Capstone Machine Learning Part 1

Tarasov Oleksiy

May 27, 2019

introduction

The target of this report is submission of MovieLens Project in accordance with the HarvardX edX online course HarvardX: PH125.9x Capstone Project. The training and test set is provided by the organiser of the course. THe purpose of the analysis is to predict rating of the films from the database MovieLens

Executive Summary

Task of the current summary was achived. The final model gives the accuracy (RMSE) of 0.866 that correspondes to the value stated in the task target 25 points: RMSE <= 0.87750

This chunk represent upload of the data from the code provided in the training code.it creates edx set, validation set, and submission file.

```
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.u
s.r-project.org")
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-proj
ect.org")

# MovieLens 10M dataset:
# https://grouplens.org/datasets/movielens/10m/
# http://files.grouplens.org/datasets/movielens/ml-10m.zip

dl <- tempfile()
download.file("http://files.grouplens.org/datasets/movielens/ml-10m.zip", d
l)

ratings <- read.table(text = gsub("::", "\t", readLines(unzip(dl, "ml-10M100
K/ratings.dat"))),
                      col.names = c("userId", "movieId", "rating", "timestam
p"))

movies <- str_split_fixed(readLines(unzip(dl, "ml-10M100K/movies.dat")),
"\\::", 3)
colnames(movies) <- c("movieId", "title", "genres")
movies <- as.data.frame(movies) %>% mutate(movieId = as.numeric(levels(movie
Id))[movieId],
                                           title = as.character(title),
                                           genres = as.character(genres))

movielens <- left_join(ratings, movies, by = "movieId")

# Validation set will be 10% of MovieLens data

set.seed(1)
test_index <- createDataPartition(y = movielens$rating, times = 1, p = 0.1,
list = FALSE)
edx <- movielens[-test_index,]
temp <- movielens[test_index,]

# Make sure userId and movieId in validation set are also in edx set

validation <- temp %>%
     semi_join(edx, by = "movieId") %>%
     semi_join(edx, by = "userId")

# Add rows removed from validation set back into edx set

removed <- anti_join(temp, validation)
edx <- rbind(edx, removed)

rm(dl, ratings, movies, test_index, temp, movielens, removed)
```

Library of the R language used for the solution of the Capstone project are activated in this chunk of the code

```
library(dplyr)
library(caret)
library(tidyr)
library(plotly)
library(tidyverse)
library(lubridate)
library(broom)
library(ggplot2)
```

Example of the data uploaded as a train dataset. Dataset has 6 columns. Columns describe userunique ID, Id of the movie, rating given by the user to certain movie, title of the movie and genres of the movie

```
as_tibble(head(edx,n=10))
```

```
## # A tibble: 10 x 6
##    userId movieId rating timestamp title             genre
s
##     <int>   <dbl>  <dbl>     <int> <chr>             <chr
>
##  1      1     122      5 838985046 Boomerang (1992)  Comedy|Romanc
e
##  2      1     185      5 838983525 Net, The (1995)   Action|Crime|Thrille
r
##  3      1     231      5 838983392 Dumb & Dumber (1~ Comed
y
##  4      1     292      5 838983421 Outbreak (1995)   Action|Drama|Sci-Fi|
T~
##  5      1     316      5 838983392 Stargate (1994)   Action|Adventure|Sci
-~
##  6      1     329      5 838983392 Star Trek: Gener~ Action|Adventure|Dra
m~
##  7      1     355      5 838984474 Flintstones, The~ Children|Comedy|Fant
a~
##  8      1     356      5 838983653 Forrest Gump (19~ Comedy|Drama|Romance
|~
##  9      1     362      5 838984885 Jungle Book, The~ Adventure|Children|R
o~
## 10      1     364      5 838983707 Lion King, The (~ Adventure|Animation|
C~
```

for the evaluation of the solution RMSE (Root Square Mean Error) Indicator will be used

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{u,i}\left(\hat{y}_{u,i} - y_{u,i}\right)^2}$$

