Capstone project report: MovieLens Recommender System

Osman CALISIR

Requirements:

- Completing the previous courses with a verified account. Here are the previous courses:
    1. R Basics
    2. Visualization
    3. Probability
    4. Inference and Modeling
    5. Productivity Tools
    6. Wrangling
    7. Linear Regression
    8. Machine Learning
- Having access to the dataset

Summary:

The purpose of this project is creating a recommender system.

After an initial data exploration, the recommender systems built on this dataset are evaluated and chosen based on the RMSE which stands for Root Mean Squared Error and it should be lower than **0.87750**.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n} e_t^2}$$

For accomplishing this goal, the **Regularized Movie + User + Genre Model** is capable to reach a RMSE of **0.8628**, which is fine.

Some important information:

• Number of rows in dataset:	9000055
• Number of columns in dataset:	6
• Number of different movies given in dataset:	10677
• Number of different users given in dataset:	69878
• Number of movies rated in dataset by genres:	
▪ Drama:	3910127
▪ Comedy:	3540930
▪ Thriller:	2325899
▪ Romance:	1712100
• The movie that has the greatest number of ratings:	Pulp Fiction
• Most given ratings in order from most to least:	4, 3, 5, 3.5, 2

The features/variables/columns in both datasets are six:

- **userId** `<integer>` which contains the unique identification number for each user.
- **movieId** `<numeric>` which contains the unique identification number for each movie.
- **rating** `<numeric>` which contains the rating of one movie by one user.
- **timestamp** `<integer>` which contains the timestamp of specific rating provided by one user.
- **title** `<character>` which contains the title of each movie including the year of the release.
- **genres** `<character>` which contains a list of pipe-separated of genre of each movie.

Here, is the first 6 rows of edx dataset

userId	*movieId*	*rating*	*timestamp*	*title*	*genres*
1	122	5	838985046	Boomerang (1992)	Comedy|Romance
1	185	5	838983525	Net, The (1995)	Action|Crime|Thriller
1	231	5	838983392	Dumb & Dumber (1994)	Comedy
1	292	5	838983421	Outbreak (1995)	Action|Drama|Sci-Fi|Thriller
1	316	5	838983392	Outbreak (1995)	Action|Adventure|Sci-Fi
1	329	5	838983392	Star Trek: Generations (1994)	Action|Adventure|Drama|Sci-Fi

Dataset Pre-Processing and Feature Engineering

After the initial data exploration, we notice that the genres are created as pipe-separated values. It is necessary to extract them to make it more consistent, robust, and precise. We also see that the title contains the year where the movie war released and this it could be necessary to predict the movie rating. Finally, we can extract the year and the month for each rating.

The pre-processing phase is composed by these steps:

1. Convert timestamp to a human readable date format.
2. Extract the month and the year from the date.
3. Extract the release year for each movie from the title.
4. Separate each genre from the pipe-separated value. It increases the size of both datasets.

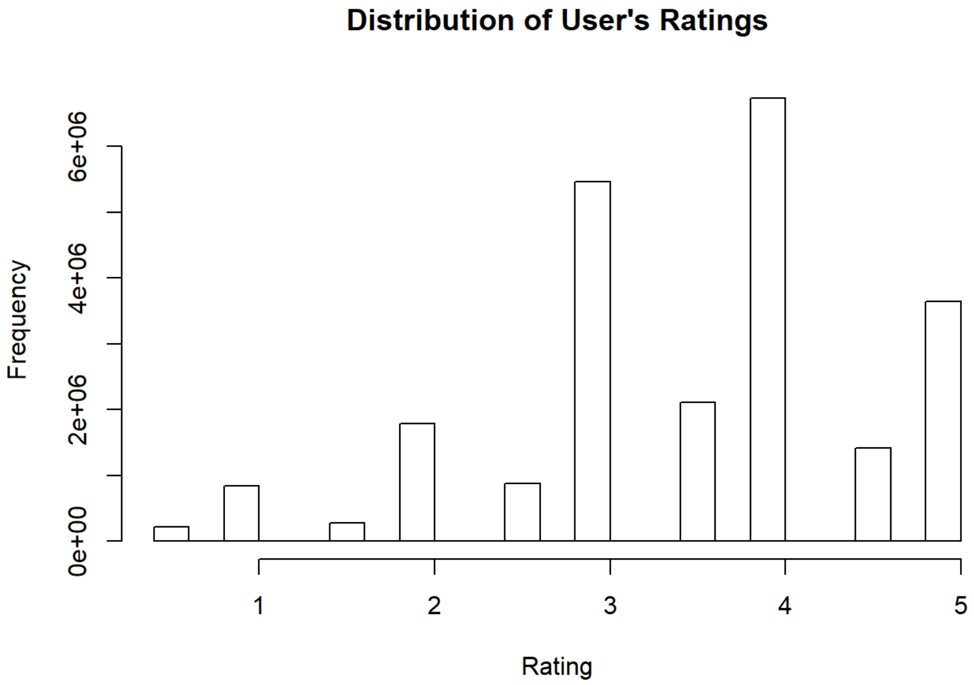
After preprocessing the data, *edx* dataset looks like this:

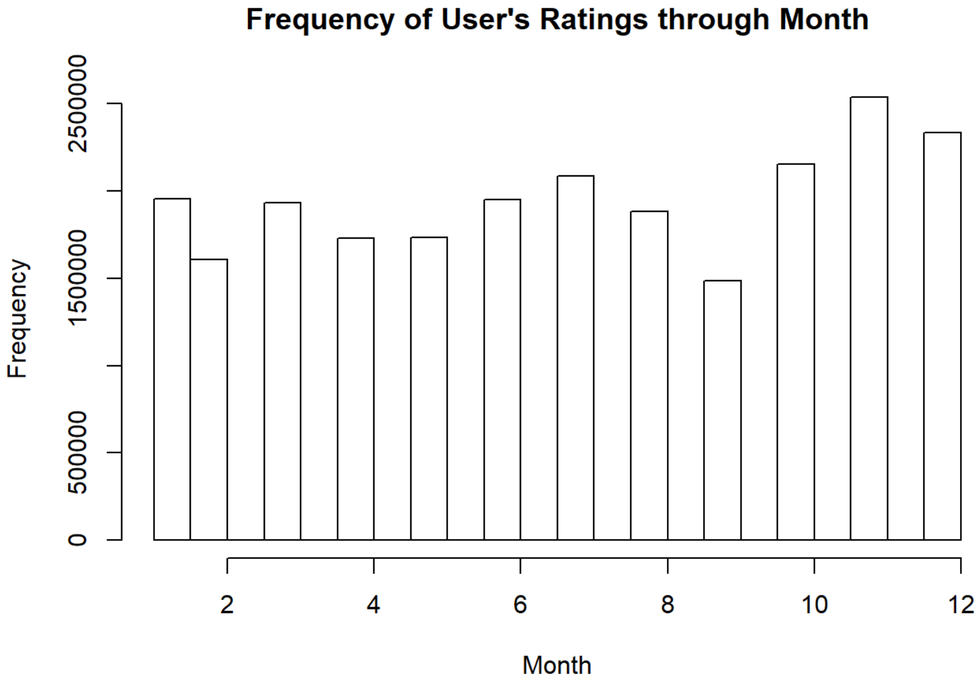
userId	*movieId*	*rating*	*title*	*genre*	*release*	*yearOfRate*	*monthOfRate*
1	122	5	Boomerang	Comedy	1992	1996	8
1	122	5	Boomerang	Romance	1992	1996	8
1	185	5	Net, The	Action	1995	1996	8
1	185	5	Net, The	Crime	1995	1996	8
1	185	5	Net, The	Thriller	1995	1996	8
1	231	5	Dumb & Dumber	Comedy	1994	1996	8

