Harvard Capstone Project : All Learners (edx HarvardX: PH125.9x)

Paul Daniel VLADU

2019-03-10

Executive Summary

This document proposes a movie recommendation system that uses the MovieLens dataset, using R programming. The purpose of the proposed solution is to predict movie ratings. The recommendation system uses the (10M version of the MovieLens dataset).

The data resides mainly into two main files (ratings.dat & movies.dat) packed into a zip file, downloaded via the R script. The current project is based on the instructions given in the “Data Science:Capstone course” (HarvardX- PH125.9x), so that the dataset is split into two main datasets:

1. **edx** as train dataset and
2. **validation** as test dataset.

Thus, the dataset **edx** will be used as train data set to develop the algorithm and the **validation** dataset will be used for the final test of the algorithm, as a test dataset, to predict movie ratings in the validation set as if they were unknown.

The proposed solution uses the insights, the models and methodology presented in the course “Data Science: Machine Learning (HarvardX - PH125.8x)”.

Analysis Section

Data Cleansing & Preparation

As described in the Executive summary section, the provided movielens dataset will be split into two datasets, where the train dataset is named **edx** and the test dataset is called **validation**, using the scripts provided in the Capstone course.

After running the script provided in the instructions of the course, we obtain the data set **edx**, with the following characteristics :

- number of rows : 9000055
- number of columns (variables) : 6

The variables (columns) in the dataset :

- movieId
- userId
- Title
- rating
- genres
- timestamp

The values of the “genres” variable could describe a simple genre of the movie, like “Romance” or “Comedy” or “Action”, or a composite genre of the movie like “Action|Comedy” or “Action|Comedy|Romance”.

The values of the “Title” variable is in the format : “Movie Title (year of release)”. From here we can process this variable and extract the year of the release for each movie, into a new variable of the dataset: example “Jumanji (1995)”

The year of release will be used later, in the analysis, in combination with the genre of the movie, such combination defining the effect “fashion” in the data.

Example of **edx** train data set:

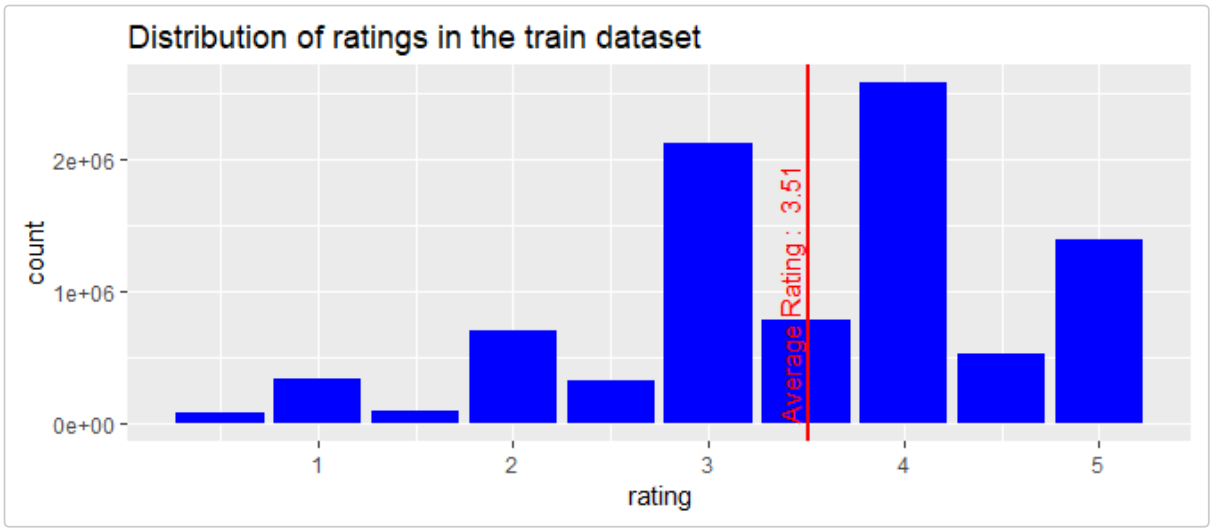
	userId	movieId	rating	timestamp	title	genres	year
1	1	122	5	838985046	Boomerang (1992)	Comedy|Romance	1992
2	1	185	5	838983525	Net, The (1995)	Action|Crime|Thriller	1995
4	1	292	5	838983421	Outbreak (1995)	Action|Drama|Sci-Fi|Thriller	1995
5	1	316	5	838983392	Stargate (1994)	Action|Adventure|Sci-Fi	1994
6	1	329	5	838983392	Star Trek: Generations (1994)	Action|Adventure|Drama|Sci-Fi	1994
7	1	355	5	838984474	Flintstones, The (1994)	Children|Comedy|Fantasy	1994

