HarvardX Professional Certificate in Data Science - MovieLens Analysis

Ferdinand Pieterse

April 26, 2019

Contents

1 Introduction

This project is produced for the capstone for HarvardX Professional Certificate in Data Science. It describes the basic process for developing a recommendation system.

Recommendation systems use ratings that users have given items to make specific recommendations. Companies that sell many products to many customers and permit these customers to rate their products, like Amazon, are able to collect massive datasets that can be used to predict what rating a particular user will give a specific item. Items for which a high rating is predicted for a given user are then recommended to that user when they use the system again.

1.1 Data Set

The GroupLens research lab (https://grouplens.org) generated a database with over 20 million ratings for over 27,000 movies by more than 138,000 users. This database is referred to as the MovieLens dataset.

We use a subset of this data containing only 10 million (10M) records to analyse and generate our report.

1.2 Goal

The goal of this project is to create a movie recommendation system by developing a machine learning algorithm that predicts movie ratings.

The methodology applied to evaluate the accuracy of the algorithm is the Residual Mean Square Error, or RMSE. We define $y_{u,i}$ as the rating for movie i by user u and denote our prediction with $\hat{y}_{u,i}$. We can interpret the RMSE similarly to a standard deviation: it is the typical error we make when predicting a movie rating and can be mathematically expressed as follows:

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

with N being the number of user/movie combinations and the sum occurring over all these combinations.

If this number is larger than 1, it means our typical error is larger than one star, which is not good.

The aim is for the RMSE generated by the algorhitm to be less than 0.87750 when run of a validation dataset derived from the 10M dataset mentioned above. The lower the RMSE, the better the model.

1.3 Key Steps

- Download the 10M dataset from grouplens.org
- Partition 10M dataset into:
- A train set which is 90% of the 10M dataset - named edx
- A test set which is 10% of the 10M dataset - named validation
- Develop 5 models of increasing complexity and calculate the RMSE of each model
- Select the model with the lowest RMSE from the 5 developed models and recommend that model as the preferred model to predict future movie ratings.

2 Analysis

2.1 Data Cleaning

To make sure we do not include users and movies in the validation set that do not appear in the edx set, we remove these entries from the validation set.

In addition we add rows removed from validation set back into edx set.

2.2 Exploratory Data Analysis

The edx data set contains `9000055` rows and `6` columns.

The names of the columns are:

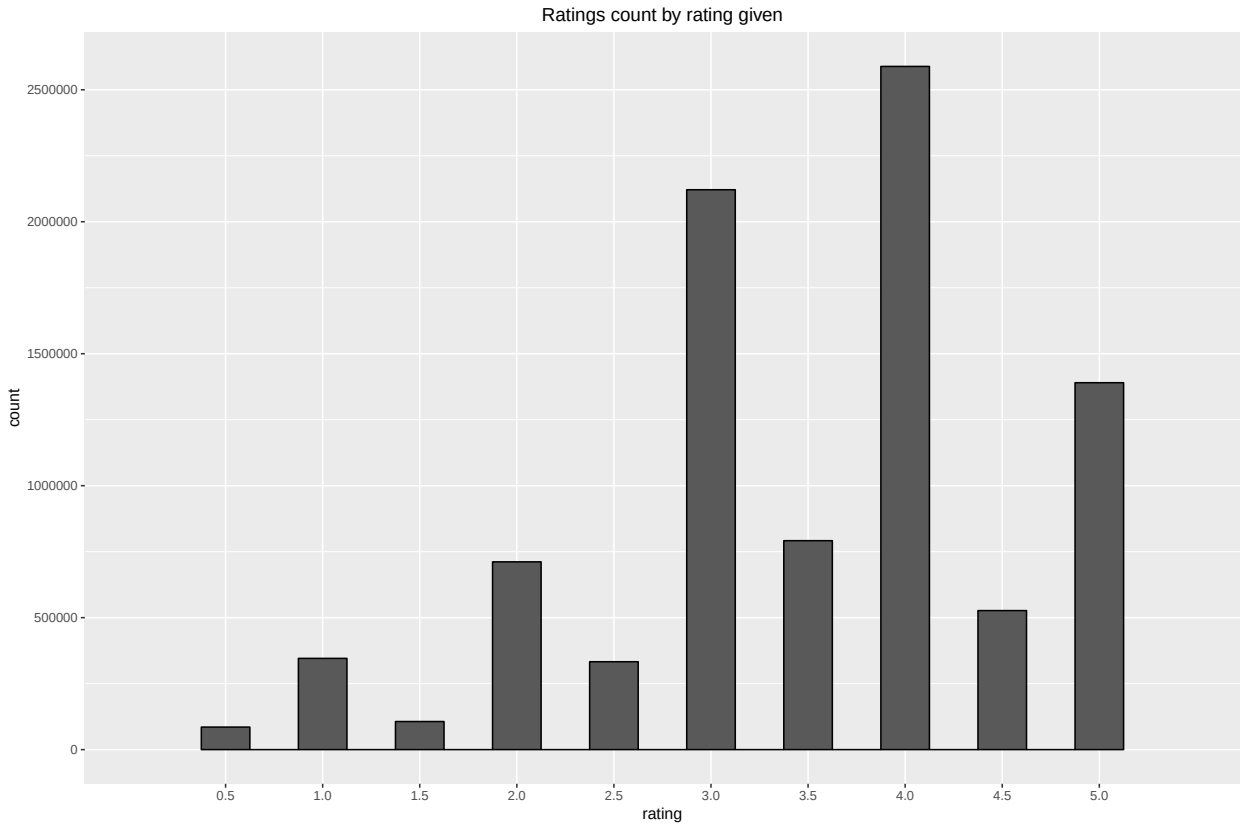
```
## [1] "userId"    "movieId"   "rating"    "timestamp" "title"     "genres"
```

We determined there are `0` missing values in the edx dataset.

The ratings given are:

Rating Given	Count
0.5	85374
1.0	345679
1.5	106426
2.0	711422
2.5	333010
3.0	2121240
3.5	791624
4.0	2588430
4.5	526736
5.0	1390114

