MovieLens Capstone Project

Manuel_5600 (edx username)

15th of March 2019

Contents

Recommendation System: Introduction/Overview/Summary **2**
Observations: Users and Movies 3
Observations: Movies 4
Observations: Users 6

Recommendation System: Methods/Analysis **8**
Average of Ratings, Movie/User Bias and Lambda 8

Recommendation System - Results/Conclusion **9**
Prediction and RMSE 9

Recommendation System: Introduction/Overview/Summary

The basic idea of a recommendation system is to give a helpful recommendation based on available data. To be more specific, the task is to predict the rating a particular user would give to a specific movie and therefore to provide matching movie suggestions to that user. The available data was the movielens data included in the dslabs package (a small subset of a much larger dataset with millions of ratings). It is a data set where each row contains a rating by a specific user for a specific movie.

According to Netflix such a recommender system is not only a nice gimmick, it has an enormous business value:

"*Consumer research suggests that a typical Netflix member loses interest after perhaps 60 to 90 seconds of choosing, having reviewed 10 to 20 titles (perhaps 3 in detail) on one or two screens. The user either finds something of interest or the risk of the user abandoning our service increases substantially. The recommender problem is to make sure that on those two screens each member in our diverse pool will find something compelling to view, and will understand why it might be of interest"* (The Netflix Recommender System: Algorithms, Business Value, and Innovation, CARLOS A. GOMEZ-URIBE and NEIL HUNT, 2015, Netflix, Inc.).

"*We think the combined effect of personalization and recommendations save us more than $1B per year"* (The Netflix Recommender System: Algorithms, Business Value, and Innovation, CARLOS A. GOMEZ-URIBE and NEIL HUNT, 2015, Netflix, Inc.).

This paper obviously had a less ambitious goal than Netflix, which was to achieve a RMSE lower or equal to 0.87750. A RMSE larger than 1 means that our typical error is larger than one star (of a 5 star rating, 1 is equal to 'not a good movie', 5 is equal to 'a great movie'). *"Historically, the Netflix recommendation problem has been thought of as equivalent to the problem of predicting the number of stars that a person would rate a video after watching it, on a scale from 1 to 5"* (The Netflix Recommender System: Algorithms, Business Value, and Innovation, CARLOS A. GOMEZ-URIBE and NEIL HUNT, 2015, Netflix, Inc.). Nowadays Netflix uses a much more sophisticated recommender system.

The recommendation system which is proposed in this paper predicts ratings by adding up the average, the movie bias, the user bias and by penalizing small sample sizes with lambda. Additional information is provided within the paragraphs that describe the method, analysis and results. The method is based on the course book "Introduction to Data Science" (https://rafalab.github.io/dsbook/acknowledgements.html). The R and Rmd files were made available to the readers but the R code has been excluded from this report to enhance readability.

However, as a first step, the problem/task must be well understood and therefore an initial familiarization with the available data was necessary. Please find below the main findings:

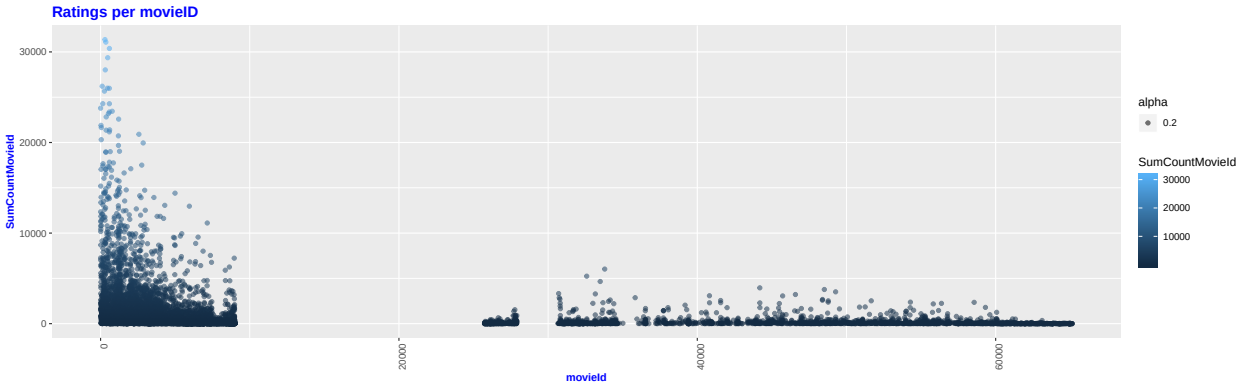
Observations: Users and Movies

Amount of Distinct Users and Movies - the data contains more users than movies and not all the users seem to have rated all the movies.

