Movielens Capstone Report

Adam Wilson

May 1, 2019

Introduction

For this project, we create a movie recommendation system using the MovieLens dataset, which is a dataset of movie ratings from Netflix. Netflix allows users to rate movies with 0 to 5 stars, and our goal is to predict a movie's rating as accurately as possible. We achieve this with machine learning techniques.

Specifically, we will use the MovieLens 10M Dataset, which can be found here: https://grouplens.org/datasets/movielens/10m/ (https://grouplens.org/datasets/movielens/10m/). It gives us 10 million movie ratings across 10,000 movies by 72,000 users.

Our key steps will be splitting our data into two groups, so that we can work on one group and then test our algorithms on another. We will perform our machine learning algorithms on models that we will make gradually more complex, but hopefully more accurate. We base our accuracy upon RMSE, which is further discussed in the next section.

Methodology

We take our MovieLens 10M Dataset and split it into two groups: the training set, **edx**, which consists of 90% of our dataset, and the test set, **validation**, which consists of the remaining 10%. We will perform machine learning algorithms on our training set and then test its accuracy on our test set.

We will judge our accuracy based on *Residual Mean Squared Error* (*RMSE*), which compares the difference between a predicted result and the actual result. We start off with a simple model and gradually make it more complex until we are satisfied with our RMSE.

Method 1: One Value Only

Our first method is to assume that all future estimates will just be the current average rating.

```
# Average Movie Rating

mu_hat
```

```
## [1] 3.512465
```

method	RMSE
Average Rating Only	1.061202

Method 2: Movie Effect

Next we will assume that there is a difference across movies, i.e. some movies just tend to be rated higher or lower than others. What we do is observe the ratings of each movie in our training set and take its difference from the mean rating. We then compare it to the ratings observed in our test set, and use these to calculate our next RMSE.

method	RMSE
Average Rating Only	1.0612018
Movie Differences	0.9439087

Method 3: Movie & User Effect

Next let’s assume there is a difference across users too, i.e. some viewers just to tend to rate their watched movies higher or lower than others. We add this observation to the previous method, so that we account for movie effect and user effect at the same time.

