Movie Recommendation System

Mazen Hamdoun

07/04/2021

Executive Summary

This documents serves to lay out the building blocks of a movie recommendation system model based on the provided data set.

The data set is comprised of movie ratings from different genres submitted by a multitude of users.

Model performance is determined by the Root Mean Squared Error (RMSE).

Data exploration and visualization of data provide a clear indication of several effects that were used to construct the model, namely:
- Movie Effects: where movies tend to be rated higher or lower than other movies
- User Effects: where users tend to rate movies higher or lower than other users
- Genres Effects: where users have genres preferences, and therefore tend to rate then higher or lower than other genres

Regularization was used to penalize scarce data for all 3 effects listed above so that the influence of such data-driven anomalies on prediction is reduced.

Data Analysis and Visualization

To assess whether the effects discussed above exist, multiple plots are produced supporting their inclusion in the model.

Data Download and Description

The dataset has millions of records with the following data as seen by revealing the first few records from the training set (edx):

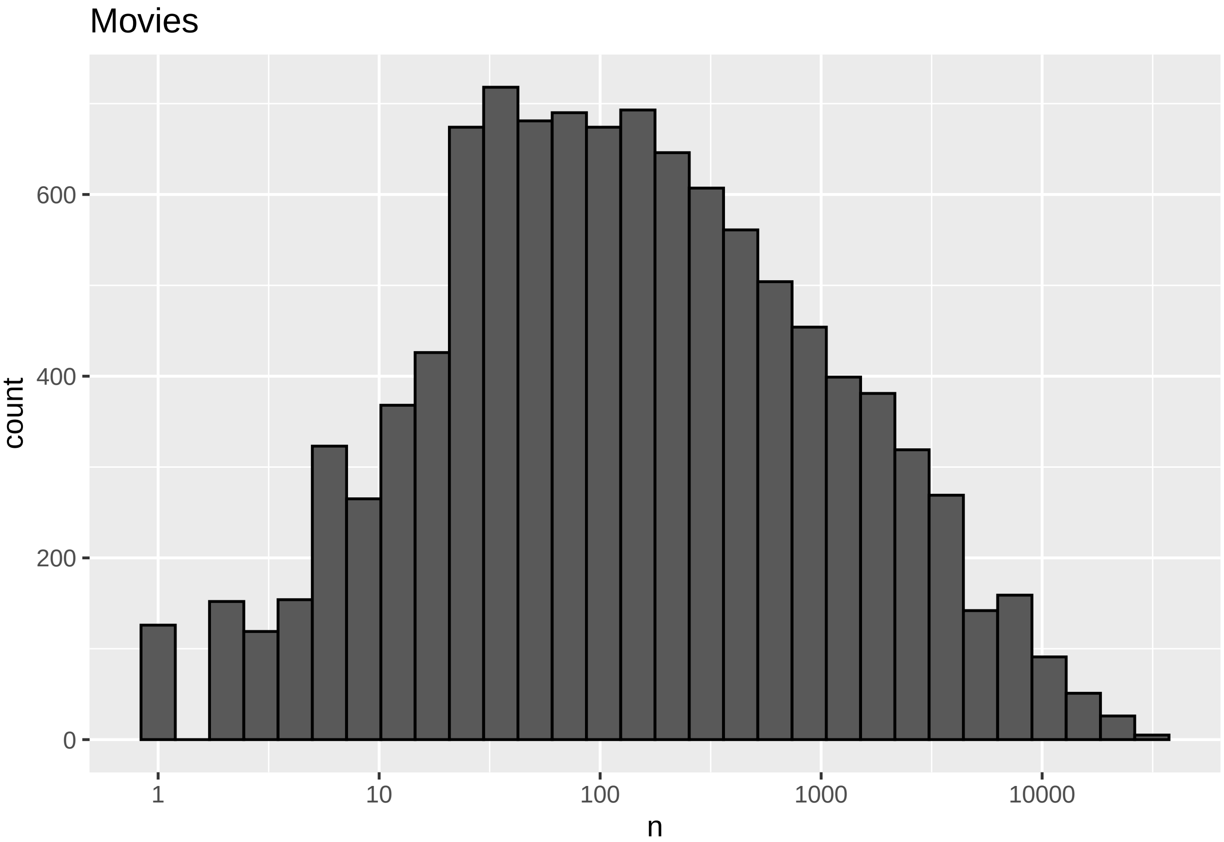
```
##    userId movieId rating timestamp                         title
## 1:      1     122      5 838985046              Boomerang (1992)
## 2:      1     185      5 838983525               Net, The (1995)
## 3:      1     292      5 838983421               Outbreak (1995)
## 4:      1     316      5 838983392               Stargate (1994)
## 5:      1     329      5 838983392 Star Trek: Generations (1994)
## 6:      1     355      5 838984474       Flintstones, The (1994)
##                           genres
## 1:                Comedy|Romance
## 2:         Action|Crime|Thriller
## 3:  Action|Drama|Sci-Fi|Thriller
## 4:       Action|Adventure|Sci-Fi
## 5: Action|Adventure|Drama|Sci-Fi
## 6:       Children|Comedy|Fantasy
```

