MovieLens Project

Ramin Ahmed

10/15/2019

Intoduction

In this project, we create a movie recommendation system using the movielens data obtained from the link provided in the project description. After doing some data wrangling using the provided code, we see that the data contains 10000054 observations with 6 features namely: userId, movieId, rating, timestamp, title and genre. Here, almost every movie is rated by multiple users and almost every user rates multiple movies. We have to predict the rating of each movie and the movies for which a high rating is predicted for a user is recommended to that user. For this, we need to create training sets (as edx) and test sets (as validation) which we obtained from the provided code.Here the validation set would contain 10% data. The idea is to train our machine learning model on the edx set and test the efficiency of our model on the validation set. We use a loss function (RMSE) to justify our model.We started with the naive approach. Since we know from experience and also confirmed through exploratory analysis that different movies are rated differently and different users rate movies differently, we decided to include the movie and user effects to our model. Further data exploration depicted strong evidence of genre effect and hence, we included it as well. While user effect and movie effect contributed significantly to the reduction of RMSE from naive approach, genre effect led to small improvements to the RMSE despite having large variations among them. Further investigation showed that these resulted from few users reviewing some movies and few movies are catagorized in some genres. This resulted in more variability and hence, we decided use regualrization on all the effects of the model which incorporates constraints to total variability of the effect sizes. Finally, this model improved the RMSE and provided a magnitude within our desired limit (RMSE <= 0.8649).

Analysis

Naive Approach

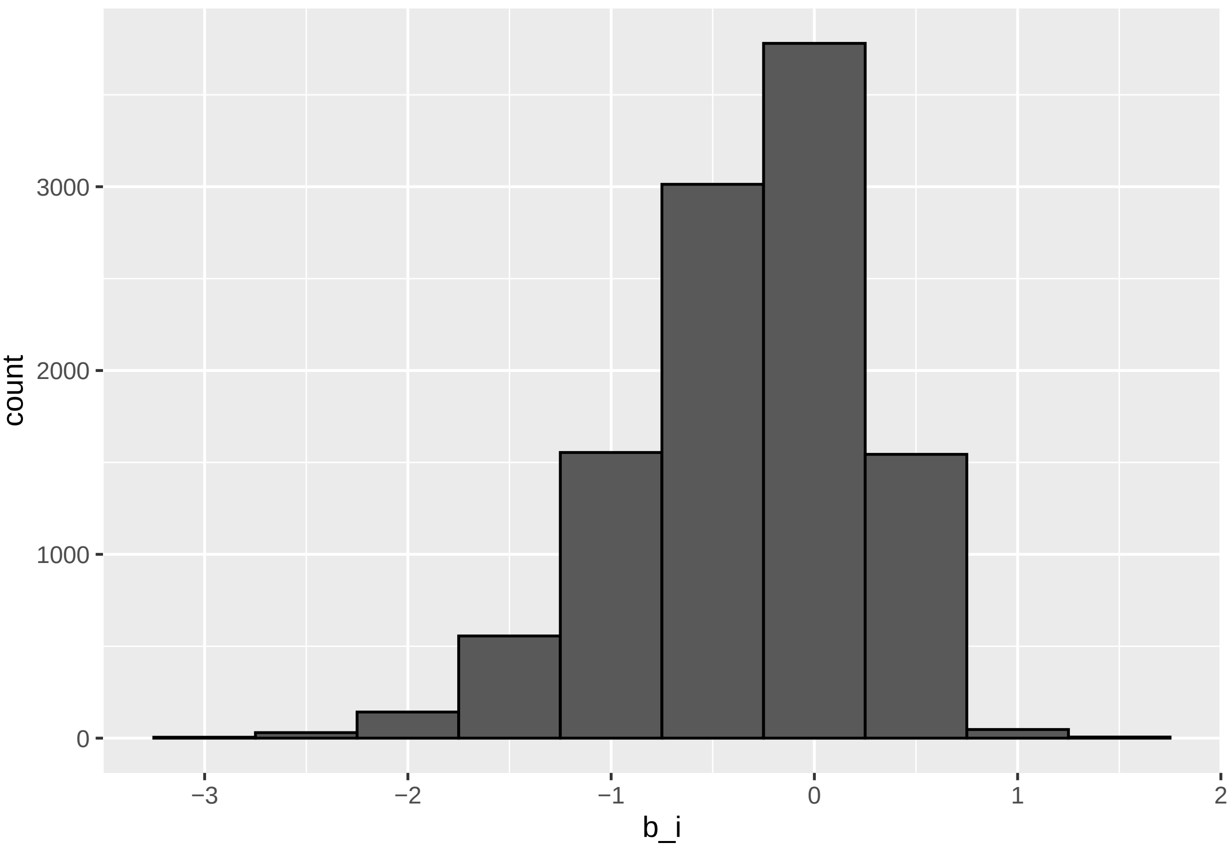
We start with a naive approach where we assume the same rating for all movies and users with all the differences explained by random variation. The model looks like this:

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

where $\varepsilon_{u,i}$ are independent errors of same distribution with mean 0 and μ is the "true" rating for all movies. This model provided an RMSE of 1.0612018.