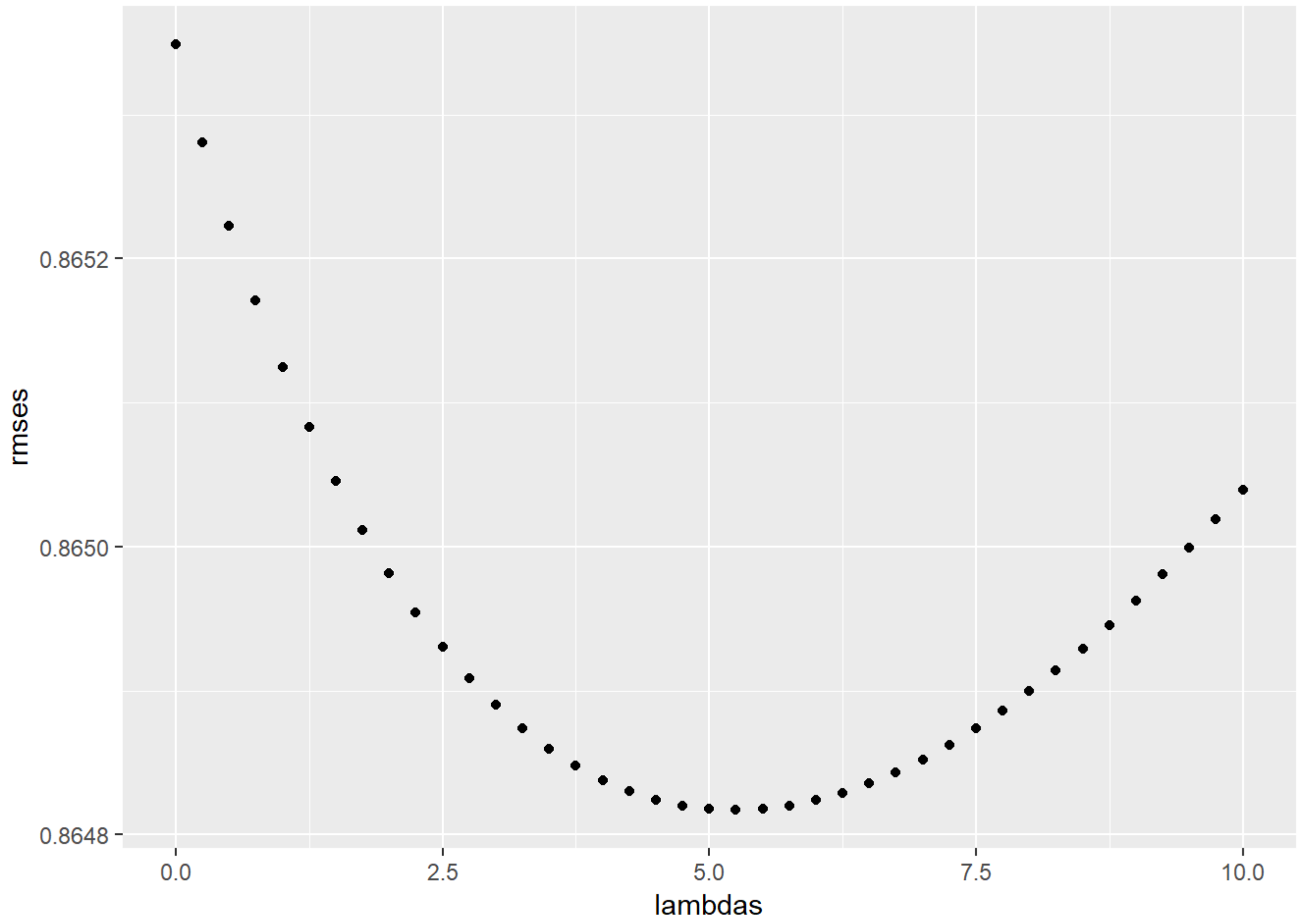
method	RMSE
Average Rating Only	1.0612018
Movie Differences	0.9439087
Movie and User Differences	0.8653488

This is a good result, but we’ll see if we can make it better with one further step, using regularization.

Method 4: Regularization of Method 3

It is possible that some movies have been ‘unfairly’ rated very high or very low, as very few people have rated them, compared to other movies which may have thousands of ratings. Regularization can account for this. We set a tuning parameter, **Lambda**, to account for scenarios in which the sample size of ratings is very small. If the sample size is very large, the penalty Lambda is effectively ignored, but if the sample size is very small, the *movie effect* is shrunken towards 0.

First we calculate which Lambda gives us the best RMSE.



```
# Best Lambda

lambdas[which.min(rmses)]
```

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

And finally, we add the best RMSE we found here to our dataframe.

method	RMSE
Average Rating Only	1.0612018
Movie Differences	0.9439087
Movie and User Differences	0.8653488
Regularized Movie & User Effect	0.8648170

Results

We calculated an average star rating of **3.512**, which we used on our first simple model, as well as to calculate differences on all following models. We have shown multiple dataframes with gradually increasing rows that show how our RMSE improves. Our final resulting RMSE was **0.865**. We also calculated a Lambda of **5.25** to achieve the minimum RMSE for our model that used penalizing least squares.

Conclusion

As we can see, using regularization changed our RMSE only slightly. This likely means that there were not many ratings that were ‘outliers’. In other words, there were not many movies with very few ratings, or if there were, they were not rated very high or low. However, I believe that it would still be good practice to perform regularization, even if the resulting RMSE change is minimal.

Our final RMSE of 0.865 means that on average, our recommendation system’s accuracy is only 0.865 stars away from the true rating in our test set. I believe this is to be of satisfactory accuracy.