MovieLens RMSE Project for course PH125.9X

Miguel Angel Bustos Sáez

11/30/2021

Table of Contents

[INTRODUCTION 1](#_Toc89342421)

[PART ONE 2](#_Toc89342422)

[Edx data set at the glance 2](#_Toc89342423)

[Global Visualization Dataset 2](#_Toc89342425)

[Edx1 it`s the ordered data, Each variable in each column 3](#_Toc89342426)

[Edx1 ranking performance visualization between 2005 to 2008, taking a sample of 1000 observations 3](#_Toc89342427)

[Edx1 ranking distribution 4](#_Toc89342428)

[Frequency of movies per Genre, the most viewed 5](#_Toc89342429)

[Action Movie Frequency, that have a rating of 5 5](#_Toc89342430)

[PART TWO 6](#_Toc89342431)

[Modeling data 6](#_Toc89342432)

[First RMSE [1] 0.9436553 7](#_Toc89342433)

[Second RMSE [1] 1.102293 8](#_Toc89342434)

[Third RMSE [1] 0.7399887 8](#_Toc89342435)

[*Customizing what`s good for the Movie Espectators* 8](#_Toc89342436)

[CONCLUSION AND RECOMMENDATION 10](#_Toc89342437)

[Biography 10](#_Toc89342438)

[About optimizing RMSE or choose samples 10](#_Toc89342439)

## 

## INTRODUCTION

Aproved the eight R programming language courses, this is the first of two projects, corresponding to finish the program and to obtain the HarvardX, Professional Certificate in Data Science. This report, it´s about the MovieLens Data Set and the submission of making a RMSE or Root-Mean Square of Error model, that can make good recommendations to movies spectators.

Because of the covid-19, many companies were obligated to offer their services or products through internet, using strong social networks campaigns as marketing strategy, but others using apps or the webpage to increment the sales transactions, good but what about make good recommendations?

The importance it´s to have the option to choose a good recommended product or service, related that were viewed or with a good evaluation criteria as a rating, star or other, for that reason it`s very important to appreciate the good recommendation products because that will push clients to buy these recommended products or services.

In this project, the main objective is to visualizing how an algorithm can reach that challange, how this project make an customized recommendation of good movies. For example, many apps, are doing this, international retailers and some national companies are determinatig the importance to implement intelligent and predicted ways to make good recommendation to customers, clients, netflix viewers, spotify listeners and many others offerings in the market.

One of the best recommendation systems is the RMSE method, this project shows a good data analysis and RMSE algorithm to make good movies recommendations.

## PART ONE

### Edx data set at the glance

### Global Visualization Dataset

This is EDX dataset table visualization, 0.5 to 5 rankings columns and year files from 1915 to 2008

table(edx1$year, edx1$rating)

### Global

*“The good of this code, is that we can easily visualize the year and rating”*

### Global Visualization Dataset

Structure view

str(edx)  
head(edx)  
dim(edx)



### Edx1 it`s the ordered data, Each variable in each column

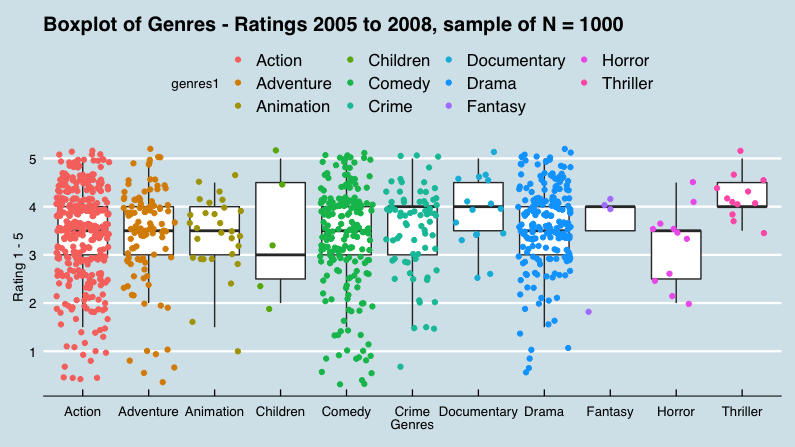
Each genre in each column, including year variable



