Documentation on MovieLens Recommendation System

Ringgold P. Atienza

August 24, 2022

Introduction

This documentation provides an overview of my proposed model for an improved movie recommendation system using **parallel matrix factorization with stochastic gradient descent** from the previous benchmark model, the **normalization of global effects**. The normalization of global effects uses the baseline predictors to cater to the problem of user and item biases, that is, the systematic tendencies of some users to give higher ratings than others and for some movies to receive higher ratings than others. Koren and Bell (2008), the Netflix Recommendation System Challenge grand-prize winner, highlighted the effectiveness of humble baseline predictors that capture the data's main effects [1]. Accordingly, while most literature mostly concentrates on sophisticated algorithms, accurate treatment of the main effects is at least as significant as coming up with modeling breakthroughs. On the other hand, matrix factorization is a form of collaborative filtering which focuses on generating latent structure of data by mapping both users and items into a latent feature space. Effectually saying, matrix factorization works with given only the ratings from users and items.

For this project, I used the residual mean square error (RMSE) as the metric to optimize the model and compare the benchmark model against the new proposed model. RMSE is the most popular metric for evaluating movie recommendation systems. It was also used in the popular Netflix Prize Challenge to find the best movie recommendation system. To put it simply, RMSE answers, "How far off should we expect our model to be on its next prediction?" [2]. Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). The RMSE is then defined as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{u,i}\left(\hat{y}_{u,i} - y_{u,i}\right)^2}$$

with N being the number of user-movie combinations and the sum occurring over all these combinations. The objective of this project is to proposed an recommendation system that beats the previous benchmark model in terms of RMSE.

Data Preparation

The MovieLens 10M dataset tags ratings for 10,000 movies by 72,000 users. The original data only provides the user ID, movie ID, rating, timestamp of the rating, movie title, and movie genres. Mutations were made to the dataset to flesh out other relevant variables such as the movie's year of release, age of the movie upon the rating, user frequency of rating, and movie frequency of being rated. The final dataset for training includes the following variables:

```
##  [1] "userId"     "movieId"    "rating"     "title"      "movieYear"
##  [6] "genres"     "reviewDate" "reviewYear" "movieAge"   "userFreq"
## [11] "movieFreq"
```

I also checked for the white entries, error encoding, and missing values. The data set is clean and has no missing values. The 10M Movielens data set is partitioned into test (20%) and train (80%) sets. The test set was used for the final-hold-out test to evaluate the performance of the final tuned model.

Data Visualization

The ratings of the movies start with the lowest value of 0.5 up to the highest value of 5.0. The distribution of the total ratings is shown in Figure 2. The distribution tells us that most of the raters rate the movies at 4.0, and very few rate movies at 0.5. It also shows that raters tend to rate movies the whole stars as compared to half stars.

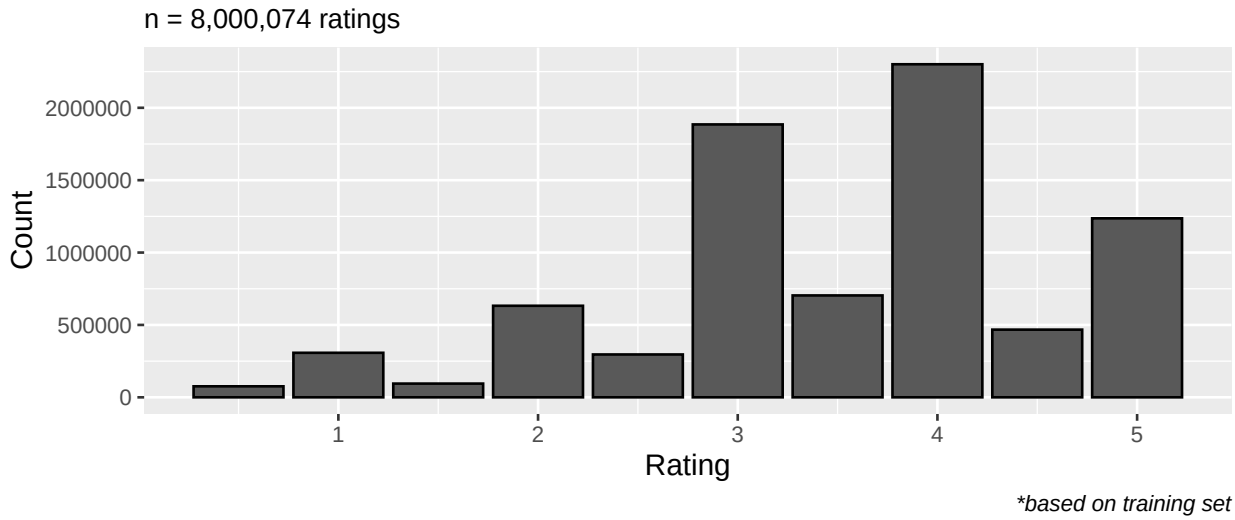

Figure 1. Total Ratings Distribution

Most of the average ratings by MovieID are at 3.0 and 3.5, as shown in Figure 3. Also, average movie ratings are between 2.84 (10% percentile) and 3.85 (90% percentile). The result indicates there are very few movies rated averagely very low (less than 10% percentile) and very high (higher than 90 % percentile).

