Movie recommendation system using the MovieLens dataset

Žiga Kljun

9/27/2021

Contents

Introduction

This report is part of a Capstone assignment from *"HarvardX Profesional Certificate Data Science Program"*. Goal of this assignment is to create a movie recommendation system based on MovieLens dataset.

The importance of recommendation systems has been growing in the last 10 years. With the rise of companies like Netflix, Amazon and Facebook, recommendation systems are more and more common in our everyday life. Recommendation systems in general, are set of algorithms, implemented to suggest their users the most relevant items. They became critical in a lot of industries since they are able to generate business advantage - both, through revenue and through time saved. Very known and similar to our case is recommendation system from Netflix, which became famous in 2006, when Netflix organised challange called *""Netflix Prize""* with goal to create the best referral system. The winner recieved 1 million dollars as a prize.

Recommendation systems are really critical in some industries, as they can generate huge revenue if they are effective, or even a way to differentiate themselves significantly from competitors. As proof of the importance of referral systems, we can mention that a few years ago Netflix organized challenges (the "Netflix Prize") where the goal was to create a referral system that would be better than its own prize algorithm. $ 1 million to win.

MovieLens datasets are stable benchmark datasets, provided by the GroupLens research lab in the Department of Computer and Engineering at the University of Minnesota. The GroupLens lab specializes in recommender systems, online communities, mobile and ubiquitous technologies, digital libraries, and local geographic information systems. In order to gather research data on personalized recommendations, they originally collected the data in 1997. It contained about 11 million ratings for about 8500 movies. For our project, we used *"MovieLens 10M Dataset"*, which includes 10 million ratings and 100,000 tag applications applied to 10,000 movies by 72,000 users. The dataset was released in 2009. It include information about users, movies, their genres and ratings from 0,5 to 5 stars, provided by users. Each user has at least 20 ratings, but no further demographic information is included.

In our project we will split the data by the 90/10 rule into the training and test sets. Then we are going to develop our best performing algorithm with the use of training set which we are going to further split into training and test sets for developing reasons. We are going to calculate accuracy of our model with root mean squere estimate or RMSE. RMSE of 0 would mean, that our model is correct 100% of the time but this is very unlikely. RMSE of 1 would mean, that our predictions are on average off by 1 star. Our goal for this project is to achieve $RMSE < 0.86490$ when prediciting unseen ratings.

Methods/Analysis

Prepare datasets for model development and and validation

The following code was already provided by the HarvardX project instructions. In this project we used R 4.1.1.

Exploratory Data Analysis

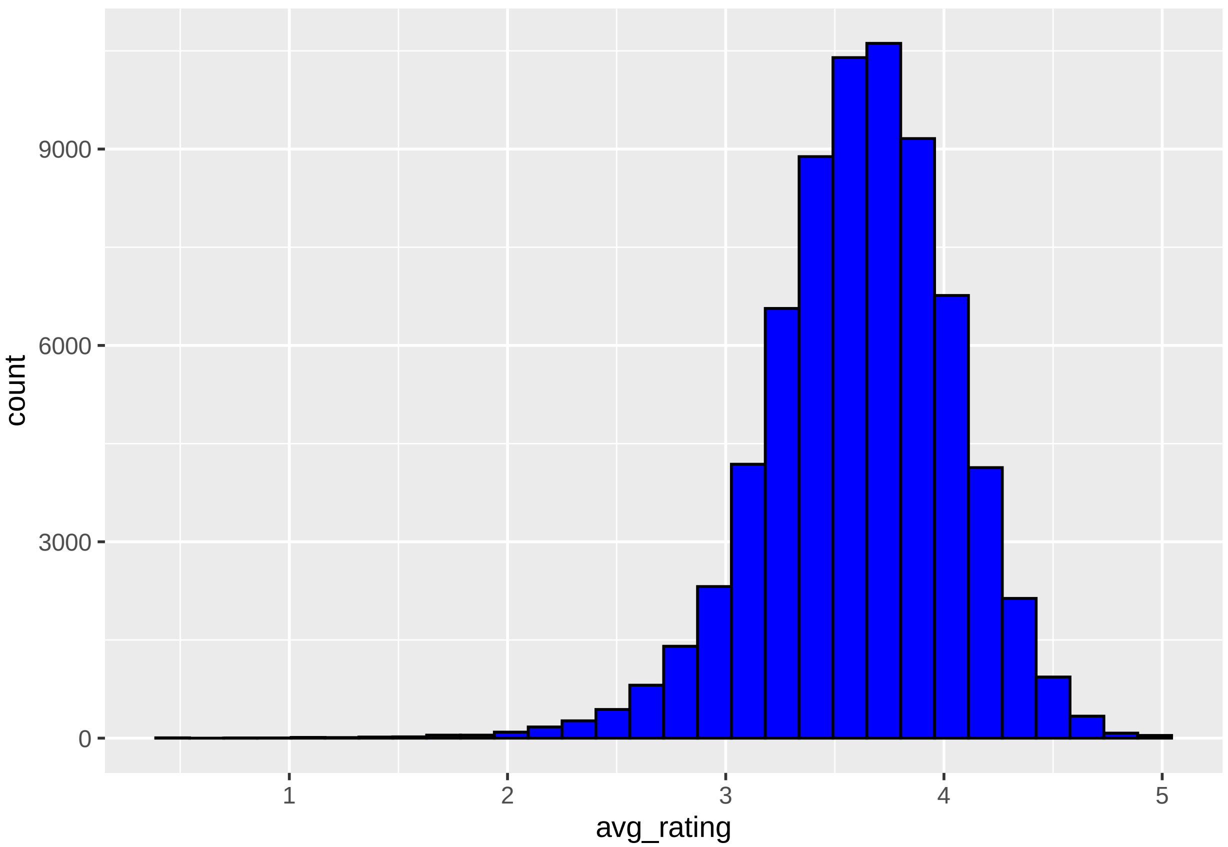
Basic information about the dataset

Prior model development, it is important, that we understand the problem and our data. Since our instructions was not to use validation sub-dataset, I will treat is separately from the rest, eventhough it is technically still a part of our MovieLens 10M dataset. I will call our "development dataset" an "edx" dataset, and validation dataset will be "validation" dataset. Edx dataset contains 9,000,055 records and validation dataset contains 999,999. Both of them have 6 rows: userId (int), movieId (num), rating (num), timestamp (int), title (chr) and genres (chr). Preview of the data is seen below:

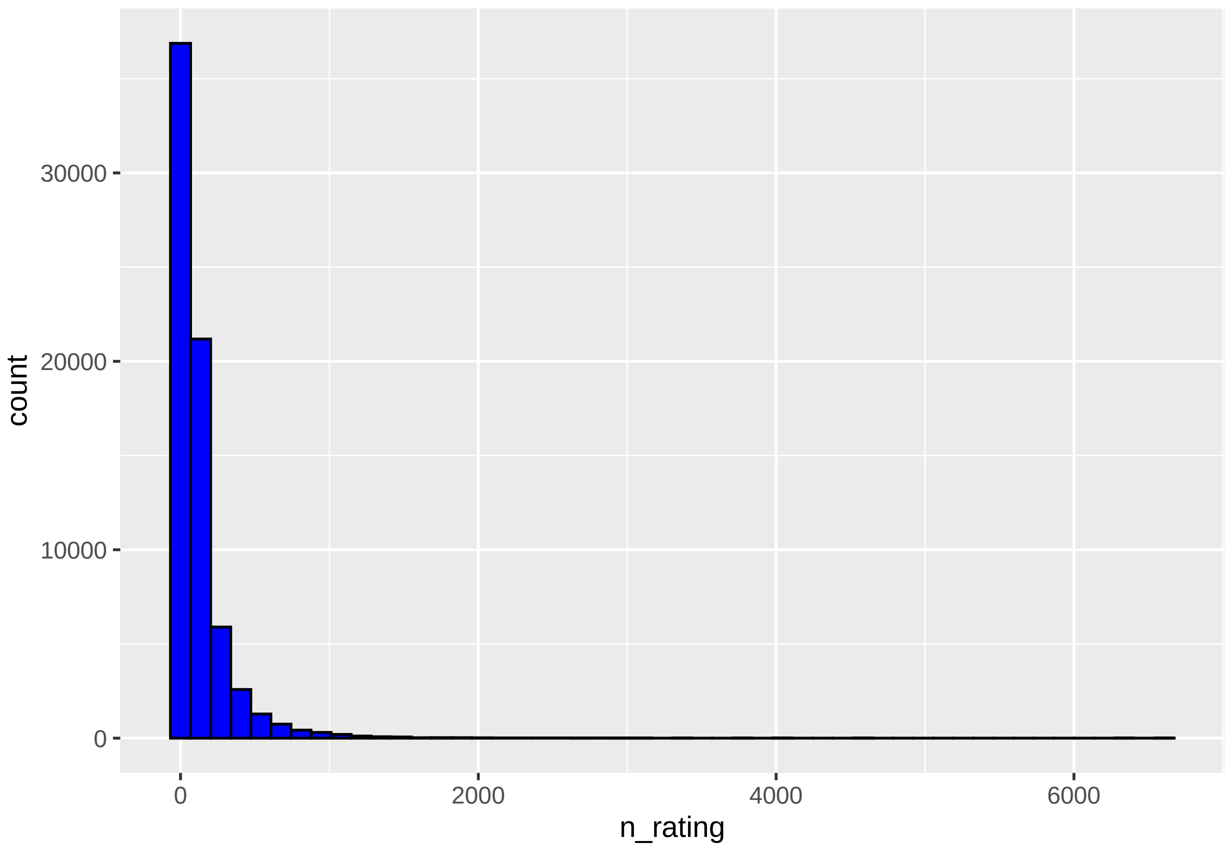
userId	movieId	rating	timestamp	title	genres
1	122	5	838985046	Boomerang (1992)	Comedy|Romance
1	185	5	838983525	Net, The (1995)	Action|Crime|Thriller
1	292	5	838983421	Outbreak (1995)	Action|Drama|Sci-Fi|Thriller
1	316	5	838983392	Stargate (1994)	Action|Adventure|Sci-Fi
1	329	5	838983392	Star Trek: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

Users

In edx dataset we have 69,878 different users which on average rated 128.7967 movies. In the plot bellow we can see the distribution of users per average movie rating. As we can see, most users tend to rate movie somewhere between 3 and 4 stars, more closely to the 4. Even if we look at the total average, it is 3.51 which confirms our finding.



If we move to the number of ratings provided by user, the results shows us, that the most users tend not to rate too many movies. 20 is minimum, but shortly after that, the amount of users starts to fall with the total movies rated rising. The plot bellow is showing the distribution of users in comparison to number of movies rated.



Genres

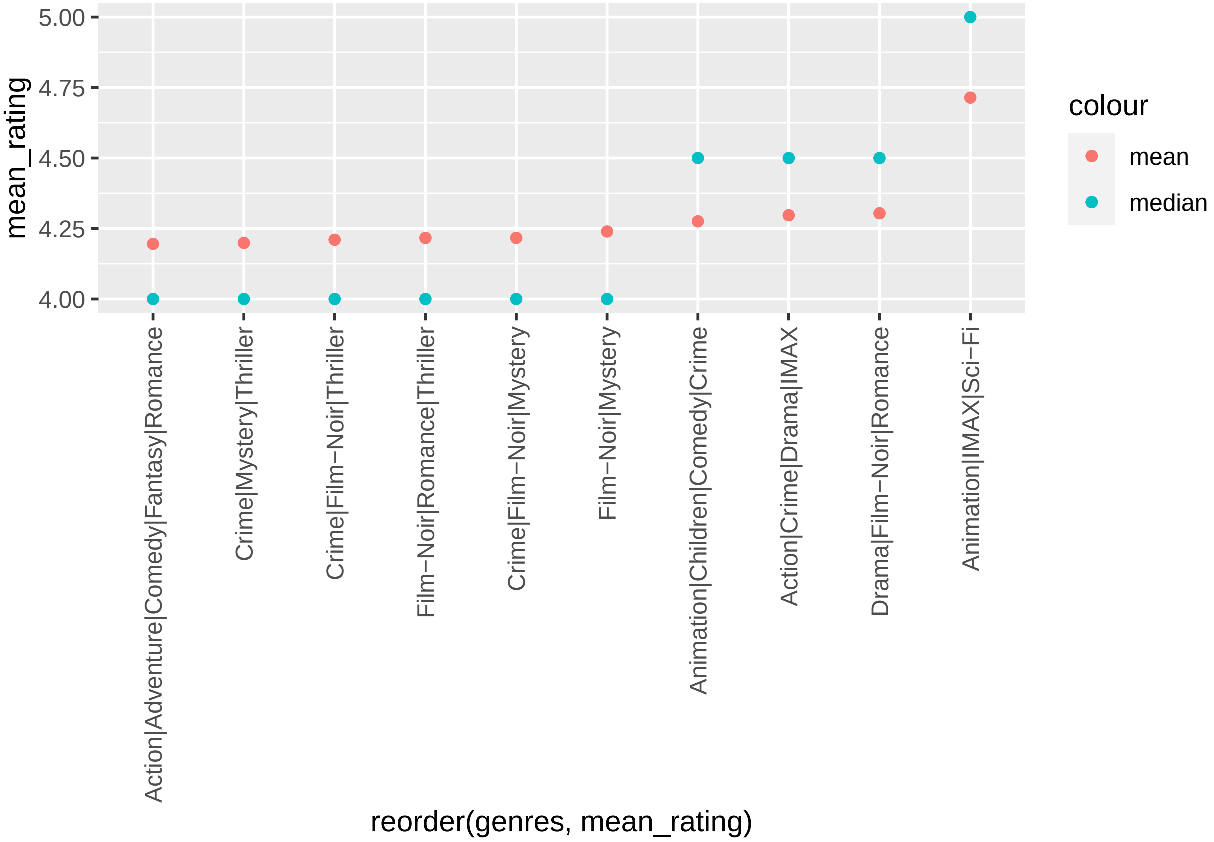
As mentioned before, in addition to user, movie and rating information, we also have the information about movie genres. In general it is believed, that movie genre has huge impact on the probability, that the user will like the movie so this could be an important column for us. In the edx dataset we have 797 different inputs for movie genres.