Data Exploration

The **edx** dataset contains 10,677 different movies rated by 69,878 different users.

The entire data set has : 7 variables (including the newly added one = year) and 9000055 rows (observations).

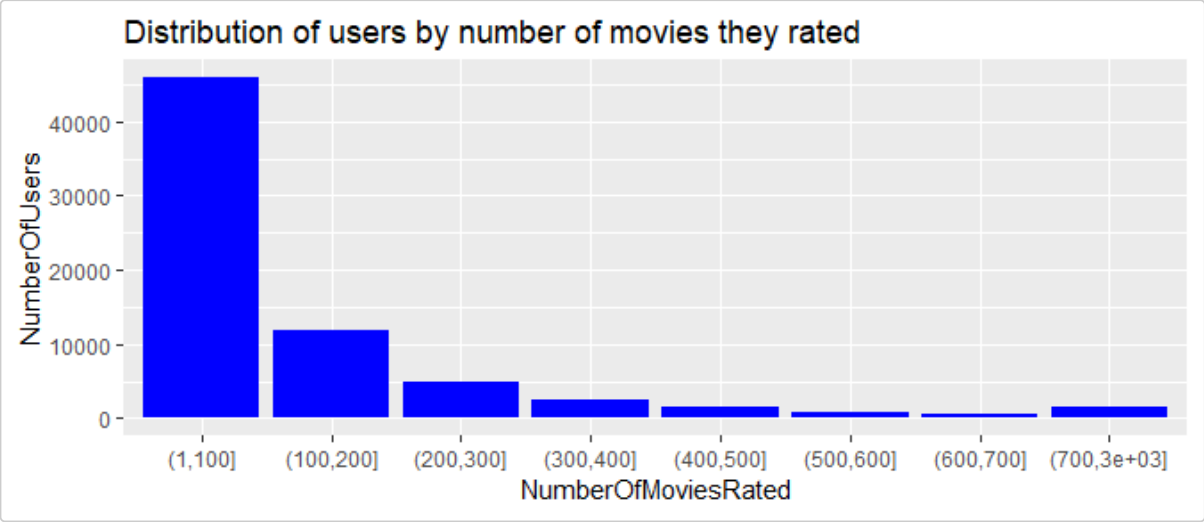
All movies in the train dataset **edx** are rated with ratings falling in the following set of values: {0.5, 1.0, 1.5 ,2.0 ,2.5 ,3.0 ,3.5 ,4.0 ,4.5 ,5.0} , with the average rating of 3.54. The graph below presents the distribution of ratings for the entire train data set.