### Edx1 ranking performance visualization between 2005 to 2008, taking a sample of 1000 observations

The latest years of the data set visualization has a lot of action ranked values

muestra1 = edx1 %>%   
 filter(year %in% 2005:2008) %>%  
 sample\_n(1000)  
  
ggplot(data = muestra1, aes(genres1, rating))+  
 geom\_boxplot(outlier.shape = NA)+  
 geom\_point(position = "jitter", aes(color = genres1))+  
 theme\_economist()+  
 labs(x = "Genres", y = "Rating 1 - 5", title = "Boxplot of Genres - Ratings 2005 to 2008, sample of N = 1000")

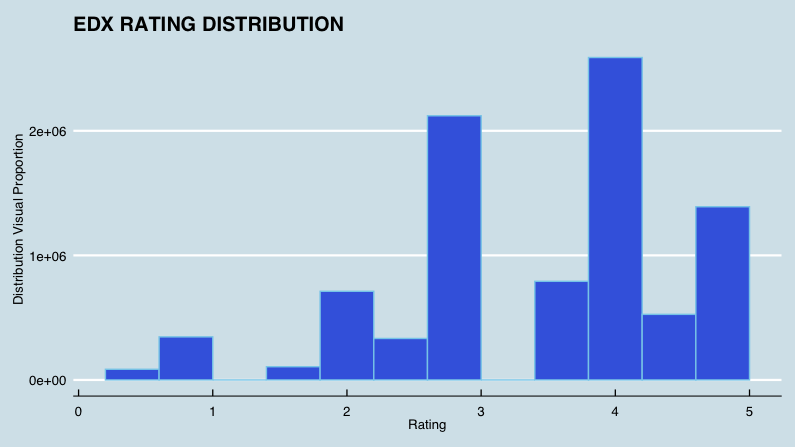


*“Ratings boxplot from 2005 to 2008”*

### Edx1 ranking distribution

Now the ranked distribution from 0.5 to 5 is visualizing through this barplot

edx1 %>%   
 ggplot(aes(rating))+  
 geom\_histogram(binwidth = 0.4, fill = "royalblue", color = "skyblue")+  
 theme\_economist()+  
 labs(x = "Rating", y = "Distribution Visual Proportion", title = "EDX RATING DISTRIBUTION")

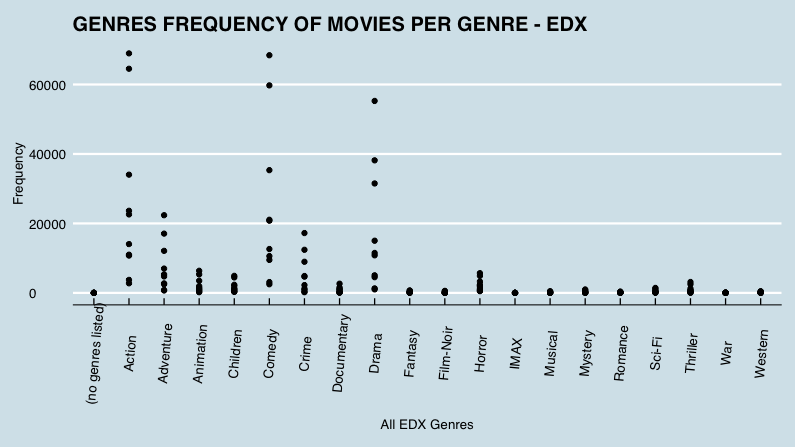


*“Four is the most frequently rating in all data set”*

### Frequency of movies per Genre, the most viewed

This shows the most movies viewed in all data set, the leaders are; Action, Comedy and Drama

genres\_frequency <- as.data.frame(table(edx1$genres1, edx1$rating))  
class(genres\_frequency)  
  
gf <- ggplot(data = genres\_frequency)  
gf + geom\_point(aes(x = Var1, y = Freq/10))+  
 labs(x = "All EDX Genres", y = "Frequency", title = "GENRES FREQUENCY OF MOVIES PER GENRE - EDX")+  
 theme\_economist()+  
 theme(axis.text.x = element\_text(angle = 85, vjust = 0.6))

