MovieLens Project

June, 2019

1 INTRODUCTION

A recommendation system studies the preferences of the users, offering suggestions on a certain content that they are interested in.

The preference systems analyze the historical data of the users (previous behavior, qualifications, places…). They can be of several types:

- User-based systems: To predict whether the user would like a particular item, the recommendation system evaluates the opinion of other users with similar preferences.
- Item-based sytems: In this case, to recommend a specific item to a user, the system will take into account the opinion of said user about other items that have been valued by the user and that are similar.
- Hybrid recommendations systems.They are a combination of both user and item-based recomendation systems.

In the movilens project, our objetive is to predict the ratings of the movies in the validation set, training the algorithm with the edx set. The measure that we will use to evaluate the proximity of our predictions to their true values is RMSE.

2 DATA REVIEW

2.1 Dimensions

```
## [1] 9000055       6
```

The data has 9000055 instances and 6 variables

2.2 Structure

```
## 'data.frame':    9000055 obs. of  6 variables:
##  $ userId   : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ movieId  : num  122 185 292 316 329 355 356 362 364 370 ...
##  $ rating   : num  5 5 5 5 5 5 5 5 5 5 ...
##  $ timestamp: int  838985046 838983525 838983421 838983392 838983392 838984474 838983653 838984885 83
##  $ title    : chr  "Boomerang (1992)" "Net, The (1995)" "Outbreak (1995)" "Stargate (1994)" ...
##  $ genres   : chr  "Comedy|Romance" "Action|Crime|Thriller" "Action|Drama|Sci-Fi|Thriller" "Action|Ac
```

Almost all the features are integer or numeric. The exceptions are genres and title, which are “character”.

2.3 Genre distribution

```
##                            freq percentage
## Drama                    733296   8.147684
## Comedy                   700889   7.787608
## Comedy|Romance           365468   4.060731
## Comedy|Drama             323637   3.595945
## Comedy|Drama|Romance     261425   2.904704
## Drama|Romance            259355   2.881705
## Action|Adventure|Sci-Fi  219938   2.443741
## Action|Adventure|Thriller 149091   1.656557
## Drama|Thriller            145373   1.615246
## Crime|Drama               137387   1.526513
```

If we study the category of genres as it is in the dataset, we could see that the percentage of ratings of each genre is quite small. Movies in the Drama category are the most rated, followed by Comedy.

2.4 Number of users and movies in the dataset

```
##   num_users num_movies
## 1     69878      10677
```

We have more users than movies. It seems right, because a user can rate several movies.