genres	number_occurances
Drama	733296
Comedy	700889
Comedy|Romance	365468
Comedy|Drama	323637
Comedy|Drama|Romance	261425
Drama|Romance	259355
Action|Adventure|Sci-Fi	219938
Action|Adventure|Thriller	149091
Drama|Thriller	145373
Crime|Drama	137387

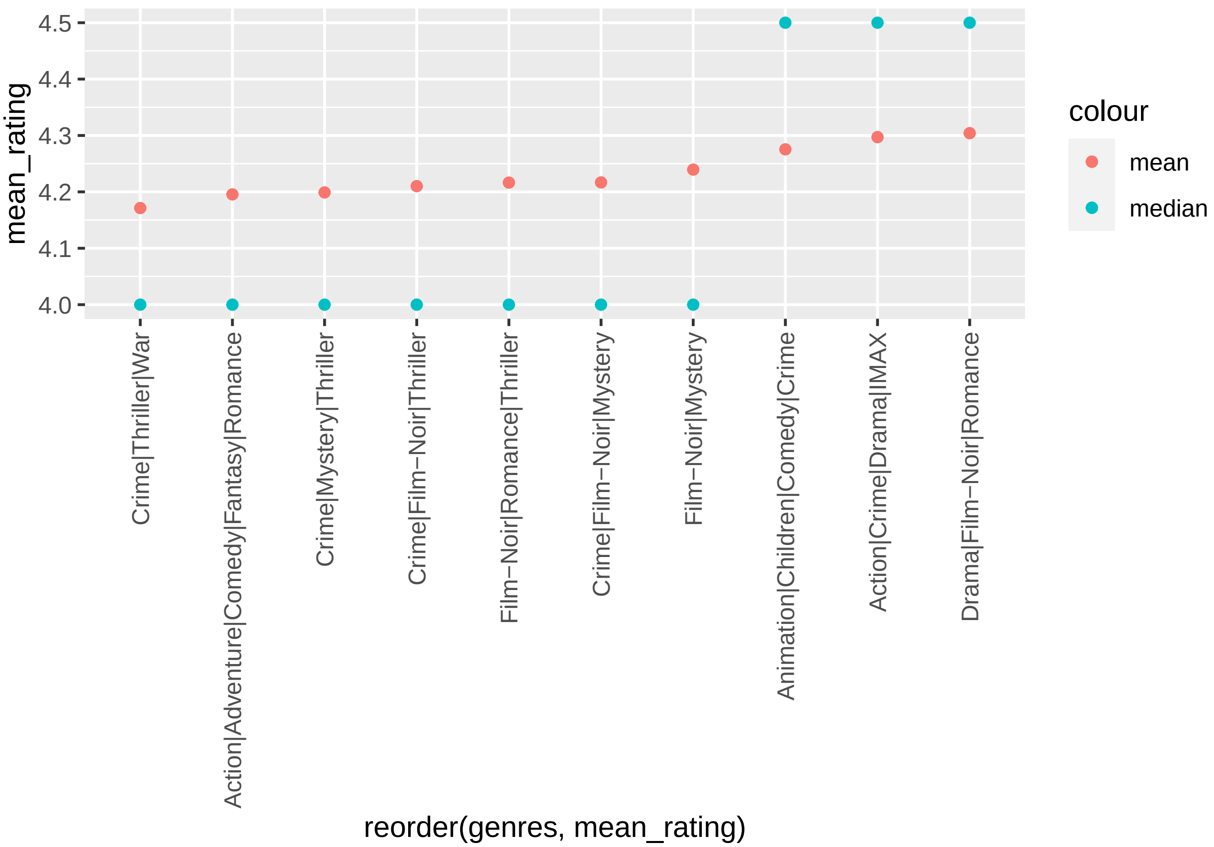
Above are shown the most popular genre entries with the number of their occurances. At first we noticed, that Drama is the most popular genre, followed by Comedy. But when looking at the third most common entry, we noticed, that is is a mix of two categories. This explains, why do we have almost 800 different genre inputs eventhough there is not so many different movie genres. Later in this project we will address this topic, but for now, we will treat each genre entry as its' own genre. In the statistics bellow we can see, that most complex combinations of genre includes as much as seven different genres. There are also three occurances where mix contains six different genres in one genre entry.

genres	number_of_genres
Action|Adventure|Comedy|Drama|Fantasy|Horror|Sci-Fi|Thriller	7
Adventure|Animation|Children|Comedy|Crime|Fantasy|Mystery	6
Adventure|Animation|Children|Comedy|Drama|Fantasy|Mystery	6
Adventure|Animation|Children|Comedy|Fantasy|Musical|Romance	6
Action|Adventure|Animation|Children|Comedy|Fantasy	5

In the plot bellow we can see best ten genre mixes based on their average rating. In red color is plotted also their median value. Based on the results we could assume, that movies with the combination of Animation, IMAX and Sci-Fi genres tends to have the best ratings.