Function RSME Definition. RMSE will be used as a target evaluation of the project.

```
RMSE <- function(true_ratings, predicted_ratings){
    sqrt(mean((true_ratings - predicted_ratings)^2))
  }
```

For the first model we will use the average rating as estimation of the model, Average value is 3.512 and it give s RMSE equal to 1.061

```
mu<-mean(edx$rating)
paste("Average rating = ",round(mu,digits = 3))
```

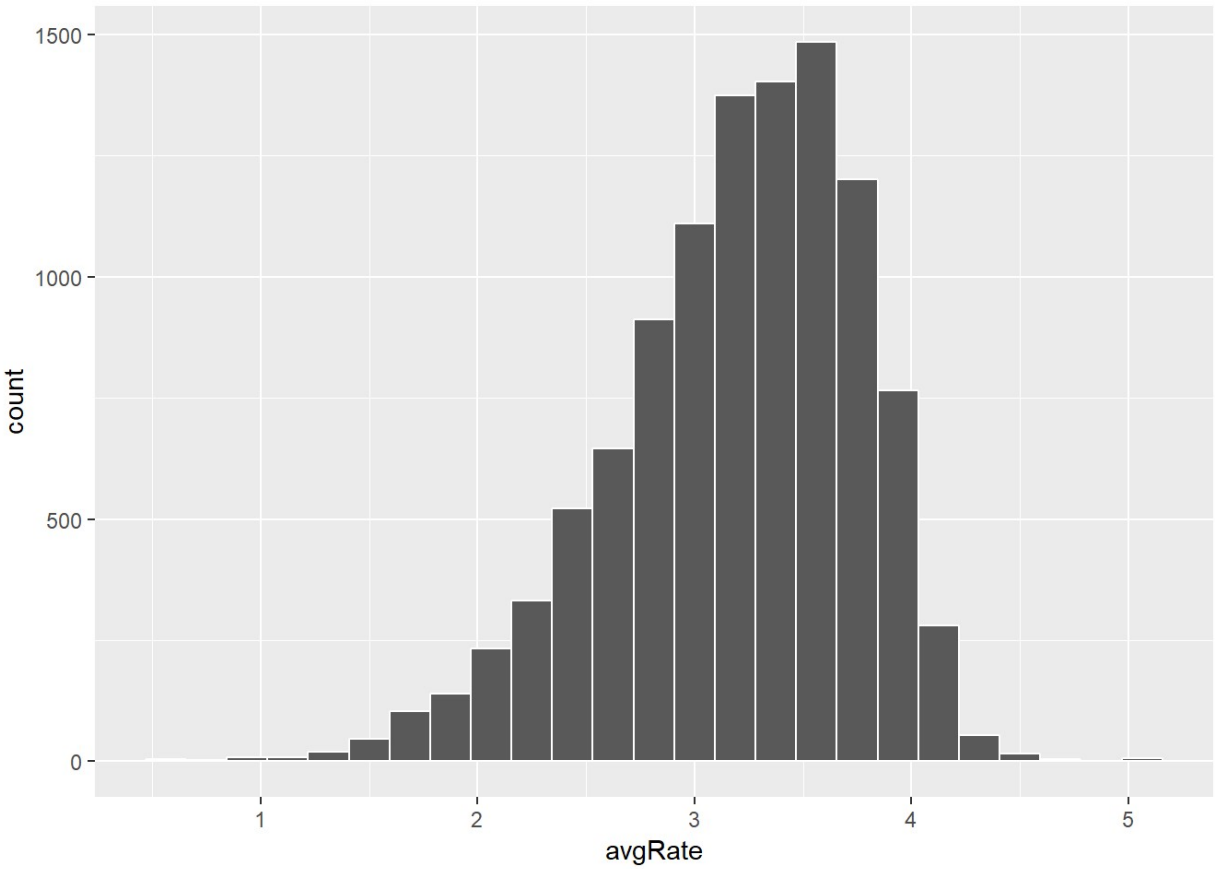
```
## [1] "Average rating =  3.512"
```

```
RMSE_naive<-RMSE(validation$rating,mu)
paste("RMSE Value by using only mean value = ",round(RMSE_naive,digits = 3))
```

```
## [1] "RMSE Value by using only mean value =  1.061"
```