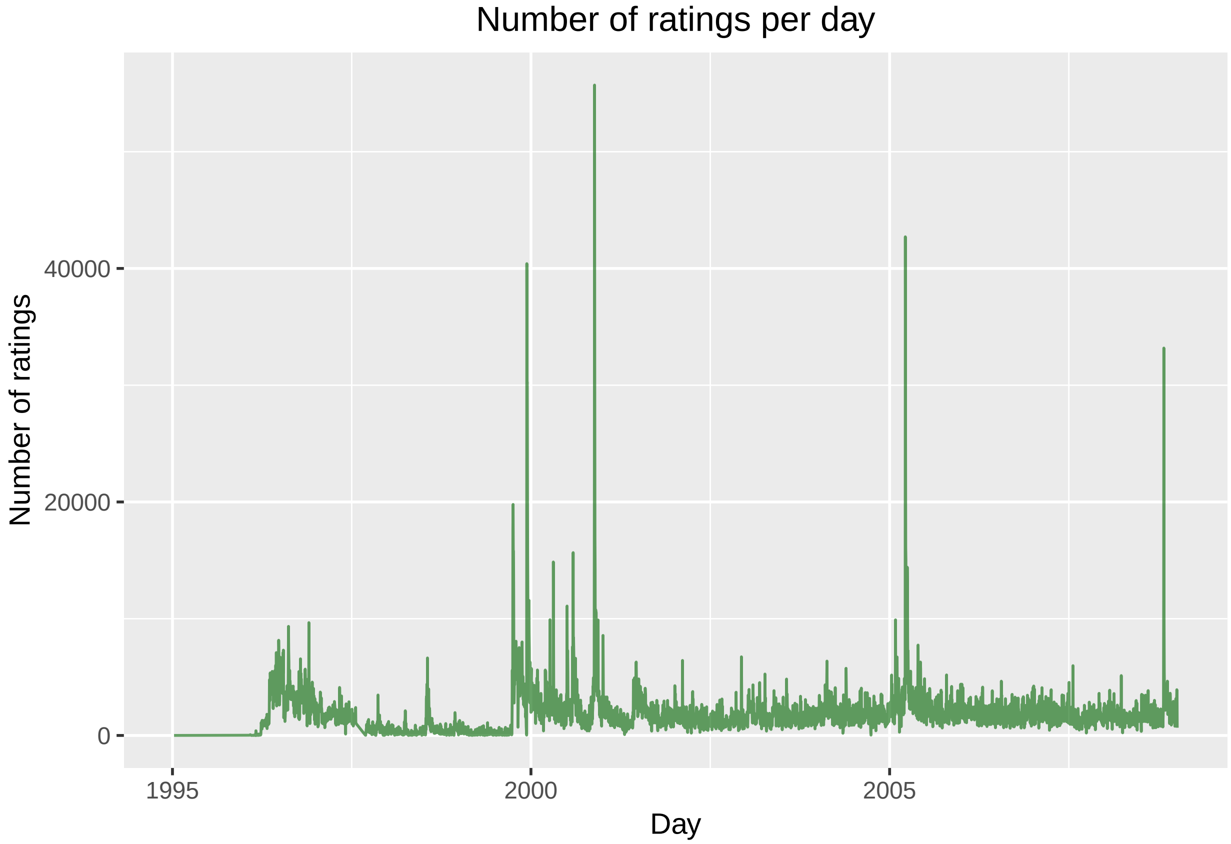
3 DATA CLEAN AND TRANSFORMATIONS

```
##    userId   movieId    rating timestamp     title    genres 
##         0         0         0         0         0         0
```

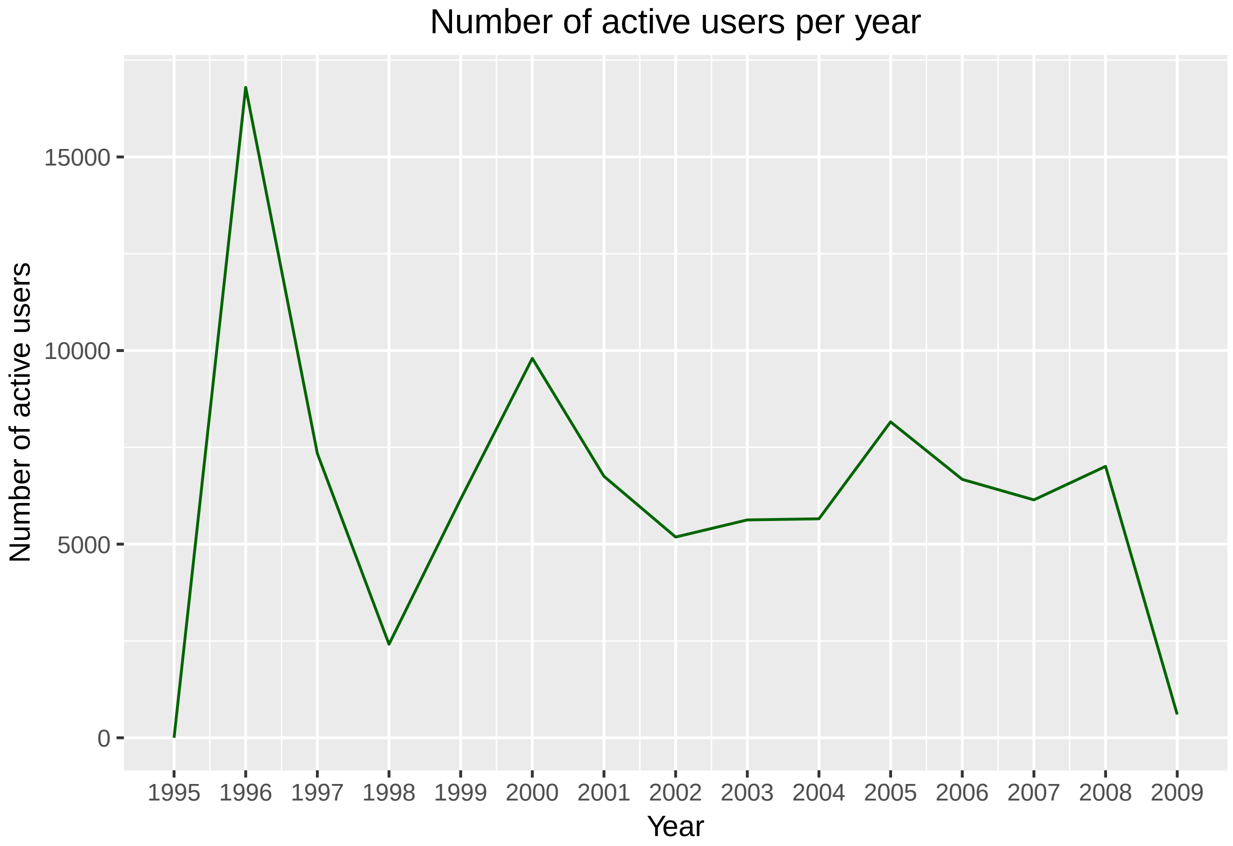
The data set is complete: there are not variables with NA values.