Visually we can get a better understanding of the rating distribution

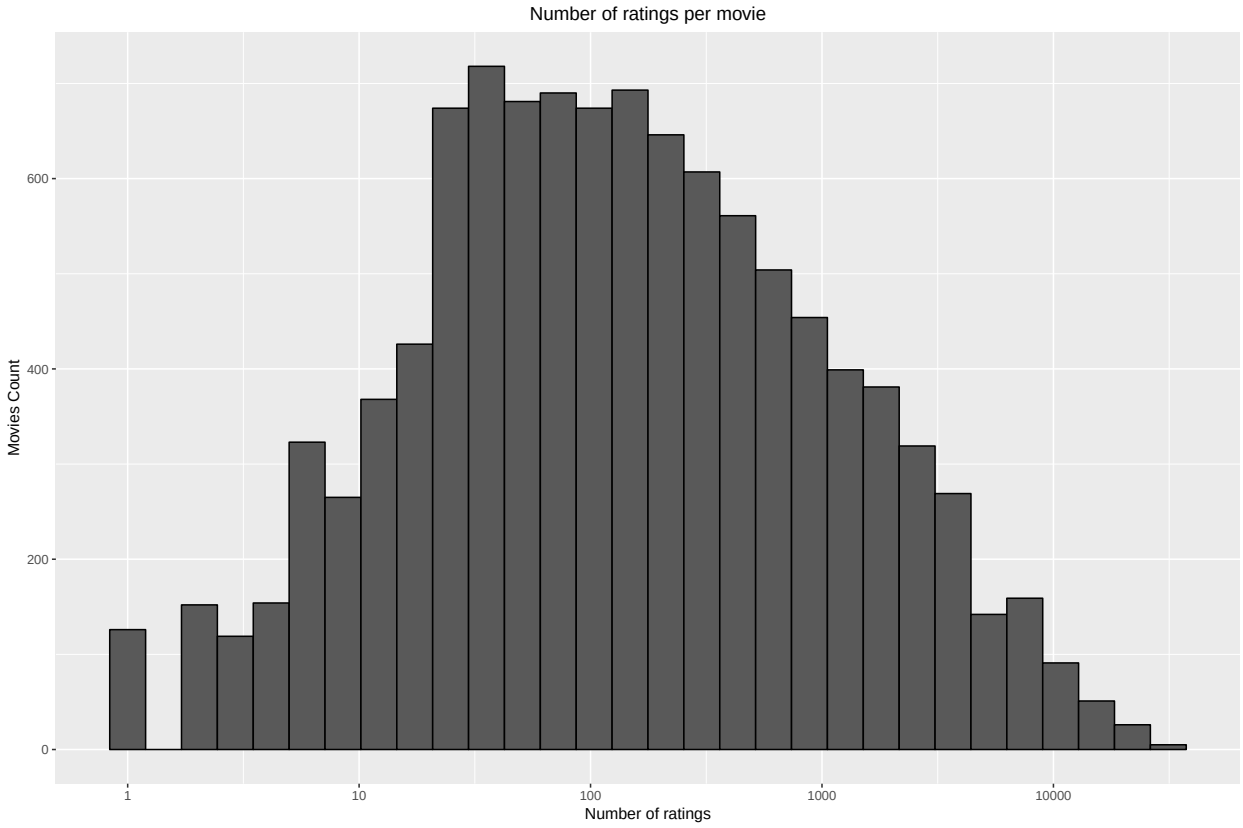


This visualisation indicates:

- movies are rated from 0.5 to 5.0 in 0.5 increments
- there are no 0 ratings
- 4 is the most popular rating
- 0.5 is the least popular rating
- In general, half star ratings are less common than whole star ratings

The edx dataset contains `10677` unique movies and `69878` unique users.