userId : User ID that defines each unique user
movieId : Movie ID that defines each unique movie
rating : Rating assigned by the specific user to a given movie ranging from 1 (worst) to 5 (best)
timestamp : An integer which can be converted to a timestamp (date and time)
title : The movie's title
genres : All genres the movie belongs to separated by a pipe ("|")

The data has been split into 2 sets: training and validation (hold-out test) sets. The hold-out test set will only be used to produce the final model RMSE.

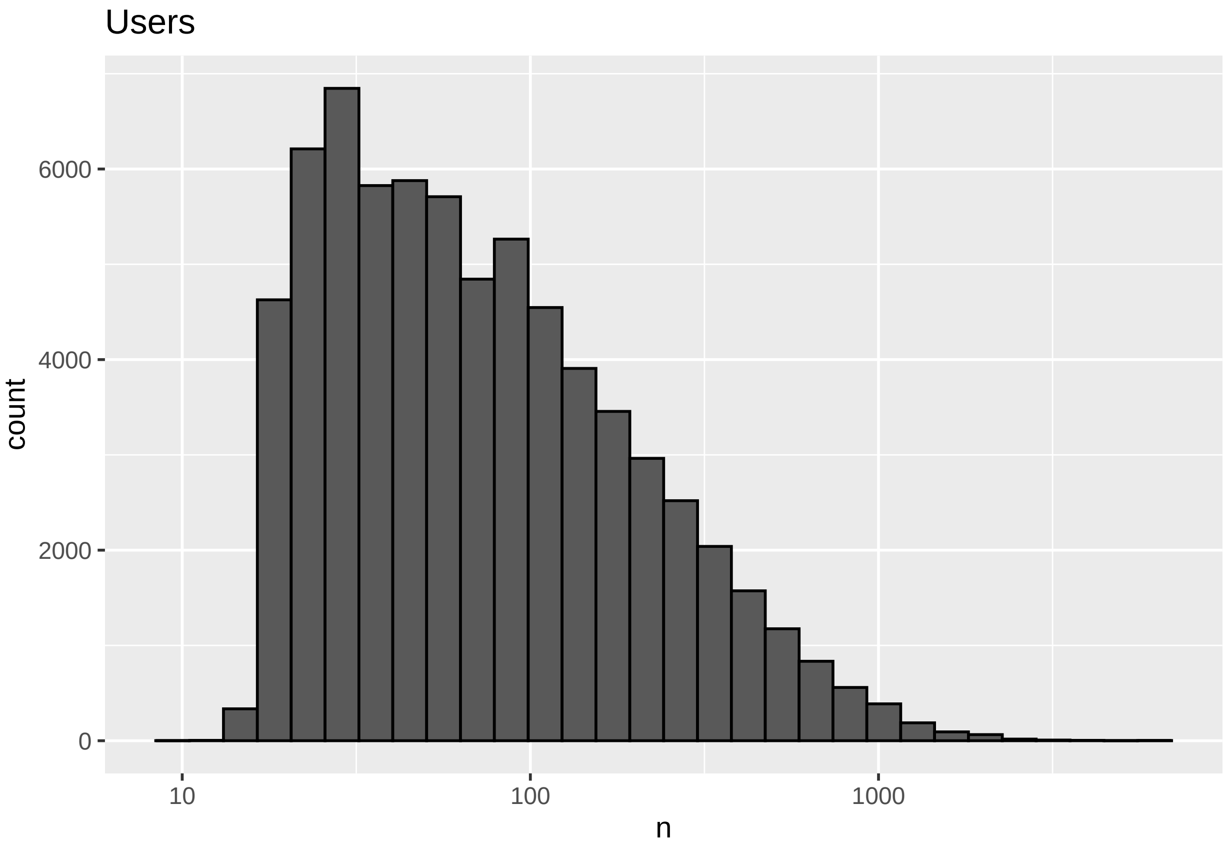
Rating Counts

The histogram shows the distribution of ratings by groups (within bins) of movieID in log base 10 scale