As we have seen previously, the Timestamp feature is a number. We are going to transform it into Datetime, more suitable for our purposes.

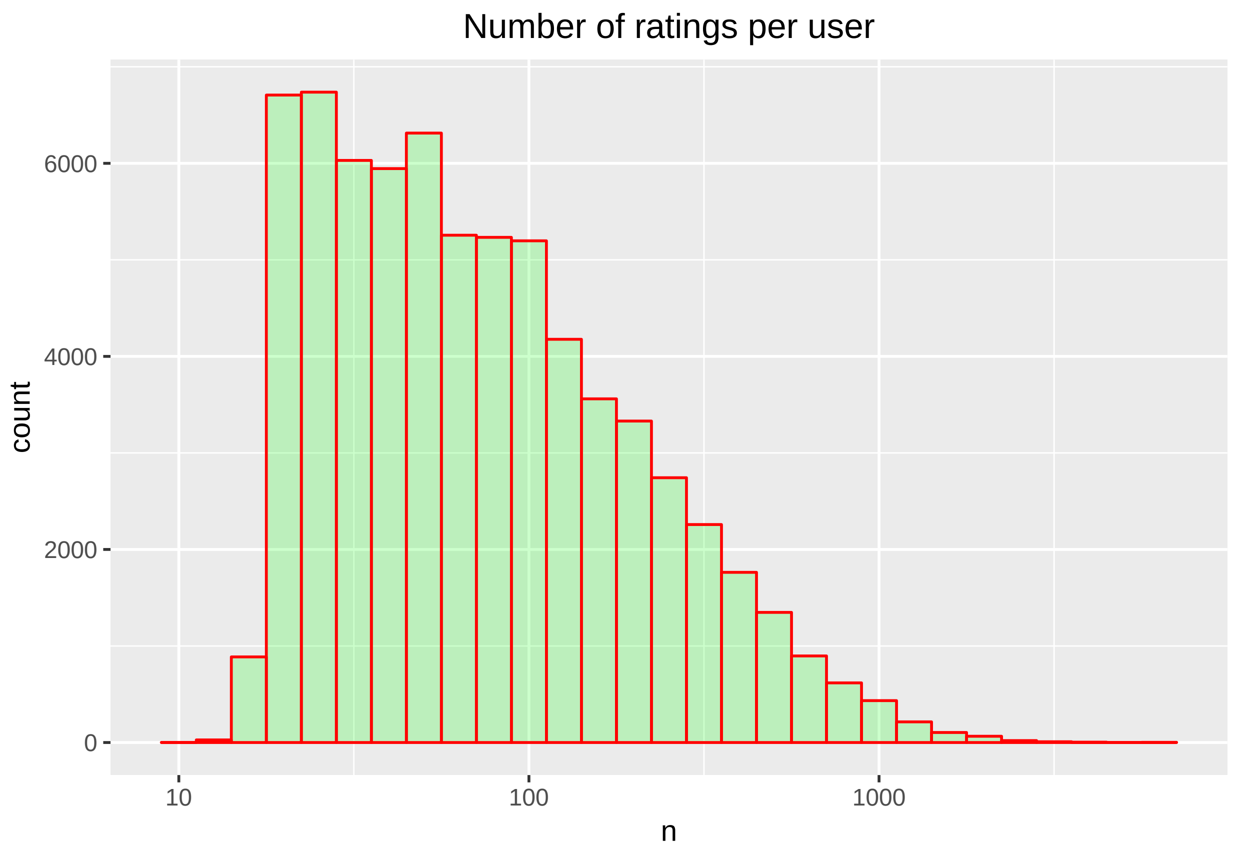
4 DATA VISUALIZATION



