MOVIE RECOMMENDATION SYSTEM

Richard Lawton

First created on 2019-12-12. Updated on 2020-01-04 Submitted in fulfilment of the requirements of HarvardX: PH125.9x Data Science: Capstone course, MovieLens assignment.

INTRODUCTION

The MovieLens 10M dataset (https://grouplens.org/datasets/movielens/10m/) provides ratings of 10677 movies by 69878 users. Any given user has rated only a small subset of the movies, and the objective of this assignment is to predict the scores for movies a user has not yet rated, as a recommendation system.

Each record of the dataset provides a rating for the movie (0 - 5 in increments of 0.5), a composite genre for the movie, a timestamp for the rating and the title of the movie, as well as a movieId and userId as keys.

The dataset is provided pre-partitioned into a training component (edx - 90%) and a validation component (validation - 10%). The predictive model is to be developed on the training component, then tested against the validation component, with the objective of minimising the difference between the predictions and the actual ratings recorded, measured as a root mean square error (RMSE) over the full validation set.

Three fairly self-evident observations provide the structure for the model. First, some movies are just better than others, and will therefore be predicted to earn a higher score. Second, some users rate more generously than others, and will therefore also be predicted to give a higher score. Third, users will have a preference for certain kinds of movies and will therefore score movies in some genres higher than others.

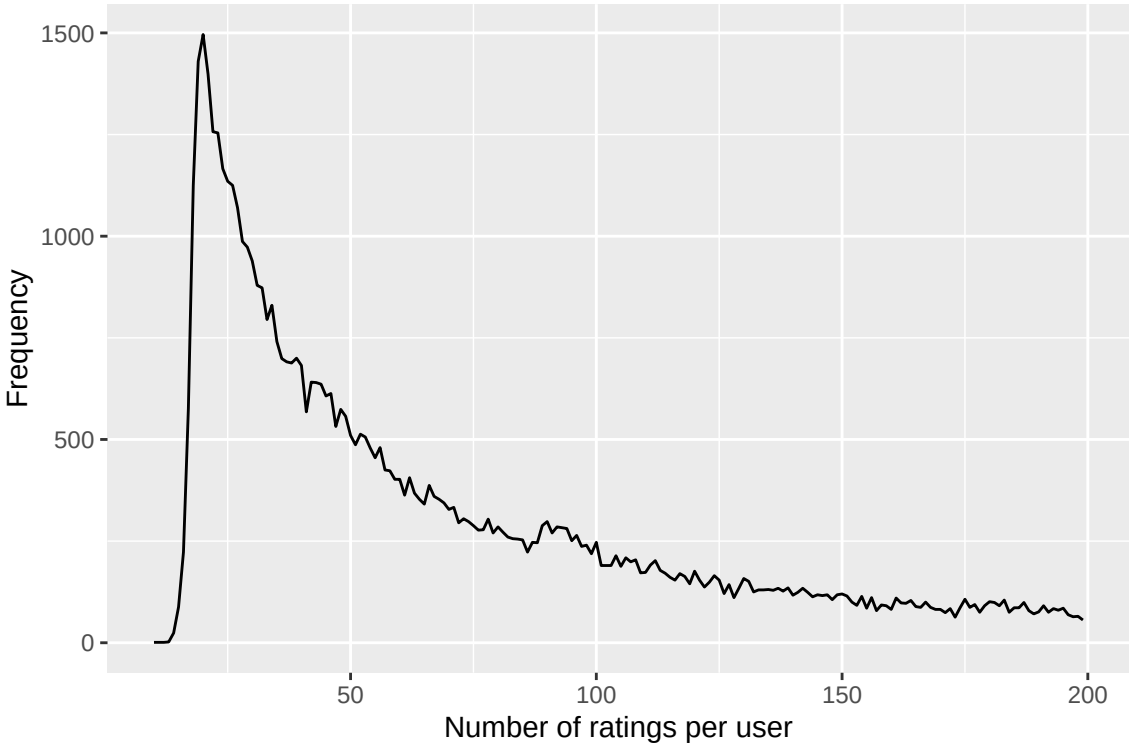
These three components were built into a predictive model and optimised using internal cross-validation on the training set. When tested on the validation set an RMSE of 0.8555 was achieved.