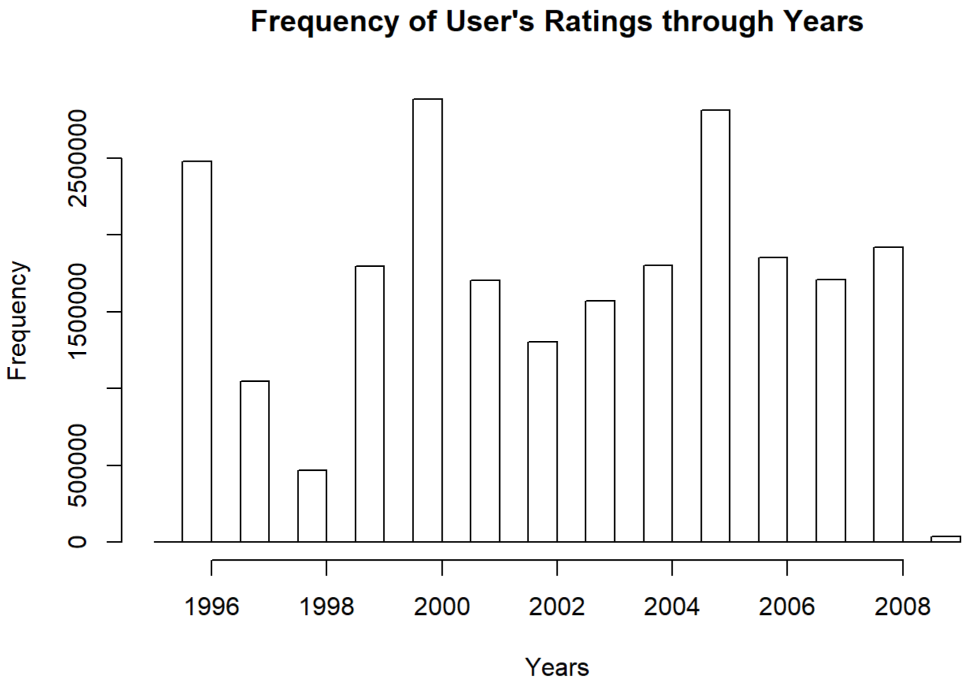
Rating Distribution

Overview of Rating Distribution

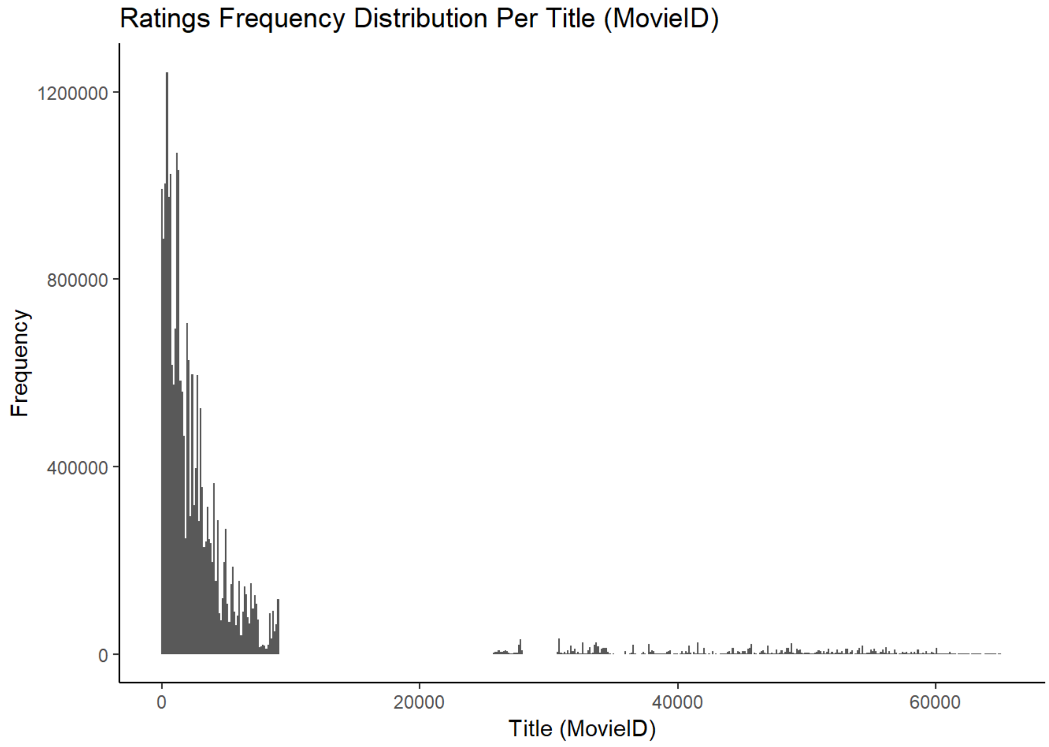
According to the histogram below, it shows that there are a small number of negative votes (below 3).



