MovieLens Capstone - Avery Clark

Avery Clark

January 1, 2020

Executive Summary

In this analysis, I used machine learning methods to build prediction models designed to predict what a user will rate a movie as a foundation for a recommendation system.

In this section I'll describe the dataset and summarize the goal of the project and key steps that were performed.

I analyzed the MovieLens 10M database and used it to attempt to build a machine learning algorithm that can predict what movies users would like to watch with high accuracy. These predictions will be trained on one dataset and tested on a separate dataset, where they will hopefully come very close to predicting how many stars (on a 0.5 to 5 star scale) a user will rate a movie.

To win the grand prize of $1 million from the Netflix challenge, a participating team had to get to a residual mean square error (RMSE) of about 0.857.

You can read more about it here: http://blog.echen.me/2011/10/24/winning-the-netflix-prize-a-summary/

My goal was to build a prediction model with a RMSE of less than 0.8649. I surpassed that goal.

The RMSE (Residual Mean Square Error) is the standard deviation of the prediction errors (residuals).

In other words, it's the average of how far the predictions deviate from what they are trying to predict.

The lower the RMSE, the more accurate the algorithm's predictions are.

I split the data into a training set (90% of data) to train the prediction models and a testing set (10% of data) to test the accuracy of the prediction model.

After running five prediction models, the lowest Residual Mean Square Error (RMSE) obtained was 0.8644501, which accomplishes the goal of reaching lower than 0.8649.

The most effective prediction model was "Regularized Movie + User + Genre Effect Model", where I used the biases per movie, per user, and per genre of the reviews in the training set and then regularized (or "rubber-banded") the results, penalizing biases of movies/users/genres with low review counts by pulling them toward the dataset average.

This report contains four sections: Executive Summary, Analysis, Results, and Conclusion.

Executive Summary describes the dataset and summarizes the goal of the project and key steps that were performed.

Analysis explains the process and techniques used, such as data cleaning, data exploration and visualization, any insights gained, and the modeling approach.

Results presents the modeling results and discusses the model performance.

Conclusion gives a brief summary of the report, its limitations and future work.

Thank you for taking the time to look at this report. I hope that you will run this code by stepping through (by pressing Ctrl + Enter) as I'm explaining it.

Analysis

I'd like to start this analysis off by asking: How important is it that your next recommendation be something you really like? Netflix thought it was so important that they happily offered $1 million for a 10% increase in the accuracy of their recommendations.

You can read more about it here: http://blog.echen.me/2011/10/24/winning-the-netflix-prize-a-summary/

Below, I'll be analyzing the MovieLens 10M database and attempt to build a machine learning algorithm that can predict what movies users would like to watch with high accuracy. These predictions will be trained on one dataset and tested on a separate dataset, where they will hopefully come very close to predicting how many stars (on a 0.5 to 5 star scale) a user will rate a movie.

To win the grand prize of $1 million from the Netflix challenge, a participating team had to get to a residual mean square error (RMSE) of about 0.857.

My goal is to build a prediction model with a RMSE of less than 0.8649.

The RMSE (Residual Mean Square Error) is the standard deviation of the prediction errors (residuals).

In other words, it's the average of how far the predictions deviate from what they are trying to predict.

The lower the RMSE, the more accurate the algorithm's predictions are.

This was run using RStudio Version 1.1.463 and R version 3.6.2 (Dark and Stormy Night) from https://www.r-project.org/

In this section, I'll explain the process and techniques used, such as data cleaning, data exploration and visualization, any insights gained, and the modeling approach. You'll see these models in action in the Results section.

90% of the data was designated for training the prediction model and 10% of the data was reserved for testing the accuracy of that model's predictions.

A simple way of thinking about this is that the model (or algorithm) will learn about the data by taking in different factors and will make a prediction of what star rating (on a 0.5 to 5 star scale) a user will rank a movie based on those factors. Different approaches will have the model/algorithm using the factors given to it in different ways to make predictions.

The model/algorithm decides to predict a review rating "Y" based on factors "A", "B", and "C" (or more). Then the model/algorithm is exposed to the testing dataset to see if what it predicts as the review rating "Y" (based on the factors in the new dataset "A", "B", and "C") is actually that accurate or not. Then from the results we can compute our RMSE (Residual Mean Square Error). This is how we test the model's accuracy.

The RMSE (Residual Mean Square Error) is the standard deviation of the prediction errors (residuals). In simpler terms, it's the average of how far the predictions deviate from what they are trying to predict (how far off the mark our model's predictions are). The lower the RMSE, the more accurate the model's predictions are.

I hope that you will step through the code with me as I explain it.

You can run all of the code by clicking Run. You can run it line by line by pressing Ctrl + Enter on your keyboard. You can also highlight a section of code and run just that by clicking Run or pressing Ctrl + Enter on your keyboard.

Let's dig in!

First, we'll build our training set (edx set) and our validation set from the MovieLens 10M dataset by splitting the data up randomly.

The training set (edx set) will hold about 90% of our data, and the validation set will hold about 10%.

We will use the training set to train our machine learning algorithm and we will test its accuracy on the validation set.

To begin this, we'll install the packages that will give us the tools to analyze the data. Notice the if statements mean the packages will not install if you have them already.

Note: this might take a couple of minutes.

Conclusion

In this section I'll give a brief summary of the report, its limitations and future work.

I split the data into a training set (90% of data) to train the prediction models and a testing set (10% of data) to test the accuracy of the prediction model.

After running five prediction models, the lowest Residual Mean Square Error (RMSE) obtained was 0.8644501, which accomplishes the goal of reaching lower than 0.8649.

The most effective prediction model was "Regularized Movie + User + Genre Effect Model", where I used the biases per movie, per user, and per genre of the reviews in the training set and then regularized (or "rubber-banded") the results, penalizing biases of movies/users/genres with low review counts by pulling them toward the dataset average.

I feel as though my report has some limitations. I could have taken more modeling approaches, such as Naive Bayes Classification and Matrix Decomposition, to potentially reach a lower RMSE.

I keep thinking about how, in the Netflix challenge, to win the grand prize of $1 million a participating team had to reach a RMSE of about 0.857.

I would like to improve this analysis in the future by finding some prediction model approaches that will give me a RMSE of less than 0.857.

Thank you for reading my report. I hope you enjoyed it. - Avery Clark