METHODOLOGY

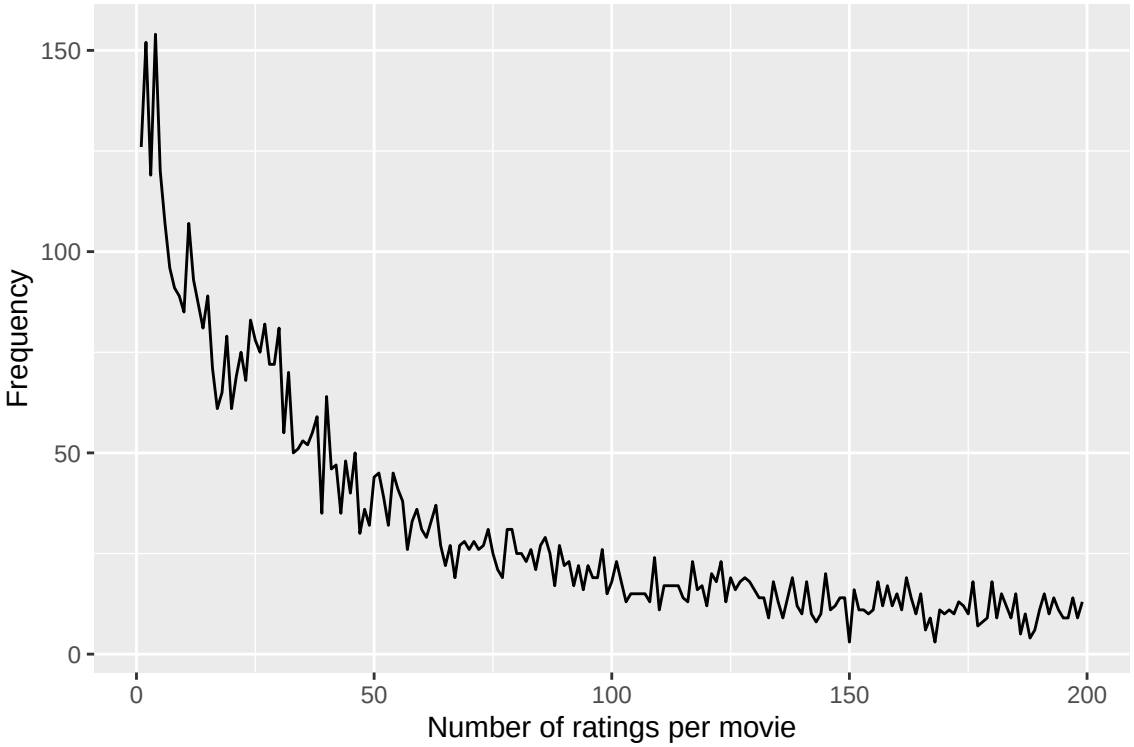
Data preparation

The data was provided pre-prepared in tidy format, and no further cleaning or organisation was required. Salient features of the data at this point:

Some users have been extremely busy - the maximum number of ratings per user is a staggering 6616. At the other end of the scale there is a cut off eliminating users with less than 10 ratings. The most common number of ratings per user is 20:



The most rated movie – *Pulp Fiction (1994)* – has 31362 ratings. At the other end of the scale *Quarry, The (1998)* has only 1. The most common number of ratings is 4:



The data is presented in tidy format, but we could also consider it in matrix form with

user in one dimension and movie in the other. It would however be an extremely sparse matrix: sparsity = 98.8%.

Memory management

The dataset is large (10 million rows) and memory management was critical, especially since the model was developed on a laptop with only 4GB memory. Two important approaches emerged in managing memory: (1) removing unnecessary duplicates of the data set as the model is developed. (2) avoiding use of anonymous functions in the mapping loops *sapply(v, fun)* which for some reason seem to lead to escalating memory leaks.

The model

Following the approach used by the winners of the Netflix Prize (http://blog.echen.me/2011/10/24/winning-the-netflix-prize-a-summary/) we break up the predicted score into 4 components, with a residual noise factor:

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

where $Y_{i,u}$ is the predicted score for movie i by user u, μ is the global mean of ratings for all movies by all users, b_i is the *bias* associated with movie i (is the movie generally held to be good or bad), b_u is the *bias* associated with user u (does the user generally score high or low), and $b_{g,u}$ is the *bias* associated with user u for genre g (does the user generally like movies of this genre). $\epsilon_{i,u}$ is the residual unexplained noise factor that we wish to minimize. The three biases b_i, b_u and $b_{g,u}$ were modelled sequentially.

