MovieLens Project Assignment

Sunil Kumar Pasupuletti

Dec 15 2021

Contents

1 Overview

Recommendation systems use ratings that users have given items to make specific recommendations. Companies that sell many products to many customers and permit these customers to rate their products, like Amazon, are able to collect massive datasets that can be used to predict what rating a particular user will give a specific item. Items for which a high rating is predicted for a given user are then recommended to that user.

Netflix uses a recommendation system to predict how many stars a user will give a specific movie. One star suggests it is not a good movie, whereas five stars suggests it is an excellent movie.

The Netflix data is not publicly available, but the GroupLens research lab generated their own database with over 20 million ratings for over 27,000 movies by more than 138,000 users.

For this project, the ask it to create a movie recommendation system using the MovieLens dataset.

The objective is to train a machine learning algorithm using the inputs in one subset to predict movie ratings in the validation set to arrive at an RMSE that's less than .86490.

1.1 Preparing the MovieLens Dataset

In this section we will explore the MovieLens dataset and its attributes, data composition, various predictors available, statistics from the data available. We will then go ahead in preparing the dataset to be used with the various methods and their performance.

Here we will extract the 10M records and construct the Movielens dataset
We will then split it into a train set, (calling it edx) and a validation set

1.2 Exploring the MovieLens dataset

Let's explore the data and fields available in the movieLens dataset.

```
## [1] "Sample observations"
```

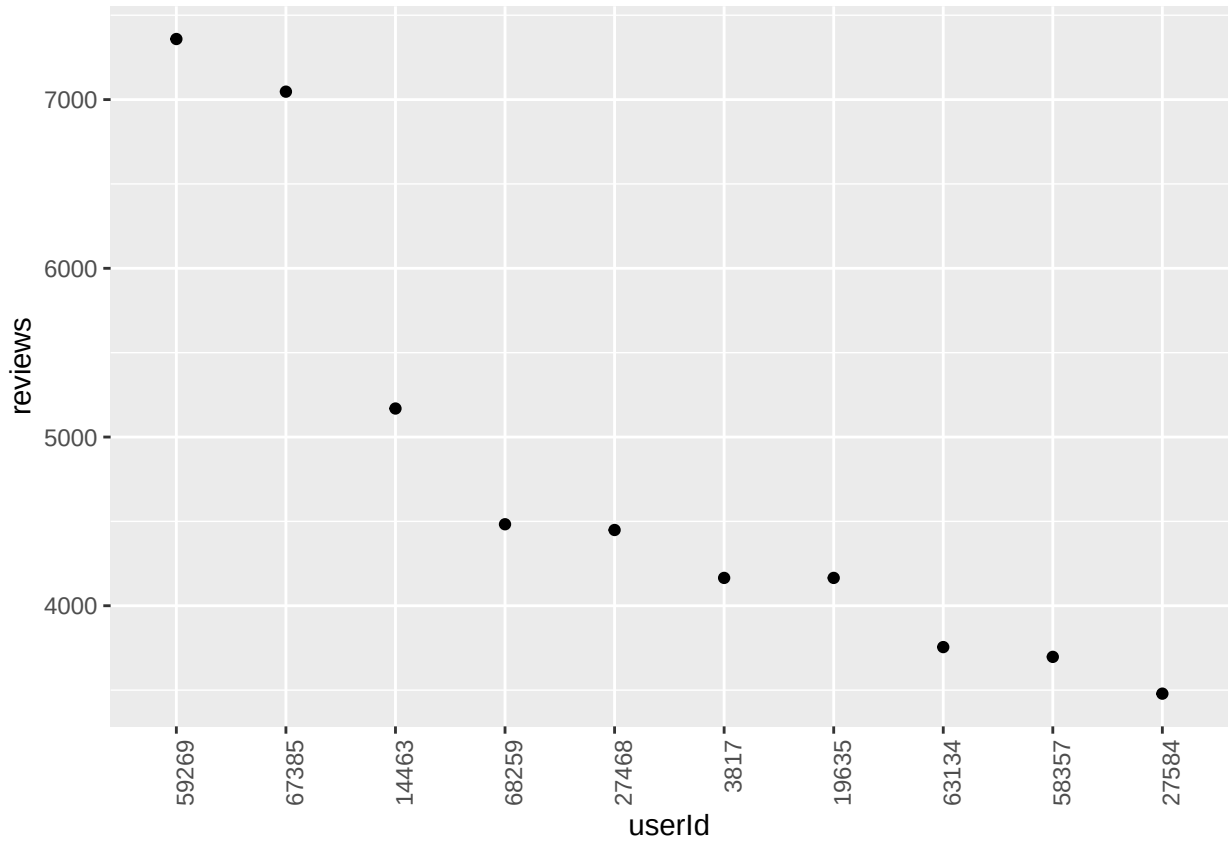
```
##    userId movieId rating timestamp                          title
## 1:      1     122      5 838985046              Boomerang (1992)
## 2:      1     185      5 838983525               Net, The (1995)
## 3:      1     231      5 838983392          Dumb & Dumber (1994)
## 4:      1     292      5 838983421               Outbreak (1995)
## 5:      1     316      5 838983392               Stargate (1994)
## 6:      1     329      5 838983392 Star Trek: Generations (1994)
##                            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
```

```
## [1] "No. of unique Movies in the dataset:  10677"
```

```
## [1] "No. of unique users in the dataset:  69878"
```