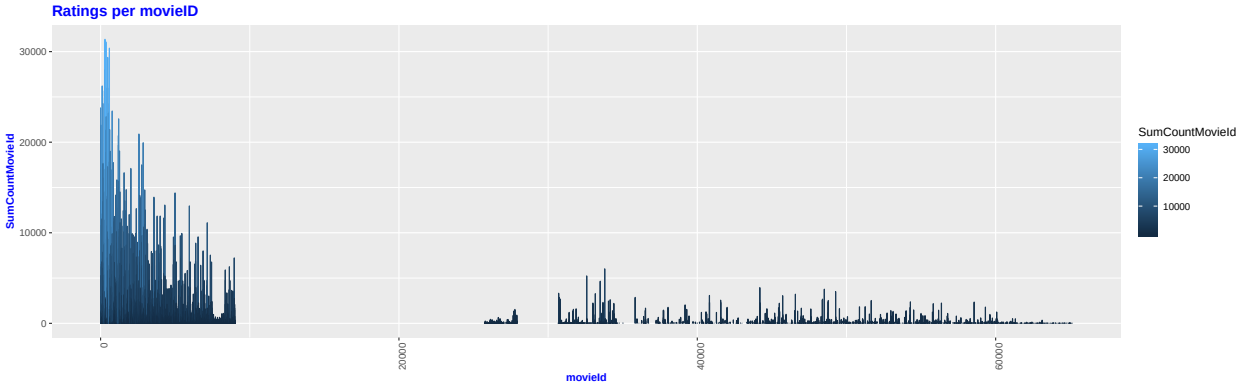
n_users	n_movies
69878	10677

Amount of Ratings per Movie - some movies get rated more frequently. Probably due to the fact, that not all the movies are equally popular. Therefore, it seems to be likely that some movies were simply watched more often. Please find the amount of ratings per movie illustrated below:

of Ratings per Movie-1.bb



of Ratings per Movie-2.bb



movieId	title	SumCountMovieId	MeanRatingMovieId
296	Pulp Fiction (1994)	31362	4.154789
3191	Quarry, The (1998)	1	3.500000
3226	Hellhounds on My Trail (1999)	1	5.000000

Note:
Minimum and maximum amount of ratings / limited to 3 rows

Observations: Movies

Top Rated Movies - the top rated movies are very well known movies. They seem to have more and better ratings than the bottom rated movies (based on the mean rating). However, the table 1t below only shows titles with more than a 100 ratings. This was necessary to avoid titles which had received only a few but exclusively excellent ratings, e.g. a single 5 star rating (as shown in table 2t). There seems to be substantial difference in the amount of ratings a specific movie receives.

movieId	title	SumCountMovieId	MeanRatingMovieId
318	Shawshank Redemption, The (1994)	28015	4.455131
858	Godfather, The (1972)	17747	4.415366
50	Usual Suspects, The (1995)	21648	4.365854
527	Schindler's List (1993)	23193	4.363493
912	Casablanca (1942)	11232	4.320424
904	Rear Window (1954)	7935	4.318651
922	Sunset Blvd. (a.k.a. Sunset Boulevard) (1950)	2922	4.315880
1212	Third Man, The (1949)	2967	4.311426
3435	Double Indemnity (1944)	2154	4.310817
1178	Paths of Glory (1957)	1571	4.308721

Note:
Table 1t with more than 100 ratings / limited to 10 rows

movieId	title	SumCountMovieId	MeanRatingMovieId
3226	Hellhounds on My Trail (1999)	1	5.00
33264	Satan's Tango (SÃ¡tÃ¡ntangÃ³) (1994)	2	5.00
42783	Shadows of Forgotten Ancestors (1964)	1	5.00
51209	Fighting Elegy (Kenka erejii) (1966)	1	5.00
53355	Sun Alley (Sonnenallee) (1999)	1	5.00
64275	Blue Light, The (Das Blaue Licht) (1932)	1	5.00
5194	Who's Singin' Over There? (a.k.a. Who Sings Over There) (Ko to tamo peva) (1980)	4	4.75
26048	Human Condition II, The (Ningen no joken II) (1959)	4	4.75
26073	Human Condition III, The (Ningen no joken III) (1961)	4	4.75
65001	Constantine's Sword (2007)	2	4.75