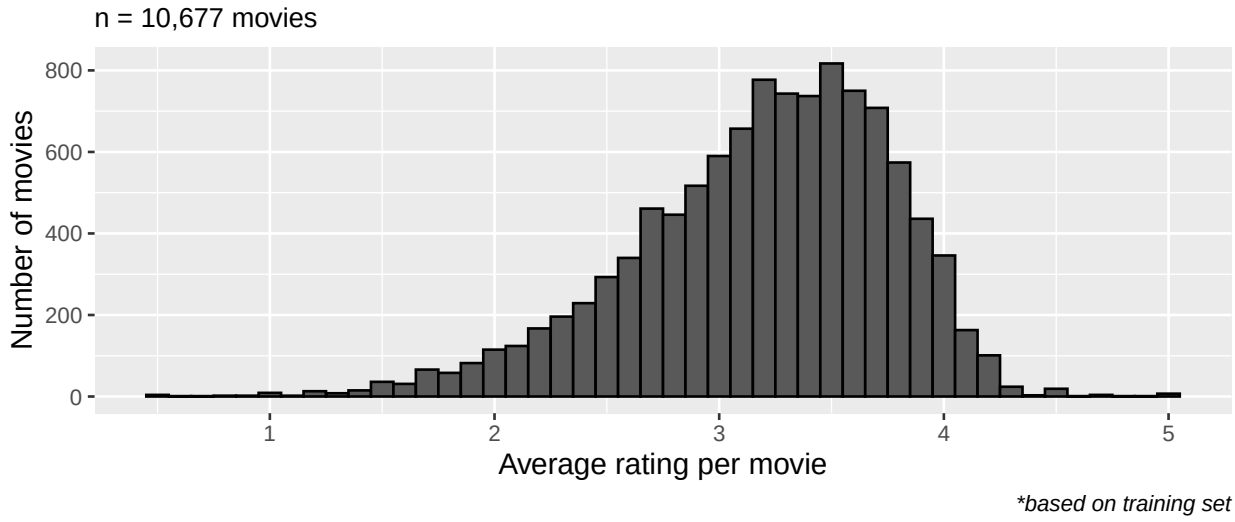

Figure 2. Distribution of Average Ratings by Movie Title

In Figure 4, most of the average ratings by UserID are at 4.0 and 3.5, which indicates that many users are inclined to rate movies at 4.0 and 3.5. Also, average movie ratings are between 3.08 (10% percentile) and 4.13 (90% percentile). The result indicates there are very few users rate very low (less than 10% percentile) and very high (higher than 90 % percentile).

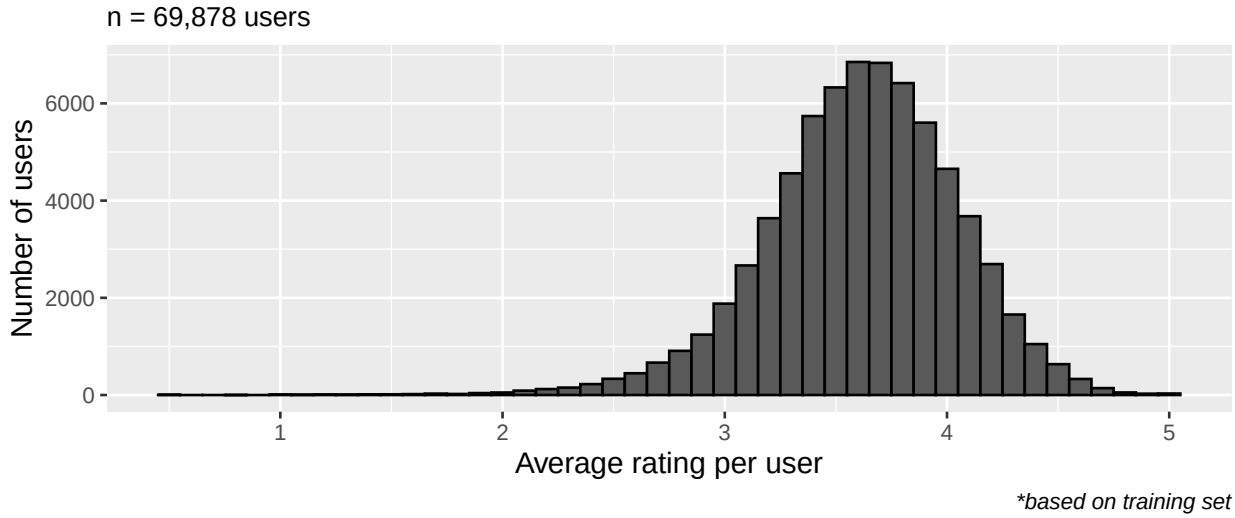

Figure 3. Distribution of Average Ratings by User ID

In Figure 5, movies released on 1995 have the highest rating reviews while the fewest is 1917. The total rated movies are between 1973 (10% percentile) and 2002 (90% percentile). The result indicates that few movies were rated from 1917 to 1972 and 2002 to 2008.