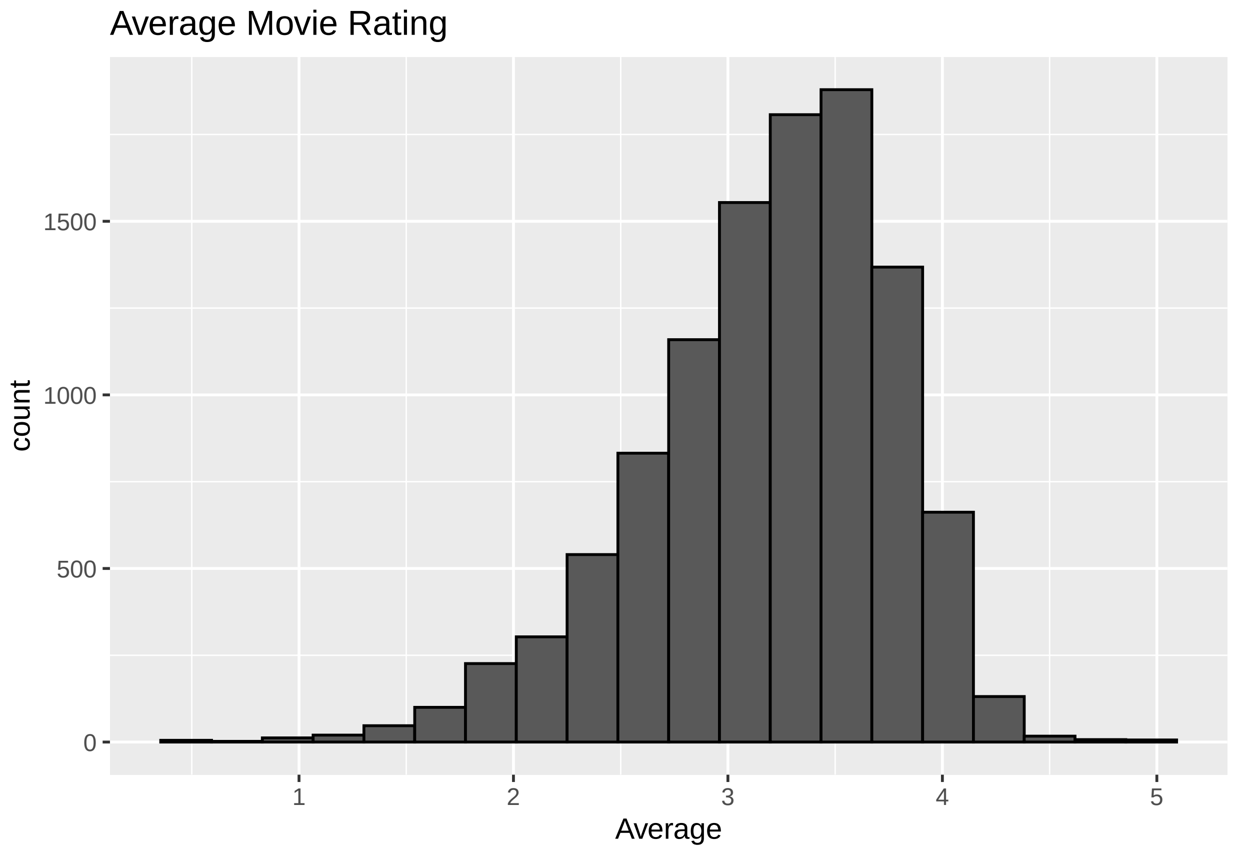
Finding: Some movies tend to have many more ratings than others

The histogram shows the distribution of ratings by groups (within bins) of userID in log base 10 scale