With Movie effects

We know that different movies are rated differently and the following plot depicts the fact.



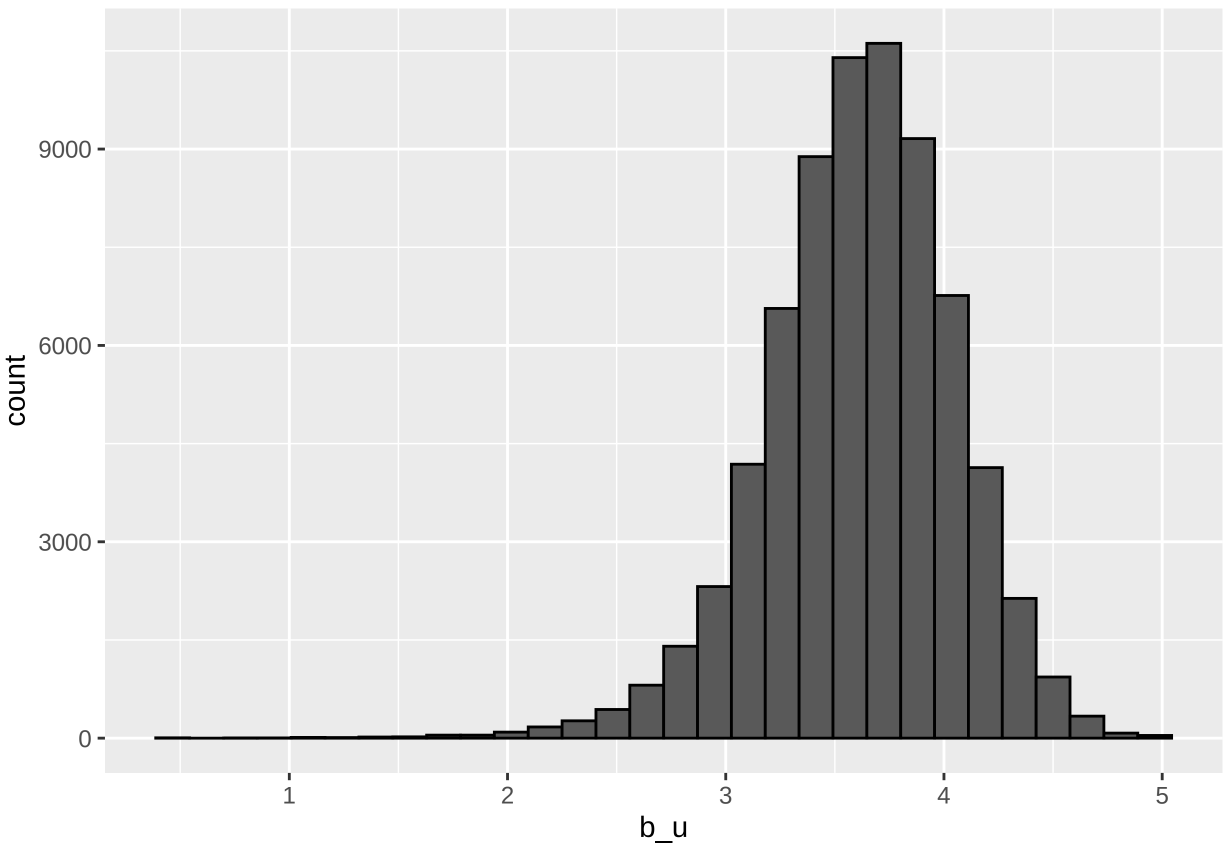
Hence, we modify our previous model by adding the term b_i to represent average ranking for movie i. The model looks like this:

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

It is important to note that due to the large size of data, lm() function will be very slow here and thereby, we do not use it in this project. Instead, we use the least sqaure estimate of each newly added components which, in this case, for b_i is the average of $Y_{u,i} - \mu$. This model provided an RMSE of 0.9439087.

With User effects

We know from experience that there are significant variablities among the ratings of different user and the following plot proves this fact.