1.Baseline model

The global mean of movie ratings was estimated as the mean of the *edx* dataframe ratings as 3.5125. Our baseline model will predict this as the rating for any movie by any user.

2. The movie effect

The bias for or against individual movies was estimated as the mean of the residuals for each movie.

$$b_i = \frac{1}{n_i} \sum_u r_{i,u} - \mu$$

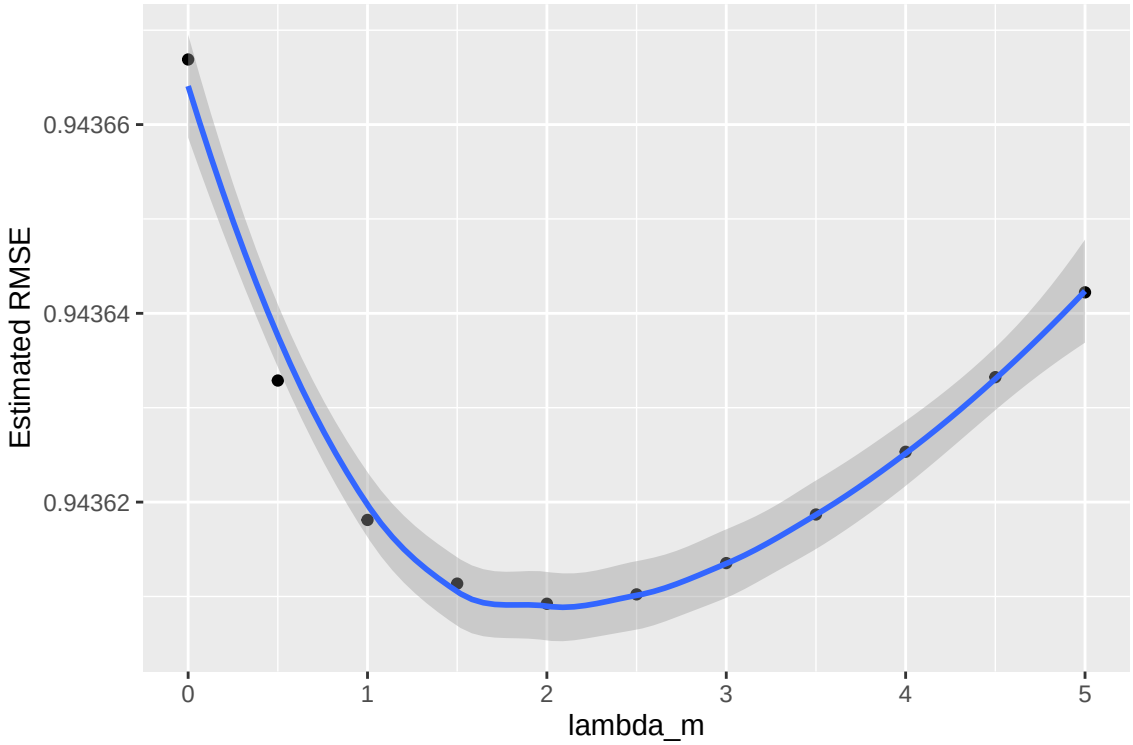
where n_i is the number of ratings of movie i, $r_{i,u}$ is the rating of movie i by user u and μ is the global mean of ratings. The sum is taken over all users that rated movie i.

In practice this estimate gives undue weight to outlier ratings, so a better model is to weight the mean by a *regularising* parameter λ_m as follows:

$$b_i = \frac{1}{n_i + \lambda_m} \sum_u r_{i,u} - \mu$$

λ_m will reduce the biases (positive or negative) of movies that have received few ratings.