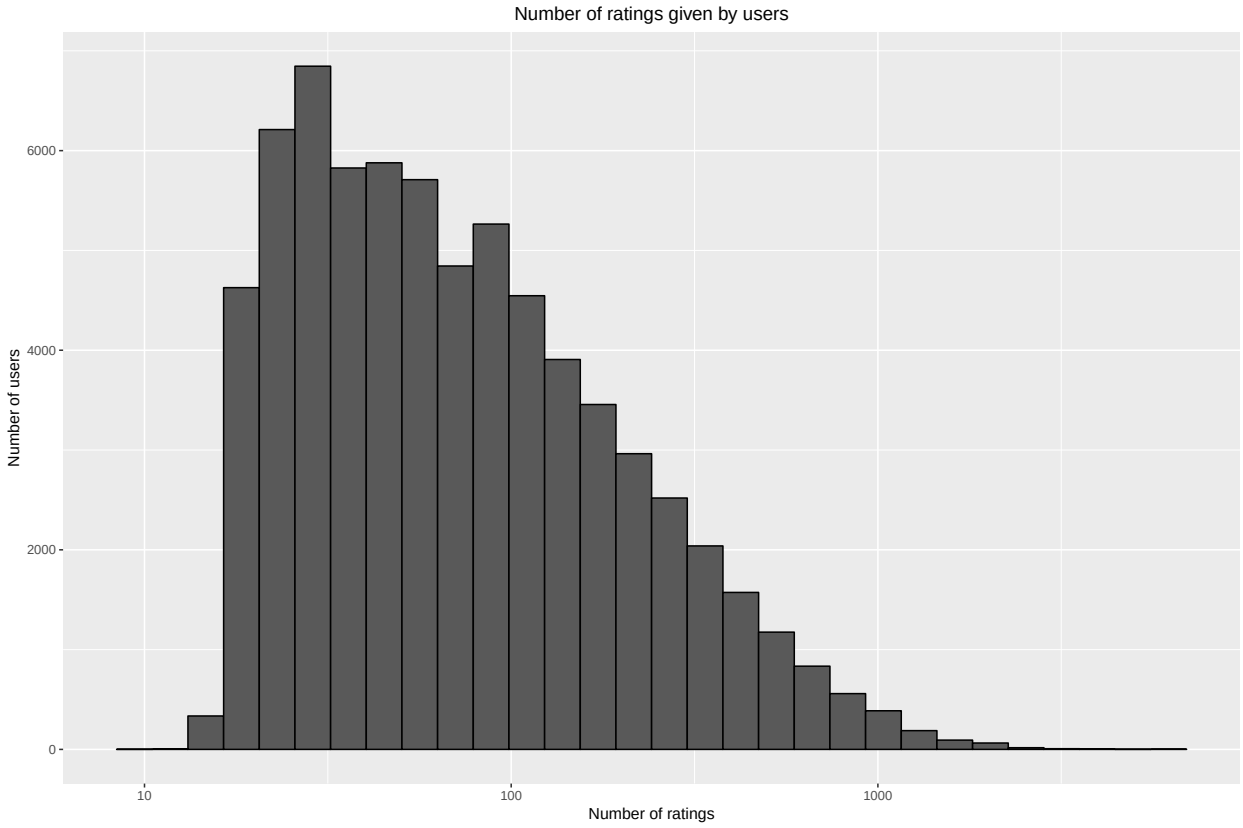
Some movies get rated more than others



Clearly as the below table indicates certain genres receive more ratings than others

Title	Ratings Count
Pulp Fiction (1994)	31362
Forrest Gump (1994)	31079
Silence of the Lambs, The (1991)	30382
Jurassic Park (1993)	29360
Shawshank Redemption, The (1994)	28015
Braveheart (1995)	26212
Fugitive, The (1993)	25998
Terminator 2: Judgment Day (1991)	25984
Star Wars: Episode IV - A New Hope (a.k.a. Star Wars) (1977)	25672
Apollo 13 (1995)	24284

Some users are more active than others at rating movies but more than 6,000 users have given between 40 and 50 ratings



2.3 Modelling Approach

We define 5 models of increasing complexity and use an R user defined funtion to calculate the value of the RMSE.

2.3.1 Naive or base model with the same rating for all movies and users

We start with the simplest possible recommendation system: we predict the same rating for all movies regardless of user. This prediction can be found using a model based approach. A model that assumes the same rating for all movies and users with all the differences explained by random variation would look like this:

$$Y_{u,i} = \mu + \epsilon_{u,i}$$

with $\epsilon_{u,i}$ independent errors sampled from the same distribution centered at 0 and μ the "true" rating for all movies. We know that the estimate that minimises the RMSE is the least squares estimate of μ and, in this case, is the average of all ratings

If we predict all unknown ratings with μ we obtain the following RMSE:

```
## [1] 1.061202
```

2.3.2 A model that takes into account an average rating by movie or a movie effect

We know from experience that some movies are just generally rated higher than others. This intuition, that different movies are rated differently, is confirmed by data. We can augment our previous model by adding the term b_i to represent average ranking for movie i:

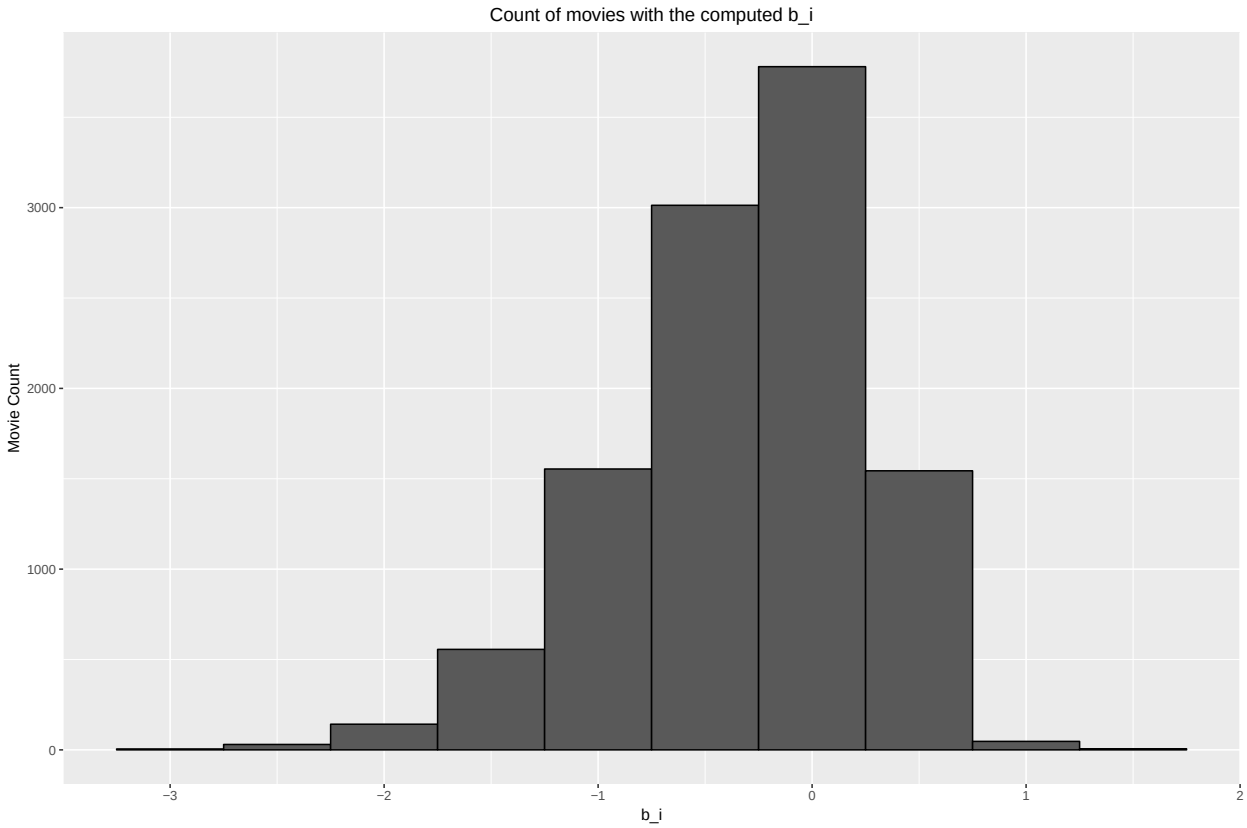
$$Y_{u,i} = \mu + b_i + \epsilon_{u,i}$$

The b's are sometimes referred to as effects.

We estimate this effect by computing μ and estimating b_i, as the average of

$$Y_{u,i} - \mu$$

We can see that these estimates vary substantially:

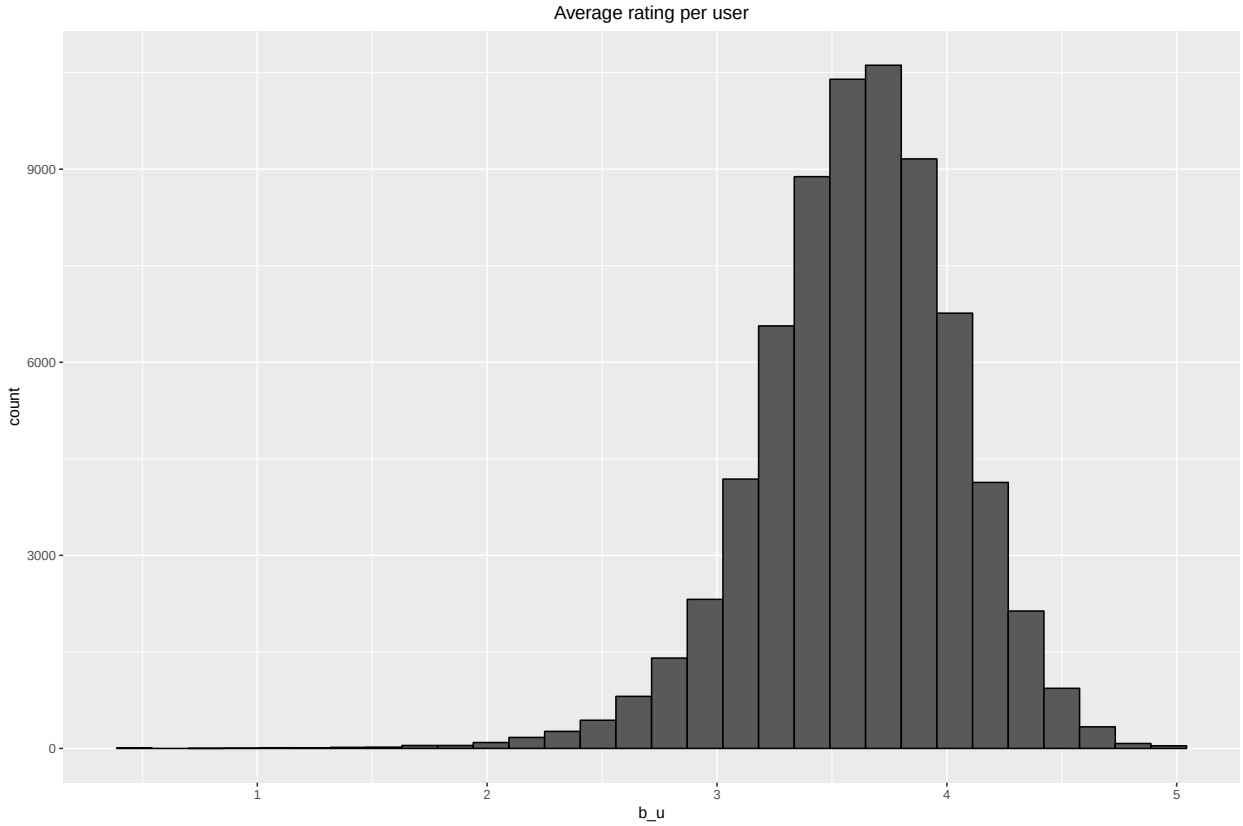


The RMSE with this model is:

```
## [1] 0.9439087
```

2.3.3 A model that takes into account an average rating by user as or a user effect as well as the movie effect mentioned in 2.3.2 above

Let's compute the average rating for user u for those that have rated over 100 movies:



This indicates there is substantial variability across users as well: some users are very cranky and others love every movie. This implies that a further improvement to our model may be:

$$Y_{u,i} = \mu + b_i + b_u + \epsilon_{u,i}$$

where b_u is a user-specific effect. Now if a cranky user (negative b_u) rates a great movie (positive b_i), the effects counter each other and we may be able to correctly predict that this user gave this great movie a 3 rather than a 5.

If we apply this model the resultant RMSE comes to:

```
## [1] 0.8653488
```

2.3.4 A model that adds regularisation to the movie effect model mentioned in 2.3.3 above

Despite the large movie to movie variation, with the movie effect model our improvement in RMSE was only about `11.73`.

This is because there are obscure movies - not often rated, but with with large predictions. With just a few users, we have more uncertainty.

These are noisy estimates that we should not trust, especially when it comes to prediction. Large errors can increase our RMSE, so we would rather be conservative when unsure.