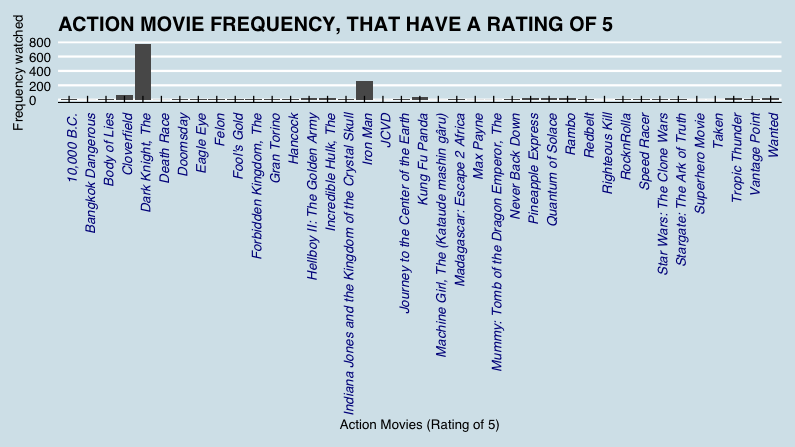


*“The most frequented movies per genre are action and comedy”*

### Action Movie Frequency, that have a rating of 5

The dark knight and the Iron Man are highly the most ranked movies of 5 ranked movies

top\_watched <- edx1 %>%   
 filter(rating == "5", genres1 == "Action", year == "2008")  
  
tg <- ggplot(data = top\_watched)  
tg + geom\_bar(aes(x = title))+  
 labs(x = "Action Movies (Rating of 5)", y = "Frequency watched", title = "ACTION MOVIE FREQUENCY, THAT HAVE A RATING OF 5")+  
 theme\_economist()+  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.8, color = "darkblue", face = "italic", hjust = 1))



*“The most action movies rated 5 viewed”*

## PART TWO

### Modeling data

The part 1 was critical to ordered the variables of genre and year and visualize it Now, that we have these variables in Edx, we can work more efficient in a RMSE The Root-mean-square error (RMSE) is the accuracy measure that calculate the prediction error rate. What are the residuals between “actual rating” and “predicted rating”? RMSE gives that answer In other words, the RMSE its the measure of how spread out these residuals are and how concentrated the data is around the line of best fit So its the difference between “Actual Ratings” and the Forecast or “predicted ratings”

test\_set & forecast\_rating

test\_set

mu <- mean(edx1$rating)  
  
set.seed(1)  
test\_set %>%   
 left\_join(edx1 %>%   
 group\_by(movieId) %>%   
 summarise(fe = mean(rating - mu)), by = "movieId")  
join <- test\_set %>%   
 left\_join(edx1 %>%   
 group\_by(movieId) %>%   
 summarise(fe = mean(rating - mu)), by = "movieId")  
  
  
forecast\_rating <- mu + join$fe   
forecast\_rating

Create bootstrap sample with createDataPartition

set.seed(1)  
test\_index <- createDataPartition(y = edx1$rating, times = 1, p = 0.5, list = FALSE)  
train\_set <- edx1[-test\_index,]  
test\_set <- edx1[test\_index,]

Visualizing the data sets

str(edx1)  
str(test\_index)   
str(train\_set)  
str(test\_set)

