MovieLens Project

Jose Aramendiz

1/28/2022

TABLE OF CONTENT

Introduction

The goal of the Movielens project is to create an algorithm to predict movie rating using the MovieLens data set. The Movilens database has over 10 million ratings for over 10000 movies and more than 72000 users. To develop the prediction algorithm, the data set was divided in to different subsets: the edx set (training set) and te validation set (test set). The validation subset is 10% of the original database and is not use for training purposes while creating the predictive model.

Since the original database is very large, it is not recommended to use the *lm* model due to memory allocation. Instead, the RMSE will be used as the measuring criteria to determine how close are the results obtained from the herein algorithm to the actual validation data set. Different models are evaluated in the edx (training set) to determine which optimized (minimized) the RMSE prior to evaluate the final RMSE model against the validation data set. Regularization was also included.

This report goes goes through the different steps needed for data analyisis, including a descriptive analysis or data exploration section, an analysis section evaluating each model used, and, finally a results and conclusion section with recommendation for future work.

NOTE TO THE GRADER: The code to elaborate this report is hidden. If you decide to take a look at the code, please refer to the .Rmd file or .R code. Thank you for your comments and feedback.

Descriptive analysis/data exploration Section

The first step of the data exploration is to determine the dimensions of the datasets. The edx (training set) contains 9000055 rows and 6 columns, the validation (test set) contains 999999 rows and 6 columns, confirming that the sets have been roughly partitioned in a 9/1 ratio. The potential predictors are userID, movieID associated with the movie title, the timestamp and genre.

```
dim(validation)
```

```
## [1] 999999      6
```

```
dim(edx)
```

```
## [1] 9000055       6
```

```
head(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
```