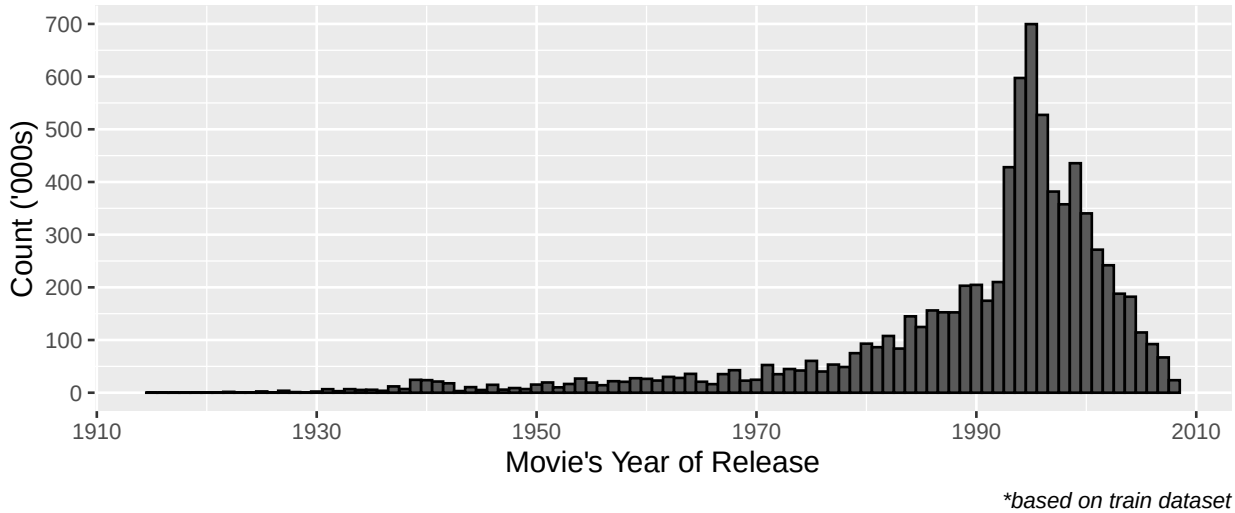

Figure 4. Distribution of Average Ratings by Movie Title

Results in Figure 6 showed that users rate movies more frequently after a year of the movie's release. Users seldom rate movies released during its year (10% percentile) and over 30 years (90% percentile). There were movies rated before the release date because some users rate movies based on the pre-screening of the movie.

Figure 5. Distribution of Average Ratings by Movie' Title's Age

In the MovieLens dataset, most movies are assigned to more than one genre. With this, we have to separate these genres aggregated in one feature to generate the distribution of the most rated genre in the dataset. The result in Figure 7 shows that Drama has the most number of rated movies.

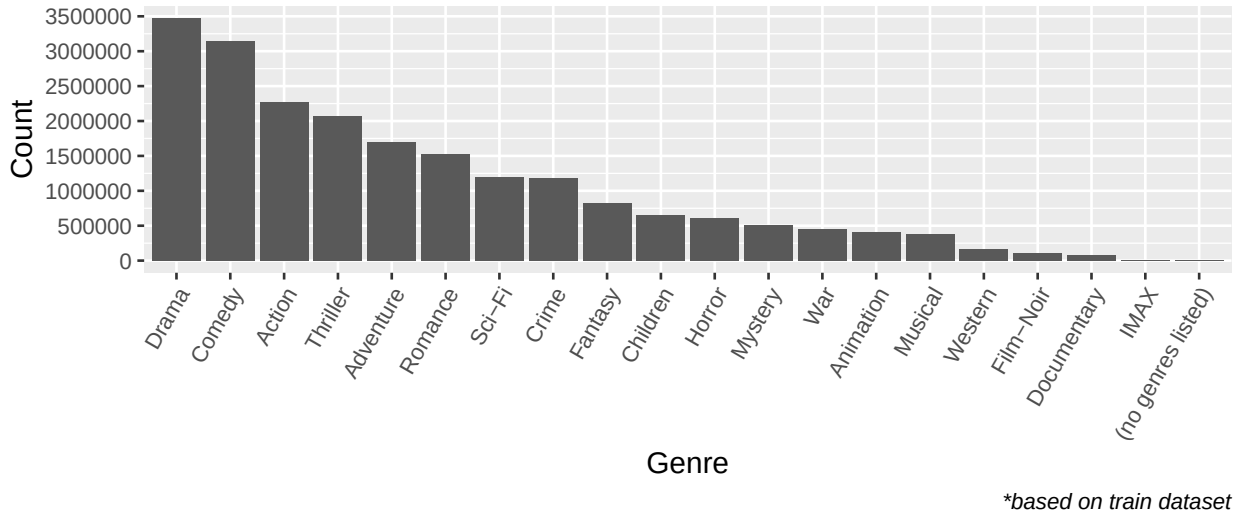

Figure 6. Distribution of Ratings by Genre

Furthermore, Film-Noir showed the highest average rating and Horror the lowest of all genres, as shown in Figure 7. On the other hand, some movie titles have no genre tag on them. Those movie titles also received a relatively low average rating from users.