Note:
Table 2t without filter / limited to 10 rows

Bottom Rated Movies - the bottom rated movies are "special" movies. They seem to have less and worse ratings than the top rated movies (based on the mean rating). However, the table 1b below only shows titles with more than a 100 ratings. This was necessary to avoid titles which had received only a few but exclusively terrible ratings, e.g. a single 0.5 star rating (as shown in table 2b). There seems to be a difference in the amount of ratings a specific movie receives.

movieId	title	SumCountMovieId	MeanRatingMovieId
6483	From Justin to Kelly (2003)	199	0.9020101
6371	PokÃ©mon Heroes (2003)	137	1.0291971
4775	Glitter (2001)	339	1.1755162
5672	Pokemon 4 Ever (a.k.a. PokÃ©mon 4: The Movie) (2002)	202	1.1782178
1826	Barney's Great Adventure (1998)	208	1.1875000
6587	Gigli (2003)	313	1.1932907
31698	Son of the Mask (2005)	165	1.3030303
6872	House of the Dead, The (2003)	209	1.3468900
1495	Turbo: A Power Rangers Movie (1997)	394	1.3591371
4241	PokÃ©mon 3: The Movie (2001)	239	1.3828452

Note:
Table 1b with more than 100 ratings / limited to 10 rows

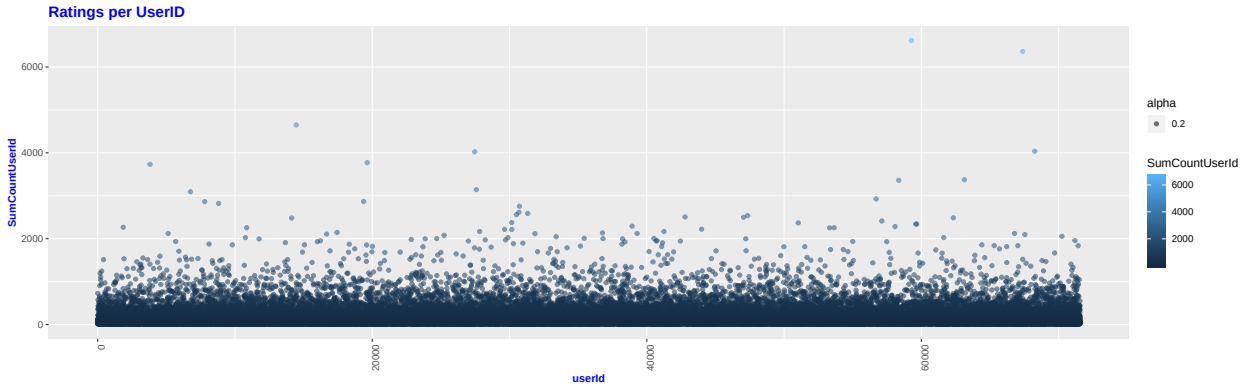
movieId	title	SumCountMovieId	MeanRatingMovieId
5805	Besotted (2001)	2	0.5000000
8394	Hi-Line, The (1999)	1	0.5000000
61768	Accused (Anklaget) (2005)	1	0.5000000
63828	Confessions of a Superhero (2007)	1	0.5000000
64999	War of the Worlds 2: The Next Wave (2008)	2	0.5000000
8859	SuperBabies: Baby Geniuses 2 (2004)	56	0.7946429
7282	Hip Hop Witch, Da (2000)	14	0.8214286
61348	Disaster Movie (2008)	32	0.8593750
6483	From Justin to Kelly (2003)	199	0.9020101
604	Criminals (1996)	2	1.0000000