In order to select the optimal value for λ_m, the *edx* dataset was further subdivided into a training set (90%) and a testing set (10%). The set of b_i's for a range of the parameter λ_m was calculated. Preliminary investigation suggested the optimal value of λ_m lay in the range 0 - 5. The results were calculated as follows:



On the basis of this plot, λ_m was set at 2. For the movie effect the benefit of regularising with the parameter λ_m is fairly small. The estimated RMSE (from our test hold-out set) was 0.94367 with no regularisation ($\lambda_m = 0$). This dropped to 0.94361 when λ_m was set to 2. The corresponding vector b_i keyed on movieId for $\lambda_m = 2$ was saved as "movie_effect" as part of our final model, and a column "b_i" was added to the *edx* dataframe for further development of the model.

3. The user effect

The general bias exhibited by individual users was estimated as the mean of the new residuals for each movie.

$$b_u = \frac{1}{n_u} \sum_i r_{i,u} - \mu - b_i$$

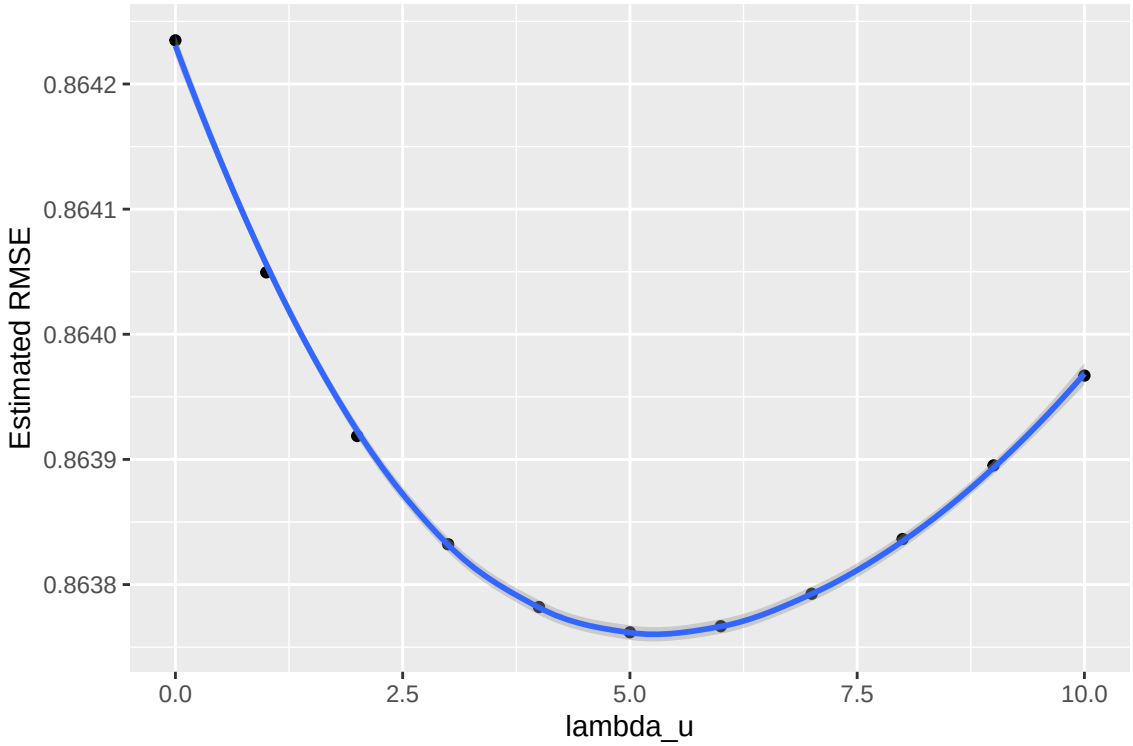
where n_u is the number of ratings by user u, $r_{i,u}$ is the rating of movie i by user u, μ is the global mean of all ratings and b_i is the bias previously calculated for movie i. The sum is taken over all movies rated by user u.

Following the same procedure we used in calculating b_i, we proceeded to weight the mean by a regularising parameter λ_u as follows:

$$b_u = \frac{1}{n_u + \lambda_u} \sum_i r_{i,u} - \mu - b_i$$

λ_u will reduce the bias given to users that have rated few movies.