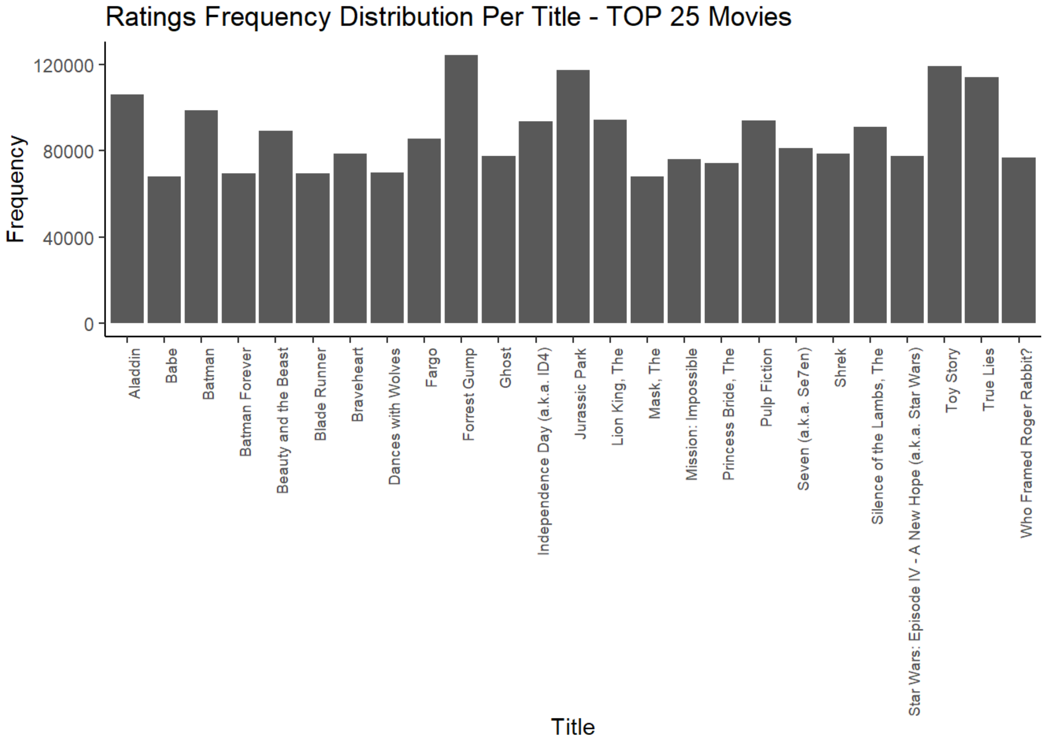




Numbers of Ratings per Movie



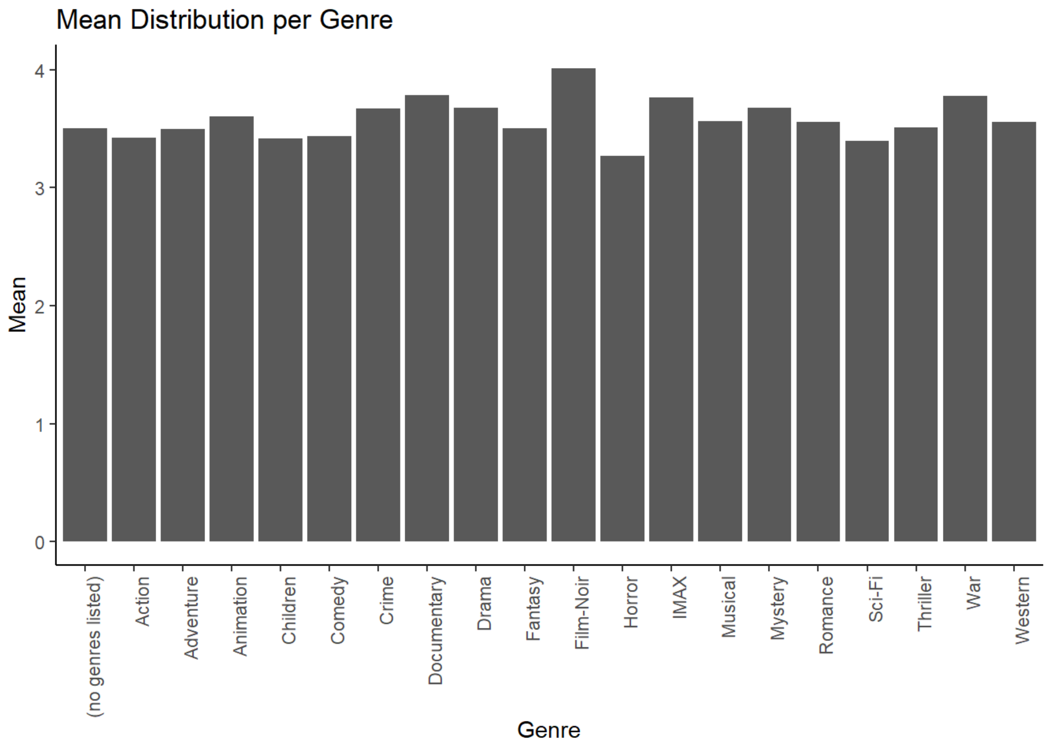
Top Rated Movies



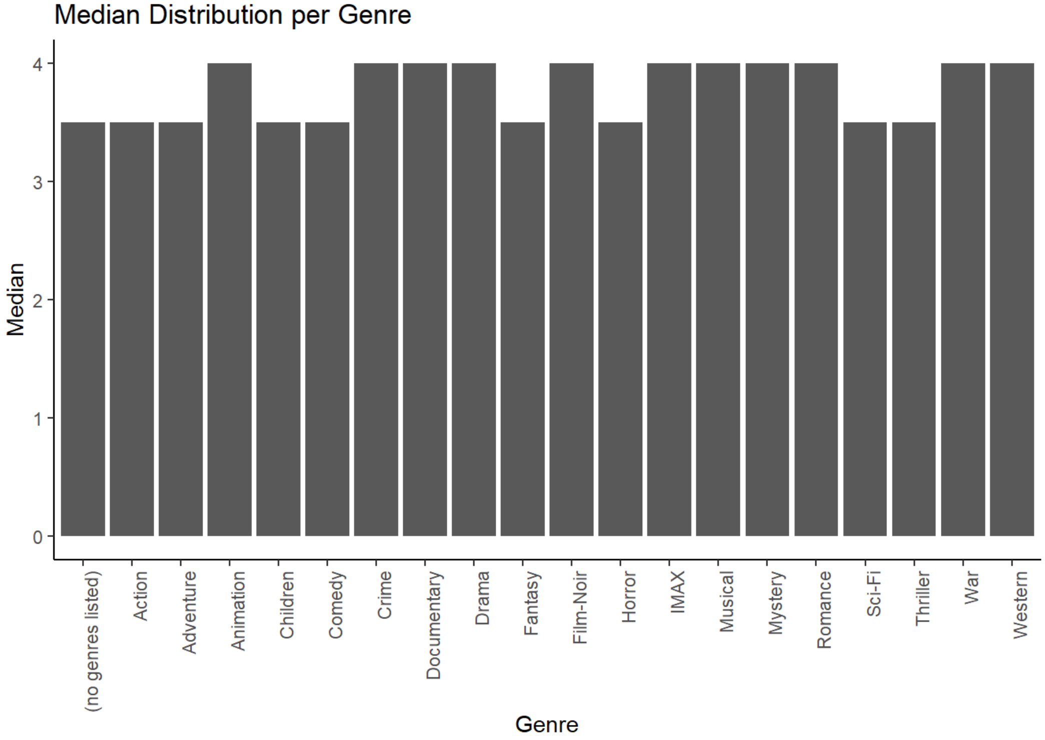
Title	*Count*
Forrest Gump	124304
Toy Story	119130
Jurassic Park	117164
True Lies	113930
Aladdin	106070
Batman	98656
Lion King, The	94435
Pulp Fiction	94008
Independence Day	93440
Silence of the Lambs	90840
Beauty and Beast	89315
Fargo	85480
Seven	81084
Braveheart	78774
Shrek	78564
Star Wars: IV	77427
Ghost	77335
Who Framed Roger Rabbit	76825

Mission: Impossible	75876
Princess Bride, The	74045
Dances with Wolves	69936
Blade Runner	69615
Batman Forever	69432
Mask, The	68200
Babe	68140

Mean Distribution per Genre



Median Distribution per Genre



Analysis - Model Building and Evaluation

Naive Baseline Model

The simplest model that someone can build, is a Naive Model that predict ALWAYS the mean. In this case, the mean is approximately 3.5.