Note:
Table 2b without a filter / limited to 10 rows

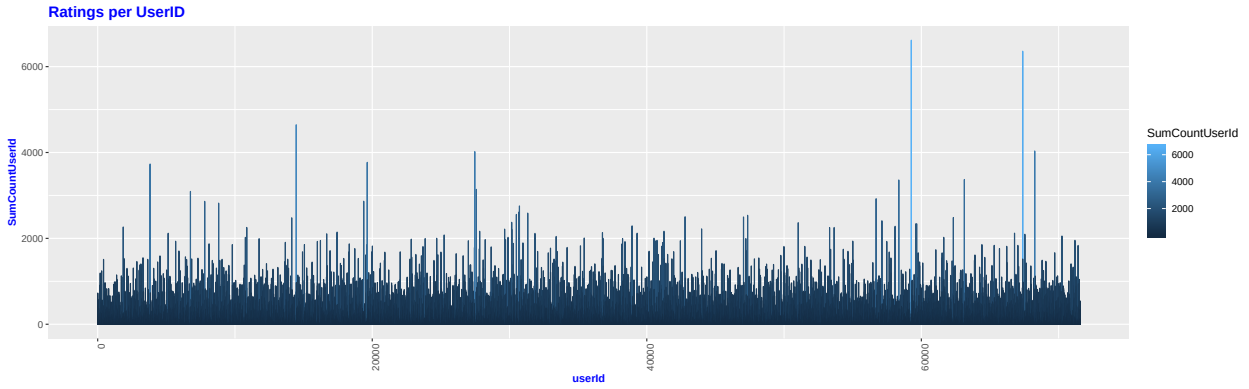
Observations: Users

Amount of Ratings per User - some users are more likely to provide a rating for a movie (probably depends on the personal preference of the user). The table shows the minimum and maximum amount of ratings provided by a specific userId. Additionally, please find the amount of ratings per user illustrated below:

of Ratings per User-1.bb



of Ratings per User-2.bb



userId	SumCountUserId	MeanRatingUserId
59269	6616	3.264586
62516	10	2.250000

Note:
Minimum and maximum amount of ratings per userId

Top Raters by Frequency - some users seem to rate movies more actively than others (probably depends on the personal preference of the user). Here are the top raters (based on the amount of ratings provided).

userId	SumCountUserId	MeanRatingUserId
59269	6616	3.264586
67385	6360	3.197720
14463	4648	2.403615
68259	4036	3.576933
27468	4023	3.826871
19635	3771	3.498807
3817	3733	3.112510
63134	3371	3.268170

Note:
Top raters by frequency / limited to 8 rows

Bottom Raters by Frequency - some users seem to rate movies less frequently than others (probably depends on the personal preference of the user). Here are the bottom raters (based on the amount of ratings provided).

userId	SumCountUserId	MeanRatingUserId
62516	10	2.250000
22170	12	4.000000
15719	13	3.769231
50608	13	3.923077
901	14	4.714286
1833	14	3.000000
2476	14	2.928571
5214	14	1.785714

Note:
Bottom raters by frequency / limited to 8 rows

Top Raters by Mean - some users seem to provide higher ratings than others (probably depends on the user's personality, e.g. not very picky). Here are the top raters (based on the mean rating) that have provided more than a 100 ratings. The limit of 100 ratings was chosen to avoid raters that decided to only provide very few ratings.

userId	SumCountUserId	MeanRatingUserId
5763	214	4.934579
59987	202	4.896040
36896	149	4.892617
19010	140	4.850000
16033	102	4.843137
48518	130	4.838462
49082	118	4.822034
20931	192	4.804688

Note:
Top raters by mean / limited to 8 rows

Bottom Raters by Mean - some users seem to provide lower ratings than others (probably depends on the user's personality, e.g. very picky). Here are the bottom raters (based on the mean rating) that have provided more than a 100 ratings. The limit of 100 ratings was chosen to avoid raters that only cared to provide very few ratings.

userId	SumCountUserId	MeanRatingUserId
24176	131	1.000000
59342	711	1.063994
4043	232	1.310345
26150	359	1.337047
20240	103	1.368932
45157	180	1.444444
9008	385	1.493507
50560	269	1.494424

Note:
Bottom raters by mean / limited to 8 rows

Recommendation System: Methods/Analysis

The recommendation system was developed based on the instructions in "Introduction to Data Science" (https://rafalab.github.io/dsbook/acknowledgements.html). Please find some additional information regarding the method used and the analysis performed within the following paragraphs.

Average of Ratings, Movie/User Bias and Lambda

The average of all ratings mu ('mean(ratings of train set)') would minimize the RMSE if the same rating was given to all the movies regardless of the user. All the differences would occur due to random variation. However, our initial data exploration did show that some movies are rated higher than others and that some users provide higher ratings than others.