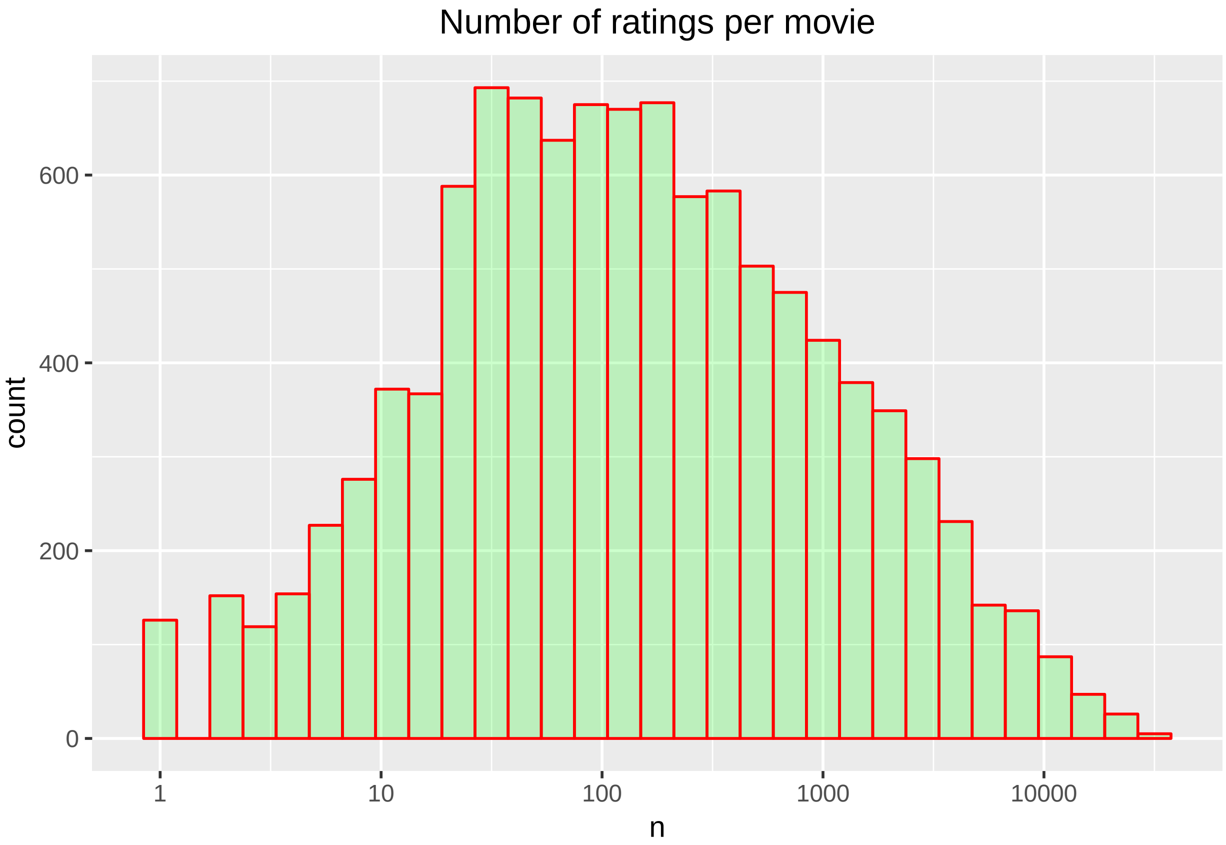
When studying the evolution of the data over time, what we could see is a big difference in the number of daily evaluations. We could ask themselves if the number of active users has change over time:



