MovieLens Project

Daniel Albert

3/6/2019

Introduction

The goal of this project was to create a program capable of taking data of movie reviews and making a prediction for what users would rate other movies. The idea is that this could be used to recommend movies to users that they might like. The datasets provided include the userId, the movieId, the rating given, the timestamp, the title of the movie, and the genres that the movie falls into.

The edx dataset includes over 9 million examples of ratings that users gave movies. We also have the validation dataset, which was used as the test set, comprised of 10% of the total data.

The main process I underwent was as follows: 1. I created the data sets following the directions in the course materials. 2. I followed the template laid out in the Machine Learning class to set up a basic recommendation system involving the average rating, the movie effect, and the user effect. 3. I ran this code on the validation set and began summarizing the information to analyze the strengths and weaknesses of the algorithm I created.

My intent was to go back and revise my algorithm after this, but I realized that I used the validation set and going back after having done so would be improper, so I decided that I needed to use what I had. Fortunately, my simple algorithm was effective and resulted in an RMSE value below the cutoff for full credit.

Methods/Analysis

Upon running the given code to set up the edx and validation data sets, I explored the edx set in order to get a sense of the data. It was extremely similar to the data sets we had been using in the machine learning class, so this introductory work did not take me very long.

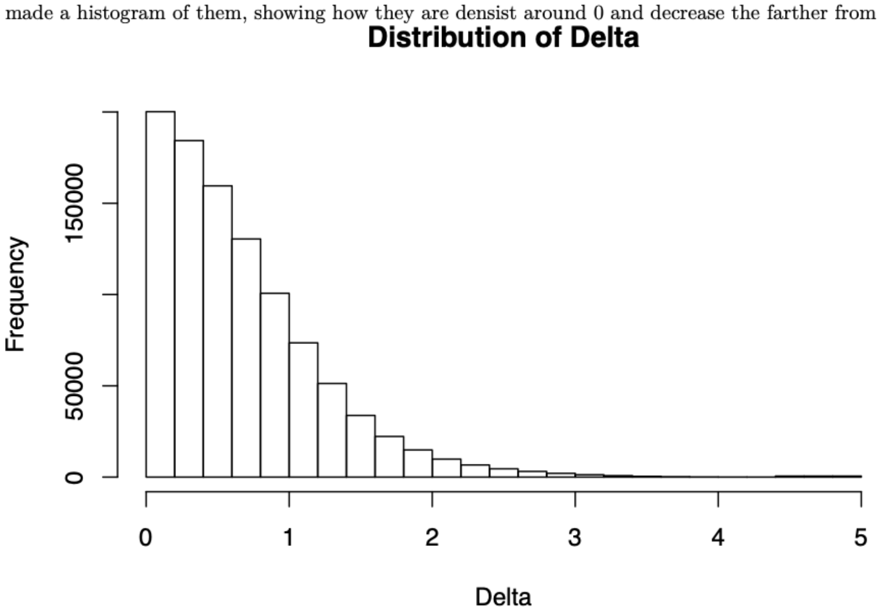
I began by defining the RMSE function that would be used to check the effectiveness of my work. I figured that this was an important step that I should not forget, so I got it done in the beginning. From there, I was able to begin putting together a strategy for developing my algorithm. Again, my experiences in the Machine Learning class formed a strong basis for this portion of my work. I found that the average rating given was around 3.5 and knew that this would be central to my work ahead. My next step was to calculate the movie effect, b_i, which represents how much better or worse individual movies are than average. This was done by grouping the edx data set by movie, subtracting the average rating from each rating, and then averaging them for each movie. This b_i value now gives us an idea of which movies are generally higher rated (with positive b_i) or lower rated (with negative b_i). I next did something very similar with the individual users and calculated a user effect, b_u, to see which users tended to rate movies higher than average or below average. This time, the edx set was grouped by user and then I found the average of the rating minus mu minus the movie effect.

With these three values, mu, b_i, and b_u, I was ready to run my first test.

Results

I used two main methods to determine the effectiveness of my algorithm. The first is the prescribed RMSE calculation. As the program shows, my RMSE value is approximately 0.8653488, which is below the target value of 0.8775. This represents a solid improvement over the basic approach of predicting the average rating for all movies, which yields an rmse of 1.061202.

My second method, which I was able to use to more effect, was to look at the difference between my prediction and the actual rating. I called this value the delta and I calculated it by subtracting the actual rating from the prediction and then taking the absolute value of it. In order to visualize the deltas, I made a histogram of them, showing how they are densist around 0 and decrease the farther from 0 we get.



To further use this imformation, and to give myself an idea of which areas to target, I then defined a few ranges based on the delta values. There are seven intervals that I named going from “excellent” to “extremely bad”.

```
##      categories numbers percents    ranges
## 1     excellent  247981     24.8   [0,.25]
## 2     very good  220176     22.0 (.25, .5]
## 3          good  176548     17.7 (.5, .75]
## 4      not good  130448     13.0  (.75, 1]
## 5           bad  195776     19.6    (1, 2]
## 6      very bad   26216      2.6    (2, 3]
## 7     super bad    2792      0.3    (3, 4]
## 8 extremely bad      62      0.0    (4, 5]
```

This table shows the breakdown of deltas as percents, allowing us to easily see that 46.8% of the predictions were within half of a star, and over three quarters were within one star. These numbers are good, though the over 20% that were off by more than one star give evidence that more work is needed.

Conclusion

The results of my preliminary process are encouraging. The RMSE is below .8775 and almost half of the predictions were within .5 stars of the actual review. It is especially worth noting that less than 3% of the predictions were off by more than 2 stars. These numbers, which could certainly be improved upon with further work, are a very solid base to build off of.

As I stated earlier, I had intended to do much more with this project than I did. There were several things that I had hoped to include, and that would be worth investigating to see how much of an impact they would have. One such thing was to use regularization to mitigate the effects of small sample sizes. Another was to use the genre information to find a “top 3” and “bottom 3” genres for each user and either boost or reduce predictions in those genres for those users. I also wanted to delve deeper into the almost 20% of predictions that were off by 1-2 stars because I saw that as the most important segment to improve.

All in all, I found this project to be very helpful to me in getting a better sense of what we’ve been learning in this program. This gave me a chance to put many of the pieces together, especially those relating to reading/writing files and working in R Markdown. One of the most important lessons that I learned, however, was not to use the given test set data until the very end. If I had created an extra partition on the edx data to use as example test sets to give myself feedback along the way, I would have been able to improve my algorithm further and implement my ideas.