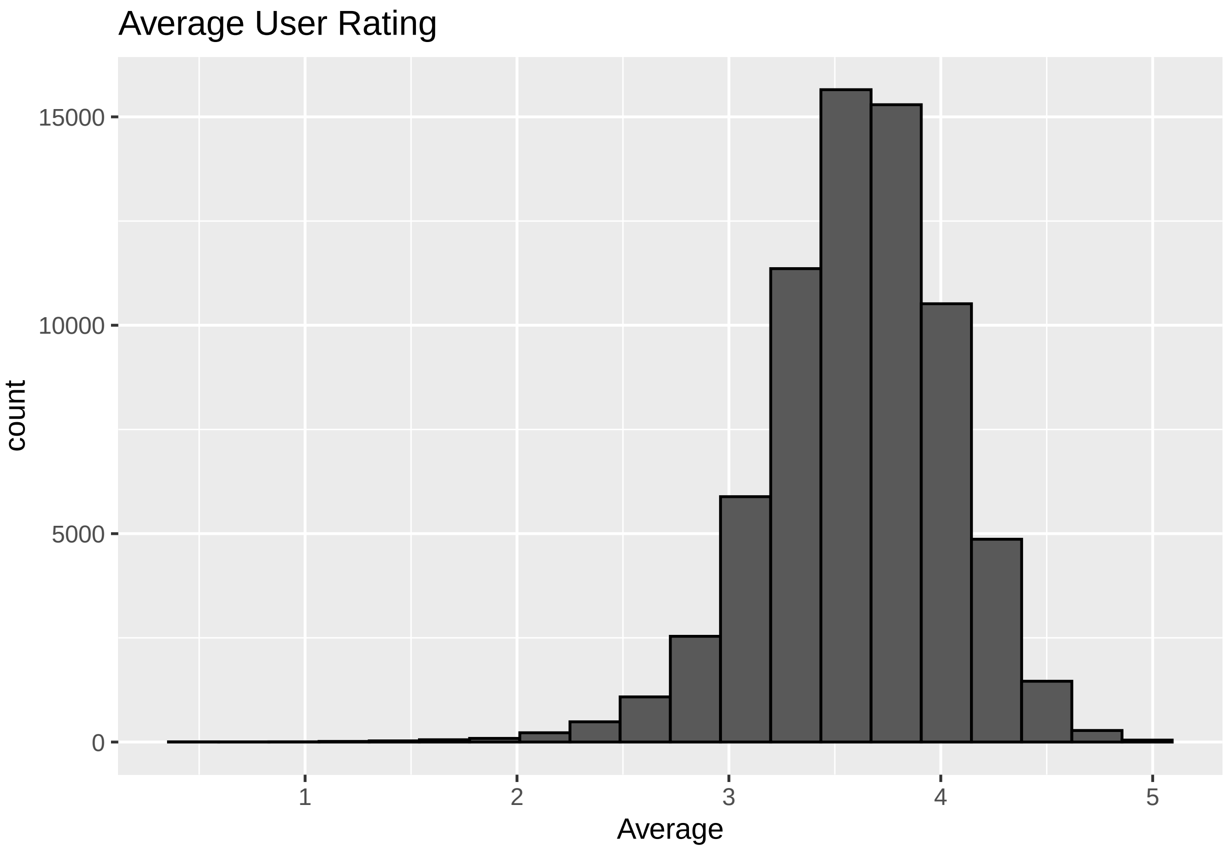
Finding: Some users tend to rate many more movies than others

Movie Effect



Finding: The distribition of average movie ratings clearly indicates that some movies have higher ratings vs others, proving the need to model this effect

User Effect



Finding: The distribition of average user ratings also indicates that some users tend to give higher ratings vs others

Genres preference Effect

Users typically have preferences for particular genres of movies, and hence tend to rate them higher.