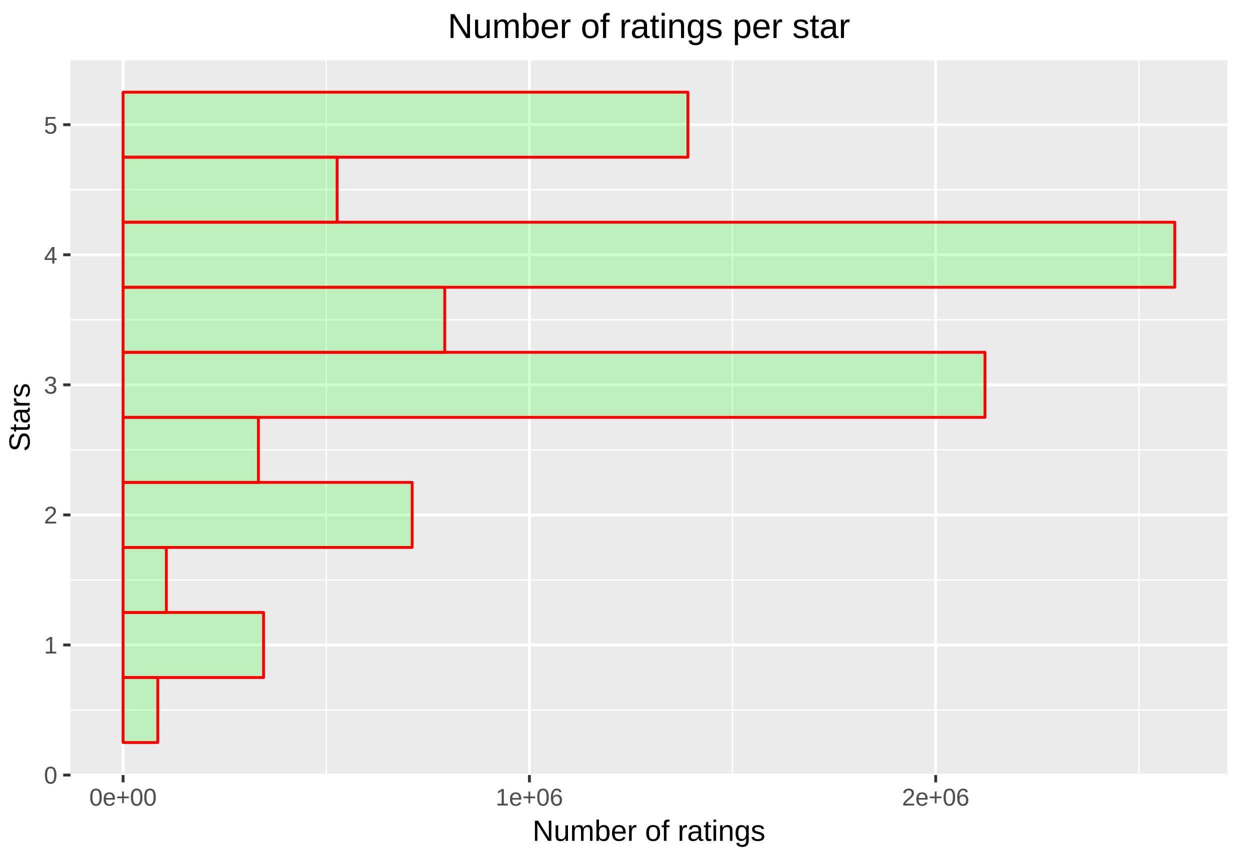
The year with the most active users was 1996. Then the number of active users suffered a big drop, until 1998, year in which the recovery began. Since then the number of users has been more stable, although having ups and downs.