Data Visualisation

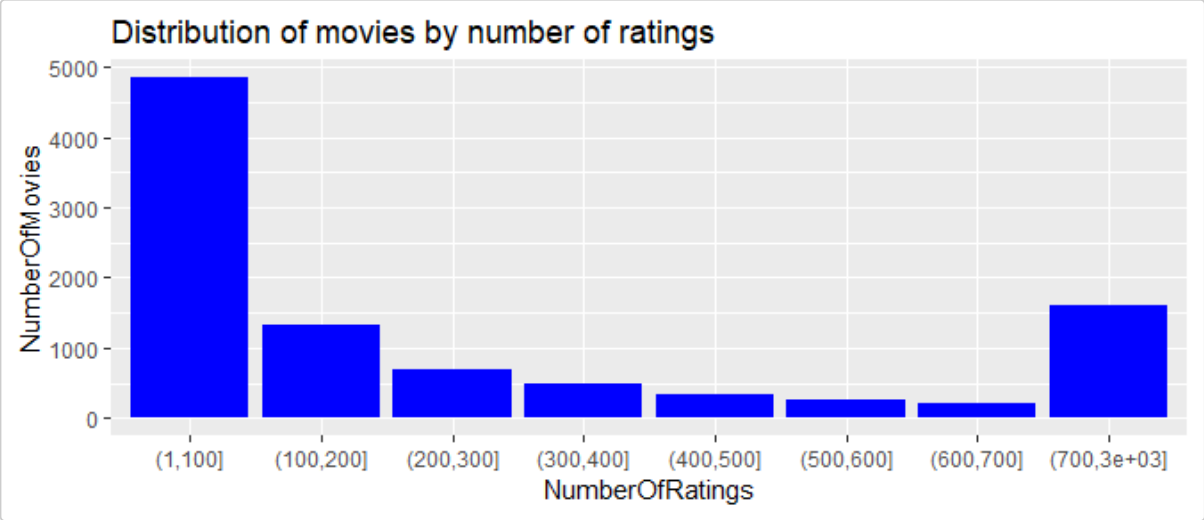
User effect

As the users gave a rating to the movies they saw, we notice that some users are very active in rating the movies, meanwhile other they have just few ratings expressed - though we believe they saw more movies than the number of movies they rated.



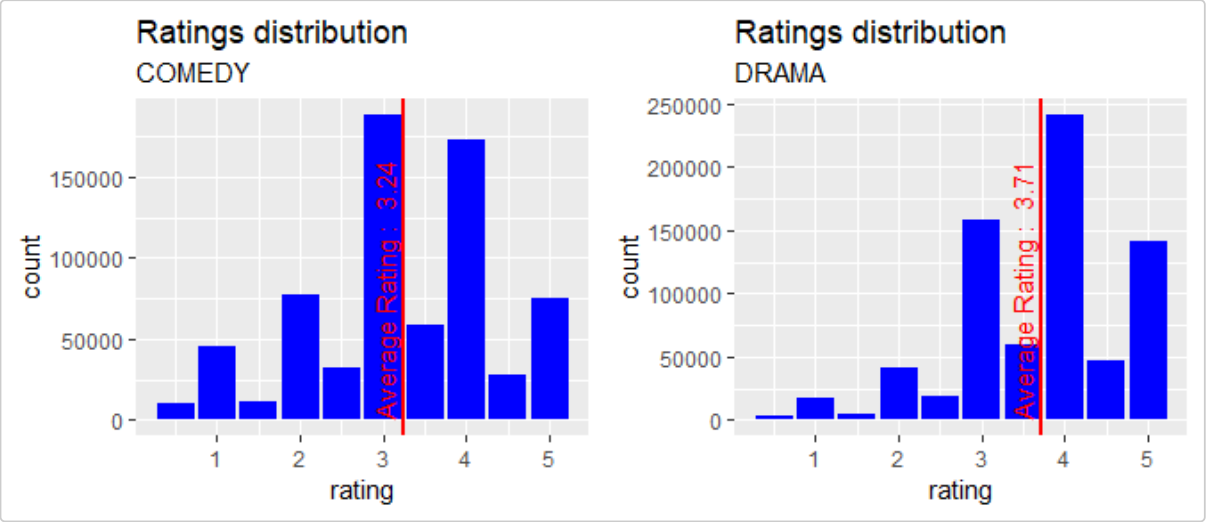
Movie effect

Also, some movies have far more rating than others, even if there are cases of different movies with very different number of rating were reported to have relatively similar audience by box office statistics.



Genre effect

The genre is also important in the distribution of ratings, and as an example we illustrate in the following graph the comparative distribution of ratings for Comedy and Drama. respectively:



It is very clear from the graph above that different genres have different distribution of ratings, and average of ratings. As in the example above we see a significant difference in average rating between the two exposed genres, a difference of almost 0.5 points (almost half a star).

Fashion effect (Year + Genre)

Another important factor of influence in ratings distribution and average variation can be noticed even inside a specific genre, if we compare the ratings, for the same genre but for different years. We can call this combination (year, genres) as the “fashion” since basically the fashion current can be determinant in the taste of public from one year to another, for the same type of item, be it a movie in a simple genre category “Comedy”, “Romance” or “Action”, or in a composite genre category as “Action|Comdey”, “Action|Comedy|Romance” and so on.

