MovieLens Project Analysis

Pedro Rodrigues

07/16/2020

Introduction

This report intends on explaining the analysis and the predictive model on the MovieLens dataset. Beginig with a massive dataset of over 10 million ratings, of 10,677 movies from almost 70 thousand anonymous users. Acording to instuctions we had to split the data into an "edx" and a "validation" set, being allowed to use the validetion one only for the final RSME calculation. The main goal is to predict reasonably well the ratings on a Validation set, thus creating a recomendation system based on data. The method used was based on the average and than correcting for estimated biases from movie, user and genre. Better movies get better ratings, crankier users give worse reviews and some genres tend to give higher expectations to who is wathing, which may lead to some variation on the ratings.

Method

I began from the basic doing a little bit of exploration on the dataset I was going to use: edx. I notice the fact that there was 797 genres variation, upon closer look it was because there was more than one genre on some movies, thus leading to more variation. Only on the head, were most was action movies there was already some variation due to genre:

```
## # A tibble: 6 x 2
##   genres                                               `Average Rating`
##   <fct>                                                           <dbl>
## 1 (no genres listed)                                               3.64
## 2 Action                                                           2.94
## 3 Action|Adventure                                                 3.66
## 4 Action|Adventure|Animation|Children|Comedy                       3.96
## 5 Action|Adventure|Animation|Children|Comedy|Fantasy               2.99
## 6 Action|Adventure|Animation|Children|Comedy|IMAX                  3.30
```

So I thought it would be a good source of information, because it would separate the similiar movies, thus providing information on what a user thinks of that style of movies. Thus designing my first model:

$$y_{m,u,g} = \mu + b_m + b_u + b_g$$

Began by spliting my edx set into train and test sets(80% - 20%) so I could test my algorithms and see any improvments. And defining a function to calculate my RMSE scores with ease:

```
RMSE <- function(true_ratings, predicted_ratings){
  sqrt(mean((true_ratings - predicted_ratings)^2))
}
```

Then I coul begin training it by calculating μ the mean of all ratings. Then calculating the movie, user and genres biases. So I had my first model and could calculate my first prediction. I got an RSME of 0.8655873, not satisfied me. So I regularized it. Trying to minimize the effect of obscure movies that didn't had too many reviews. So I began minimizing this, instead of the typical RSME equation:

$$\frac{1}{N}\sum(y_{m,u,g} = \mu + b_m + b_u + b_g)^2 + \lambda(\sum b_m^2 + \sum b_u^2 + \sum b_g^2)$$

We could use calculus to derivate that, and discover that is possible to calculate them using some variation of the formula:

$$b_x(\lambda) = \frac{1}{\lambda + n_x}\sum_{x=1}^{n_x}(y_x - \mu)$$

After using my train and test set to tune in the best possible value for λ, I found an RSME of 0.8649403, and I came to the conclusion that once I trained it using the whole edx data set to try on the validation one I would improve a bit more, thus reaching my Final Model.

Results

The final model code:

```
# Lambda that was tuned on the train test subsets of edx
lambda <- 4.75

# Average of the whole dataset
mu_f <- mean(edx$rating)

# Movie Bias
b_i <- edx %>%
  group_by(movieId) %>%
  summarize(b_i = sum(rating - mu_f)/(n()+lambda))

# User Bias
b_u <- edx %>%
  left_join(b_i, by="movieId") %>%
  group_by(userId) %>%
  summarize(b_u = sum(rating - b_i - mu_f)/(n()+lambda))

#Genre Bias
b_g <- edx %>%
  left_join(b_i, by='movieId') %>%
  left_join(b_u, by='userId') %>%
  group_by(genres) %>%
  summarize(b_g = sum(rating - b_i - b_u - mu_f)/(n()+lambda))

# Final Prediction
predicted_ratings <- validation %>%
  left_join(b_i, by='movieId') %>%
  left_join(b_u, by='userId') %>%
  left_join(b_g, by='genres') %>%
  mutate(pred = mu_f + b_i + b_u + b_g) %>%
  .$pred

#Final RMSE
RMSE(predicted_ratings, validation$rating)
```

```
## [1] 0.8644514
```

I got very satisfied with this result, even though it still can be improved on. It was a very simple model that doesn't require much from the machine even though it uses this massive dataset.

Conclusion

There it is, a simple predictive model of ratings, formulating a recomendation system, with reasonable RMSE scores that doens't require much from the machine. Using simple bias calculation based of the winners from the Netflix contest of recomendation systems. It isn't very creative, but it works, I'm sure better minds than mine could do a better job. I'm thinking on studing further on the fields of PCA and SVD and make a recomendation system using such tools. But that is work for another day.