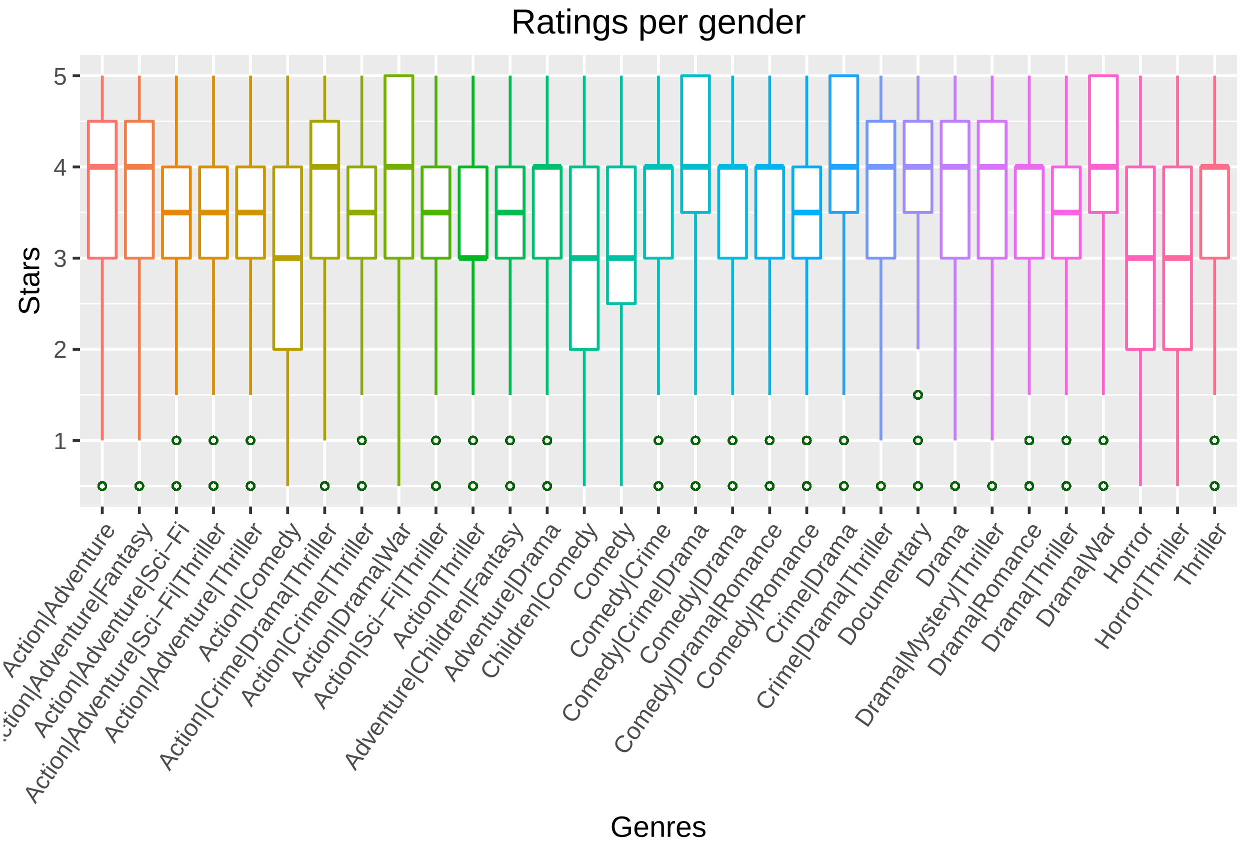
The number of ratings per user is quite heterogeneous. Some of the users are very active with a high number of ratings.