The following table presents the “fashion” influence in rating for the genre “Comedy”, for a those years where more than 100 ratings were counted.

Genre Commedy - Average rating by year

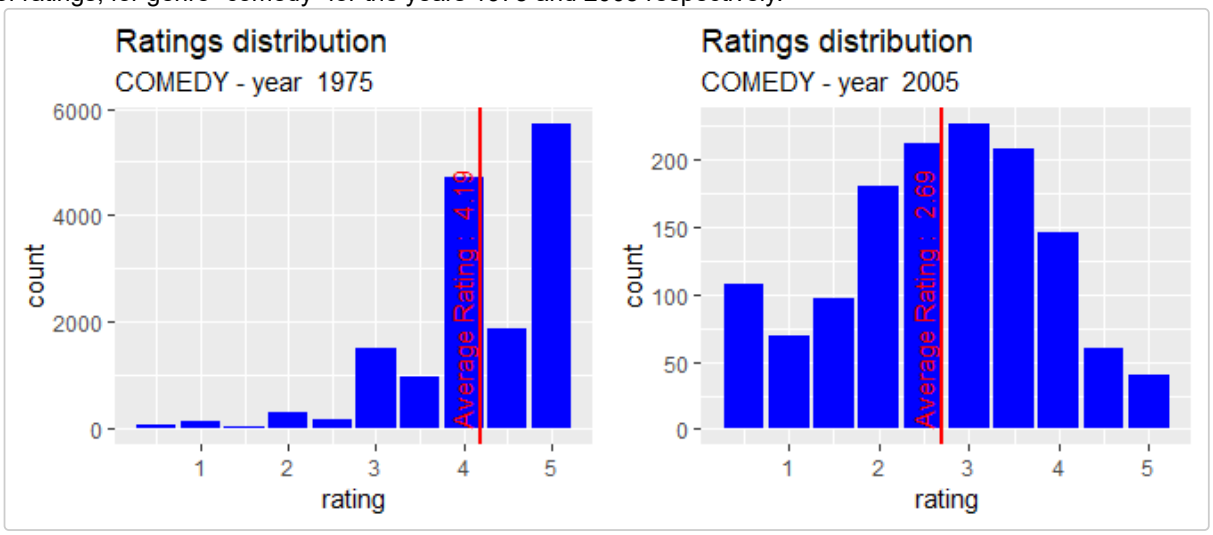
year	NumberOfRatings	avarage_rating
1975	15595	4.19
1936	850	4.12
1938	2205	4.05
1971	2775	4.00
1942	614	3.99
1930	397	3.99
1932	397	3.99
1940	329	3.99
1939	490	3.95
1958	237	3.95
2006	8444	3.46
2007	2972	3.31
2004	12797	3.28
2000	32820	3.21
2003	13274	3.04
2001	29923	3.02

year	NumberOfRatings	avarage_rating
2002	10076	2.96
2008	2294	2.96
2005	1351	2.69

We notice in the above table the effect “fashion” over the years for the same genre “Comedy”. We see the average ratings for the year of 1975 was 1.5 point(s) bigger than 2005 - that means a significant difference in taste over a span of -30 years, coming from “fashion” effect.

We also notice that the average rating for movies during 1975 and previous years is significantly higher than the average rating of the movies released -30 year(s) later, in the genre “Comedy”.

For exemplification of the effect “fashion” on ratings, the following graph presents comparatively, the distribution of ratings, for genre “comedy” for the years 1975 and 2005 respectively.



Insights gained & modeling approach

Insights

From data exploration and visualization we have the following insights:

1. Insight 1: **Movie effect** : there is a movie effect observed in data that we can use in the model to predict ratings for similar movies.

The Movie effect practically considers the notoriety of the movie, its impact on the audience and how many ratings it receives during a certain period. As a general observation, the bigger the number of ratings for a movie at a certain moment, the closer the subsequent ratings are to the average rating.