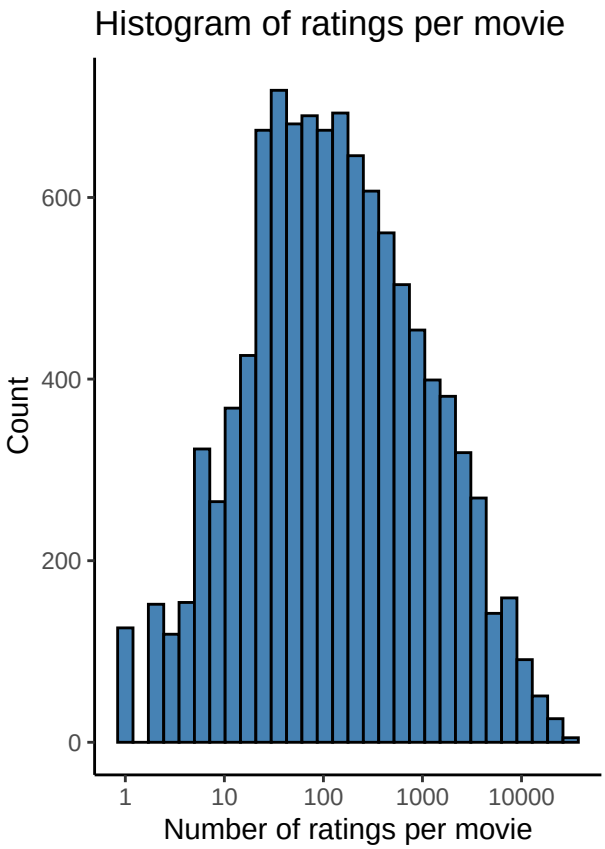
```
mean(is.na(edx))
```

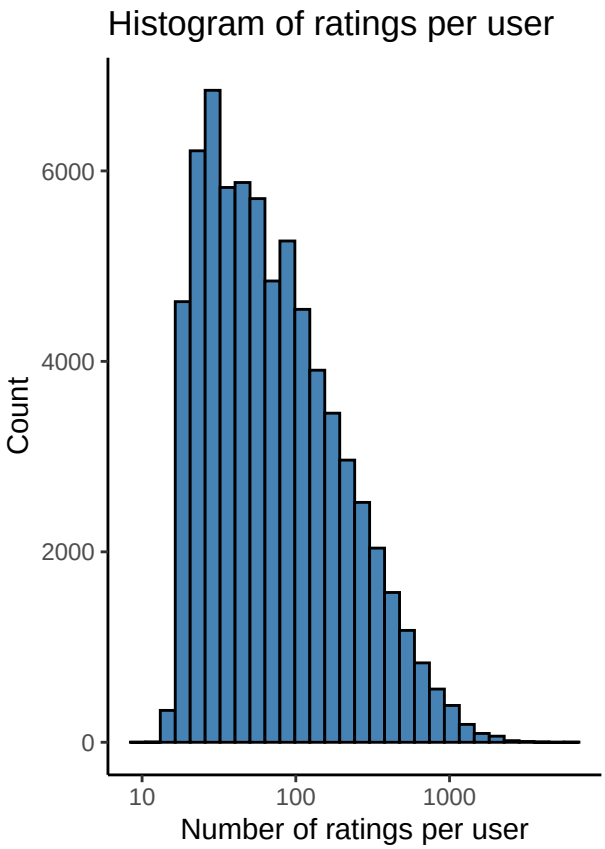
```
## [1] 0
```

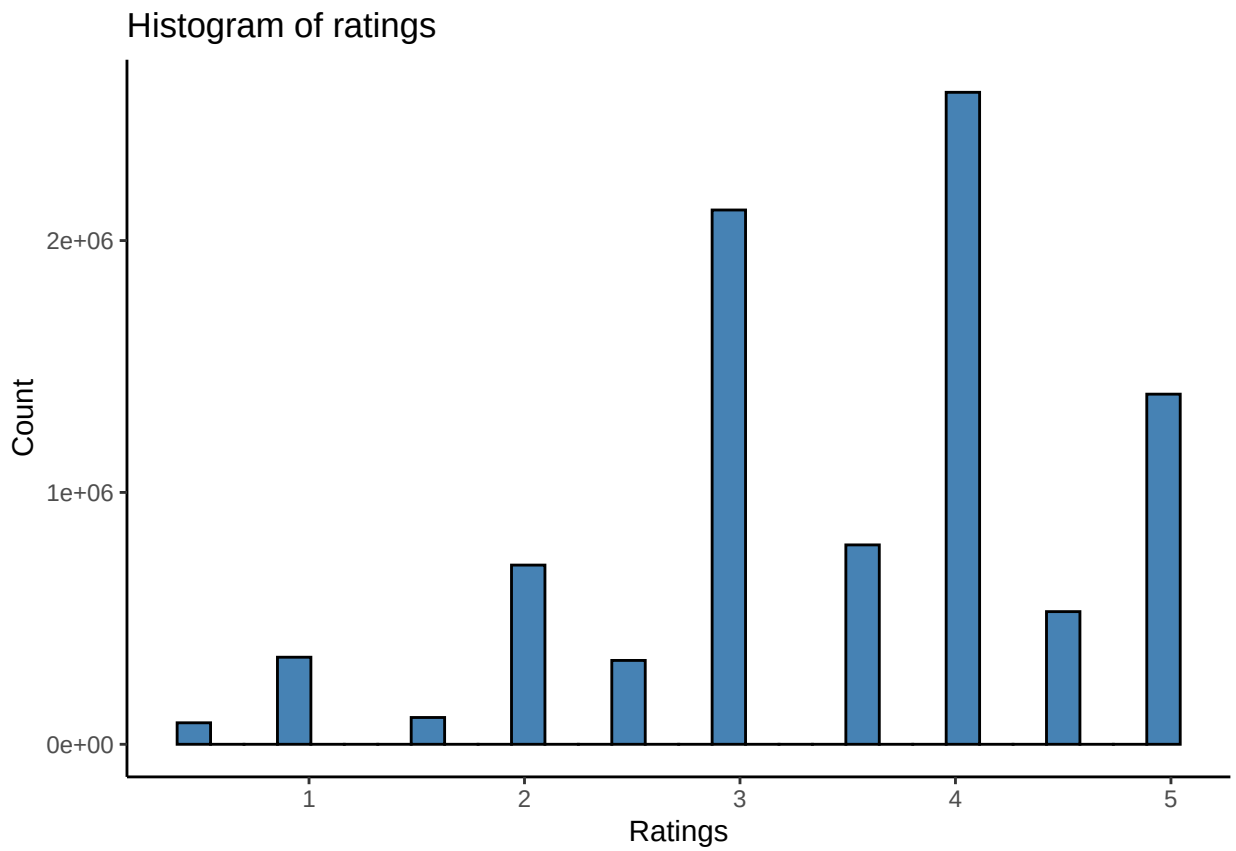
```
mean(is.na(validation))
```

```
## [1] 0
```

