Movielens - Capstone Project

Srikanth Sridharan

April 18, 2020

Introduction

The purpose of this project is to predict the rating an user would provide to a movie. This predicted rating can be used to line up movie recommendations for the user for future viewing.

The **grouplens** website is dedicated towards social computing research and has few datasets targeted towards this that are available for general public. For more details on grouplens visit https://grouplens.org/. One of the datasets is **movielens** that has movie and rating details collected over time. This data is extracted from the website of movielens project. Please see https://movielens.org/ for more information on movielens. There are a variety of movielens data sets in grouplens. The data set with 10 million ratings is used for this project.

In this project, the movielens dataset is split into a training set and a testing set with 90% of the data set being used for training and the balance 10% being used for testing the accuracy of the model. The accuracy of the model is verified by calculating the **Root Mean Square Error**. The ratings of the validation set, predicted through the model, is compared with the actual rating in this set by, calculating the RMSE.

Analysis

User ratings vary based on their perception of a movie. A block buster movie may have ratings predominantely above 4 on a scale of 1 to 5, with 5 being the highest. However, certain users may be critical of it due to a variety of reasons such as movie length, ingenuinity in story, etc. Hence, the user rating would depend on the past rating provided by the user. In this project, the training dataset is named as **edx** while the testng dataset is called **validation**. The edx data set is used to train a model and the validation data set is used to test the accuracy of the model. The average rating of a movie and the average user bias on various movies is used to predict the user rating for a movie.

The movielens 10 million ratings dataset can be downloaded from the link http://files.grouplens.org/datasets/movielens/ml-10m.zip. This zip file contains two files, each for the list of movies (movies.dat) and the ratings (ratings.dat) for those movies. The movies and rating in both the files are tied together through the MovieId field. The zip file contents were downloaded and stored in a temporary object so that it could be extracted and modified to be fit for data analysis.

The **ratings** object contains the ratings from the ratings.dat file. The data in both the files are delimited with a double colon "::". Hence, the data is extracted to seperate columns using this delimiter. The column names are set for clarity and programming purposes. The below summary shows the summary of ratings dataset after performing these steps.

```
summary(ratings)
```

```
##      userId        movieId         rating        timestamp
##  Min.   :    1   Min.   :    1   Min.   :0.500   Min.   :7.897e+08
##  1st Qu.:18123   1st Qu.:  648   1st Qu.:3.000   1st Qu.:9.468e+08
##  Median :35741   Median : 1834   Median :4.000   Median :1.035e+09
##  Mean   :35870   Mean   : 4120   Mean   :3.512   Mean   :1.033e+09
##  3rd Qu.:53608   3rd Qu.: 3624   3rd Qu.:4.000   3rd Qu.:1.127e+09
##  Max.   :71567   Max.   :65133   Max.   :5.000   Max.   :1.231e+09
```

Similar to the ratings, the movies data in movies.dat is extracted to the **movies** data frame. The column names of this dataset is set appropriately to match the ratings dataset. Below is a summary of the movies data frame after performing these steps.

```
summary(movies)
```

```
##     movieId         title              genres
##  Min.   :    1   Length:10681       Length:10681
##  1st Qu.: 2755   Class :character   Class :character
##  Median : 5436   Mode  :character   Mode  :character
##  Mean   :13121
##  3rd Qu.: 8713
##  Max.   :65133
```

The movies and ratings datasets are combined to a single data frame to make data analysis and modeling easier. The movielens object contains this merged dataset. Below is a sample from the movielens data frame after merging the two datasets.

```
##    userId movieId rating timestamp                         title
## 1       1     122      5 838985046              Boomerang (1992)
## 2       1     185      5 838983525               Net, The (1995)
## 3       1     231      5 838983392          Dumb & Dumber (1994)
## 4       1     292      5 838983421               Outbreak (1995)
## 5       1     316      5 838983392               Stargate (1994)
## 6       1     329      5 838983392 Star Trek: Generations (1994)
## 7       1     355      5 838984474       Flintstones, The (1994)
## 8       1     356      5 838983653           Forrest Gump (1994)
## 9       1     362      5 838984885       Jungle Book, The (1994)
## 10      1     364      5 838983707         Lion King, The (1994)
##                                      genres
## 1                          Comedy|Romance
## 2                    Action|Crime|Thriller
## 3                                  Comedy
## 4               Action|Drama|Sci-Fi|Thriller
## 5                    Action|Adventure|Sci-Fi
## 6               Action|Adventure|Drama|Sci-Fi
## 7                    Children|Comedy|Fantasy
## 8                  Comedy|Drama|Romance|War
## 9                Adventure|Children|Romance
## 10 Adventure|Animation|Children|Drama|Musical
```