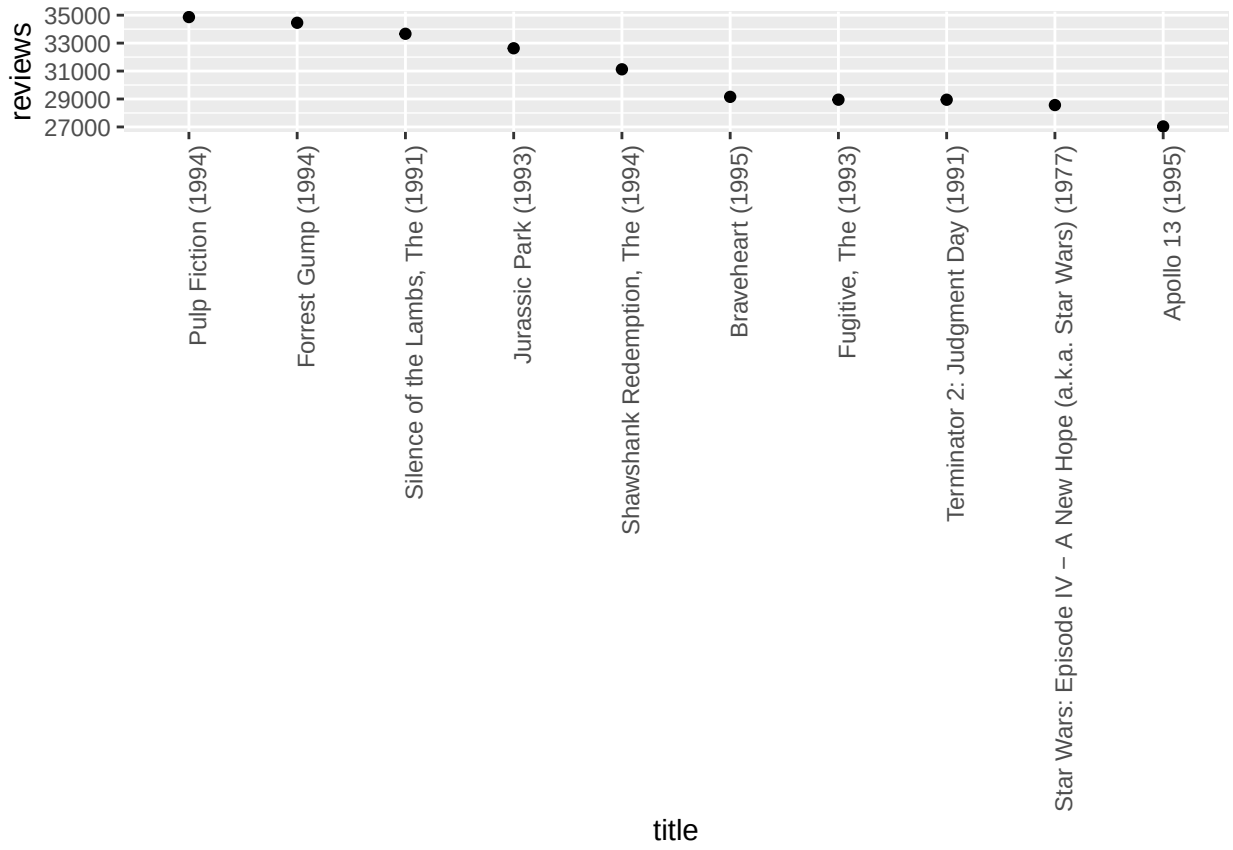
```
## [1] "No. of reviews by user (Top 10): "
```

```
## Selecting by reviews
```




```
## [1] "No. of reviews by movie (Top 10): "
```

```
## Selecting by reviews
```




```
## [1] "Best rated movies(with >10k reviews) (Top 10): "
```

```
## Selecting by Rating
```

```
## # A tibble: 10 x 3
##    title                                                reviews Rating
##    <chr>                                                  <int>  <dbl>
##  1 Shawshank Redemption, The (1994)                       31126   4.46
##  2 Godfather, The (1972)                                  19814   4.42
##  3 Usual Suspects, The (1995)                             24037   4.37
##  4 Schindler's List (1993)                                25777   4.36
##  5 Casablanca (1942)                                      12507   4.32
##  6 Godfather: Part II, The (1974)                         13281   4.30
##  7 Dr. Strangelove or: How I Learned to Stop Worrying and Love t~   11774   4.30
##  8 One Flew Over the Cuckoo's Nest (1975)                 14435   4.29
##  9 Raiders of the Lost Ark (Indiana Jones and the Raiders of the~   21803   4.26
## 10 Star Wars: Episode IV - A New Hope (a.k.a. Star Wars) (1977)     28566   4.22
```

