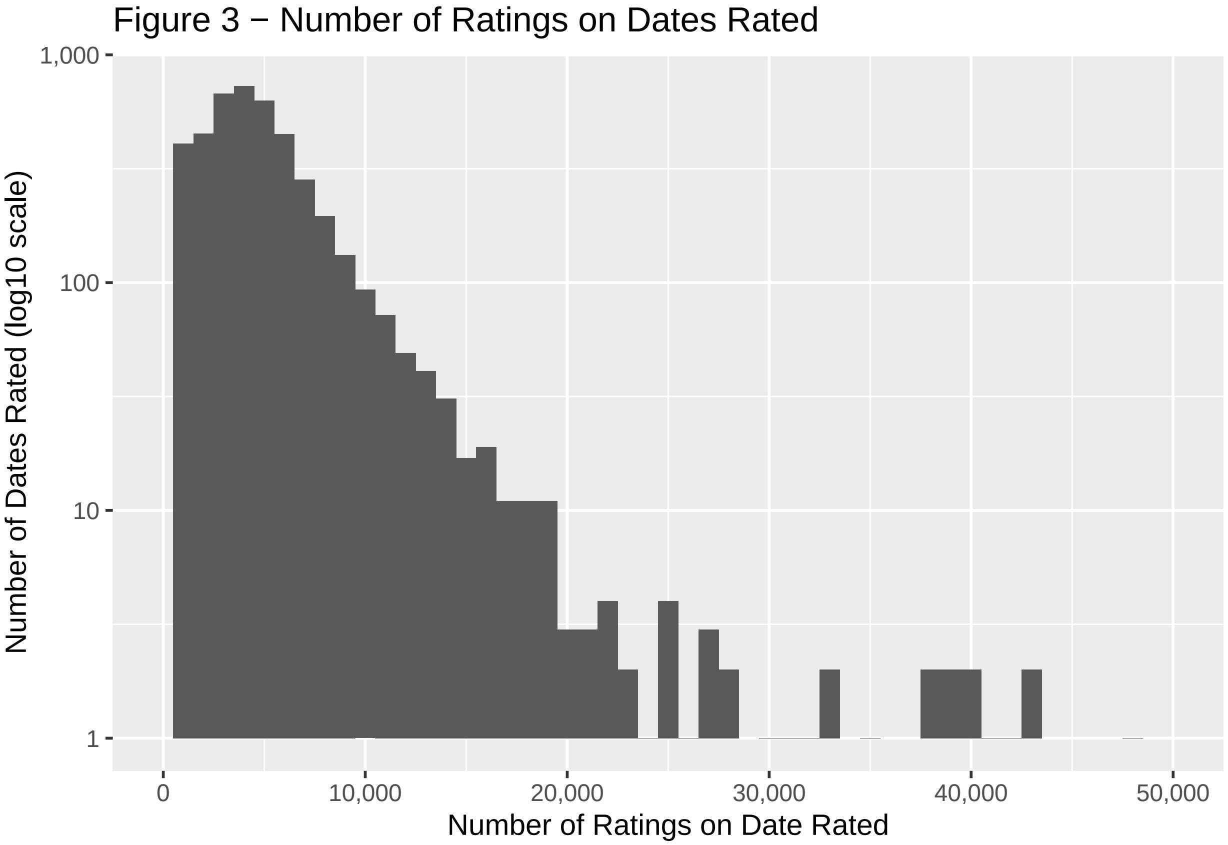


Figure 4 is a bar plot of the number of ratings at each rating level. The plot is skewed to the right and shows more higher ratings are given than lower ratings (there are more 3, 4, 5 ratings than 1, 2 ratings). There are also a larger number of whole number ratings than ratings with 0.5 increments. Additionally, the plot includes a vertical line at x =3.5268887 marking the overall average rating for *edx_tidy*. If edx_tidy ratings were normally distributed the mean would be 2.75.