In order to select the optimal value for λ_u, the same subdivision of the *edx* dataset was used. The set of b_u's for a range of the parameter λ_u was calculated and used to estimate the RMSE using the testing set. Preliminary investigation suggested the optimal value of λ_u lay in the range 0 - 10. The results were calculated as follows:



On the basis of this plot, λ_u was set at 5. For the user effect the benefit of regularising with the parameter λ_u is more evident. The estimated RMSE (from our test hold-out set) was 0.86423 with no regularisation ($\lambda_u = 0$). This dropped to 0.86376 when λ_u was set to 5. The corresponding vector b_u keyed on userId for $\lambda_u = 5$ was saved as "user_effect" as part of the final model, and a column "b_u" added to the *edx* dataframe for further development of the model.

4. The genre effect

Each movie in the MovieLens 10M dataset is pre-classified according to genres selected from the following list:

Comedy, Romance, Action, Crime, Thriller, Drama, Sci-Fi, Adventure, Children, Fantasy, War, Animation, Musical, Western, Mystery, Film-Noir, Horror, Documentary, IMAX.

A given movie is associated typically with a combination of several of these genres eg *Adventure | Children | Comedy.* A total of 797 different genre combinations are represented in the dataset.

The task at hand is to estimate the effect on a rating of the users bias towards the genres represented by the movie. One approach would be to estimate a bias $b_{g,u}$ for a user u towards each elemental genre g. The overall genre bias for a movie i rated by user u would then be calculated as the sum of of biases for each genre associated with that movie:

$$b_{i,u} = \sum_g b_{g,u}$$

where $b_{i,u}$ is the bias effect for movie i with user u and $b_{g,u}$ is the bias of user u towards genre g, to be summed over all genres represented by movie i.

The simpler approach that we opted to use takes the genre combinations as the variable in our analysis. It could well be argued that the elemental genres are very broad categories (for example "war" might cover everything from *War and Peace* to *Star Wars*, and "comedy" everything from Woody Allan to Keystone Cops). Perhaps the genre combinations represented in the dataset give a more nuanced handle on user preferences? In our model then, $b_{g,u}$ will reference a genre combination, and be estimated following the procedure used for movie and user effects:

$$b_{g,u} = \frac{1}{n_{g,u}} \sum_i r_{i,u} - \mu - b_i - b_u$$

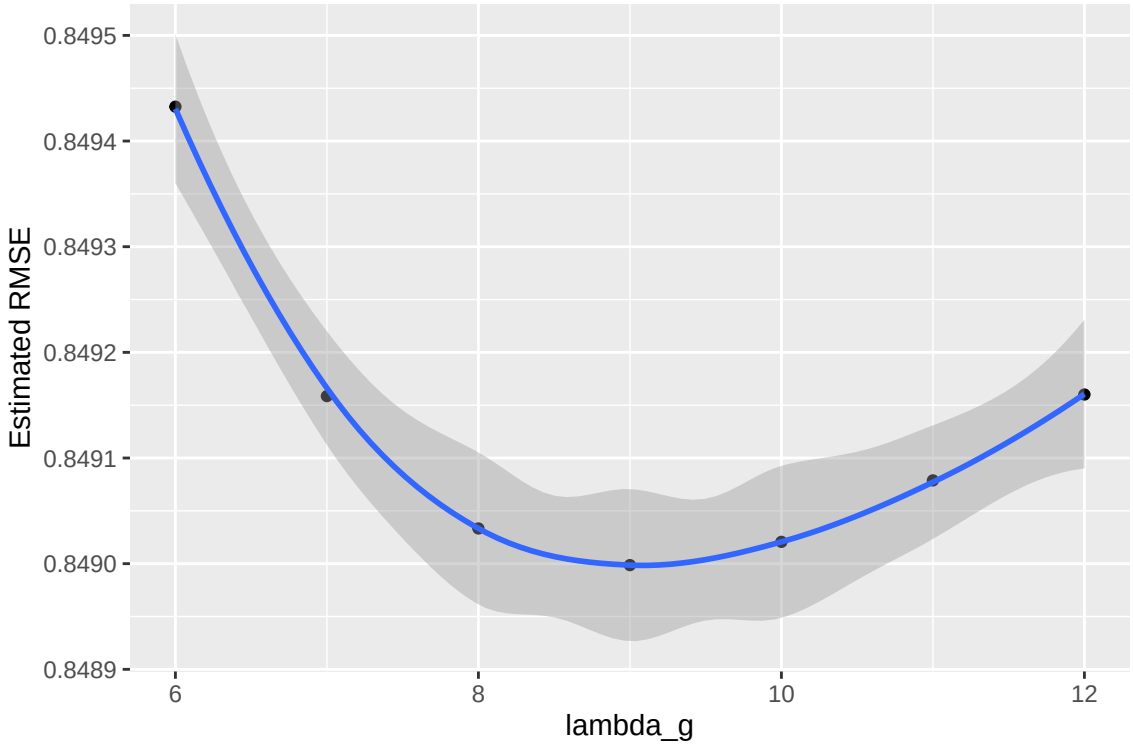
where $n_{g,u}$ is the number of ratings by user u of movies with genre combination g, and $r_{i,u}$ is the rating of movie i by user u, μ is the global mean of all ratings, b_i is the bias previously calculated for movie i and b_u is the bias previously calculated for user u. The sum is taken over all movies with genre combination g that have been rated by user u.