2 Approach

We will be using common machine learning techniques/algorithms to train a sample of data (called the train set) to generate predictions. These predictions are then compared against the remaining sample of data (called the test set).

To help us make a decision on the best model/method to predict, RMSE (Residual mean squared error) is compared across the methods. The lower the RMSE is, the better the prediction is.

RMSE is interpreted similar to the standard deviation. In our case, if this is larger than 1, it translates into

an error that is larger than one star.

Let's split the edx dataset into train and test sets to apply the methods.

We will use the train set to train the various models and test set is used to test the same. Upon testing a model with the best RMSE, we would do a final validation/test of this model on the validation set.The expectation is that the RMSE thus obtained on the validation set is a much better number to make accurate predictions.

2.1 Method 1 - Making a random prediction

We will build a simple model here. This method makes a random prediction which is then used to calculate the RMSE. This model is expected to perform much lower than the objective and is being constructed here to be measured against the other models that follow below

Table 1: RMSE Results

method	RMSE
Just a hunch	1.841099

2.2 Method 2 - Using the mean rating to predict

This method uses the mean of ratings to make the prediction. We then calculate the RMSE to understand its effectiveness

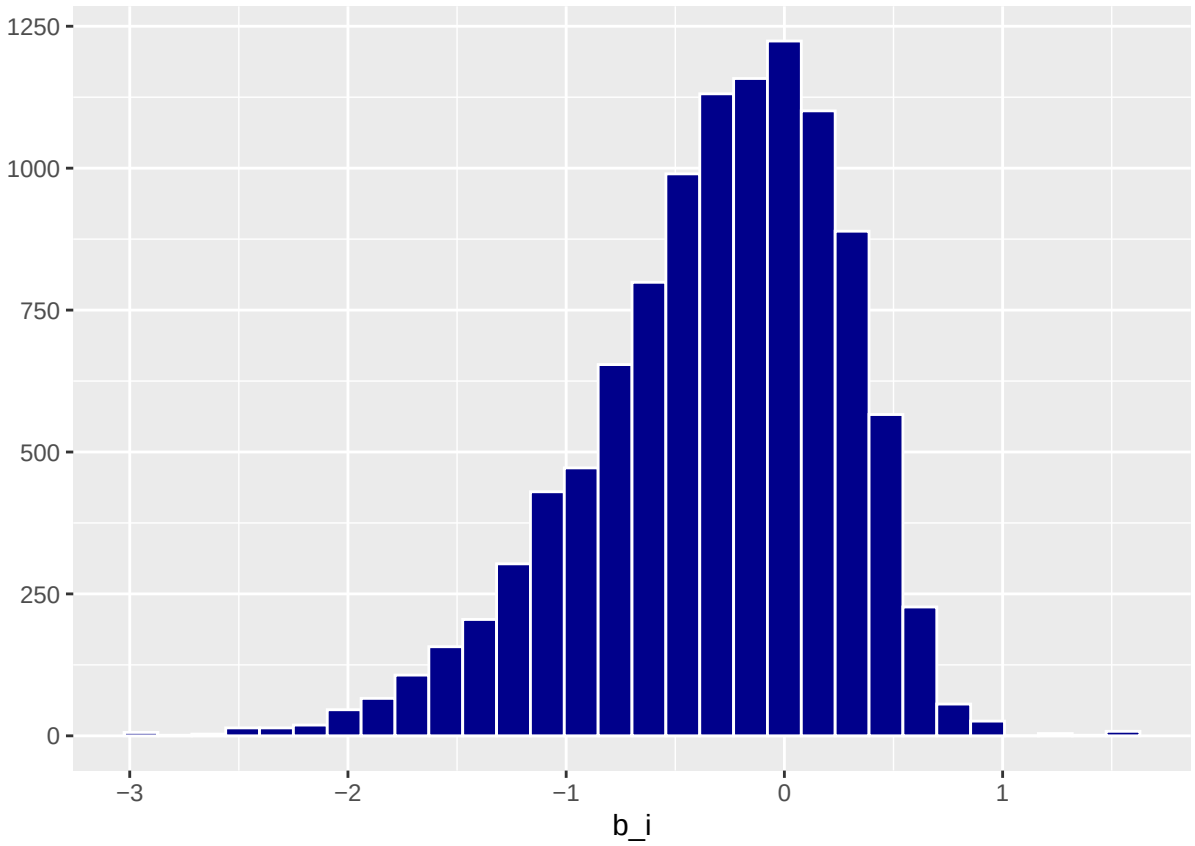
Table 2: Results

method	RMSE
Just a hunch	1.841099
All about Average	1.060704

As expected RMSE obtained here is better than Method 1

2.3 Method 3 - The Movie Effect

From the data, we could interpret that some movies are rated higher than others. We call this movie effects/bias.



