Capstone Project

Chen Shih-Chieh

December 28, 2019

Introduction

In this project, I will be creating an algorithm for predicting how a user would rate a movie from a set of data called edx. The goal of the project is to obtain an RMSE of less than or equal to 0.86490 on a validation set. To achieve this goal, I plan to utilize all the available data: userId, movieId, timestamp, genres, and release year (imbedded within the movie titles). I first processed the categorical data (userId, movieId, genres) by average the effects of each category. I also regularized the categorical predictors in order to accommodate categories with small sample sizes. Afterwards, the continuous predictors, release year and timestamp order–a generated variable showing the seconds since the user's first rated their movie.

Methods/Analysis

The categorical predictors were created based on the machine learning course. They include the baseline[a], movie effects[b], user effects[c], and genre effects[d].

a.The average rating was calculated from all the available ratings. This can be considered a baseline factor, the rating that people would generally give if there were no other factors. This value is labeled mu.

b.The baseline was subtracted from ratings and the remainders used for modeling. The remainders were grouped by their movieId and the average was calculated for each group. Some movies are generally well liked while others are not, with many in between. These averages will account for those differences. These values are labeled b_i.

c.The baseline and movie effects were subtracted from ratings and the remainders used for modeling. The remainders were grouped by userId and the average was calculated for each group. Some users rate movies more strictly than others. These averages will account for those differences. These values are labeled b_u.

d.The baseline, movie effects, and user effects, were subtracted from ratings and the remainders used for modeling. The remainders were grouped by genres and the average was calculated for each group. These values are labeled b_g. It should be noted that instead of having a single genre, many movies contain multiple genres. For example, a movie, "Boomerang," was categorized as Comedy and Romance. Each unique combination of genres were considered their own group of genre. Each combination may have a unique interaction effect that is different from its components. Considering each combination as their own group preserves this interaction. With 797 unique combinations of genres and 10677 different movie titles, each combination is relatively well represented.

The above values were then regularized. A series of lambda values were tried. Each lambda value was tested using 10 fold cross validation and the lambda value that provided the lowest RMSE was selected.

Like before, previously created models were subtracted from the ratings in the raw data and the remainders used for further analyses. The data was further processed to extract release year and timestamp order–the

time since a user rated their first movie. The release year of each movie was extracted from their respective titles. Timestamp order was created by first, grouping timestamp by userId. For each group, their timestamps are centered–The mean timestamp for each group is subtracted from the timestamp–and shifted so that the smallest value is zero–this is achieved by adding the smallest value of each group to the centered timestamp of that group.

The second set of predictors is a linear model generated from timestamp order and release year. The model uses the residuals from subtracting baseline, movie effect, user effect, and genre effects. The purpose of the model is to account for the effect of release year[a] and timestamp order[b].

a.For the release year, the linear model should capture how movies are rated as they get older.

b.It is possible that as users rate more movies, their rating criteria become more stringent.

Since both release year and timestamp order are continuous variables, linear regression was used to take advantage of the extra information provided by continuous, rather than categorical, variables.

Results

Below is a table comparing the RMSE of the different models from the cross validation analyses. Based on this, it seems that all models will hit the 0.86490 goal that I was trying to achieve. It also seems that the models are not that different from each other, with the largest difference being 0.0000209.

Model	RMSE
Main Effects	0.8563594
Main Effects(Regularized)	0.8563547
Main Effects(Regularized) + LM	0.8563338

Finally, for the validation set, the regularized main effect with the linear model was used and we achieved a RMSE of 0.648306.

Conclusion

In this project, we looked at a set of data that provided us with users ratings of movies. The goal of this project was to create a model that could predict the ratings of users with a RMSE of 0.86490. The variables that were available to me was the user Id, movieId, movie title (with release year embedded), movie genre, timestamp (date and time when the rating was submitted in unix time), and the ratings (out of 5). For my model, I used userId, movieId, genre, the release year (embedded in the title) and the timestamp order (timestamp centered then shifted so the lowest value is at zero). For the categorical variables (userId, movieId, and genre), a model was created based on the average for each category within each variable. The model was regularized to account for low sample categories. For the continuous variables (release year and timestamp order), a linear model was creating using the lm function. The model achieved a RMSE of 0.648306.

While the model did hit the target RMSE, I feel that there is the possibility to create a better model if SVD was able to be implemented. Unfortunately, I was unable to implement SVD due to computing limitation. If a more power machine could be aquired, it would be interesting to see how the movies break down into their principle components and the possible analyses we could do from them.