On the other hand, 69878 unique users provided ratings for movies and 10677 unique movies were rated, which can give a huge number of possible combinations (> 746 millions). However, as we observed previously the edx set is just above 9 million rows, suggesting that not every user rate every movie, and in fact some users tended to rate more movies than others and in the same line some movies were rated more than others,as shown in the two histograms below. Also, it is possible to establish that integer ratings were more frequent than half-integers.


Histogram of ratings per movie
Count
600
400
200
0
1
10
100
1000
10000
Number of ratings per movie


Histogram of ratings per user
Count
6000
4000
2000
0
10
100
1000
Number of ratings per user



The genre variable contains the classification of genres movies with 20 different types and from the count of ratings it is clear that some genres are more popular that others, being the most rated Drama and Comedy, while documentary and IMAX are the least rated.

```
## # A tibble: 20 x 2
##    genres               count
##    <chr>                <int>
##  1 Drama              3910127
##  2 Comedy             3540930
##  3 Action             2560545
##  4 Thriller           2325899
##  5 Adventure          1908892
##  6 Romance            1712100
##  7 Sci-Fi             1341183
##  8 Crime              1327715
##  9 Fantasy             925637
## 10 Children            737994
## 11 Horror              691485
## 12 Mystery             568332
## 13 War                 511147
## 14 Animation           467168
## 15 Musical             433080
## 16 Western             189394
## 17 Film-Noir           118541
## 18 Documentary          93066
## 19 IMAX                  8181
## 20 (no genres listed)       7
```

Defining RMSE

The term root mean square error (RMSE) is the square root of mean squared error (MSE). RMSE measures the differences between values predicted by a hypothetical model and the observed values. In other words, it measures the quality of the fit between the actual data and the predicted model.

To measure how close the predictions were to the true values in the validation set we will use the RMSE, defined by the following function:

```
#############################################################################
## The RMSE function that will be used in this project is defined as   ##
#############################################################################

RMSE <- function(true_ratings, predicted_ratings){
  sqrt(mean((true_ratings - predicted_ratings)^2))
}
```

Analysis section

Baseline Model

As explained before, due to the size of the dataset, modeling the data using a function like *lm* is not recommended. It can crash your computer. A first approach is to evaluate the simplest model to set a baseline. this model predicts the same rating regardless independently of user, movie or genre. This model would look like this:

Yu,i = mu + Eu,i

Where (Y) represents the expected rating of the movie (i) from user (u), (mu) as the average movie rating and (E) the random variability of the ratings.