Selecting the test\_set\_r and the forecast\_rating

valid <- test\_set %>%   
 semi\_join(edx1, by = "movieId") %>%   
 semi\_join(edx1, by = "userId") #Return all rows from x with a match in y  
  
length(test\_set)   
  
test\_set\_r <- test\_set$rating  
forecast\_rating <- mu + join$fe

## First RMSE [1] 0.9436553

Using the test\_set\_r: 0.5 to 5.0, and forecast\_rating: different values (2.939235 3.738641 3.419572…)

mu <- mean(edx1$rating)  
mu  
  
set.seed(1)  
difference <- test\_set\_r-forecast\_rating  
rmse <- sqrt(mean(difference^2))  
  
rmse  
[1] 0.9436553  
  
or:  
  
sqrt(mean((test\_set\_r-forecast\_rating)^2))  
  
[1] 0.9436553

## Second RMSE [1] 1.102293

Using the test\_set\_r: 0.5 to 5.0, and now forecast\_rating: using ceiling function, that rounds up to the nearest integer, with more realistic number. This way shows that the residual reaches: [1] 1.102293.- And this is still too far:

set.seed(1)  
forecast\_rating2 <- ceiling(forecast\_rating)  
sqrt(mean((test\_set\_r-forecast\_rating2)^2))  
[1] 1.102293

## Third RMSE [1] 0.7399887

## *Customizing what`s good for the Movie Espectators*

Creating two vectors called test\_sample and forecast\_sample, we obtain #[1] 0.7399887 residuals value choosing in the range from 3.5 to 5.0 range.

Making good recommendations movies, over 3.5 ranked movies, is the way to get an optimal RMSE, for that reason, it`s neccessary to fix some parameters, that make a better recommendations:

Just replace the parameter, each of these, here we have some vectors parameters:

parameter1 <- 3.0   
parameter2 <- 3.5  
parameter3 <- 4.0  
parameter4 <- 4.5  
parameter5 <- 5.0

What will be the RMSE results, in the recommendation is more than these ratings?, here we have some vectors parameters results:

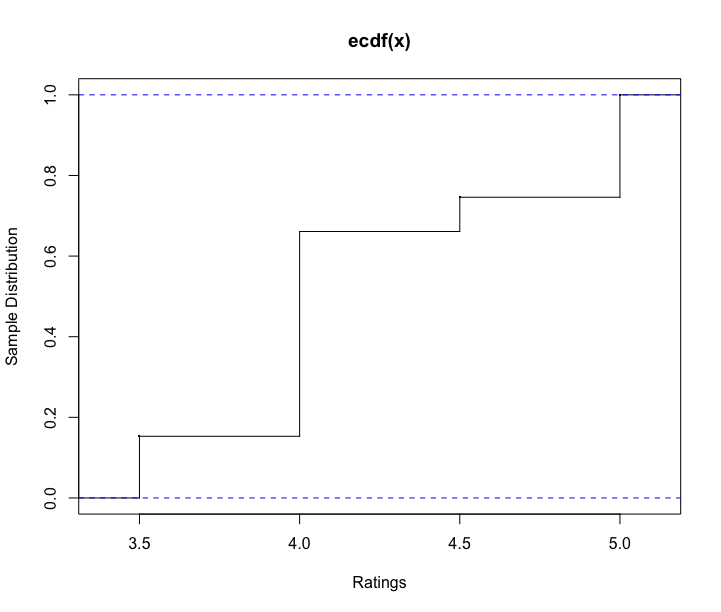
parameter1 : approximately [1] 1.000417 RMSE result  
parameter2 : approximately [1] 0.7399887 RMSE result  
parameter3 : approximately [1] 0.6379002 RMSE result  
parameter4 : approximately [1] 0.3106445 RMSE result  
parameter5 : approximately [1] 0 RMSE result, this is because it`s the limit ranking

The best RMSE it`s obtained using parameter3 [1] 0.6379002 rmse result, anyway , just using parameter2, we obtained a great recommendation [1] 0.7399887 RMSE result.