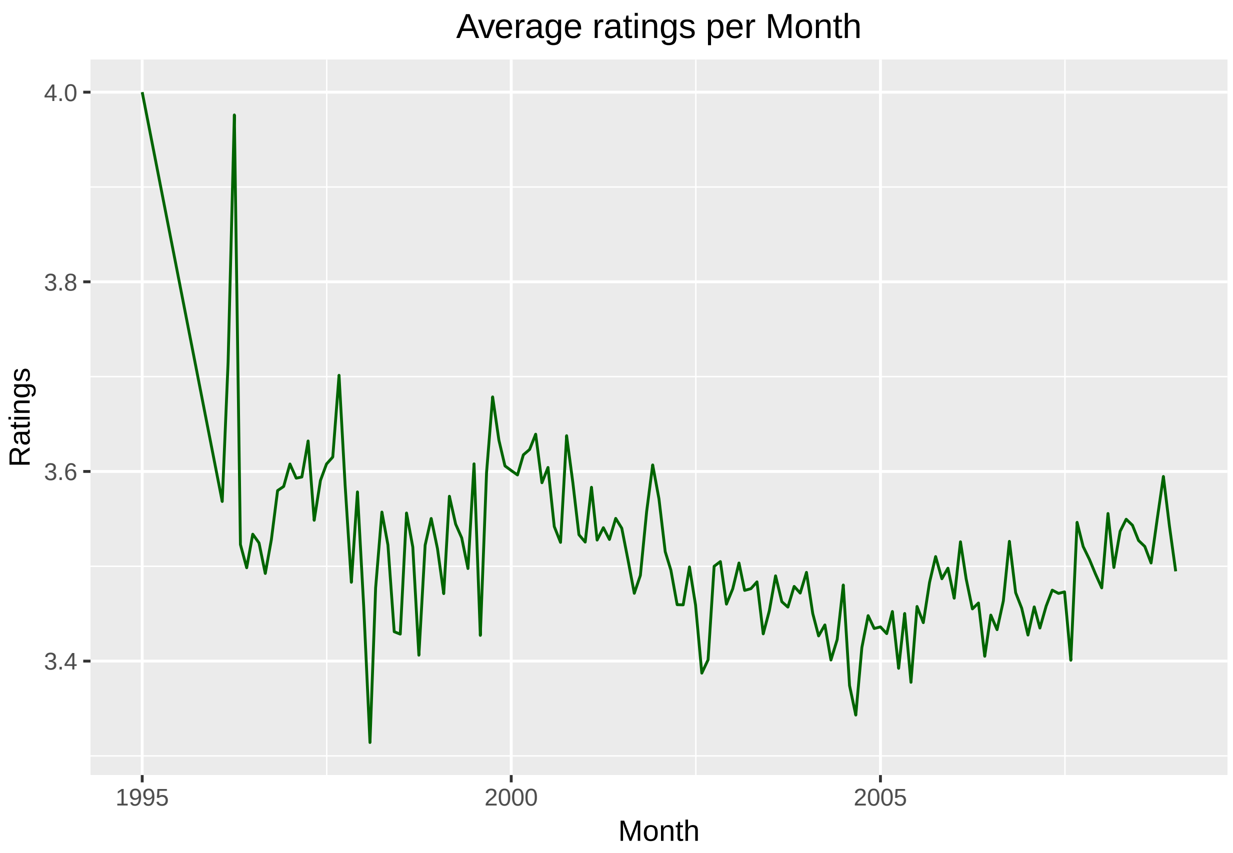
Some movies have a lot of ratings (some of which almost 500), but the vast majority have few ratings. More than 10,000 films have less tha 50 ratings.



Regarding the ratings, users use to assign integer numbers. The most frequent puntuations are 4, 3 and 5 (in this orden). We can conclude that users tend to rate movies they like or they usually watch movies they like.



We have only taken into account genres with more than 50000 ratings. The dispersion is accentuated, particularly in some genres. In most genders, the median is 3.5 or 4.



Studying the changes in monthly average ratings over time, we can see that there is sligth evidence of a time effect. Excluding the first year, in which the use of the application was not very widespread, we see that the average of the ratings vary only between 3.3 and 3.7.

5 RESULTS

As we explained in the introduction, we could try three approaches: User-effect ,Item-effect and hybrid recommendation model:

5.1 Users-effect recommendation model

If we only take into account user effect, we have this kind of model:

```
  Yu,i=mu+bu+eu,i
```

Where mu is the media of the ratings of all recomended movies, bu is the user effect for user u and eu,i is the error for user u and movie i.

```
## [1] "User-effect model  RMSE: 0.979599"
```

5.2 Item-effect recommendation model

In this case, the model is: Yu,i=mu+bi+eu,i

where mu is the media of the ratings of all recomended movies, bu is the user effect for user u and eu,i is the error for user u and movie i.

```
## [1] "Item-effect model  RMSE: 0.943315"
```

In base of RMSE metric, this is a better model than the previous one.