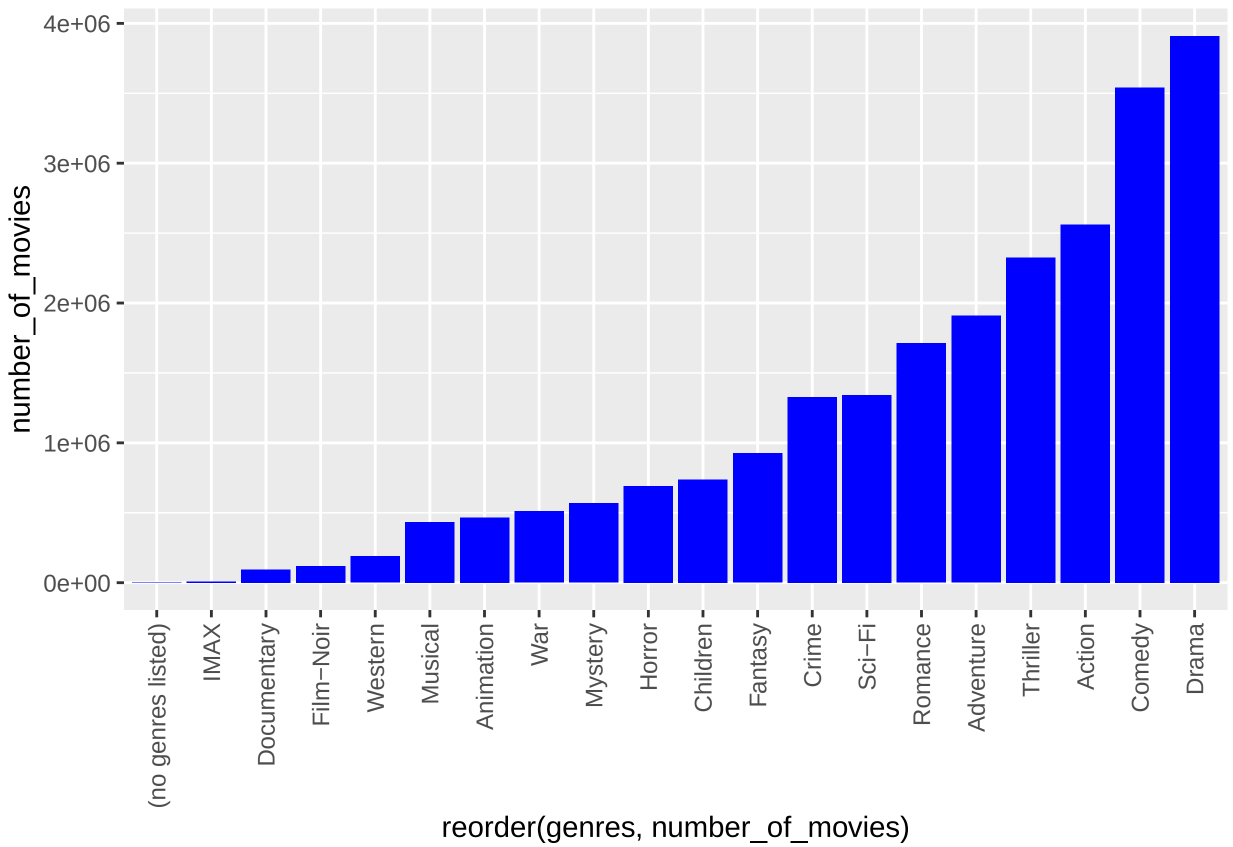
To get a bit more robust results, we drawn the same plot, but this time only for genres, that contains at leas 50,000 ratings. This time we get a bit different results, with Drama, Film-Noir and Romance mix at the top.



Even after this analysis, the results can still be a bit confusing, so I decided to separate genre entries to the single genre. Bellow we can see all the individual genres with their average rating and the amount of times genre is present in our MovieLens dataset.

genres	avg_rating	n_ratings
(no genres listed)	3.642857	7
Action	3.421405	2560545
Adventure	3.493544	1908892
Animation	3.600644	467168
Children	3.418715	737994
Comedy	3.436908	3540930
Crime	3.665925	1327715
Documentary	3.783487	93066
Drama	3.673131	3910127
Fantasy	3.501946	925637

Now we can actually plot, which genre has the most movies. The results are seen in the plot bellow. We see, that the most movies fall into Drama genre, followed by the Comedy and Action.