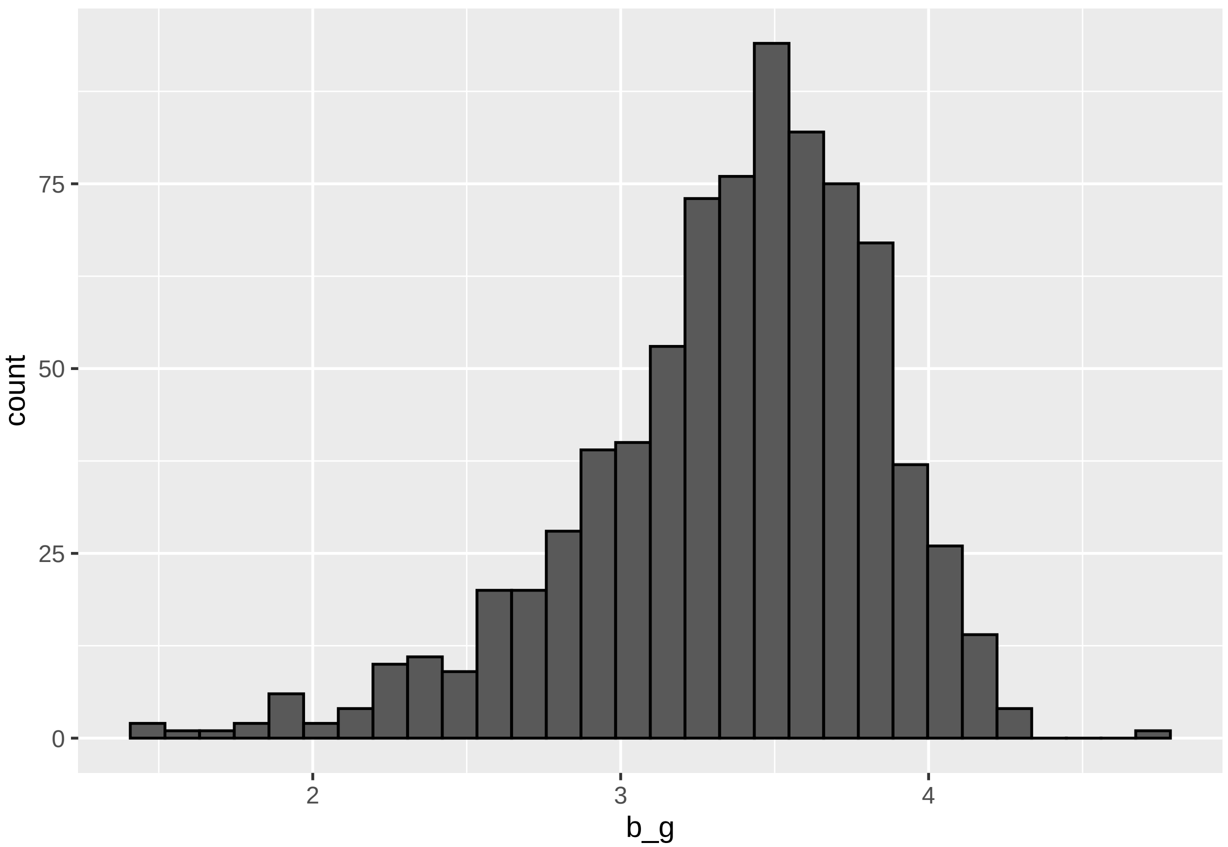
Hence, we modify our previous model by adding the term b_u to accomodate user-specific effect. The model looks like this:

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

Here, the least square estimate of b_u is the average of $Y_{u,i} - \mu - b_i$. This model provided an RMSE of 0.8653488.

With Genre effect

We know, from experience, that the move ratings vary with genre and the follwing graph depicts a strong evidence of genre effect.



Hence, we modify our previous model by adding the term b_g to accomodate genre-specific effect. The model looks like this:

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

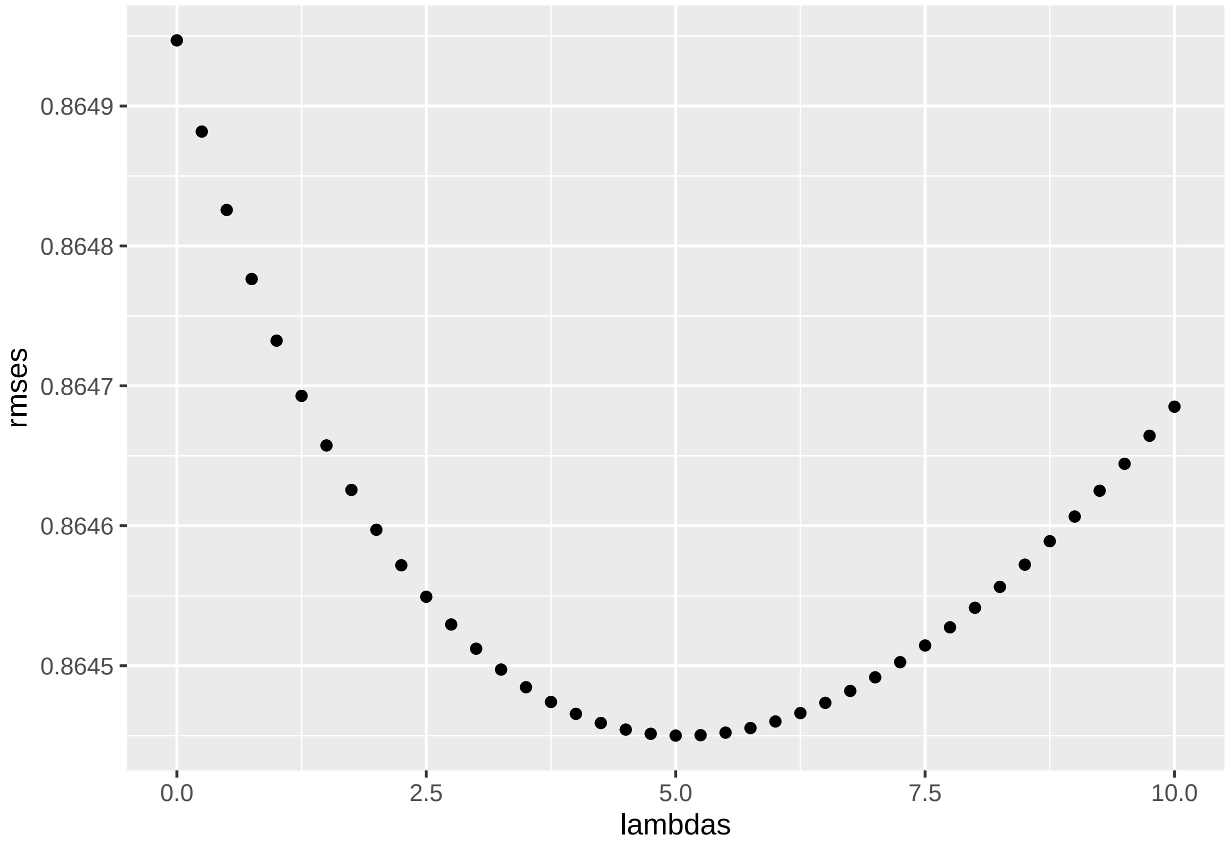
Here, the least square estimate of b_g is the average of $Y_{u,i} - \mu - b_i - b_u$. This model provided an RMSE of 0.8649469.

Regularization

Despite having large variations among the genres, the improvement to RMSE was insignificant which resulted from very few movies being catagorized under some genres, some users reviewing few movies and some movies having few reviews. This added more variability to your model. Hence, we use regularization which contraints the total variability of effect size. For this we use penalized least square which adds penalty on top of the regualr least squares when many estimates are large. For example, for the regularization of movie effect alone, we minimize

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

Performing further analysis, we plot a varying range of $\lambda's$ against their resulting RMSEs.



From this graph (and from analysis), we can see the the optimal value of λ is 5. Using this λ, we construct the regularlized movie, user and genre model.

Results

We see an improvement in the RMSE after each effect is added. The final regularized model provides an RMSE of 0.8644501 which is within our desired limit. The following is the summary of the models and their performance when tested against validation sets.

```
## Warning: package 'knitr' was built under R version 3.5.3
```

Table 1: Summary of model performance

method	RMSE
Just the average	1.0612018
Movie Effect Model	0.9439087
Movie + User Effect Model	0.8653488
Movie + User + Genre Effect Model	0.8649469
Regularized Movie + User + Genre Effect Model	0.8644501

Conclusion

In this project, we created a movie recommendation system based on given data. At first we analyzed the observations and features of the data. Through exploratory data analysis, we created a machine learning model which incorporates movie effects, user effects and genre effect. Since very few movies being catagorized under some genres, some users reviewing few movies and some movies having few reviews, the data significant variability in it. Hence, we decided to use the regularized movie, user and genre effect to our model. We think we could have further reduced the RMSE had we seperated each genre and incorporated their interaction when the genre catagory demanded. So, as future work we recommend doing this. This will hopefully help to further reduce the RMSE.