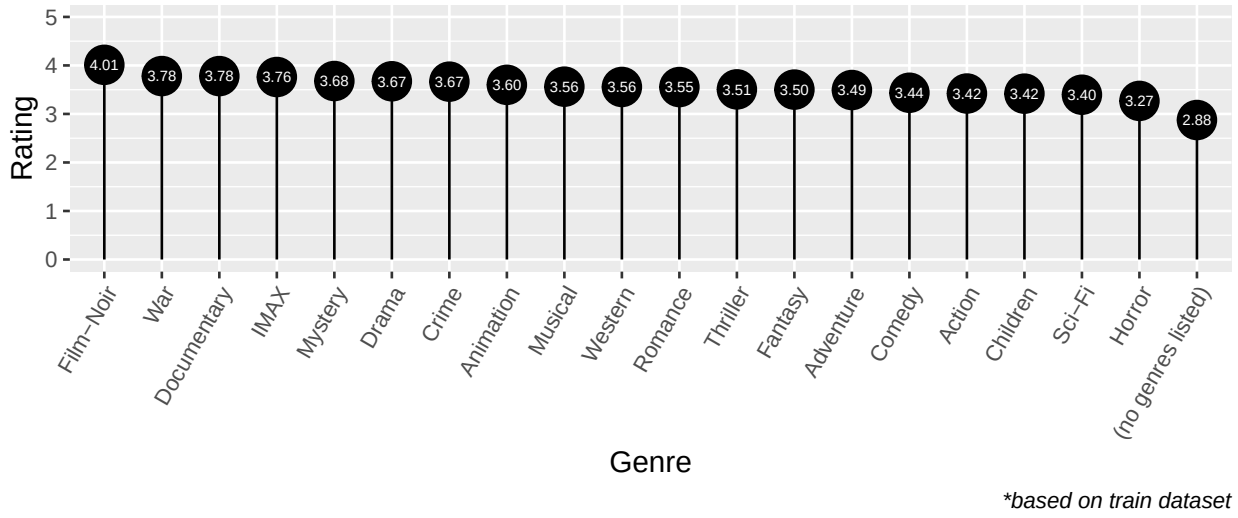

Figure 7. Average Rating by Genre

Benchmark Model: Baseline Predictors

The baseline predictors for the normalization of global effects encapsulate the effects of user and item biases i.e., systematic tendencies for some users to give higher ratings than others and for some items to receive higher ratings than others, temporal effects, frequency effects, and genre effects. In these baseline predictors, I introduced the genre effects, which were not considered in the BellKor Solution to the Netflix Grand Prize (Koren, 2009). The predicted ratings of a specific movie are composed of several parts. To provide an overview of baseline predictors, we have:

- A **baseline rating** (mean overall user-movie rating): $b_{ui} = \mu$
- A **user-specific effect** (e.g., tendency of a user to rate movies lower than the average user): $b_{ui} = \mu + b_u$
- A **movie-specific effect** (e.g., a movie is great so its ratings are higher than the average): $b_{ui} = \mu + b_u + b_i$

In other words, we can decompose a 3.5-star rating of a specific movie into, e.g.,: 3.5 = [3.1 (the base line rating) - 0.5 (the user-specific effect) + 0.9 (the movie-specific effect)].

To gain more accurate estimation of b_u and b_i, the regularization term, $\lambda_3(\sum_u b_u^2 + \sum_i b_i^2$, avoids overfitting by penalizing the magnitudes of parameters. Using the least square problem, we can solve efficiently in finding the appropriate λ for the model. A way to estimate the parameters is by decoupling the calculation of b_i from the calculation of the b_u. First, for the item i, we find the tuning parameter for λ, using cross-validation:

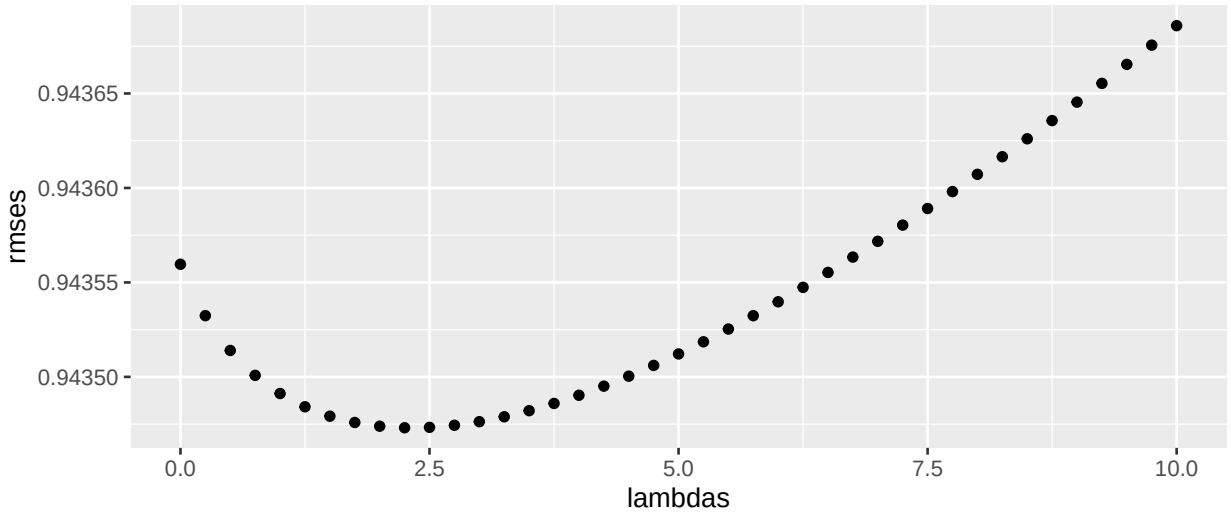

Figure 8. Penalized Least-Square Cross Validation for Movie Effect.

```
## The results indicates that the lambda for movie effect is = 2.25
```

First for the user u, we find the tuning parameter for λ, using cross-validation:

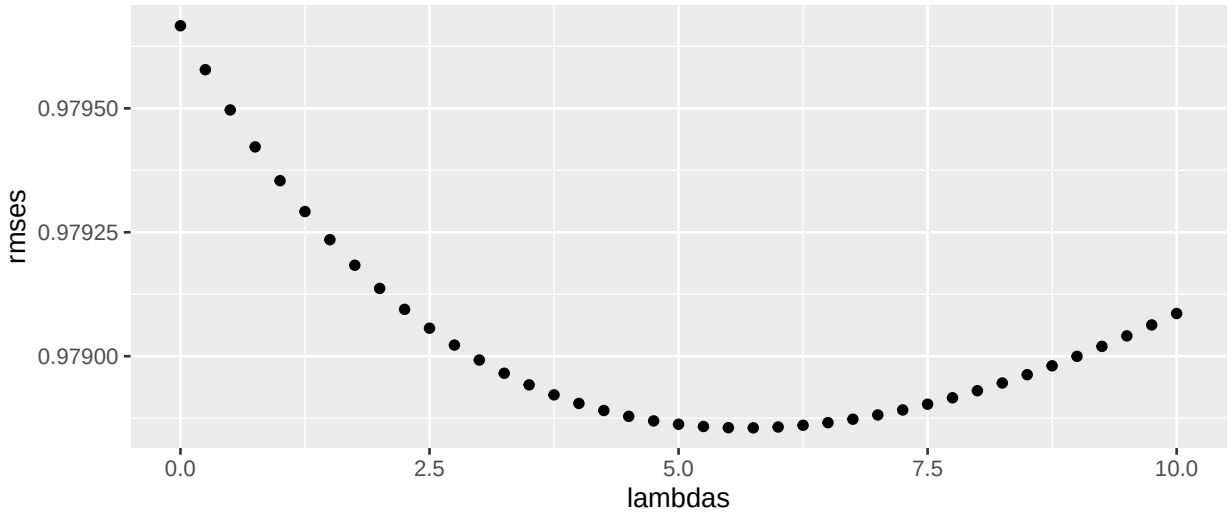

Figure 9. Penalized Least-Square Cross Validation for User Effect.

```
## The results indicates that the lambda for user effect is = 5.75
```

Since the tuning parameter λ was calculated, we can now run the baseline predictors and the results is shown in the table below:

Stepwise	RMSE
Global Average + Movie effect + User Effect	0.8658615

The predictions of the normalized global effects already provide high accuracy for a movie recommendation system. However, it can still be improved by including baseline predictions [3]:

- **Temporal effects**: A factor that allows the user's rating to depend on the year (or days) the movie since the user's first rating and the number years (or days) the movie's first rated by anyone.
- **Frequency effects**: A factor that allows user's rating to depend on the number of people who have rated the movie (movie's popularity can affect the user's rating) and the number of ratings the user has rated (e.g., some user become harsher critic over time).
- **Genre effect**: A factor that allows the user's rating to depend on the genre of the movie.

Before tuning the baseline predictors, 20% of the train set was partitioned as the validation set.

Stepwise 1: Movie Frequency Effect

I utilized the stepwise method to model the algorithm for the baseline predictors using RMSE as a metric for finding the best model. The stepwise method then considered the movie's age effect (β_a), movie year of release affect (β_y), user frequency effect (β_f), movie frequency effect (β_m), and genre effect (β_g) for the previous baseline predictors: global average + movie effect + user effect ($\mu + \beta_i + \beta_u$).

As shown in Figure 10, the previous baseline predictors with movie frequency effect ($\mu+\beta_i+\beta_u+\beta_m$) produced minimal loss compared to other variables. The baseline with the movie frequency effect ($\mu + \beta_i + \beta_u + \beta_m$) was then chosen as the new baseline for the next stepwise model.

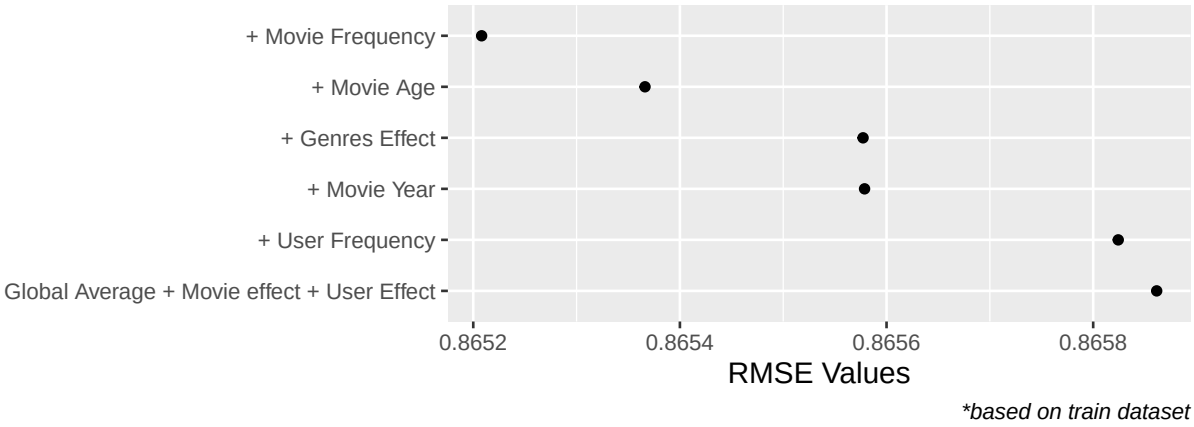

Figure 10. 1st Stepwise: Least-Square Method.