These vector parameters, can be changed in this code, and we will have different RMSE results, this code including >= symbol, so it is the best way to make recommendations to clients and obtain a better RMSE, making good recommensations movies, over 3.5 ranked value, that value is parameter 2:

set.seed(1)  
index1 <- edx1$rating >= parameter2 #<- You must here write the parameter vector 1, 2, 3, 4 or 5  
test\_sample <- edx1$rating[index1] %>% sample(size = 3000)  
test\_sample  
  
set.seed(1)  
index2 <- join$rating >= parameter2 #<- You must here write the parameter vector 1, 2, 3, 4 or 5  
forecast\_sample <- join$rating[index2] %>% sample(size = 3000)   
forecast\_simple\_mu <- join$fe[forecast\_sample]+ mu   
forecast\_sample

plot.ecdf(test\_sample, forecast\_sample, ylab = "Sample Distribution", xlab = "Ratings", col.01line = "blue", verticals = TRUE)



*“The sample N = 3000 ranking distribution, as ECDF Empirical Distribution Function”*

To obtain a RMSE value

sqrt(mean((test\_sample-forecast\_sample)^2))

[1] 0.7399887

## CONCLUSION AND RECOMMENDATION

With the intention to make good recommendations and not bad recommendations, the conclusion is recommend good movies, this is equal or greater than 3.5 rated.

For that reason, if the intention is to make a good recommendation, the critical decision is recommend a good movie to the viewers, that recommendation is over 3.5, and the gap will be more close, as a residual of 0.7399887 RMSE.

Nowadays many companies are offering online products and services, they won`t recommend a bad products, books, movies, licences, etcetera, these companies will recommend products well evaluated, to guarantee a great experience, happens the same in this project, it is necessary make good recomendations.

The recommendation is to recommend all movies that are equal or more than 3.5 ranked, so the spread will be roughly 0.7399887 RMSE and that guarantee the clients will have a great experience.

## Biography

Data Science HarvardX - [Professional Certificate in Data Science](https://www.edx.org/es/professional-certificate/harvardx-data-science)

Raphael Irizarry onnline dsbook - [Introduction to Data Science](https://rafalab.github.io/dsbook/)

Raphael Irizarry book - [Introduction to Data Science, CRC](https://www.amazon.com/-/es/Rafael-Irizarry/dp/0367357984)

Raphael Irizarry - [Webpage](https://rafalab.github.io/)

ImperialX, Imperial College Business School - [EDX Course, Data Analysis Essentials](https://www.edx.org/es/course/data-analysis-essentials)

Root Mean Squared Deviation - [Wikipedia](https://en.wikipedia.org/wiki/Root-mean-square_deviation)

## About optimizing RMSE or choose samples

The ratings members provided for recent movies provided more predictive power than older ratings - [2007 NETFLIX PRIZE](https://gibsonbiddle.medium.com/a-brief-history-of-netflix-personalization-1f2debf010a1)

They can sample a few videos before settling on one - [TECHDIRT](https://www.techdirt.com/articles/20120409/03412518422/why-netflix-never-implemented-algorithm-that-won-netflix-1-million-challenge.shtml)

Root-mean square of the differences between the predicted values and observed values - Even though it is feasible to conduct manual classification, it is generally very time consuming and error prone when dealing with a large number of samples - [NATURE](https://www.nature.com/articles/s41438-021-00560-9)

Having an accurate understanding of starch supports critical ration development and drives economic decisions. Invest in proper sampling, sample frequently to understand true variation - [HOARDS](https://hoards.com/article-30870-starch-brings-testing-concerns-to-corn-silage.html)

What is a Recommender System? A recommender system is an intelligent system that predicts the rating and preferences of users on products - [ANALITICALSINDIAMAG](https://analyticsindiamag.com/singular-value-decomposition-svd-application-recommender-system/)