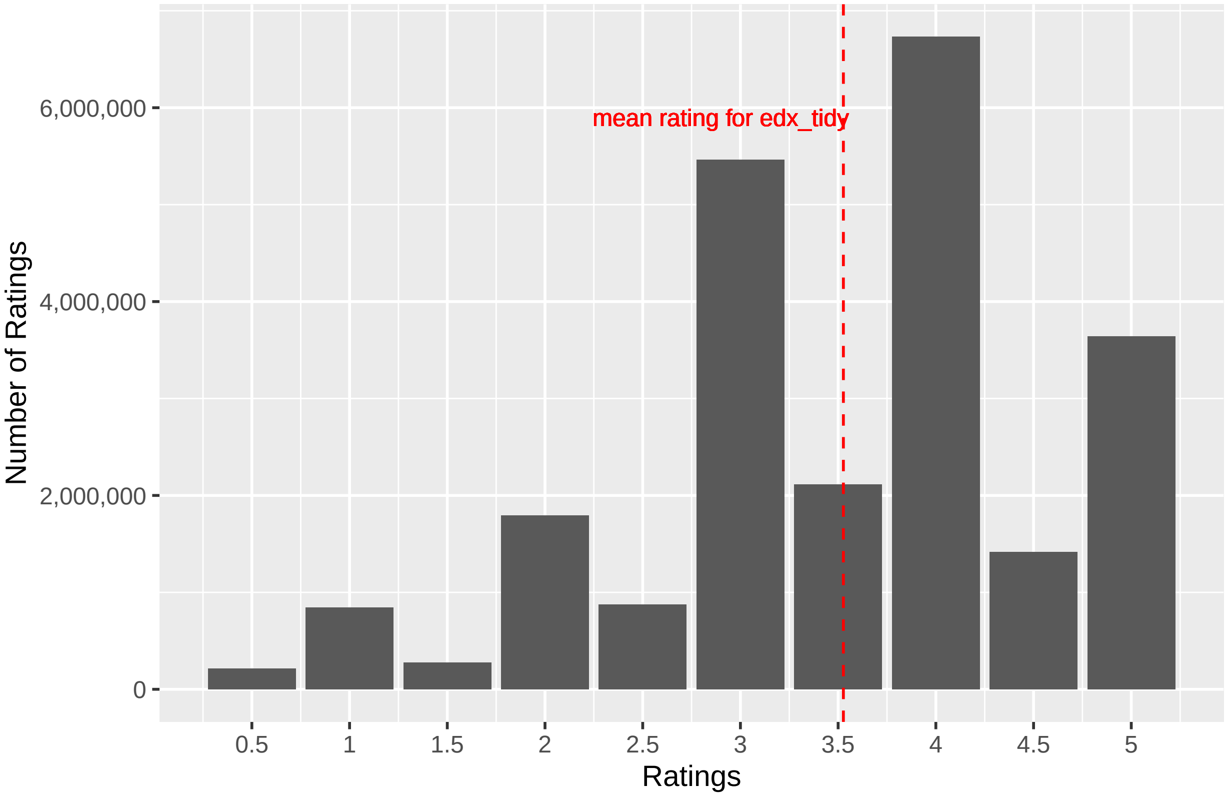