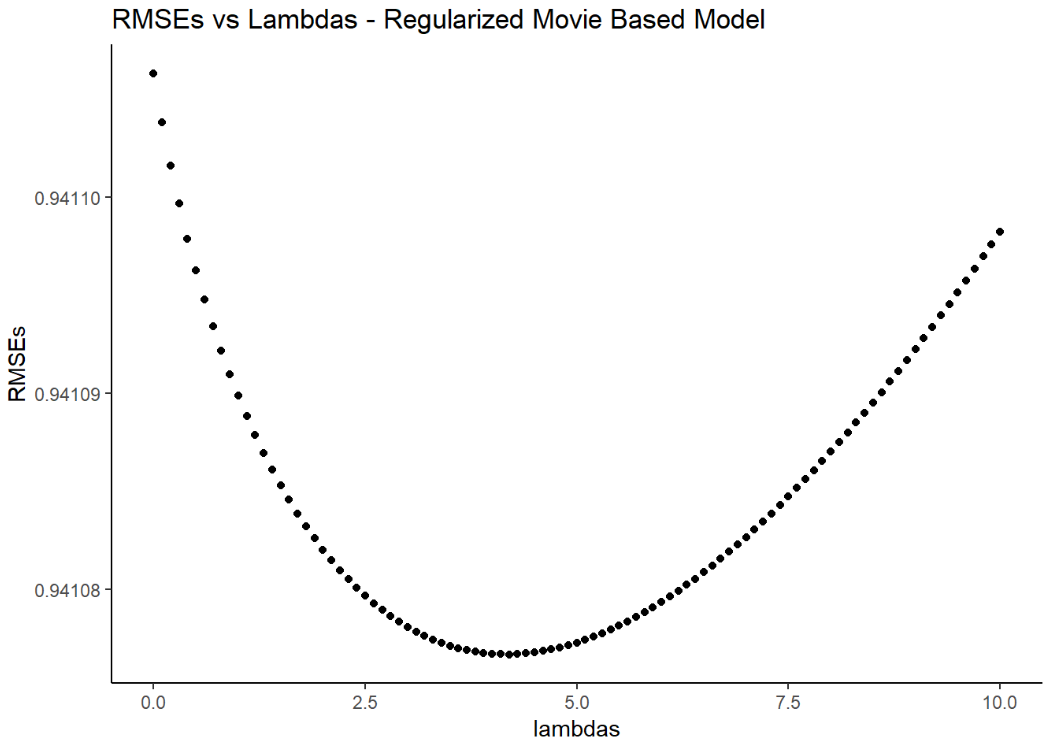
```
## [1] "The mean is: 3.52700364195256"
```

Naive Mean-Baseline Model

The formula used is:

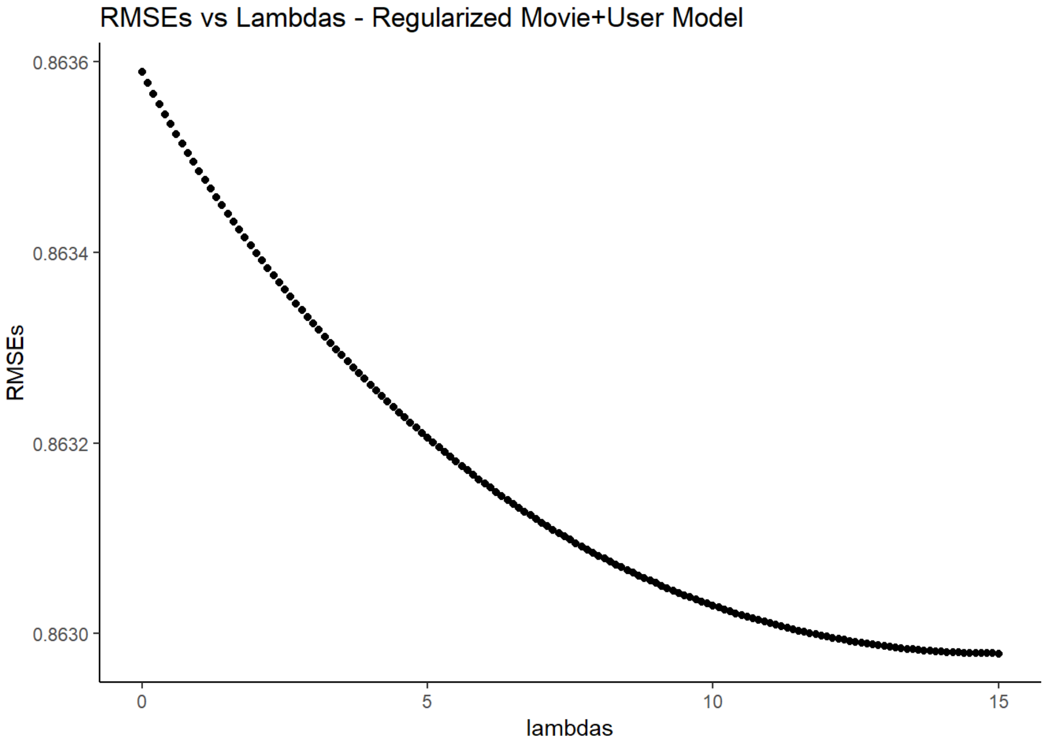
$$Y_{u,i} = \hat{\mu} + \varepsilon_{u,i}$$