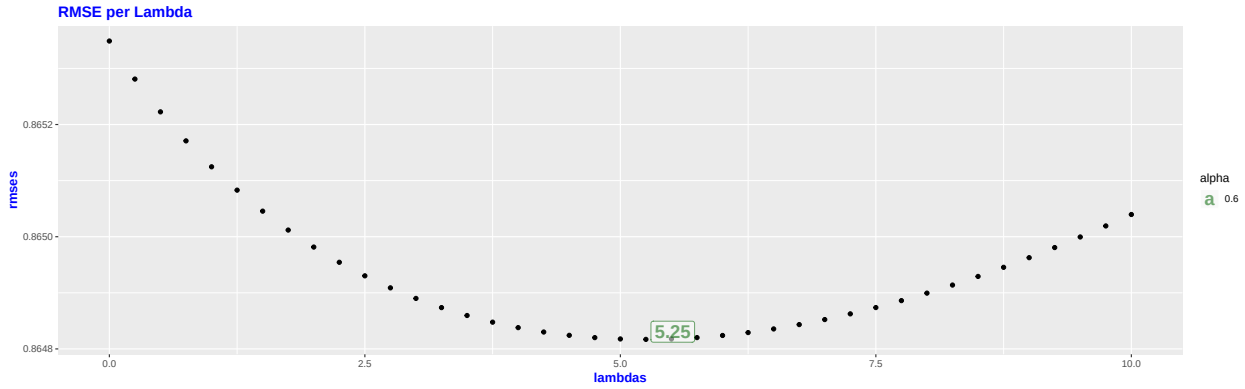
Therefore, a movie bias (b_i) and a user bias (b_u) were introduced. The movie bias (b_i) accounts for the fact that some movies are generally rated higher than others. It is simply the mean difference of the ratings from the average (mu) grouped by movieId; 'b_i = mean(ratings of train set - mu)'. The user bias (b_u) is necessary because different users have different personalities (e.g. more or less likely to give a high rating). It is calculated as the mean difference of the ratings from the sum of the average (mu) and the movie bias (b_i) grouped per userId; 'b_u = mean(ratings of train set - mu - b_i)'.

As mentioned before some of the very well or very badly rated titles did receive only a few ratings (some even just 1). Therefore, Lambda (l) was used to penalize large ratings that were based on a tiny sample size; e.g. 'b_i = sum(ratings of train set - mu) / (n() + lambda))'. This method is called Regularization. The use of a movie bias (b_i), the user bias (b_u) and lambda (l) seems to be consistent with the data exploration performed in the preparatory step of this analysis.

The goal was to minimize the RMSE. Therefore, the value for lambda that resulted in a minimized corresponding RMSE was chosen. The Root Mean Square Error or RMSE is the typical error that is made when predicting a movie rating. A RMSE larger than 1 means that our typical error is larger than one star (of a 5 star rating). The goal was to achieve a RMSE <= 0.87750.

Please find below the chosen lambda that appears to minimise the Root Mean Square Error (RMSE):

```
## [1] 5.25
```



Recommendation System - Results/Conclusion

Finally, the minimized lambda was used to successfully predict the ratings. Furthermore, the predicted ratings resulted in a RMSE lower than the mentioned limit. Please find below some further explanations regarding the calculation of the predicted ratings and the final RMSE that was achieved.

Prediction and RMSE

The edx data set was provided as a training set. It was generated from the movielens data set included in the dslabs package (a small subset of the original dataset with millions of ratings). To receive the predicted ratings we simply had to add up the mean rating based on the edx data set, the movie bias for each movieId based on the edx data set and the user bias for each userId based on the edx data set. Additionally, the movie and user bias (b_u and b_i) needs to be adjusted by using lambda (to penalize ratings that were based on a small sample size, e.g. 1 rating).

The RMSE is the typical error that occurs when predicting a movie rating based on the edx data set (training set) in comparison to the validation data set (generated from the movielens data set as well). Please recall that a RMSE larger than 1 means that our typical error is larger than one star (of a 5 star rating) and the goal was to achieve a RMSE <= 0.87750. Please find below the obtained RMSE:

```
## [1] 0.864817
```

In a nutshell, the simple method described above is able to predict the rating a user would give to a movie with a typical error <= 0.87750 stars. In a rating system that ranges from 1 to 5 stars, this already seems like a useful recommendation.