Stepwise 2: Movie Age Effect

As shown in Figure 11, the previous baseline with movie age effect ($\mu+\beta_i+\beta_u+\beta_m+\beta_a$) produced minimal loss compared to other variables. The baseline with the movie age effect was then chosen as the new baseline for the 3rd stepwise model. The next stepwise reiteration then considered the movie year affect (β_y), genre effect (β_g), and user frequency effect (β_f).

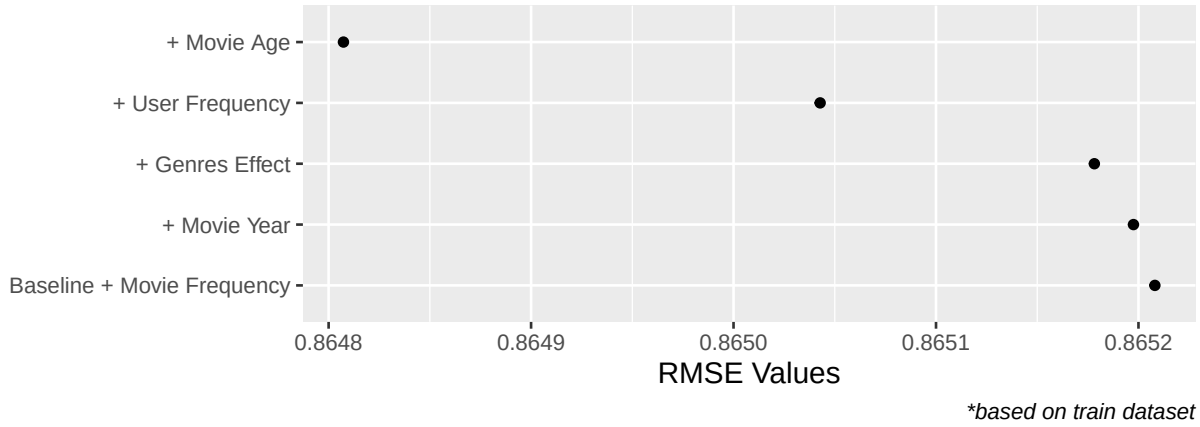

Figure 11. 2nd Stepwise: Least-Square Method.

Stepwise 3: User Frequency Effect

As shown in Figure 12, the previous baseline with user frequency effect ($\mu+\beta_i+\beta_u+\beta_m+\beta_a+\beta_f$) produced minimal loss compared to genres. The global average + movie effect + user effect + movie frequency effect + movie age effect + user frequency effect ($\mu + \beta_i + \beta_u + \beta_m + \beta_a + \beta_f$) became the baseline for the 4th stepwise reiteration. The stepwise method then considered the genre effect (β_g), and movie year effect (β_y).

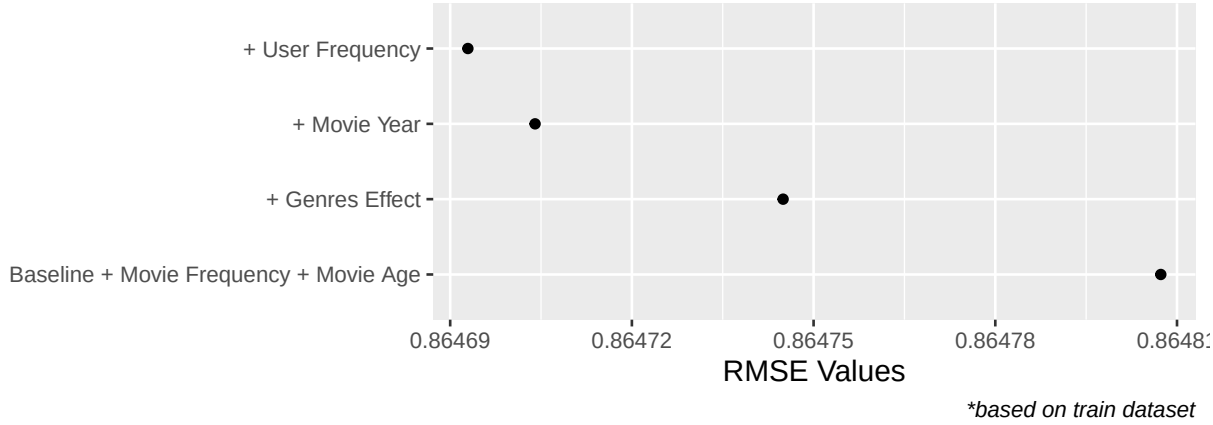

Figure 12. 3rd Stepwise: Least-Square Method.

Stepwise 4: Movie Year Effect

As shown in Figure 13, the previous baseline with user frequency effect ($\mu + \beta_i + \beta_u + \beta_m + \beta_a + \beta_f + \beta_y$) produced minimal loss compared to genres. The previous baseline with the movie year effect was then chosen as the new baseline for the next stepwise reiteration.