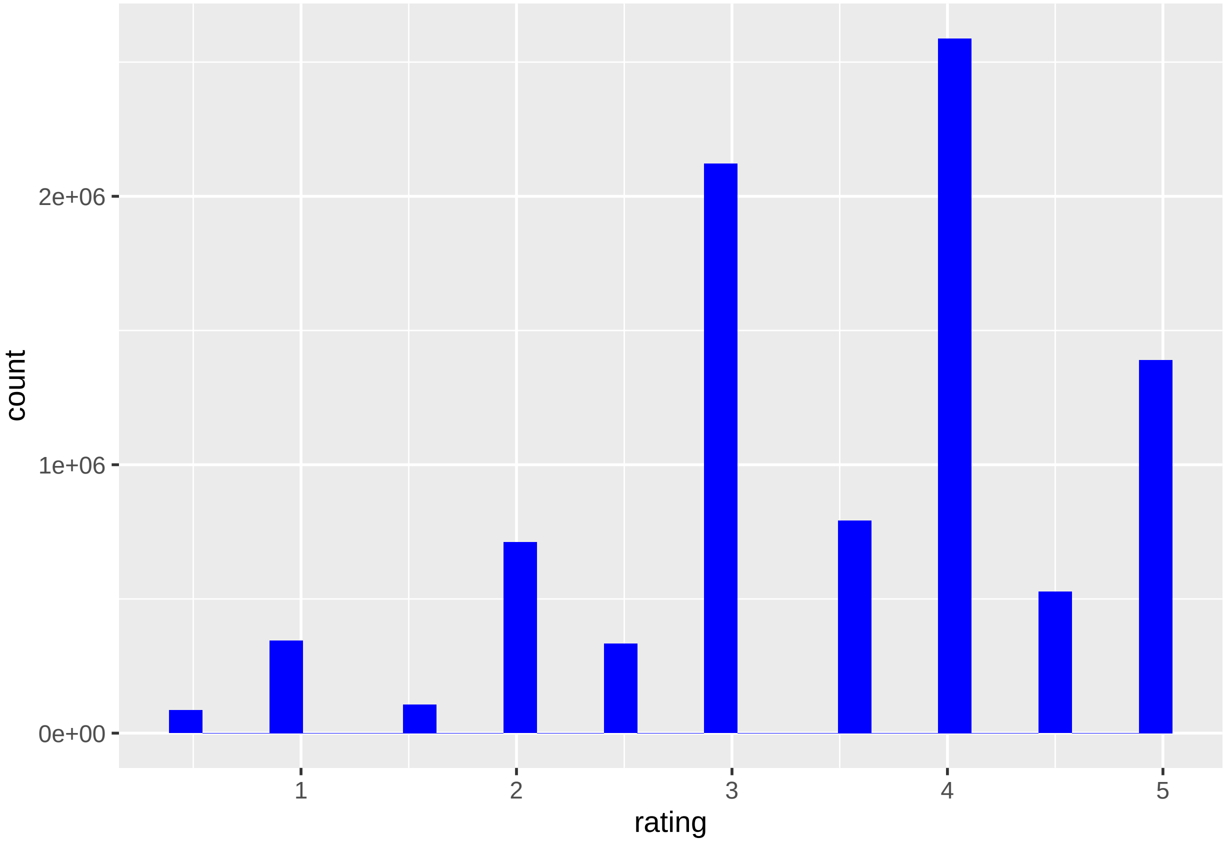


Movies

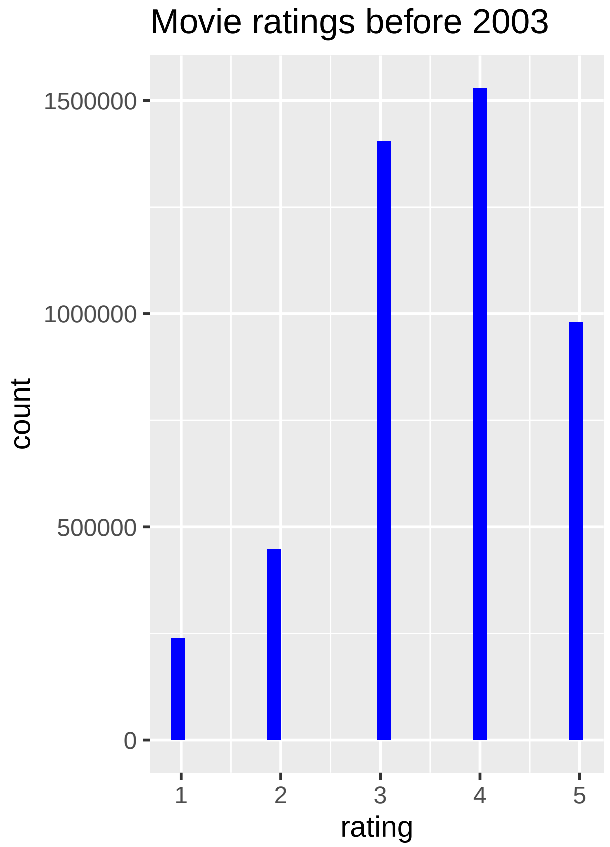
Until now, we have been focusing on more general areas, such as genres. Now we will look also in the individual movies. Bellow we can see the most rated movies with their average rating. We can see, that Pulp Fiction, Forrest Gump and Silence of the Lambs are on top. This should not be surprising, since they are all very well known movies.

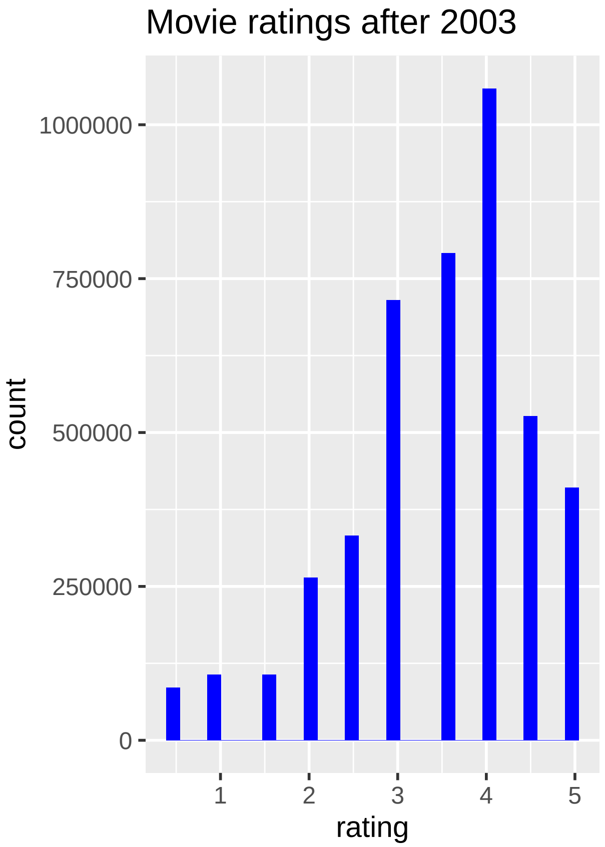
movieId	title	nr	avg_rating
296	Pulp Fiction (1994)	31362	4.154789
356	Forrest Gump (1994)	31079	4.012822
593	Silence of the Lambs, The (1991)	30382	4.204101
480	Jurassic Park (1993)	29360	3.663522
318	Shawshank Redemption, The (1994)	28015	4.455131

Before we already saw, how are moview average ratings distributed - average was 3.51, and the results were a bit skewed to the four stars. Now lets take a look at the distribution of actual star ratings. Bellow we can see, that if we drew the line, it through the top of the bins, it would be similar to our previous plot.