Regularized Movie – Based Model



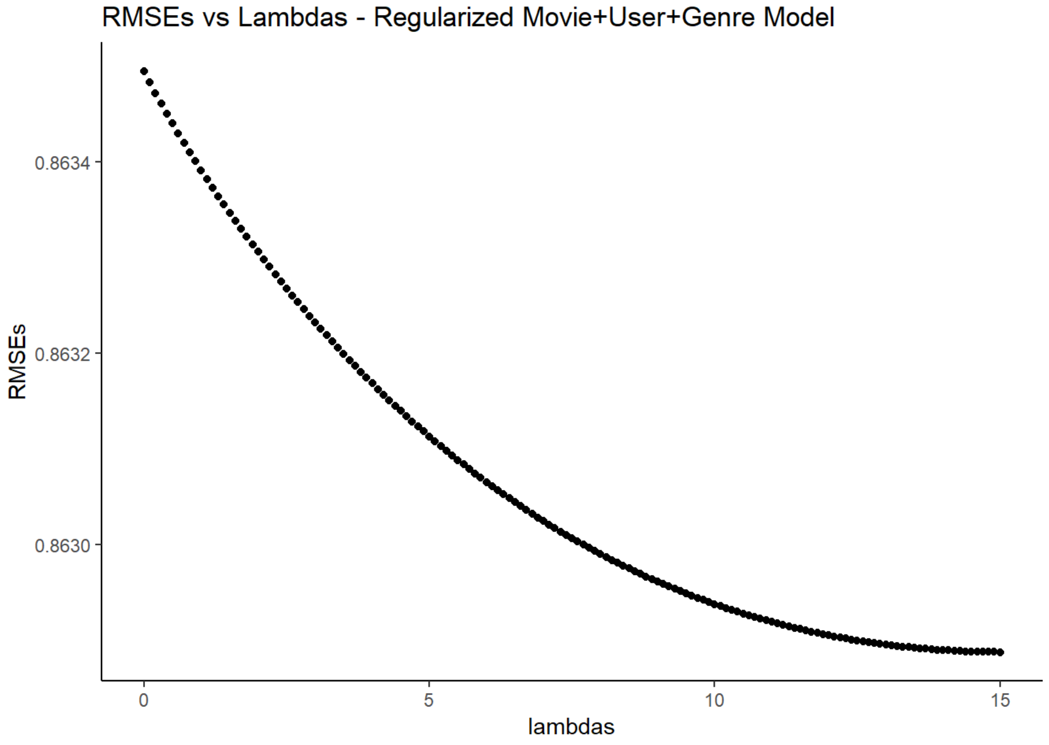
The RMSE on the *validation* dataset is **0.8635** and this is very good. The Movie + User Based Model reaches the desired performance but applying the regularization techniques, can improve the performance just a little.

Regularized Movie + User Model



The RMSE on the validation dataset is **0.8629**. The Regularized Movie + User Based Model improves just a little the result of the Non-Regularized Model

Regularized Movie + User + Genre Model



The RMSE on the validation dataset is 0.8628 and this is the best result of the built models. The Regularized Movie + User + Genre Based Model improves just a little the result of the Non-Regularized Model. As the Non-Regularized Model, the genre predictor doesn't improve significantly the model's performance.

Results

This is the summary results for all the model built, trained on *edx* dataset, and validated on the validation dataset.

Model	*RMSE*
Naive Mean-Baseline Model	1.0524433
Movie-Based Model	0.9411063
Movie + User Based Model	0.8635899
Movie + User + Genre Based Model	0.8634946
Regularized Movie-Based Model	0.9410767
Regularized Movie + User Based Model	0.8629791
Regularized Movie + User + Genre Based Model	0.8628874

Conclusion

After training different models, it is very clear that movieId and userId contribute more than the genre predictor.