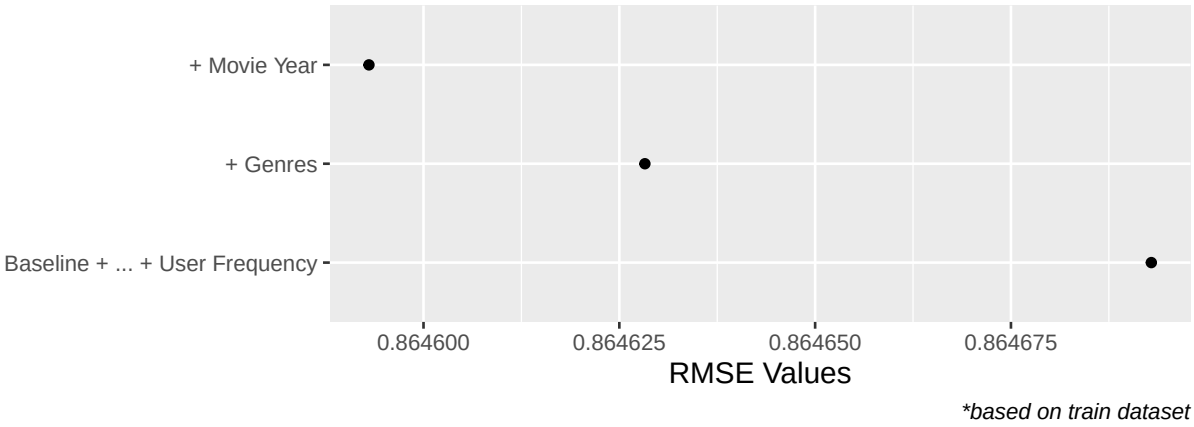

Figure 13. 4th Stepwise: Least-Square Method.

Stepwise 5: Genre Effect

The genre effect provides an improvement on the training data (see Figure 14); therefore, we now have the final model for our baseline prediction: $(\mu + \beta_i + \beta_u + \beta_m + \beta_a + \beta_u + \beta_y + \beta_g)$.

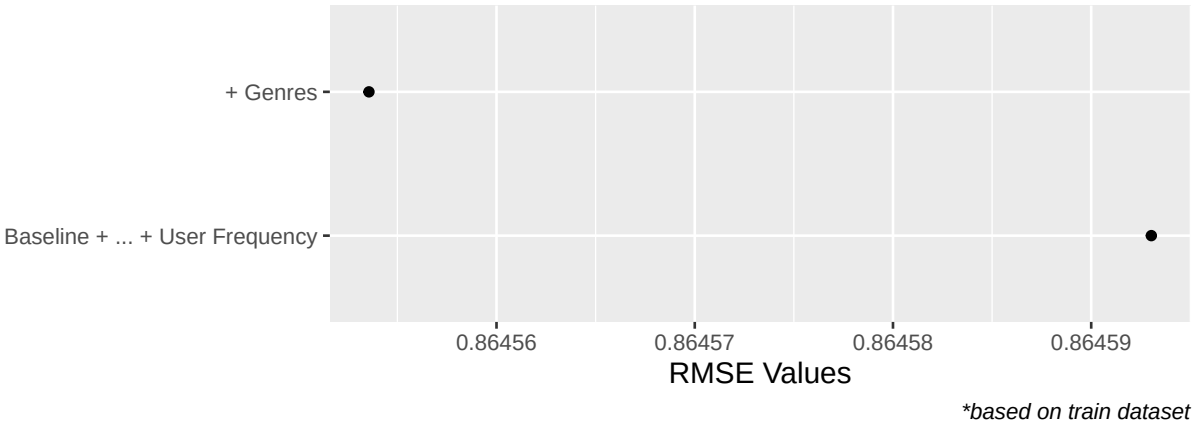

Figure 14. 5th Stepwise: Least-Square Method.

Final Hold-Out Test for Baseline Prediction (Normalization of Globel Effects)

The result of the final model is shown in the table below. An RMSE below 0.8670 is a good model for the 10M Movielens dataset [4]. The baseline prediction result was set as the benchmark for the proposed model.

Model	RMSE
Baseline Predictor	0.8632295

Parallel Matrix Factorization with Stochastic Gradient Descent

The proposed model is based on the library package developed by Chin et. al (2016) in their open-source tool the **LIBMF: A Matrix-factorization Library for Recommender Systems** [5]. The LIMBF package provides solution for real-valued matrix factorization such as the 10M Movielens recommendation with improved performance speed and efficiency through using parallel computation of multi-core machine in training for the stochastic gradient descent.

Before we can train for the model, it is important to do parameter tuning. The goal of parameter tuning is to find the optimal parameters to find the minimum value of RMSE or the global minimum of stochastic gradient descent. To find the global minimum, we need to set the dimension of the latent factors, regularization parameter, learning rate, cross-validation, and the number of iterations. After several iterations on different set-ups for tuning, the final parameters are the following:

- Learning rates: learning rate is set to .1.
- Regularization: L1 set at .1.
- Cross-validation (nfold): cross-validation is set to 5-folds.
- Iterations (niter): the number of iterations is set to 20.
- Parallel computing: the number of threads is set to 4.

The results indicates that the proposed model is better than the benchmark as shown in the Table below. The parallel matrix factorization with stochastic gradient descent result shows better RMSE compared to the baseline predictor by 8.92%.