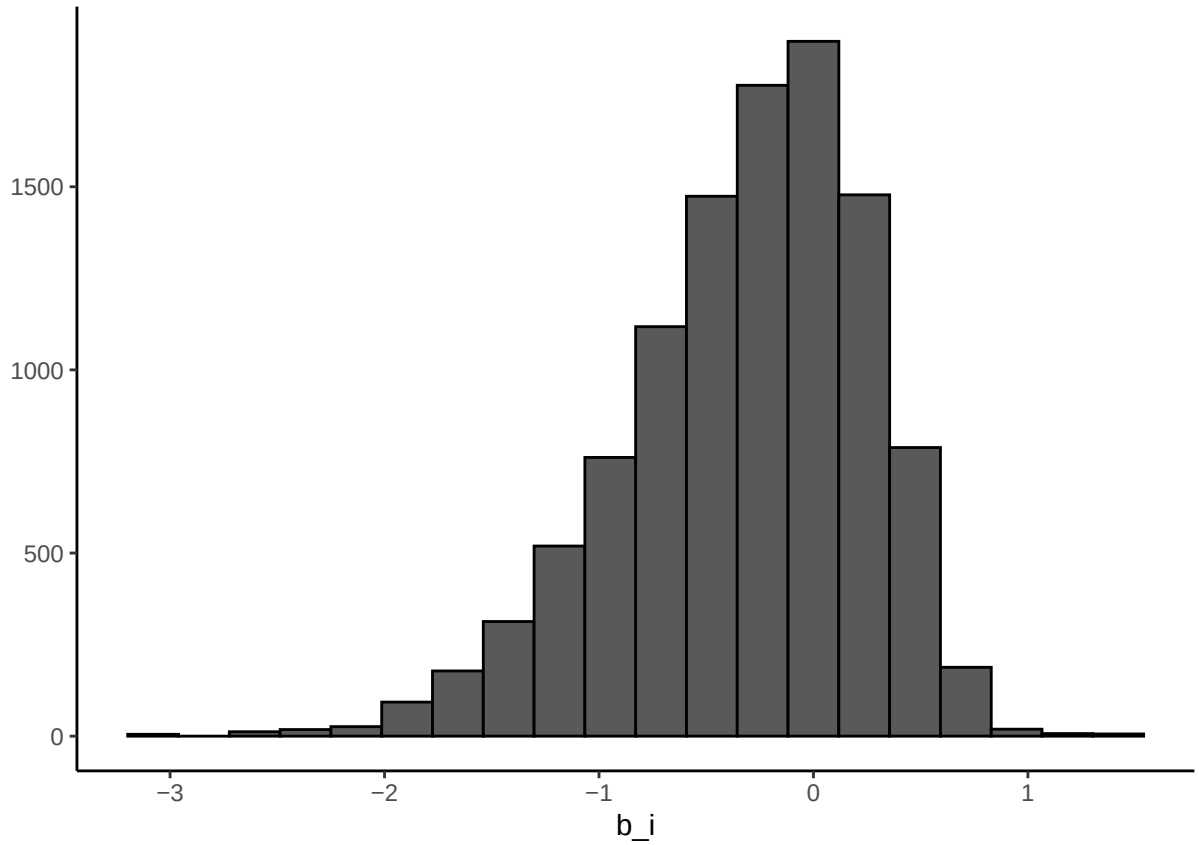
Method	RMSE
Just the rating average	1.060331

The movie average (mu) is 3.5124 and the estimated RMSE for this simple model is 1.06.

Movie Bias or Movie Effect Model

The baseline model does not consider the movie bias effect. Not all movies are good, and not all are bad. Thus, it is possible for some movies to get higher rating than others. We can add to the previous model the movie bias effect (b) that stands for the average rating of the movie (i) regardless of the user.

As we can observe from the following plot, whereas most of the movie ratings are concentrated towards the average movie rating centered to zero, there are other movies substantially deviated from the average. This deviation motivates the inclusion of a movie effect bias parameter to the model.



The Movie bias or movie effect model can be represented as follows:

Yu,i = mu + bi + Eu,i

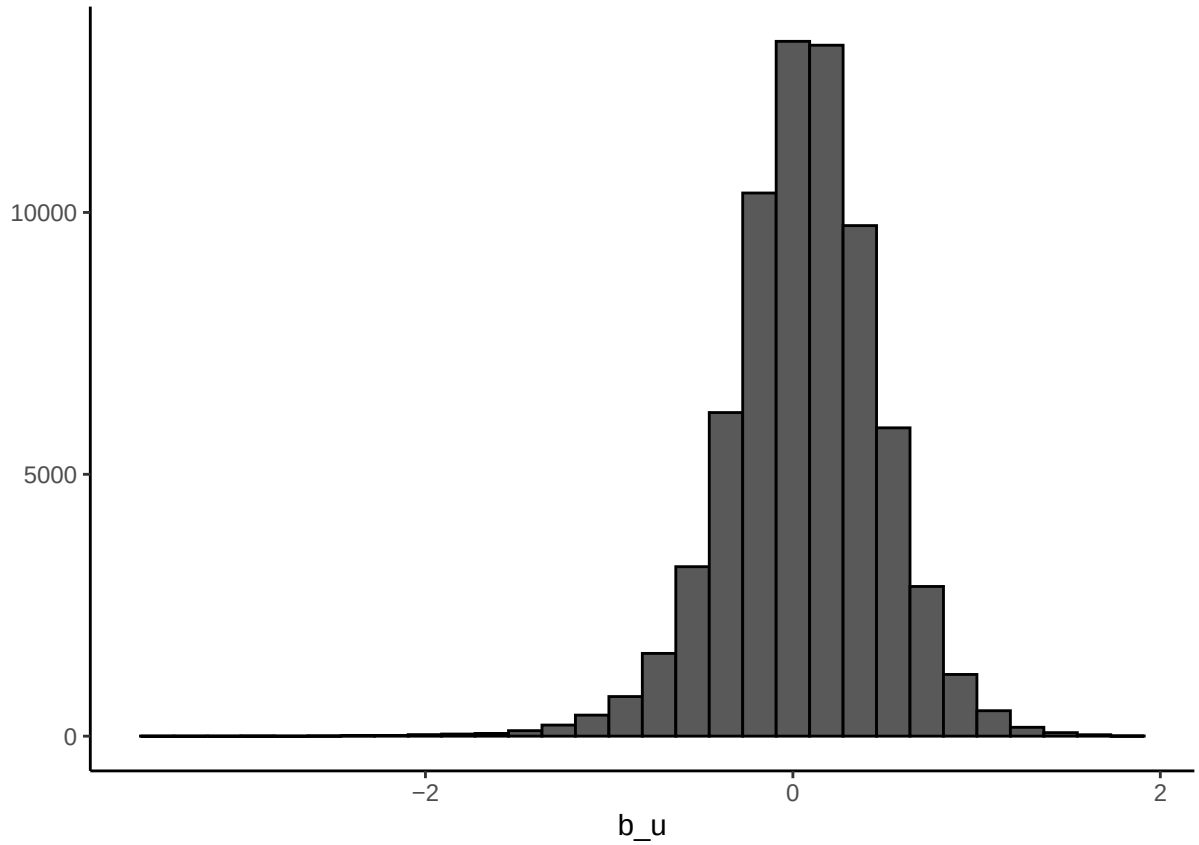
To develop the code, the least square estimate bi is determined as the average of Yu,i - mu for each movie i.

As shown below, we can see that this already improved the model, reducing the RSME to 0.943.

Method	RMSE
Movie Bias Effect Model	0.9439087

User Specific Effect

As learned before, movie effect generates a variability in rating. The same effect is experience with users since some users rate