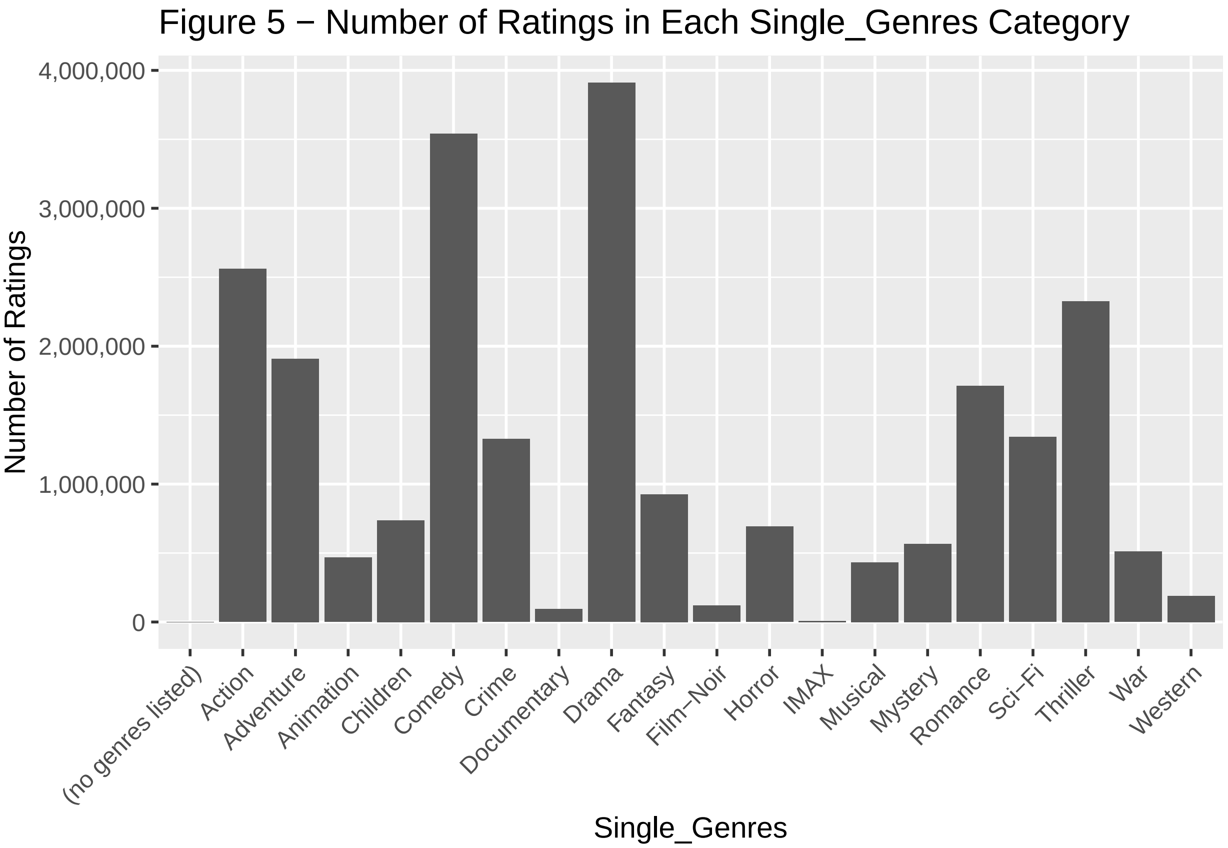