Model	RMSE
Baseline Predictor	0.8632295
Parallel Matrix Factorization	0.7928365
Difference	-0.0703930

Show Recommendation

This section shows the recommendation results of the proposed model for a specific user. The user profile (e.g., User ID = 2022) and the number of recommendations (e.g., 15) were set. The first part shows the user's top 10 best and worst rated movies to show the user's movie preference. The user's top recommendations were then offered. Finally, recommendation for the top global movie and an example of the top genre-specific movie is also provided.

```
#Set N, User ID
current_user <- 2022 #user must have at least 1 rating data
```

```
#set N, number of movie recommendation
n_recom <- 15
```

Best Rated Movies by the User

Movie ID	Title	Year Released
34	Babe	1995
43	Restoration	1995
80	White Balloon, The (Badkonake sefid)	1995

(continued)

Movie ID	Title	Year Released
94	Beautiful Girls	1996
213	Burnt by the Sun (Utomlyonnye solntsem)	1994
215	Before Sunrise	1995
235	Ed Wood	1994
265	Like Water for Chocolate (Como agua para chocolate)	1992
273	Frankenstein (Mary Shelley's Frankenstein)	1994
306	Three Colors: Red (Trois couleurs: Rouge)	1994

Worst Rated Movies by the User

Movie ID	Title	Year Released
194	Smoke	1995
1059	William Shakespeare's Romeo + Juliet	1996
1416	Evita	1996
1834	Spanish Prisoner, The	1997
2297	What Dreams May Come	1998
3252	Scent of a Woman	1992
3257	Bodyguard, The	1992
1562	Batman & Robin	1997
1633	Ulee's Gold	1997
1639	Chasing Amy	1997

Top Recommendation for the User

Movie ID	Title	Year Released
4348	Whatever Happened to Harold Smith?	1999
7585	Summertime	1955
1900	Children of Heaven, The (Bacheha-Ye Aseman)	1997
5911	Urgh! A Music War	1981
25975	Life of Oharu, The (Saikaku ichidai onna)	1952
8484	Human Condition I, The (Ningen no joken I)	1959
4376	Down From the Mountain	2000
4454	More	1998
7388	Brother Sun, Sister Moon (Fratello sole, sorella luna)	1972
7456	Valentin (Valentín)	2002
1780	Ayn Rand: A Sense of Life	1997
2512	Ballad of Narayama, The (Narayama bushiko)	1983
1131	Jean de Florette	1986
1177	Enchanted April	1992
6896	Shoah	1985

Top Recommendation for the User (Movies already rated by the user are excluded)

Movie ID	Title	Year Released
4348	Whatever Happened to Harold Smith?	1999
7585	Summertime	1955
1900	Children of Heaven, The (Bacheha-Ye Aseman)	1997
5911	Urgh! A Music War	1981
25975	Life of Oharu, The (Saikaku ichidai onna)	1952
8484	Human Condition I, The (Ningen no joken I)	1959
4376	Down From the Mountain	2000
4454	More	1998
7388	Brother Sun, Sister Moon (Fratello sole, sorella luna)	1972
7456	Valentin (Valentín)	2002
1780	Ayn Rand: A Sense of Life	1997
2512	Ballad of Narayama, The (Narayama bushiko)	1983
1131	Jean de Florette	1986
6896	Shoah	1985
7106	Black and White in Color (Noirs et blancs en couleur)	1976

Global Top Rated Movies

Movie ID	Title	Year Released
26048	Human Condition II, The (Ningen no joken II)	1959
5194	Who's Singin' Over There? (a.k.a. Who Sings Over There) (Ko to tamo peva)	1980
58185	Please Vote for Me	2007
50477	Testament of Orpheus, The (Testament d'Orphée)	1960
7452	Mickey	2003
63179	Tokyo!	2008
63808	Class, The (Entre les Murs)	2008
3226	Hellhounds on My Trail	1999
42783	Shadows of Forgotten Ancestors	1964
26073	Human Condition III, The (Ningen no joken III)	1961
64275	Blue Light, The (Das Blaue Licht)	1932
61695	Ladrones	2007
64418	Man Named Pearl, A	2006
33264	Satan's Tango (Sátántangó)	1994
51209	Fighting Elegy (Kenka erejii)	1966

Global Top Rated Movies by Genre (Comedy)

Movie ID	Title	Year Released
5194	Who's Singin' Over There? (a.k.a. Who Sings Over There) (Ko to tamo peva)	1980
5849	I'm Starting From Three (Ricomincio da Tre)	1981
3022	General, The	1927
1193	One Flew Over the Cuckoo's Nest	1975
1148	Wallace & Gromit: The Wrong Trousers	1993
720	Wallace & Gromit: The Best of Aardman Animation	1996
4973	Amelie (Fabuleux destin d'Amélie Poulain, Le)	2001
53355	Sun Alley (Sonnenallee)	1999
745	Wallace & Gromit: A Close Shave	1995

(continued)

Movie ID	Title	Year Released
950	Thin Man, The	1934
3030	Yojimbo	1961
750	Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb	1964
3307	City Lights	1931
51209	Fighting Elegy (Kenka erejii)	1966
64408	Sun Shines Bright, The	1953

References

[1] https://www2.seas.gwu.edu/~simhaweb/champalg/cf/papers/KorenBellKor2009.pdf

[2] https://towardsdatascience.com/what-does-rmse-really-mean-806b65f2e48e

[3] http://blog.echen.me/2011/10/24/winning-the-netflix-prize-a-summary/

[4] https://www.researchgate.net/publication/303698729_A_Neural_Autoregressive_Approach_to_Collaborative_Filtering

[5] https://www.csie.ntu.edu.tw/~cjlin/papers/libmf/libmf_open_source.pdf