It turns out there are about 4.3 million $b_{g,u}$'s calculated by this process. Of these over 2.8 million have $n_{g,u}$ equal to 1, which means that user u has only rated one movie of genre combination g. We would therefore expect the outlier problem to be fairly significant, and the regularisation process to be particularly important for the genre effect. Following the previous logic, we introduce the parameter λ_g:

$$b_{g,u} = \frac{1}{n_{g,u} + \lambda_g} \sum_i r_{i,u} - \mu - b_i - b_u$$

and optimise again using the test set for cross-validation. Of course a large number of user-genre combinations will not be represented in the *edx* set. Our model has calculated a b_i for every movie in the dataset, and a b_u for every user. However we want to be able to predict the rating for a movie by a given user where this user has not previously rated movies of that genre combination – i.e. $b_{g,u}$ does not exist. For these cases our model must use the value zero, because we have no better information.

Preliminary investigation suggested the optimal value of λ_g lay in the range 6 - 12. The results were calculated as follows:



On the basis of this plot, λ_g was set at 9. The estimated RMSE (from our test hold-out set) was 0.91482 with no regularisation ($\lambda_g = 0$). This dropped significantly to 0.849 when λ_g was set to 9.The sparse matrix $b_{g,u}$ for $\lambda_g = 9$ was saved in long form as "genre_effect", as part of our final model.

At this point we are ready to calculate predictions for ratings on the basis of movie id, user id and genres, and test against the validation set, using the final model:

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

RESULTS

Predictions were generated for the validation set and compared with actual ratings recorded:

Model	RMSE
$Y_{i,u} = \mu$	1.0612
$Y_{i,u} = \mu + b_i$	0.9439
$Y_{i,u} = \mu + b_i + b_u$	0.8649
$Y_{i,u} = \mu + b_i + b_u + b_{g,u}$	0.8555

The final RMSE of 0.8555 implies that our predictions are likely to lie within $\pm$ 1.8 of the actual value. On a possible rating range of 0-5 this is not particularly impressive, on the

face of it.

A couple of factors can explain this residual uncertainty. First the response of a given user to a given movie is about more than *genre*. The principal actors, the style of the dialogue, the pace, the music, even the length of the movie could all affect the rating given. Our dataset does not provide any information on these variables.

A second, more intractable problem is that movie ratings involve a large measure of spontaneity. The residual noise $\epsilon_{i,u}$ will therefore include a 'halo effect': how do extraneous circumstances affect a rater's score? How reproducible is an individual rating? (How differently would a user rate the same movie on a different day?) It is possible that an individual rating would have a reproducibility not much better than $\pm$ 1.

The Netflix prize was awarded for an RMSE score of 0.87 (albeit on a different dataset). Our model then does lie somewhere close to the limits of what is possible.

CONCLUSION

A movie recommendation model has been developed from the MovieLens 10M dataset. The model consists of indexed vectors of movie biases and user biases and a sparse matrix of user-genre biases. This model was applied to a validation set of a million observations and gave predictions with an overall RMSE of 0.8555.

The project was limited in development of user-genre interaction, and there is ample scope for further investigation through matrix factorisation, and cluster analysis, for example. In addition no use was made of the rating timestamp. Future work could investigate whether ratings follow a time-trend in years following the release of the movie, enabling us to add a time bias to the model.