movies higher than others. The following chart shows the user variability rating.



This model considers both, the movie and the user effect, and estimate the user effect as the average of the ratings per user. In that sense, the model can be establish as follows:

Yu,i = mu + bi + bu Eu,i

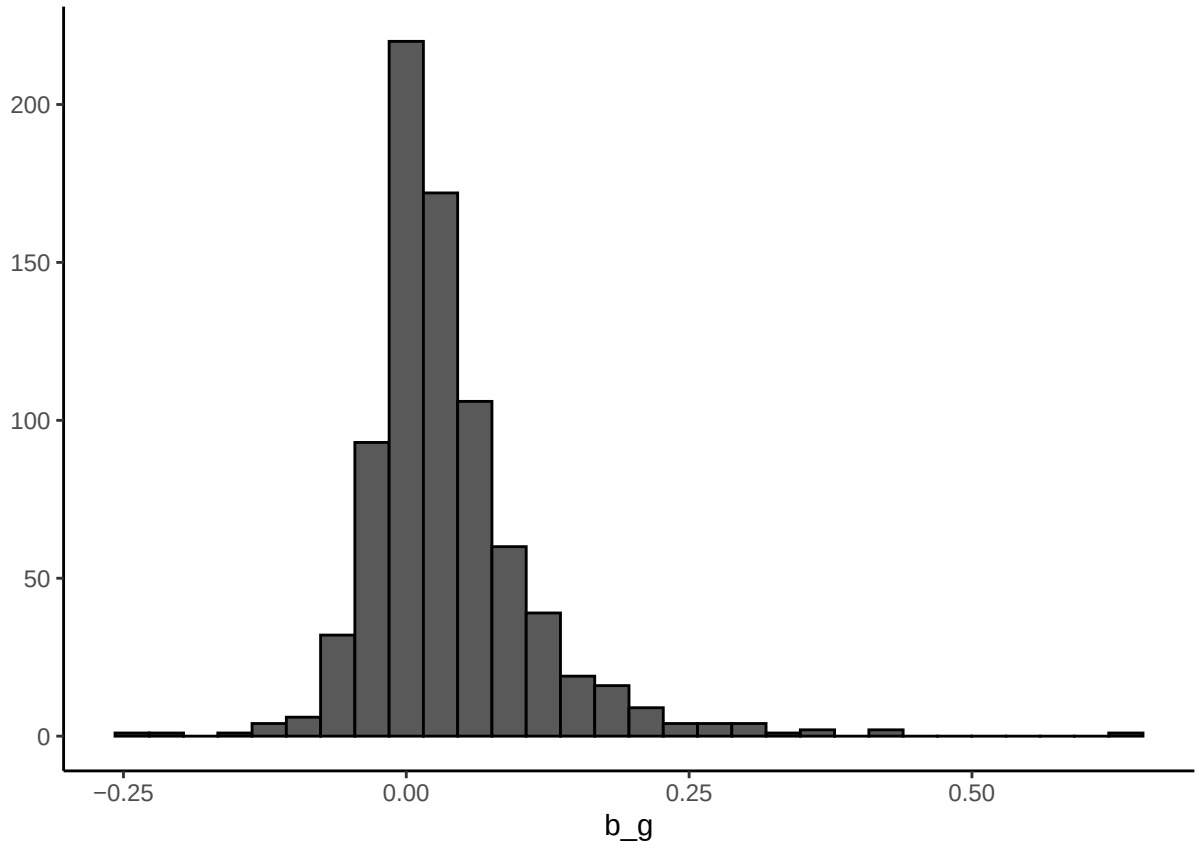
To develop the code, the least square estimate (bu) is determined as the average of Yu,i - mu_hat - bi for each movie (i) and user (u).