Figure 4 − Number of Ratings in Each Rating Category

Figure 5 is a bar plot of the number of ratings in each genre category showing the differences between the number of ratings from genre to genre. Certain genres (Drama, Comedy, Action, Thriller and Adventure) have many ratings while others (IMAX, Documentary, Film-Noire, Western) have many less. The genres are arranged alphabetically on the x axis.



Figures 6, 7 and 8 are plots of the average ratings for movies, users and dates_rated, respectively.

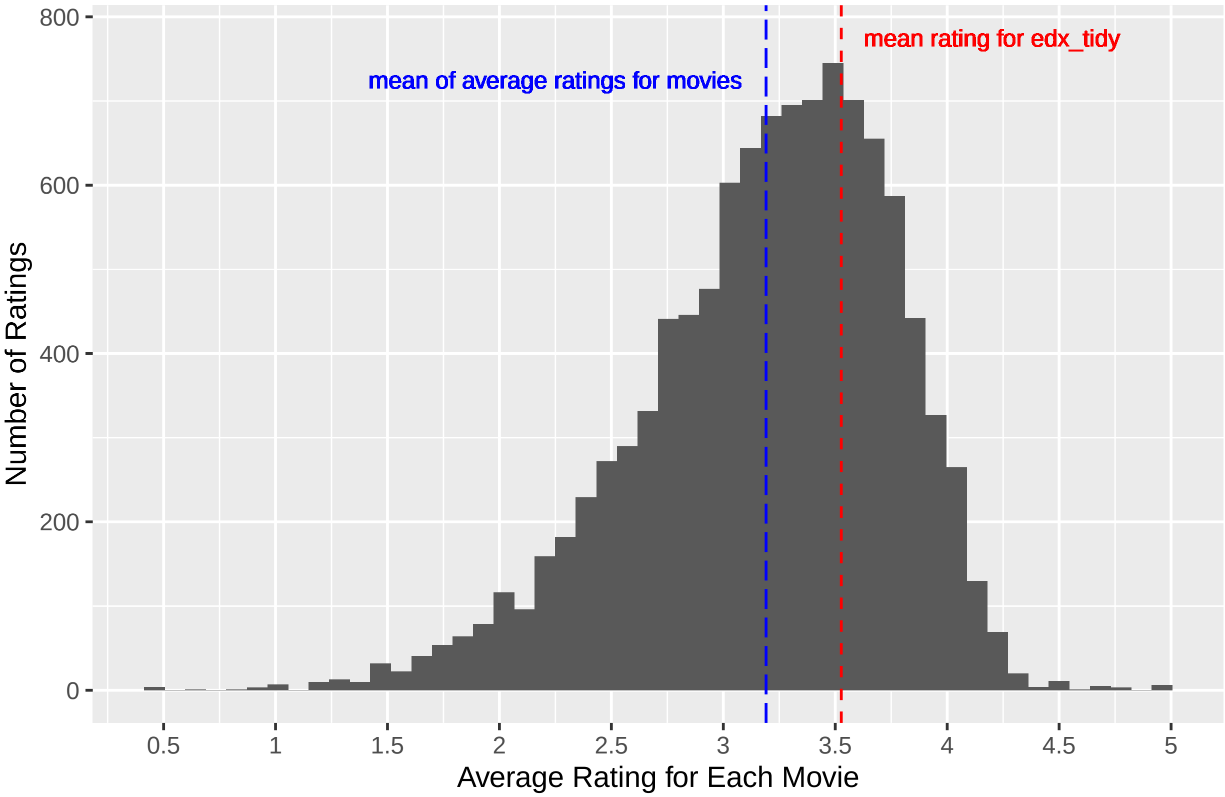


Figure 6 − Number of Average Ratings for Movies

Figure 7 − Number of Average Ratings for Users

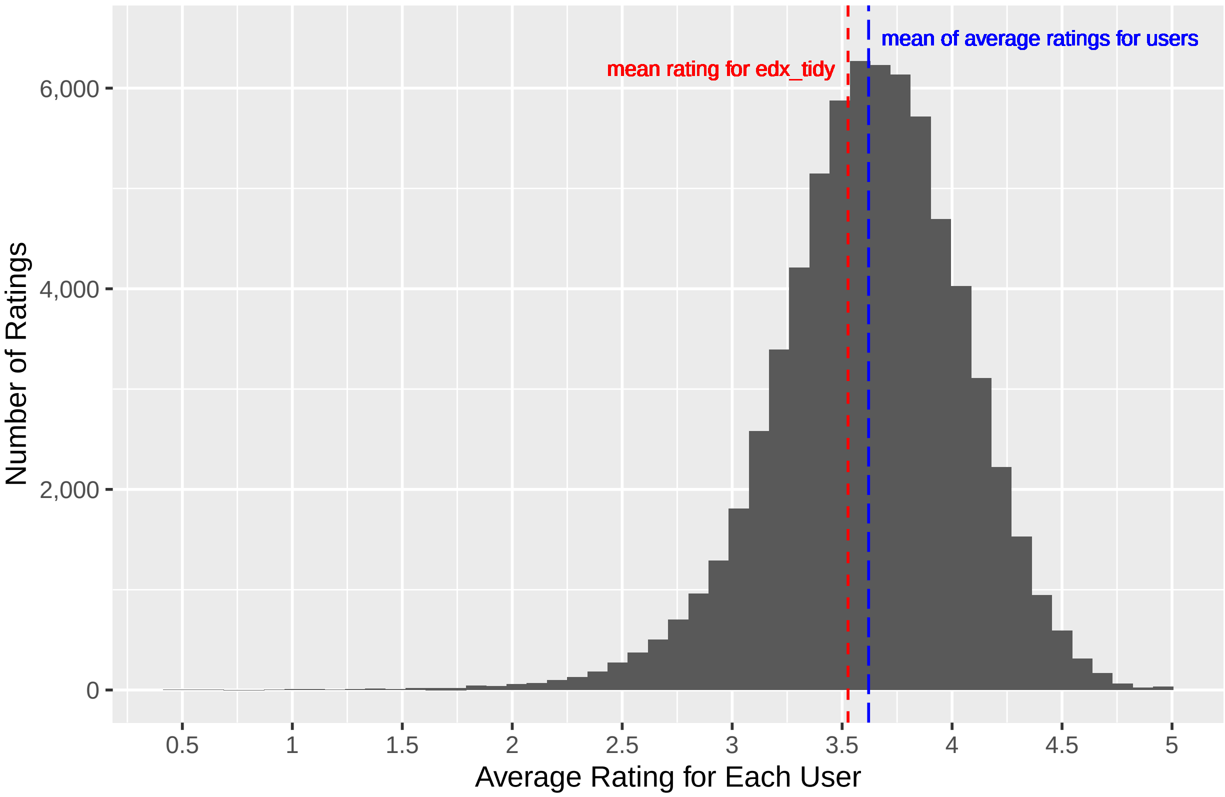
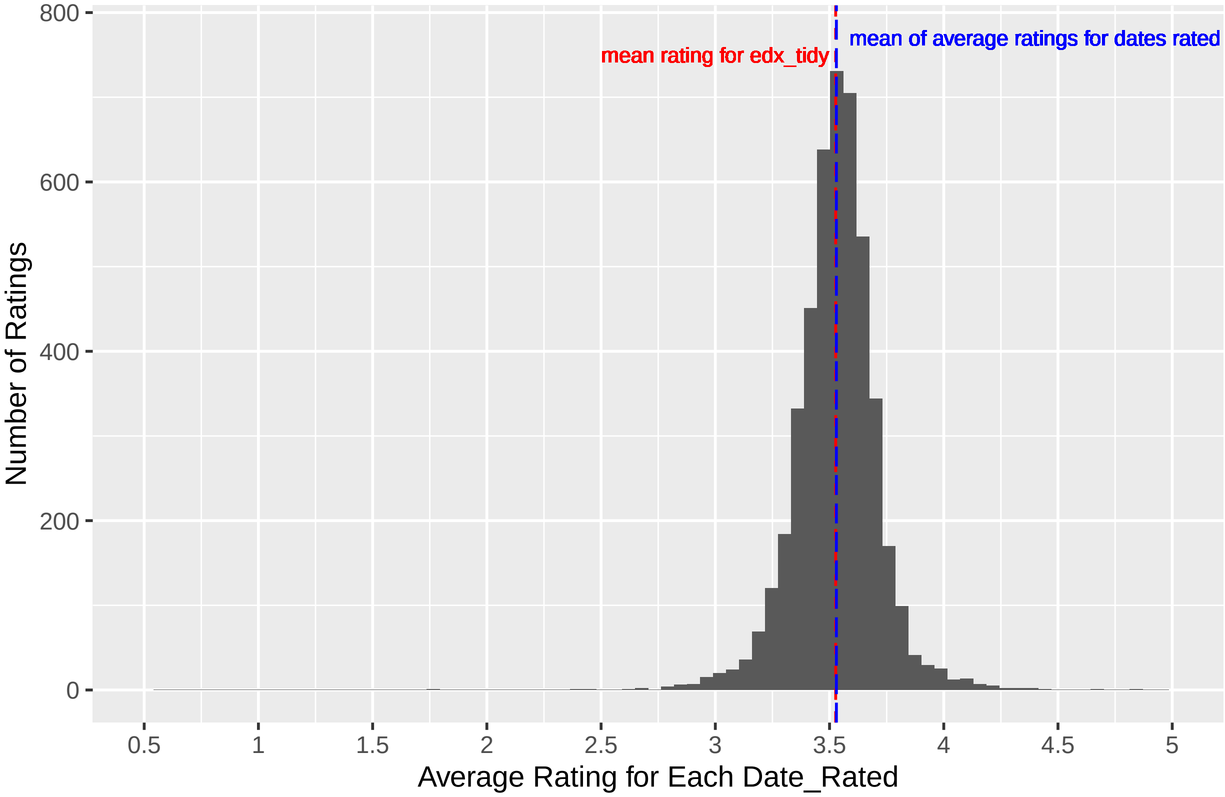


Figure 8 − Number of Average Ratings for Dates Rated



The plots in Figures 6, 7 and 8 are close to normally-distributed and are characterized by the means and variances presented in Table 4.