From the plot above, we could see that these estimates vary quite a bit

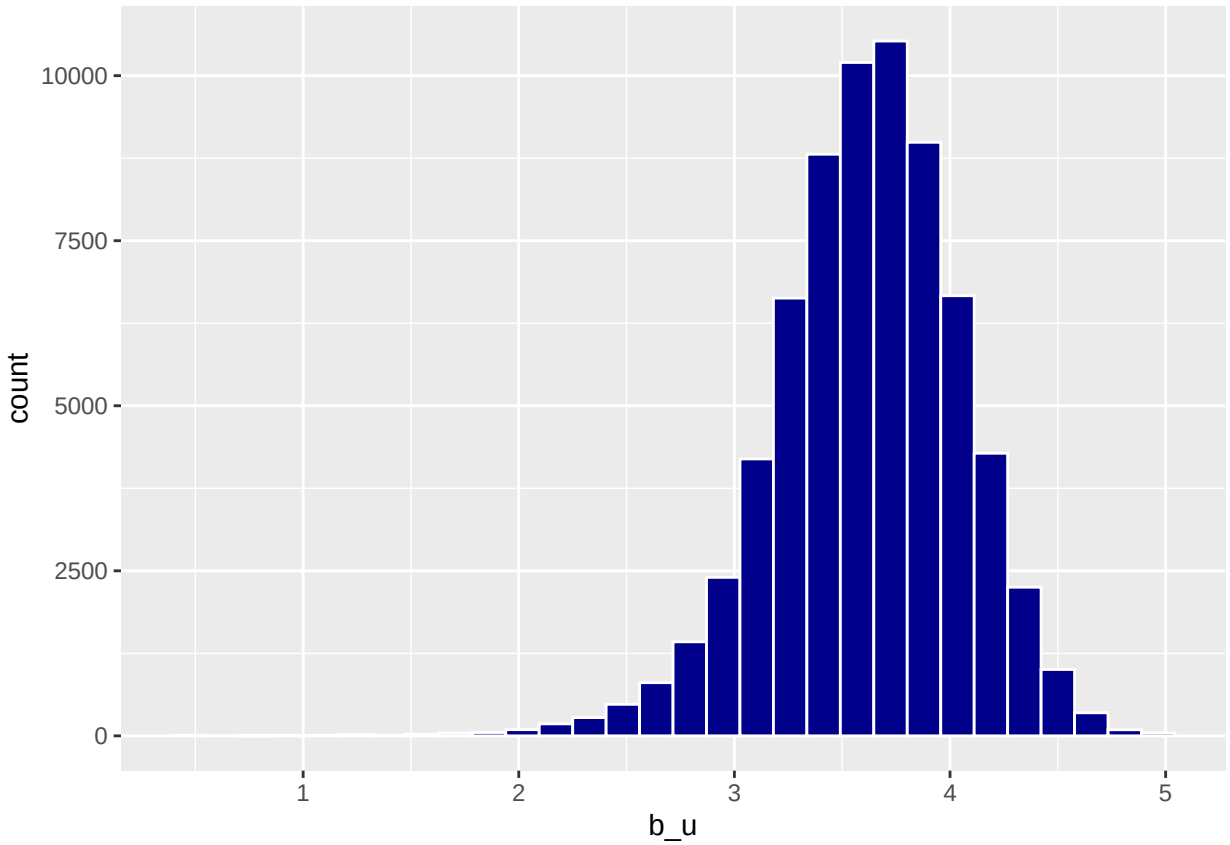
Table 3: Results

method	RMSE
Just a hunch	1.8410994
All about Average	1.0607045
The Movie Effect	0.9437144

RMSE obtained here is better than Method 2, but we can improve it further

2.4 Method 4 - The User Bias

We will compute the average rating for users that have rated 100 or more movies



As can be seen from this plot as well, there is a substantial variation across users as well. A negative b_u would indicate that a great movie is rated good