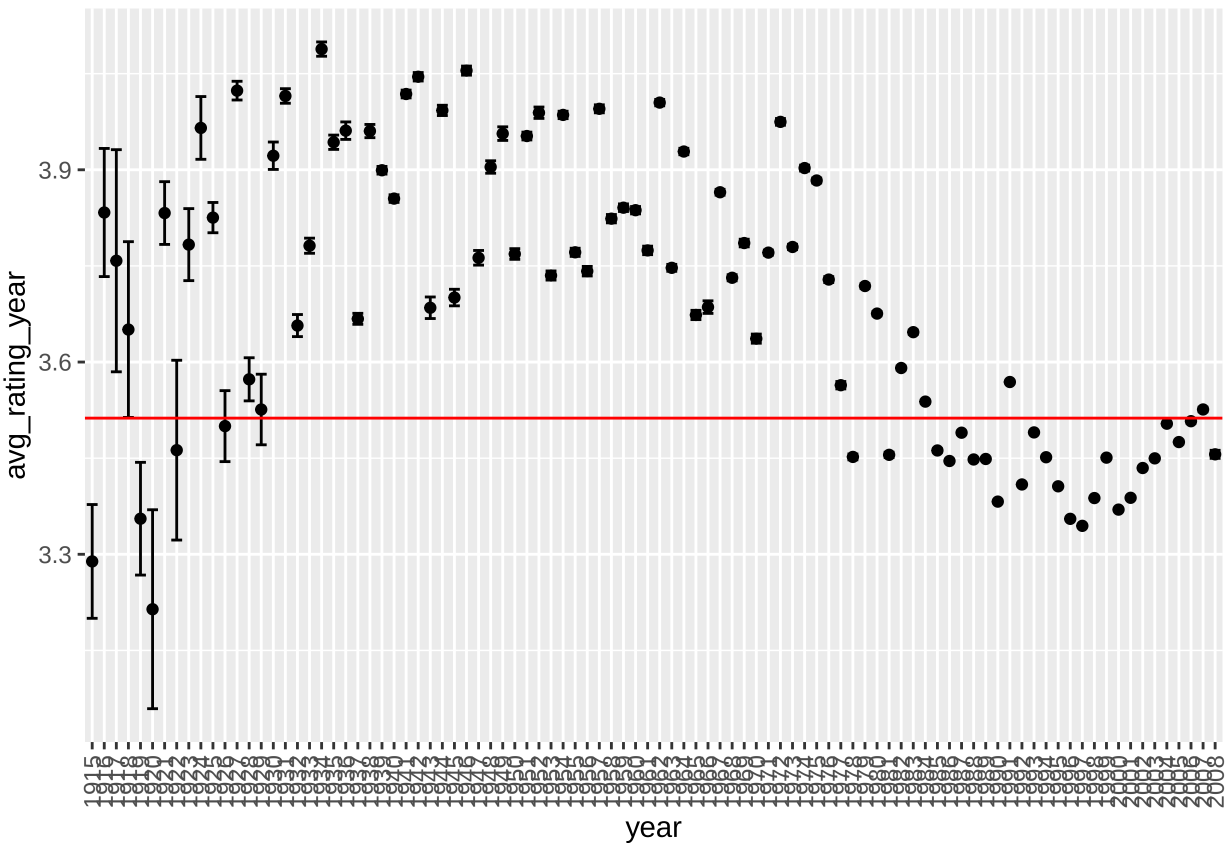


When looking at the full star ratings, as expected, four stars are the most common, followed by the three. But we noticed, that the amount of half stars are significantly lower than the ones with full star. Are users more likely to give full star review, rather than the half one? When looking at the data, however, we noticed, that half stars were only introduced in 2003, which explains this huge gap between full and half stars ratings. To clarify the situation, I splited data to before and after 2003. Now the results are more clear. In the plots bellow, you can see distribution of ratings before and after 2003. As we found out before, there are no half star ratings before 2003. Other than that, the results are quite expected based on our previous plot. However, we can see, that half stars in general are not less common than full star ratings, since they fall between the full star ratings next to then. However, four star rating is still the most common rating in both plots.





Now we know, how are movie ratings distributed by individual ratings (how many stars). But how are they distributed by each year? Were movies in 1950s higher than today? On the graph bellow we can see average ratings for each year. For each year we have also calculated standard error. The standard error (SE) of a statistic is the approximate standard deviation of a statistical sample population. The standard error is a statistical term that measures the accuracy with which a sample distribution represents a population by using standard deviation. In statistics, a sample mean deviates from the actual mean of a population; this deviation is the standard error of the mean. It is calculated as standard deviation devided by squere root of the sample size. Red line represent overall average rating of 3.51.



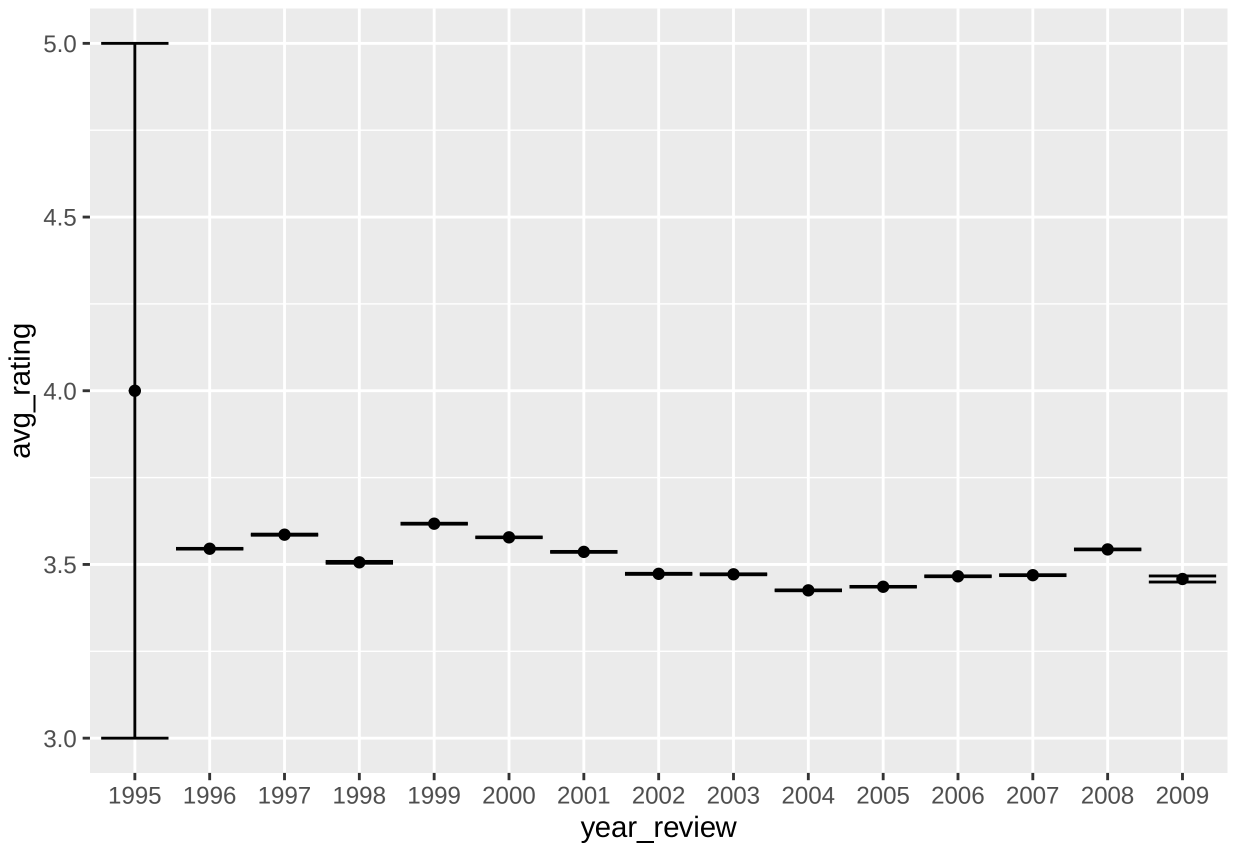
On the plots we can, that the ratings from 30 years or more ago, tend to have higher ratings on average. Going back, we also notice, that the standard error is larger than in more recent years. Based on this observations we could assume, that now we have more movies than before. The plot bellow, where numbers of movies per year are presented, actually confirms this. However, we noticed, that even though the number of movies had been rising up until 1995, it actually start declining short after that. Plot also explains, why most recent movies are much closer to total average than older movies. This is due to recent movies being more common which leads to a greater impact on total average.

Based on the findings above, the below summary is not very surprising. 50th percentile (median) and 75th percentile are the same (both four stars), with 25th being 3 stars.