There are many examples of movies creating a current of opinion that is convergent into a certain direction as fashion, ideals or social habits. Some movies have a strong personality and they might present a new perspective that is generally accepted or assimilated, raw models or examples of humanity that are able to traverse borders, cultures and continents.

2. Insight 2: **User effect** : there is a user effect noticed in data that we can use in the model to predict ratings for movies rated by similar users.

The user effects is more related to the user, and it is supposed to consider the taste of the user, her/his orientation and tendency in rating an element of art, like a movie.

3. Insight 3: **Fashion effect** : there is a fashion effect noticed in data, described by the combination (year, genres) that we can use in the model to predict ratings for movies falling into a certain genre category, released in a certain year, based on the overall taste for different genres in different years of release

The fashion effect found for this dataset is defined by the combination of the year when the movie was released with the genre of that movie. It is supposed to consider the general public changes in taste from a period to another, for the same type of item : in this very instance, type described by the genre.

The taste of the public changes over the years, at both levels, collectively as well as individually, and that is a determinant for movie producers to change their methods/approach when producing movies. The movie producers thus, they propose changes and new perspectives in their movies that at their turn become a catalyst for new changes in the public tastes and consequently ratings.

We may noticed that many categories of movies in the past - like “Romance” or “Drama” - they were more descriptive,more analytical, with dense dialogues meanwhile recent productions have a tendency to be more focused on special effects, visual and audio, even though the core message is the same for both old and new movies: the good will overpower the evil, the success comes with effort and dedication, and so on.

4. Insight 4: **regularization** : as some movies have more than 100 ratings meanwhile others they have less than 10, and, some users are rating more than 100 movies meanwhile some other are rating less than 10, it is very clear that movie effect as well as user effect have equal relative importance in the total general rating effect.

For this reason, regularization comes into place to adjust this importance of movie effect and user effect in contributing to the total overall effect - practically permitting to penalize large estimates coming from small sample sizes.

Modeling

As mentioned in the executive summary, the modeling will consider the methodology and modeling described in the Harvard course “Data Science: Machine Learning (HarvardX - PH125.8x)”, and it considers the insights above mentioned as they were gathered and obtain from the Analysis section.

The final model, that considers the insights and effects above mentioned, is in the following form:

Y = $\mu + b\_i + b\_u + b\_y\_g + \epsilon$

with the predicting ratings being the result of the following equation :

$$\hat{Y} = \mu + b\_i + b\_u + b\_y\_g$$

and the rezidual :

$$\epsilon = Y - \hat{Y}$$

Where :

Element	Description
Y	the real rating
$\hat{Y}$	the predicted rating
μ	the general average rating for the train dataset **edx**
b_i	the movie effect (item effect)
b_u	the user effect
b_y_g	the fashion effect (described by the combination **year& genres**)
ϵ	the error

As mentioned in the executive summary, this model effectiveness will be measured using RMSE :

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2}$$

General tendency

The methodology used to build this model is organised in cascade, at each level fitting the residual obtained by removing the influence identified at the precedent level. At the first level, the main influence is quantified through the general average of the ratings, for the entire dataset : μ.

[1] 3.512465

Thus the model at this level becomes : $Y = \mu + \epsilon$

Movie (item) effect

The second level in the cascade calculates the movie (item) effect in the residual from the preceding level of this cascading methodology :

$$\epsilon = rating_i - \mu$$

for every *i* movie.

Thus the model at this level becomes : $Y = \mu + +b\_i + \epsilon$ where *b_i* is obtained with the following R code :

```
bi <- edx %>% group_by(movieId) %>% summarise(b_i = mean(rating - mu))
```

User effect

The third level in the cascade considers the user effect in trying to fit the residual from the previous level :

$$\epsilon = rating_u - \mu - b\_i$$

for every *u* user.