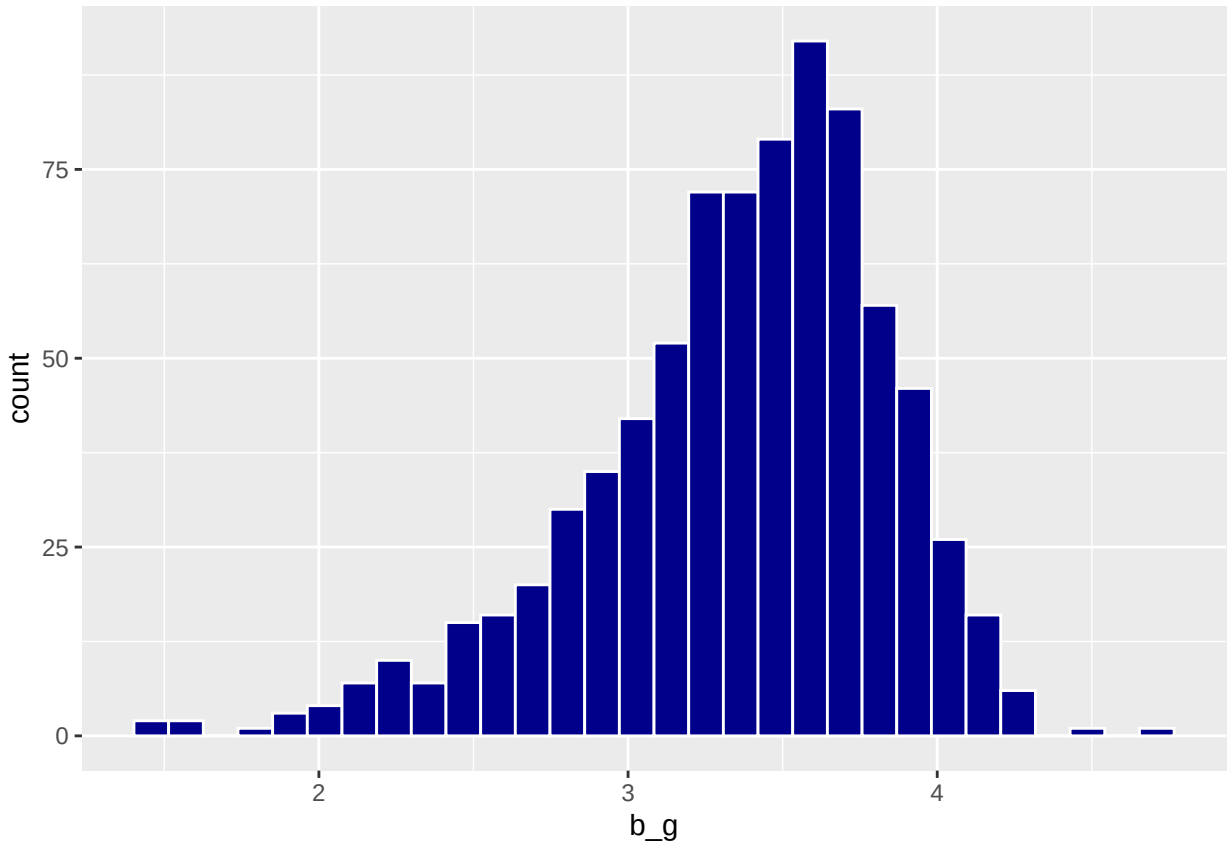
Table 4: Results

method	RMSE
Just a hunch	1.8410994
All about Average	1.0607045
The Movie Effect	0.9437144
The Movie+User Effect	0.8661625

RMSE is much improved with this method. This shows the importance of these predictors and how the inclusion of such effects can be leveraged in building a good model that can make better predictions

2.5 Method 5 - The genres effect

This method explores the effect genres bring into providing a rating. The genre field includes every genre that would apply to that movie.Let's demonstrate below if there is an evidence of a genre effect



As can be seen from the above graph some genres are rated better than others. Below is a list of best rated/worst rated genres

```
## Selecting by b_g
```

```
## # A tibble: 10 x 2
##    genres                                      b_g
##    <chr>                                     <dbl>
##  1 Animation|IMAX|Sci-Fi                      4.71
##  2 Action|Adventure|Animation|Comedy|Sci-Fi   4.5
##  3 Drama|Film-Noir|Romance                    4.31
##  4 Action|Crime|Drama|IMAX                    4.31
##  5 Animation|Children|Comedy|Crime            4.27
##  6 Film-Noir|Mystery                          4.24
##  7 Crime|Film-Noir|Mystery                    4.23
##  8 Film-Noir|Romance|Thriller                 4.21
##  9 Crime|Film-Noir|Thriller                   4.20
## 10 Crime|Mystery|Thriller                     4.20
```

```
## Selecting by b_g
```

```
## # A tibble: 10 x 2
##    genres                                        b_g
##    <chr>                                       <dbl>
##  1 Documentary|Horror                           1.46
##  2 Action|Animation|Comedy|Horror               1.5
##  3 Action|Horror|Mystery|Thriller               1.60
##  4 Comedy|Film-Noir|Thriller                    1.62
##  5 Adventure|Drama|Horror|Sci-Fi|Thriller       1.76
```

```
##  6 Action|Children|Comedy                                1.87
##  7 Action|Adventure|Children                             1.92
##  8 Adventure|Animation|Children|Fantasy|Sci-Fi           1.94
##  9 Action|Adventure|Drama|Fantasy|Sci-Fi                 2.01
## 10 Action|Adventure|Children|Comedy|Fantasy|Sci-Fi       2.04
```

Table 5: Results

method	RMSE
Just a hunch	1.8410994
All about Average	1.0607045
The Movie Effect	0.9437144
The Movie+User Effect	0.8661625
The Movie+User+Genre Effect	0.8655979

Though we got a lower RMSE than the previous method, Genre effect didn't bring in a substantial improvement. We will now proceed to employ a concept called regularization