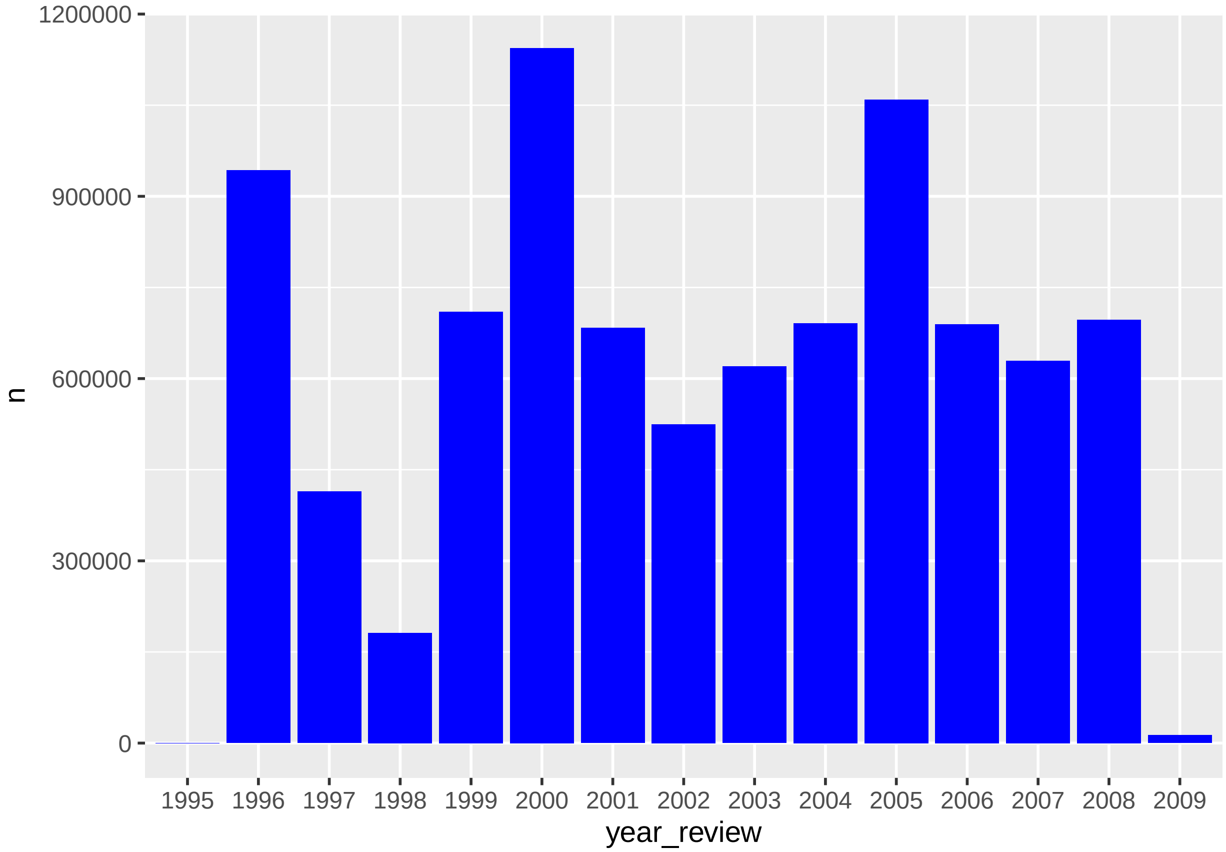
```
##   0%  25%  50%  75% 100%
##  0.5  3.0  4.0  4.0  5.0
```

Timestamp

Timestamp in data represents time and date of the individual review. The earliest review was collected in 1995. As we can see on the plot below, this was also the year, when average rating (around four stars) and also the standard error were the highest. Over time, the average ratings started to move more closely around the total average rating with smaller standard errors.



When looking at the rating distribution per timestamp year, the results are not as predictable. We noticed, that the number of ratings per timestamp fluctuate year over year. We have three peaks (1996, 2000, 2005) and two years with very low number of ratings (1998, 2009). Other years are relatively similar (from 400,000 to 700,000 ratings in one year).


1200000
900000
600000
300000
0
n
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
year_review

Recommendation system modeling

Preparing the data

In this section of the report, I will focus on recommendation system development - starting with data preparation. As mentioned in the beginning, we must not use validation dataset while still developing our models and algorithms. Thus, we first split the edx dataset into train and test datasets. We will use the train dataset to train our model and then based on our algorithm we will predict rating values in test dataset. We will compare this values with the actual ones and calculate RMSE. When we will develop our best permorming model, we will test it with the validation data.

Naive prediction

We will start with the naive prediction. We will predict, that all unseen ratings are just the average rating (3.51). This prediction will not very good, but can serve as a good baseline for our future development. We used following formula:

$$Rating_{movie} = Mean_{overall}$$

method	RMSE
Overall average	1.059086

Our RMSE in this case is 1.06, which means, that our prediction, on average, misses the actual rating by 1.06 stars. This is not terrible for beginning, but can definitely be better.

Movies' averages

For our next alghoritms, instead of total averages, we will use averages from individual movies. In general we will calculate the average for each movie and then, when predicting unseen rating, we will just look at which movie we are looking at and then assume, that the rating will be movie's average:

$$Rating_{movie} = Mean_{movie}$$

method	RMSE
Overall average	1.0590864
Movies' averages	0.9430345

We managed to improve our RMSE - now, on average, we are missing with our prediction for less than one star. But we are still far from our goal of 0.86490.

Users' averages

When rating the movie, movie itself is not the only variable than can impact the decision. Some users tend to rate all movies higher than the others, regardless of how much they actually like the movie. To take this into account, we will calculate how each user rates movies on average compared to the total average. Then, when predicting ratings, we will add this bias to our existing predictions from movie averages:

$$Rating_{movie} = Mean_{movie} + Bias_{user}$$

method	RMSE
Overall average	1.0590864
Movies' averages	0.9430345
+Users' averages	0.8845046

We managed to improve our prediction even more. RMSE fell to 0.885, which is already very close to our desired value. But we will still try to improve our ratings.

Genres

Even though we came very close to our desired RMSE value, we have still some variables, we need to take into our consideration. Now, when predicting, we take into account the overall movie average and user's bias toward high or low ratings. But often, this is not enough to predict whether the movie likes or dislikes the movie. It is also up to movie's genre. For example,thrillers are on average better rated than animations. To some degree, this observation is already baked in in the movie averages, but we will still add it to our model to see, if there are any improvements:

$$Rating_{movie} = Mean_{movie} + Bias_{user} + Bias_{genre}$$

method	RMSE
Overall average	1.0590864
Movies' averages	0.9430345
+Users' averages	0.8845046
+Genres' averages	0.8842123

We can see, that we managed to further improve our model. Not by much, but this was expected due to movie's genre bias is already partly taken into account with movie's average.