Method	RMSE
Movie + User Effects Model	0.8653488

Notably, the model calculated a very good RMSE of 0.865 compare to the first two.

Movie + User + Genre effect

As presented before, the movie ratings vary per genre, so there is a variability as well that can cause a bias effect in the model, so the following model will take into consideration this effect. The chart below represents the genre variability rating and the RMSE result including this effect in the model.



Method	RMSE
Movie + User + Gender Effects	0.8649469

Movie + User + Independent Genre effect

As one can see the effect of genre to reduce RMSE in the previous model is minimum (0.8649), maybe because the model treated some genres together and not independently (ie: "Action|Adventure|Animation|Children|Comedy"). To solve this issue, the new model treats the genres independently: a movie is or not of a certain genre.

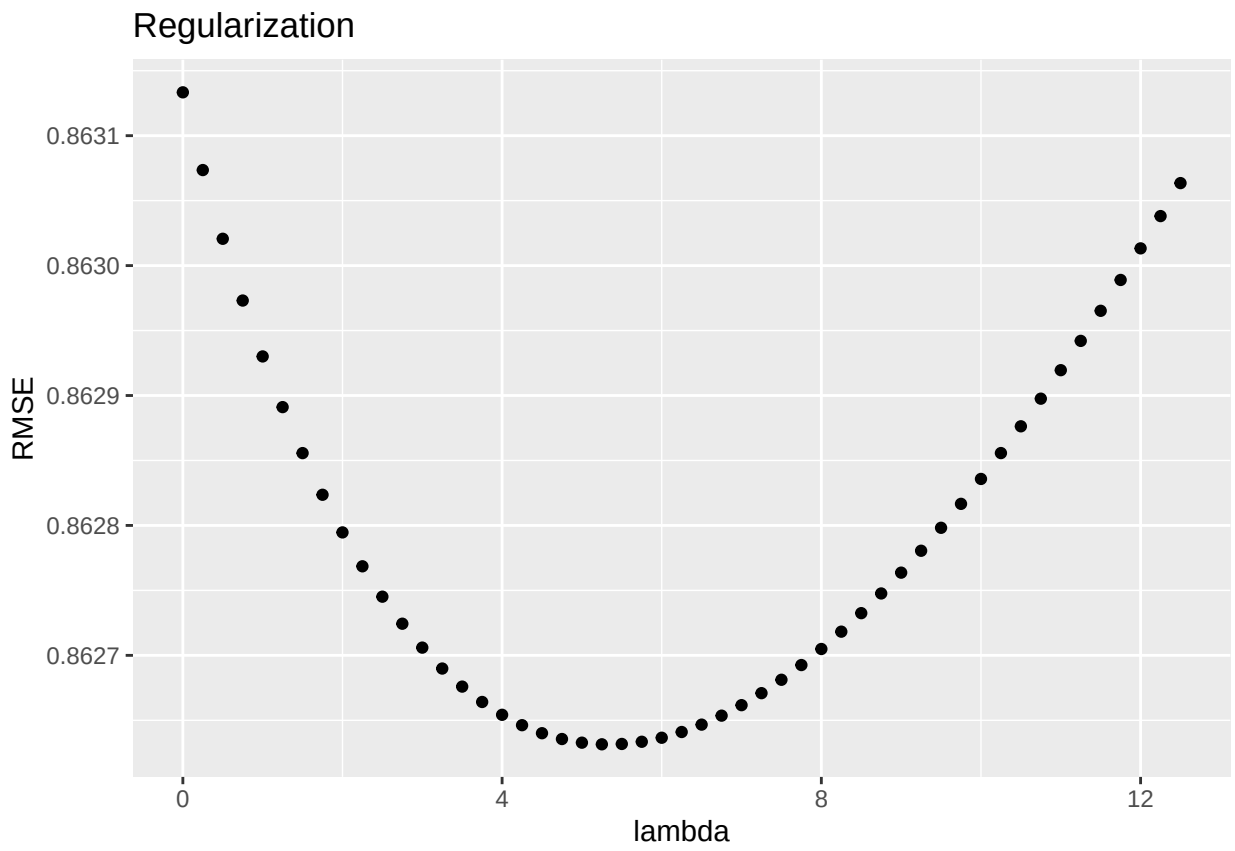
Method	RMSE
Movie + User + Genres Ind. Effects Model	0.8631334

Treating the genre effect independently reduced the RMSE slightly more than before to a value of 0.8631.

Regularization

A final consideration take into account that rating estimate for a movie rated many times is more likely to be more precise than the estimate of a movie rated few times. Not penalizing low estimates can lead to mistakes in the overall prediction. Regularization can help to control this condition.A penalty factor (called λ) is introduced to the model. As the sample size increases, the penalty effect decreases and since λ is a tuning parameter and we can use cross-validation to estimate the λ that minimizes the RMSE for the model.

The following plot represents the behavior of the RMSE as the λ changes.




```
## [1] 5.25
```

The λ which optimizes the model (minimizes RMSE) is 5.25 in this case and the result of the final model with regularization shows a good performance when used in the validation set as shown bellow

Method	RMSE
Regularized Movie + User + Genre Ind Effect Model	0.8626315

Results section

To predict movie ratings we create diverse models that considered the effects of movies, users, genres and interactions between them. The best model (regularized) considered all, yielding to an RMSE of 0.8626. The movie effect has the highest impact in the reduction of the RMSE. The later indicates that the movie itself is a key factor to describe the rating. The following table summarize the RMSE results including all models.

Method	RMSE
Just the rating average	1.0603313
Movie Bias Effect Model	0.9439087
Movie + User Effects Model	0.8653488
Movie + User + Gender Effects	0.8649469
Movie + User + Genres Ind. Effects Model	0.8631334
Regularized Movie + User + Genre Ind Effect Model	0.8626315

Conclusion Section and Future Work

The main objective was to develop a model to predict movie ratings from a large database containing millions of evaluation. To develop the different models it was considered the impact of movies, users, and, movie genres to the ratings. To avoid over fitting the database was divided into to subsets, edx (training set) and validation data set (test set). Regularization method was implemented to the final model with the lambda tuning parameter equal to 5.25 which was the value that optimizes the final model (minimizes RMSE). The final model (best-fitted) yielded an RMSE of 0.8626, which is a good results when compare to the RMSE scale guideline of this course.

Although, the final model achieved a good RMSE value, it would be interesting to evaluate the impact of other effects in the model such as the user-genre effect because one can expect that users rate genres differently. Also, the movie release year could be evaluated as some users may prefer old-style movies than newer movies and if more information about those user (e.g. age and gender) can be included it may improve the prediction model.