The seed is set to 1 to ensure consistency in results across systems.

Since the validation dataset can only be used once to predict the ratings. Hence, the edx dataset is further partitioned into train and test datasets in the ratio 9:1. The train datset is used to train the models, while the test dataset is used to verify the model and tune it further to reduce RMSE. The average rating of each movie in the train dataset is calculated. This is used as the primary predictor to calculate the user rating. The user bias from the average rating of a movie is calculated for each movie and for each user in the train dataset. This gives an indication of whether the user is liberal and give ratings in accordance with the wider population or whether they are thoughtful in their ratings.

The average rating for a movie is obtained using formula

$\mu_m = \sum_{i=1}^{n} \frac{r_{im}}{n}$

where,

μ_m is the average movie rating for a movie m

r_{im} is i^{th} rating for the movie m

n is the total number of ratings for the movie m

The average user bias for an user is obtained using formula

$b_u = \sum_{m=1}^{n} \frac{r_{mu} - \mu_m}{n}$

where,

b_u is the average rating bias by user u

r_{mu} is the rating by the user u for the m^{th} movie

n is the total number of ratings by the user u

The prediction model is the sum of average rating of the movie and the general bias of the user towards a movie. It is represented by the below formula.

$r_{um} = \mu_m + b_u$

Based on the above model, the ratings are predicted and the RMSE is calculated to understand how well this model performs.

The RMSE of the previous model is 0.8646843. There could be other considerations included in deciding the model. Did the average user rating for movies change over the years? Is there a general trend of rating the movies over the years irrespective of the director or the protagonist? These could be considered for fitting the model to impove accuracy. But it could also result in over-fitting. However, the above model can be regularized to penalize large values. Using L2 regularization on the user bias, the optimal value for lambda is found that minimizes the RMSE is found. A lambda value of 5 minimized the RMSE to 0.864253. The average user bias for an user with the lambda value is calculated using the below formula.

$b_u = \sum_{m=1}^{n} \frac{r_{mu} - \mu_m}{n + \lambda}$

where,

b_u is the average rating bias by user u

r_{mu} is the rating by the user u for the m^{th} movie

n is the total number of ratings by the user u

μ_m is the average movie rating for a movie m

λ is the L2 regularization factor

The prediction model is the same as mentioned before. i.e

$r_{um} = \mu_m + b_u$

Results

The prediction algorithm was run on the validation dataset. This resulted in a RMSE of 0.8649708.

The prediction model is a basic model that does not over-fit the testing data due to L2 regularization that penalized large values. Since the actual ratings provided by the users will be in multiples of 0.5, there will be hardly any predicted ratings that absolutely will match with the actual rating. Hence, there will always be some residual error. However, models can be updated to bring the predicted rating as close as possible to the actual rating.

Conclusion

This report has described a basic model that uses the user bias and average movie rating in predicting the user rating. Depending on the capacity of the system, a much more elaborate model can be built that includes the bias based on genre as well as the timestamp. An analysis of the movielens dataset showed ratings were in integer prior to 2003-02-12. "Half" ratings seem to have been introduced only after this date. Including these in the prediction model may reduce the RMSE further. However, in this project only the user and movie are distributed in both the edx and validation datasets. Hence, the final model is optimal for predicting the rating.

Also, the RMSE value varies based on the seed value. The RMSE value of the final model is arrived by setting seed to 1. If the seed value is changed, based on that the RMSE value can increase or decrease.