Users' genres preferences

Sometimes, consideration of movie's average, users tendency to rate movies low or high and genre's bias is not enough. For example, if particular user prefers documentaries and dislikes romantic comedies, he will not like the movie *"Pretty Woman"*, even though user tends to rate movies quite high and *"Pretty Woman"* is on average rated quite highly. To take this genre bias into account, we will calculate user's biases towards different genres and then add this bias to our existing prediction:

$$Rating_{movie} = Mean_{movie} + Bias_{user} + Bias_{genre} + Bias_{user-genre}$$

method	RMSE
Overall average	1.0590864
Movies' averages	0.9430345
+Users' averages	0.8845046
+Genres' averages	0.8842123
+Users' genres preferences	0.8524354

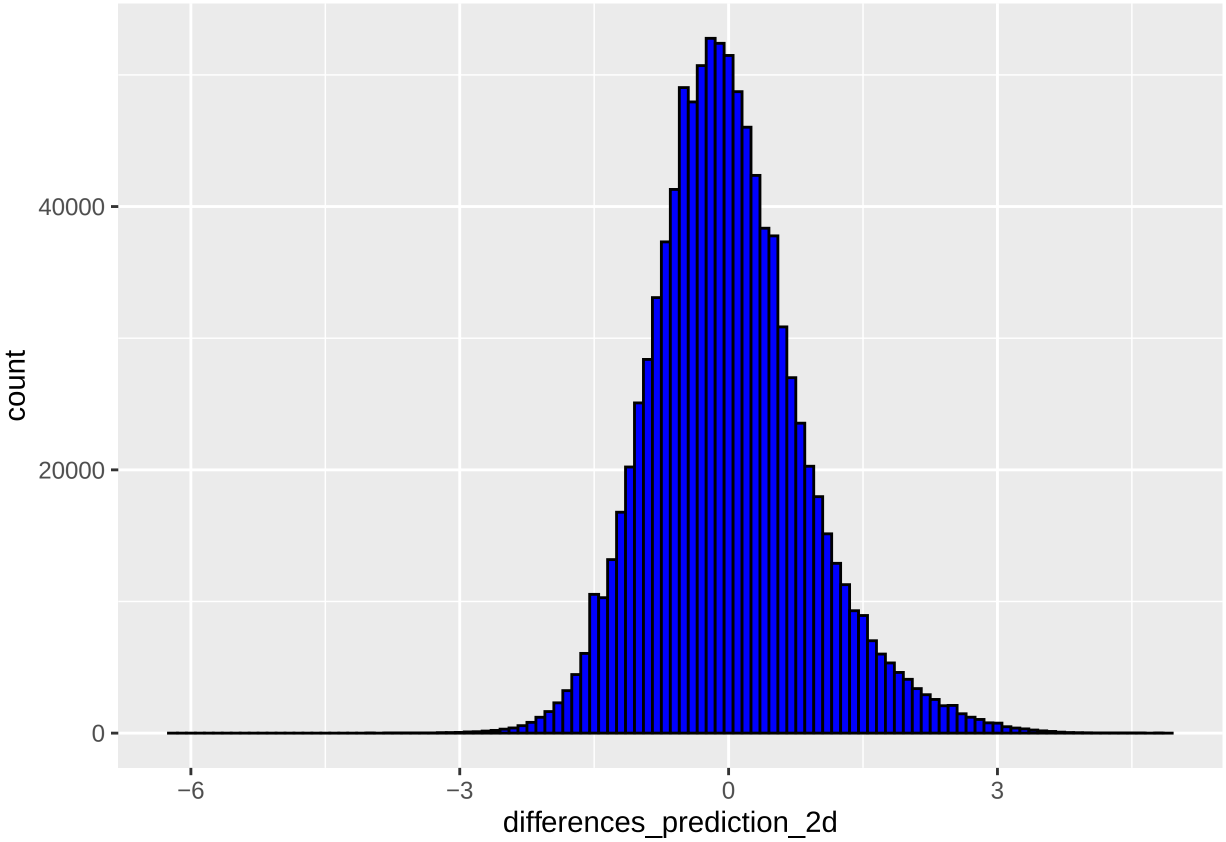
We did it! After adding user's genre bias, we managed to get RMSE smaller than 0.86499. Our RMSE on edx dataset is 0.852. Users' preferences regarding genres seems to have relatively big additional impact on top of the properties already taken into account. Now we can run the same algorithm on the validation dataset to see, if our model is really performing as well as it looks here.

Final results

Now, when we finally have our final model, we need to evaluate it, to see, how good it is. We will do that using RMSE. For our training set, we will use edx dataset, which we already used for development of our model. Our test set will be validation dataset, which was unused up until this evaluation. We will try to predict ratings in validation dataset based on other columns and ther calculate RMSE between the actual ones and the one we predicted. This is the result:

method	RMSE
Final RMSE	0.8516689

As expected, we get the same RMSE as in our development testings which is 0.852 or 0.8516689 to be more exact. In the plot bellow we can also see the distribution of errors, which is normal with a center close to 0. If we look at the exact data, the graph has weak skewness of 0.5 and median value is a little less than 0 at -0.07.This means, that our model tends to rate movies a bit lower than they are in reality, but not by much.



Conclusion

The opbjective of this assesment was to develop a recommendation system, based on the MovieLens 10M dataset, which will predict movie ratings with RMSE smaller than 0.86490. Prior to modeling, we performed an analysis of our dataset to gain a better understanding of the data provided. After that we started to develop our model. We spit our train dataset into train and test datasets for developing purposes so our final test set remained unseen. In the process of developing our model, we took into account movies' averages, users' tendancies to give higher or lower ratings, genres' average ratings and users' liking of individual genres. When our model was showing good enough results in our development process, we decided to perform the final test on the real test set. It performed similar on the final test set as it did on our development test set. We got RMSE of 0.8516689.

Even though we reached our goal, the model is still far from being perfect. RMSE of 0.85 in reallity is not very good, but we also lack quite o lot of information, which could be beneficial to our model (such as age, gender, geo location etc.). We could be also improving our model with existing information. We could do some additional feature engineering or regularization. We could also use other, more CPU or GPU-demanding machine learning techniques, which could give us better results. But for that, our personal computer would probably not be enough.

References

Irizarry, R. (2020). Introduction to Data Science: Data Analysis and Prediction Alghoritms with R. CRC Press

Rocca, B. (2019). Introduction to recommender systems. Available at: https://towardsdatascience.com/introduction-to-recommender-systems-6c66cf15ada (Accessed: 29 September 2021).

Statistics How To (2021). RMSE: Root Mean Square Error. Available at: https://www.statisticshowto.com/probability-and-statistics/regression-analysis/rmse-root-mean-square-error (Accessed: 29 September 2021).