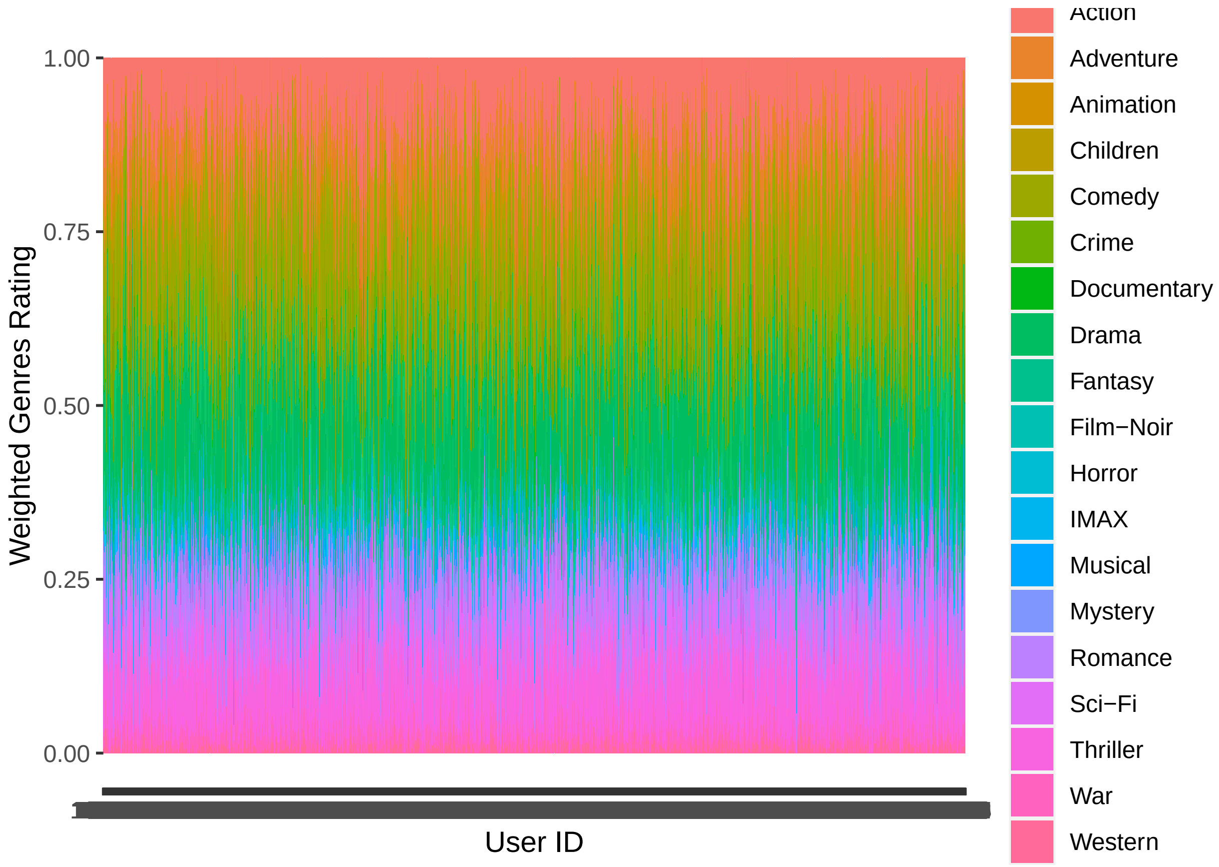
The following plot shows a standardized plot of average ratings by genres of movies for a sample of 10,000 users. Users tend to watch more movies of genres they particularly prefer. Therefore, ratings are weighted by the number of movies per genres relative to total ratings for each user.

```
uid <- edx %>% distinct(userId)

#Sampling 10,000 users randomly for visualization
set.seed(1,sample.kind = "Rounding")
ug_rating <- edx_tidy[which(edx_tidy$userId %in% sample(uid$userId,10000,FALSE,NULL)),]  %>% group_by(u
               summarise(Average=mean(rating),n=n())

ug_rating <- ug_rating %>% group_by(userId) %>% summarize(sum_wt=sum(n)) %>%
              inner_join(ug_rating,by="userId") %>%
              mutate(wtd_genres=(ug_rating$Average*ug_rating$n/sum_wt*100))

ug_rating %>% ggplot(aes(as.character(userId),wtd_genres,fill=genres)) + geom_bar(stat = "identity",pos
```



Finding: Genres color bands are not aligned, therefore weighted ratings indicate that each user has specific movie genres preferences

The following genres by user Look-Up Table is created to allow for aggregation of results for the Genres effect later.

```
##   userId                          t   genres
## 1      1             Comedy|Romance   Comedy
## 2      1      Action|Crime|Thriller   Action
## 3      1  Action|Drama|Sci-Fi|Thriller   Action
## 4      1     Action|Adventure|Sci-Fi   Action
## 5      1 Action|Adventure|Drama|Sci-Fi   Action
## 6      1     Children|Comedy|Fantasy Children
```

Methodology

The model starts from the premise that the average rating is the initial prediction.