2.6 Method 6 - Regularization

The number of users also play a role in skewing the predictions. For instance, some of the best and/or worst movies are rated by number of users, as less as one. This should be treated as noise and should not be considered in our prediction The penalty factor is generally denoted as lambda below and we will be performing cross-validation to select the best lambda We will be applying regularization on movie, user and genre effects

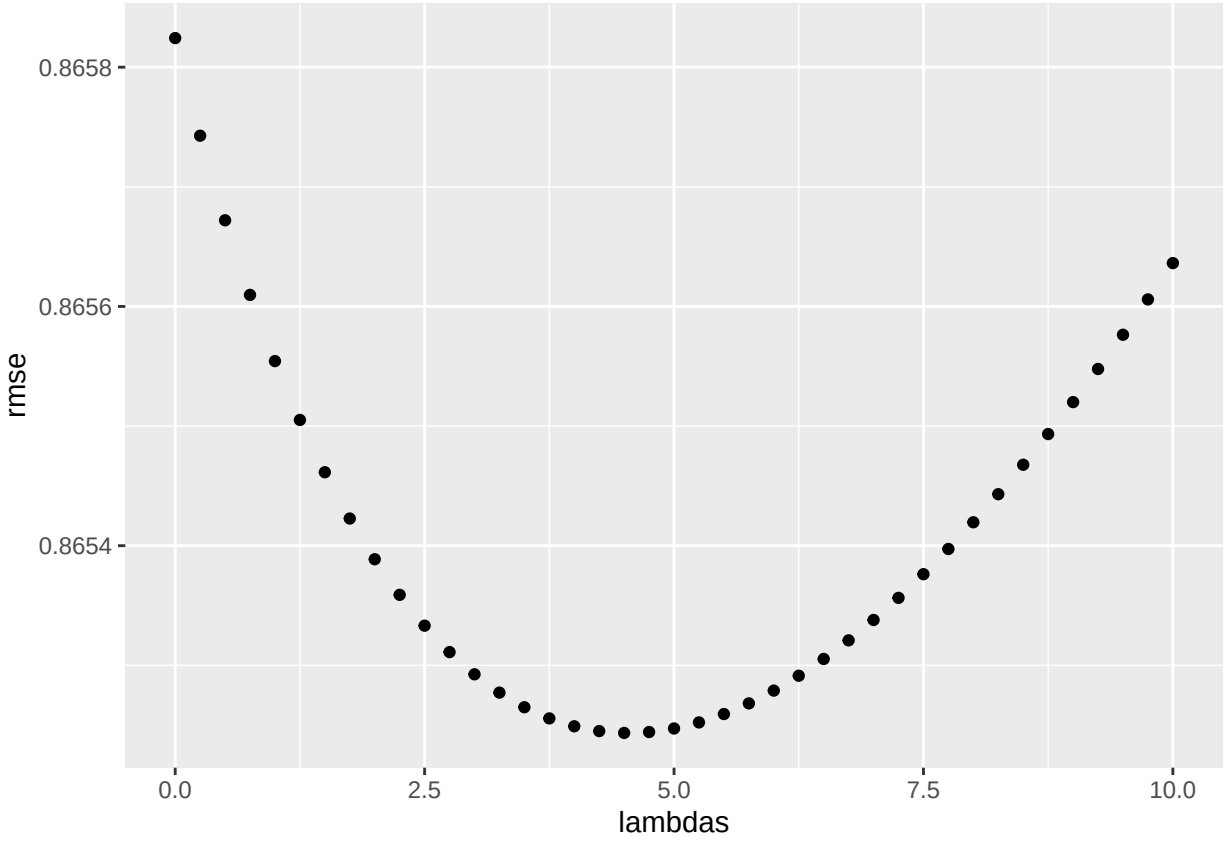


Table 6: Results

method	RMSE
Just a hunch	1.8410994
All about Average	1.0607045
The Movie Effect	0.9437144
The Movie+User Effect	0.8661625
The Movie+User+Genre Effect	0.8655979
The Movie + User + Genre Effect (Regularized)	0.8652435

Regularization as a method penalizes estimates with smaller sample sizes. This penalized estimates in fact perform better than the least square estimates. However the RMSE has not improved substantially even after applying regularization

2.7 Method 7 - Recommender system (Matrix Factorization)

Matrix factorization is a powerful technique that captures patterns in learning certain characteristics that describe users and items. These factors are stored in two matrices, P – user factors and Q – item factors.