Thus the model at this level becomes : $Y = \mu + +b\_i + b\_u + \epsilon$ where *b_u* is obtained with the following R code :

```
b_u <- edx %>% left_join(b_i, by='movieId') %>%

group_by(userId) %>%

summarize(b_u = mean(rating - mu - b_i))
```

Fashion effect

The fashion effect is introduced into the model by efforts to fit the residual (error) resulting at the previous level, grouping data by two variables (year & genres), with the following code

```
b_y_g <- edx %>% left_join(b_i, by='movieId') %>% left_join(b_u, by='userId') %>%
group_by(year, genres) %>% summarize(b_y_g = mean(rating - mu - b_i - b_u) )
```

Regularisation

Regularization is implemented in the model, by adjusting the mean with a parameter

$$\lambda$$

(lambda) that has more effect on samples with small size and a lesser/negligible effect on large samples, with the following general formula :

$$Mean_{regularized} = \frac{1}{n+\lambda}\sum_{i=1}^{n} X_i$$

To determine the value of lambda, a bagging approach will be used on edx data train set. The trainset will be partitioned 10 times into two subsets :

1. the edx subset of test : circa 10% of the edx data set
2. the edx subset of train : the rest of the edx data set after eliminating the subset of test

For each edx subset of train a model fitting the data in the subset will be built, for a range of lambda values, model that will be tested then on the edx subset of test, and retaining the lambda value that gives the best RMSE. The R algorithm and code for obtaining the best regularization lambda value is available in the accompanying script file.

The following table illustrates the process of splitting the edx trains dataset into two subsets (10% for tests and 90% for train) that are used in determining the optimal value of λ (lamdba).

iteration	**edx original train set**										**lar**
	subset_1	subset_2	subset_3	subset_4	subset_5	subset_6	subset_7	subset_8	subset_9	subset_10	
1	test	train									4.0
2	train	test	train								4.5
3	train		test	train							5.0
4	train			test	train						4.5
5	train				test	train					4.5
6	train					test	train				4.5
7	train						test	train			4.5
8	train							test	train		4.5
9	train								test	train	4.5
10	train									test	4.5

Following this process, it was found the value of lambda that gives the best RMSE result on the edx subsets train and tests :

$$\lambda = 4.5$$

With regularization, the entire modeling R program becomes as following:

```
lambda <- 4.5
mu <- mean(edx$rating)

b_i <- edx %>%
  group_by(movieId) %>%
  summarise(b_i = sum(rating - mu)/(n() + lambda))

b_u <- edx %>%
  left_join(b_i, by='movieId') %>%
  group_by(userId) %>%
  summarize(b_u = sum(rating - mu - b_i)/(n() + lambda))

b_y_g <- edx %>%
  left_join(b_i, by='movieId') %>%
  left_join(b_u, by='userId') %>%
```

```
  group_by(year, genres) %>%
  summarize(b_y_g = mean(rating - mu - b_i - b_u) )
```

Result section

And now, the model applied on **validation** data set to predict the ratings for this original test dataset, results with the following RMSE:

```
predicted_ratings <-
  validation %>%
  left_join(b_i, by = "movieId") %>%
  left_join(b_u, by = "userId") %>%
  left_join(b_y_g, by = c("year","genres")) %>%
  mutate(pred = mu + b_i + b_u + b_y_g) %>%
  .$pred

RMSE <- sqrt(mean((predicted_ratings - validation$rating)^2))

paste('RMSE = ', RMSE)
```

[1] “RMSE = 0.864046097966375”

The resulting RMSE is :

RMSE = 0.86404

Conclusion section

As we noticed during the entire process of building this recommendation system, considering movie effect, user effect and fashion effect - as combination of genre and year of release for the movie - could lead to a good recommendation model with a good measure of effectiveness : RMSE = 0.86404

However, the accuracy might still be an aspect to improve since a very good RMSE might not guarantee a high accuracy.

The model could be improved by refining the analysis, considering the behaviour of the user in different periods of time and for different genres as well as the moment when the rating is given by the user,moment defined by the timestamp in the dataset. That moment describes the number of ratings the respective movie had accumulated at the time when a certain user decides to give a rate for the movie, and the more ratings a movie accumulates the more probable is the new ratings will be closer to the average rating for that movie.

references

1. Ref. : EDX course-Data Science: Machine Learning (harvardX: PH125.8x)
2. Ref. : EDX course-Data Science: Capstone (harvardX: PH125.9x)
3. Ref. : Book - Data Analysis for the Life Science (authors: Rafael A. Irizarry & Michael I. Love)