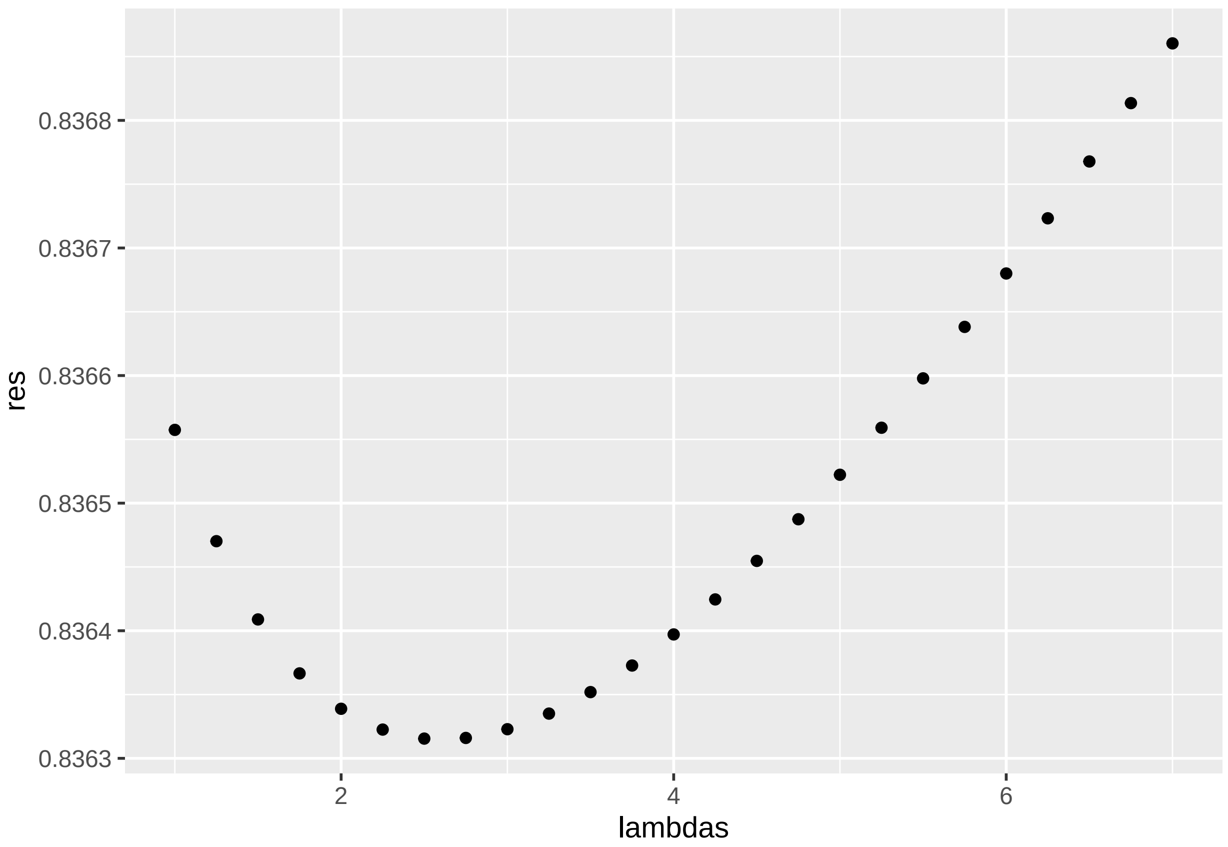
We gradually determine and add effects (movie, user and genres) to allow for more accurate prediction.

The training data set (edx) is split into training and test sets to assess model performance

Regularization is used to reduce the undesirable effects of lack of data (users/movies/genres with very few ratings) on the model's predictive ability.

The regularization parameter Lambda is optimized by splitting training data (train_set) into 2 subsets: - optimization training set (train_BS) - optimization testing set (test_BS)

The following plot determines optimal Lambda (having the minimum RMSE) to be used in the model:



Now, we train the model with our optimal lambda on the training set (train_set) and test model output on our test set (test_set).

We commence by showing model performance when incorporating the movie effect to the average rating initially:

RMSE (movie effect) = 0.9348064

Model performance **improves** after adding the ***user effect***:

RMSE (adding user effect) = 0.8528816

Model performance improves **further** after adding the ***genres effect*** :

RMSE (adding genres effect) = 0.836786

We now train the final model including all effects on the full training data set (edx):

Conclusion (Results)

Running the model on the hold-out set (validation), we can be confident enough that its performance is satisfactory based on final performance.

Note: Not all userId/genres combinations in the validation set also exist in the training set, hence we consider no improvement (ie we add zero the user/genres effect) to our prediction.

This provides a model result (RMSE) of: 0.8535974

However, if we remove user/genres combinations not existing in the training set, the final result is even better.

The final model result (RMSE) = 0.8357163