Table 4

Average Rating by	Mean of Average Ratings Distribution	Variance of Average Ratings Distribution
Movie	3.19	0.325
User	3.62	0.186
Date_Rated	3.53	0.034

The distribution of average rating by movie has the largest variance (0.325), followed by the distribution of average rating by user (0.186) and then the distribution of average rating by dates_rated (0.034). The means and variances of these distributions in comparison to the mean of the ratings in *edx_tidy* (3.53) inspired the use of the individual movie means as the first term and the individual user effects as the second term in the design of the movie rating prediction model.

Model Approach and Design

The edX Data Science: Machine Learning course introduced the following recommendation system model as:

$Y_{iu} = \mu + b_i + b_u + \epsilon_{iu}$

where:

- Y_{iu} is the prediction for a specific movie i and a specific user u,
- μ is the mean of all the ratings in the dataset,
- b_i is a movie specific adjustment used to capture how different movies are rated,
- b_u is a user specific adjustment used to capture how different users rate movies, and
- ϵ_{iu} is an error term containing all of the remaining random variation.

Here μ is the average of the ratings in the dataset. The movie effect, b_i, is estimated by $b_i = Y_{iu} - \mu$ where the average of all the ratings is subtracted from the specific movie/user rating. The user effect, b_u is estimated by $b_u = Y_{iu} - \mu - b_i$ where the average of all the ratings and the movie effect are both subtracted from the specific movie/user rating. The movie effect (b_i) and user effect (b_u) are movie and user specific adjustments which attempt to capture how ratings differ for different movies and users, respectively.

This Capstone MovieLens Project movie rating model is:

$Y_{iu} = \mu_i + b_u + \epsilon_{iu}$

where:

- Y_{iu} is the prediction for a specific movie i and a specific user u,
- μ_i is the mean of a specific movie's ratings,
- b_u is a user specific adjustment attempting to capture how different users rate movies, and
- ϵ_{iu} is an error term containing all of the remaining random variation.

This model uses the edX Data Science: Machine Learning course model as a guide but rather than using the average of all of the ratings in the data as its first term this model uses each movie's average rating, μ_i. The intuition for this comes from Figure 6. It shows the large variation in average rating for individual movies therefore the individual movie effect is important in determining ratings. Starting with each movie's average rating was expected to produce a more accurate prediction than the average for the entire dataset. The model then adds an individual user effect which is calculated from the data using $b_u = Y_{iu} - \mu_i$. The large variation in individual user average ratings demonstrated in Figure 7 supports this choice.

In order to train this model the *edx_tidy* dataset was divided into training and test sets: *edx_tidy_train* and *edx_tidy_test*.

In order to show how the model's predictions compare to the actual ratings the following root mean squared loss function was used.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}$$

The closer the RMSE is to zero the better the model's predictions. For this model the goal is a RMSE less than 0.86490.

The model was trained on *edx_tidy_train* and predictions made from *edx_tidy_test*. These predictions were then compared to the actual ratings in *edx_tidy_test* and produced a RMSE of 0.85811. This was below target RMSE of 0.86490 so model was ready to test with *validation_tidy* data.

Results

The trained model produced a RMSE of 0.85811 when tested on *edx_tidy_test* data. When the trained model was used to make predictions from the *validation_tidy* data and those predictions were compared to the actual ratings in *validation_tidy* a RMSE of 0.85698 was achieved. This model meets the goal of beating a RMSE of 0.86490.

Conclusion

This edX Data Science: Capstone Course MovieLens Project produced a movie rating prediction model from data downloaded from the MovieLens Datasets of GroupLens. The model used the average rating for individual movies plus an individual user effect determined from the *rating*, *movieId* and *userId* variables of the *edx_tidy* data. It successfully made rating predictions from the *validation_tidy* data and produced a RMSE of 0.85698 when the movie rating predictions were compared to actual movie ratings.

Limitations of the model include its very basic design and use of only two of the five variables from the data for prediction. There may be additional rating information contained in the *single_genres*, *date_rated* or the *title* variables which were not included in the model. The *title* variable contains the name of the movie and the year of movie release. The year of movie release may provide additional rating variation information and could be made into a variable but was not done so by this work.

Future work may include:

- construction of a new "movie release year" variable,
- exploration of the rating variation information in the *single_genres*, *date_rated*, and movie release year variables, and
- attempting to design a model using k-nearest neighbor or principal component analysis methods.

References

1. http://grouplens.org/datasets/movielens/10m/