This recosystem method is an efficient matrix factorization approach. As referred above, It is typically used to approximate an incomplete matrix using the product of two matrices in a latent space. In our case, we will be using the movie-user matrices for matrix factorization.

Table 7: Results

method	RMSE
Just a hunch	1.8410994
All about Average	1.0607045
The Movie Effect	0.9437144
The Movie+User Effect	0.8661625
The Movie+User+Genre Effect	0.8655979
The Movie + User + Genre Effect (Regularized)	0.8652435
Matrix Factorization/Recosystem	0.8338963

3 Results

We tried out various methods above and find that as we progress further with understanding the predictors in the dataset and catering to bias and illustrating concepts like regularization, the RMSE value showed a steady improvement. Method 6, where we illustrated matrix factorization/recosystem gave the best RMSE of all the methods demonstrated here. The real test is for this method to provide the desired RMSE when we test it out on the validation set The expectation is that the RMSE achieved below would meet the project objective.

Table 8: Results

method	RMSE
Just a hunch	1.8410994
All about Average	1.0607045
The Movie Effect	0.9437144
The Movie+User Effect	0.8661625
The Movie+User+Genre Effect	0.8655979

method	RMSE
The Movie + User + Genre Effect (Regularized)	0.8652435
Matrix Factorization/Recosystem	0.8338963
Final Recommendation on Validation using recosystem	0.8323383

As can be seen from the results, the reco model performed as expected and gave us a better RMSE, meeting the project objective.

From the predictions thus obtained with this model, below is the list of top 10 best and worst movies

Table 9: Best 10 Movies

title
Shawshank Redemption, The (1994)
Shawshank Redemption, The (1994)
Schindler's List (1993)
Terminator 2: Judgment Day (1991)
Usual Suspects, The (1995)
Office Space (1999)
Shawshank Redemption, The (1994)
Star Wars: Episode VI - Return of the Jedi (1983)
Silence of the Lambs, The (1991)
Forrest Gump (1994)

Table 10: Worst 10 Movies

title
Time Walker (a.k.a. Being From Another Planet) (1982)
Faces of Death 6 (1996)
Faces of Death 5 (1996)
Zombie Lake (Le Lac des morts vivants) (1981)
Armageddon (1998)
Yu-Gi-Oh! (2004)
Faces of Death 6 (1996)
Big Momma's House 2 (2006)
Armageddon (1998)
College Road Trip (2008)

4 Conclusion

The most important aspect of developing or applying a method/model is to understand the dataset and the various fields available.

As we start analysing the various predictors available in the dataset, it became clear that there are certain predictors that can be leveraged to make better predictions. RMSE is a key metric that was used to select the final model for prediction.

The final method that is selected is the recosystem/Matrix Factorization which implements the LIBMF algorithm that yielded an RMSE of .83

Tuning the Recosystem for best parameters would have further resulted in an even better RMSE, but due to

computational/memory limitations, the above solution was demonstrated just using the default parmeters

4.1 Constraints

- Owing to the large size of the dataset, some of the popular models such as linear model, glm, knn etc. couldn't be tried out on a personal laptop with limited computing power and memory
- Tuning the Recosystem for the best parameters is time/memory intensive on a personal laptop

4.2 Future Work

Matrix factorization is a very powerful method for arriving at recommendations based on past ratings. There are many implementations and approaches for matrix factorization. It is important to find the best approach for the dataset at hand, as big datasets would require efficient implementations. In our case recosystem was a good choice as it could efficienctly process the movielens datasets and deliver results with the available RAM/computing power of a personal laptop. There is however future scope to explore and practice other competitive models such as collaborative filtering and tensorflow recommendation systems.

5 References

- Harvardx Data Science Professional certificate **(Instructor: Rafael Irizarry)**
- Matrix Factorization
- rdocumentation
- The Comprehensive R Archive Network
- Netflix Challenge