5.3 Hybrid-effect recommendation model

In this case we are going to take into account bot user and movie effects. The model is:

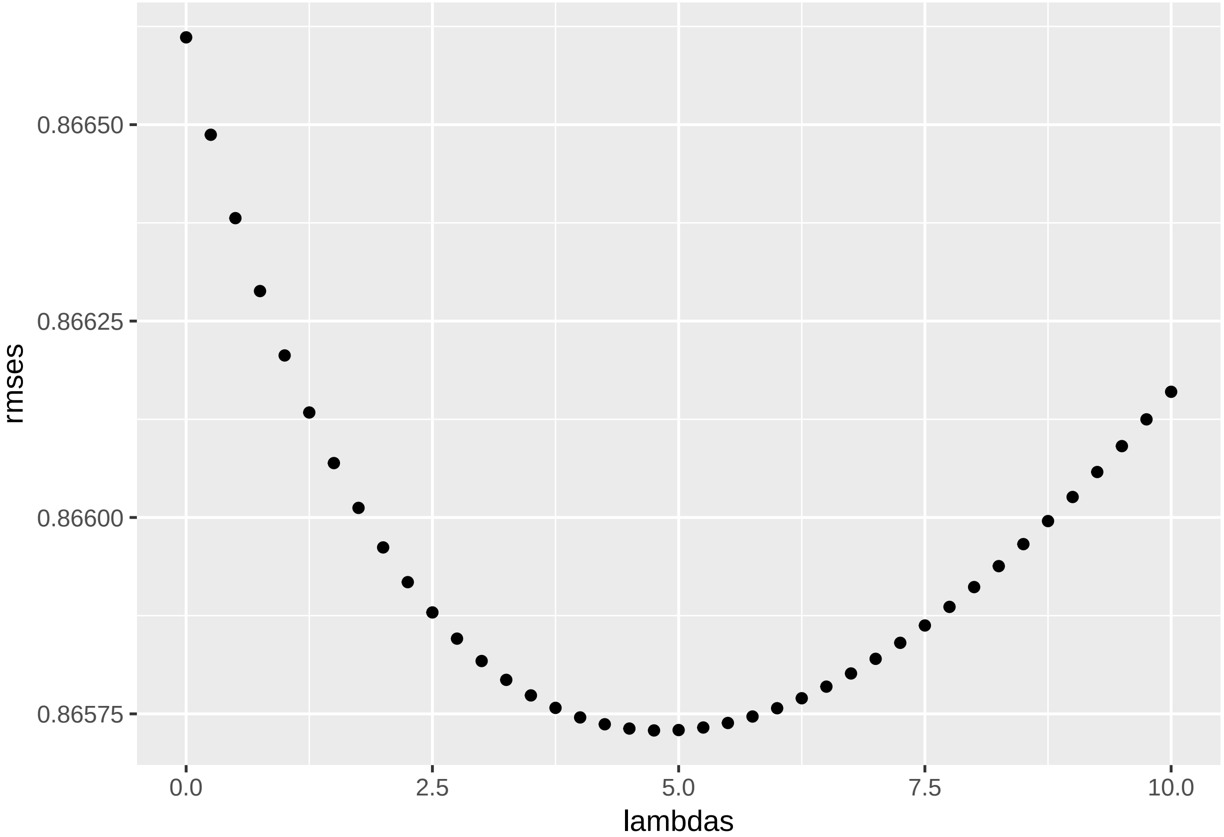
Yu,i=mu+bi+bu+eu,i

```
## [1] "Hybrid model RMSE: 0.886814"
```

The improvement in the RMSE metric is remarkable. Can we get a better result?

5.4 Hybrid-effect recommendation model with regularization

We can try to modify the model to avoid that those films with few ratings have as much influence on the result as those with many ratings. For this we will calculate the difference between the rating and the average, divided by the number of ratings and a parameter lambda. We will estimate this parameter using cross-valition.



```
## [1] "Lambda: 4.750000"
```

```
## [1] "Hybrid model with regularization RMSE: 0.865729"
```

Comparing the four models, with which we obtain the best result is with the las one. Once the best model is chosen, we apply it to the validation data set:

```
## [1] "RMSE_validation: 0.864820"
```

6 CONCLUSION

The previous analysis has been conditioned in a certain way by the size of the data set. In particular, I have not been able to perform dispersion analysis with the entire data set, and selecting subsets the results have been contradictory. In addition to the effects of user and item (movie, in this case), it seems that there may be slight effects due to gender and time. To simplify the calculation, in the model we will only take into account, in principle, user and item effects.

Of the four tested models, with which better results are obtained (based on the values of RMSE) is a hybrid model, taking into account both the effects of the user and the movies, but after a process of regularization. with this model, the RMSE obtained applaying this model to the validation data set is:

```
## [1] "RMSE: 0.864820"
```