The general idea behind regularisation is to constrain the total variability of the effect sizes. Instead of minimising the least square equation, we minimise an equation that adds a penalty λ:

$$\frac{1}{N}\sum_{u,i}(y_{u,i}-\mu-b_i)^2+\lambda\sum_i b_i^2$$

The values of b_i that minimise this equation are:

$$b_i(\lambda)=\frac{1}{\lambda+n_i}\sum_{u=1}^{n_i}(Y_{u,i}-\mu)$$

where $n_i$is the number of ratings made for movie i

Let's re-run the model with the movie effect using these regularised estimates of b_i using $\lambda = 3$ and compute the resultant RMSE:

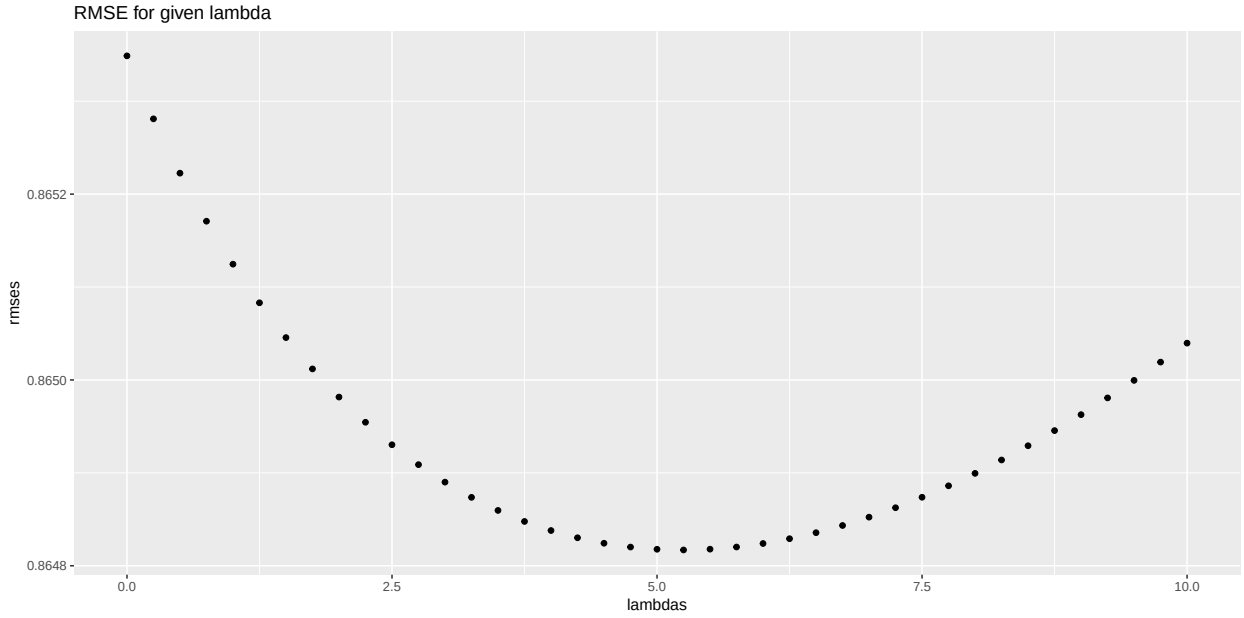
```
## [1] 0.9438538
```

2.3.5 A model that adds regularisation to the user and movie effect model mentioned in 2.3.4 above

We can use regularisation for the estimate user effects as well. We are minimising:

$$\frac{1}{N}\sum_{u,i}(y_{u,i}-\mu-b_i-b_u)^2+\lambda\left(\sum_i b_i^2+\sum_i b_u^2\right)$$

λ is a tuning parameter. We can use cross-validation to choose it.



From the above it is apparent the optimal λ to use is `5.25`

Which results in a RMSE of:

```
## [1] 0.864817
```

3 Findings / Results

The table below summarises the RMSE's of the 5 models we developed:

Model	RMSE
1. Base	1.0612018
2. Movie Effect Model	0.9439087
3. Movie and user effect Model	0.8653488
4. Regularised movie effect Model	0.9438538
5. Regularised movie and user effect Model	0.8648170

4 Conclusion

It is clear that the model taking into consideration both the movie effect and the user effect outperforms the base model as well as the model considering only the movie effect.

The regularised model taking into consideration both the movie effect and the user effect delivered the lowest RMSE and would therefore be our choice to deploy in predicting future unknown movie ratings.