Films have different overall rating. Histogramm shows distibution of the ratings.

```
edx%>%group_by(movieId)%>%summarise(avgRate=mean(rating))%>%arrange(avgRat
e)%>%select(avgRate)%>%ggplot(aes(avgRate))+geom_histogram(color="white",bin
s=25)
```



Adding the effects of the films rating to the model, as a difference to the average rating value gives

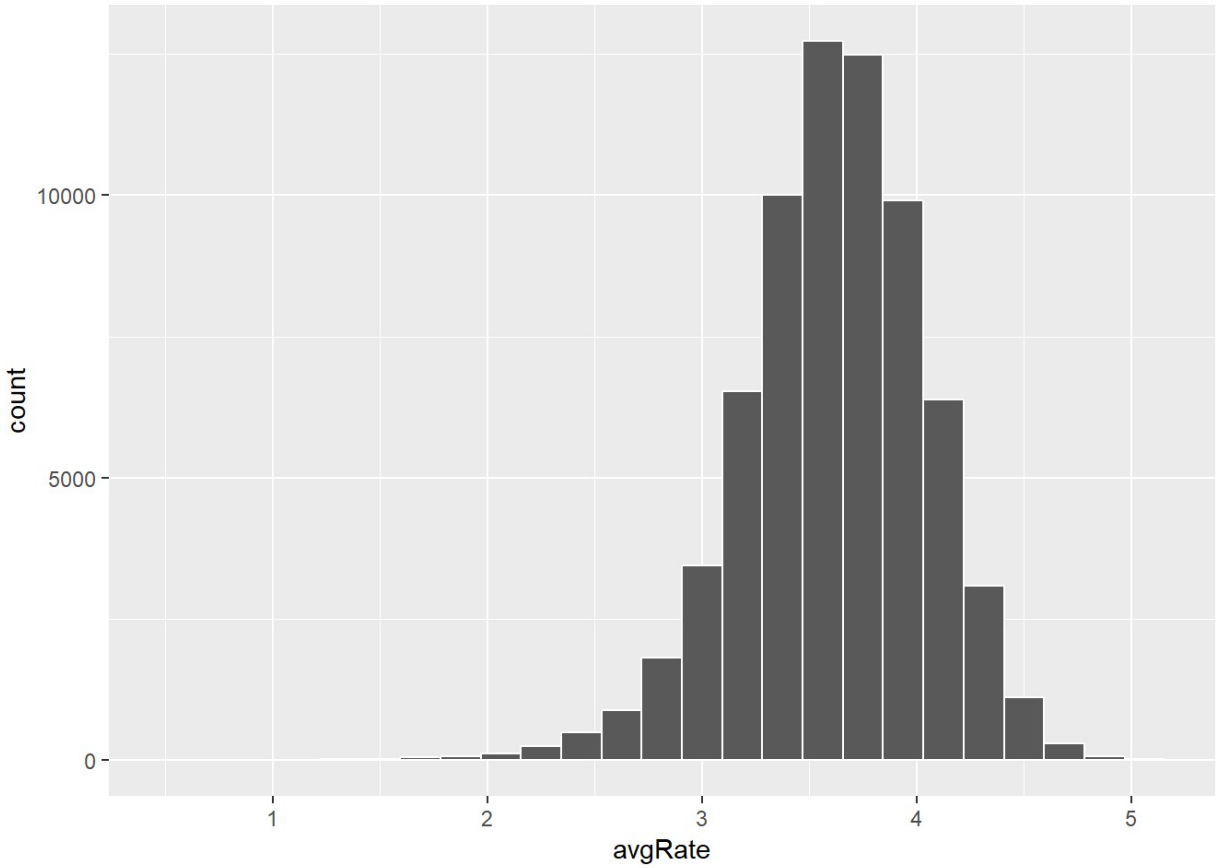
more precise model

```
movie_avgs <- edx %>%
  group_by(movieId) %>%
  summarize(b_i = mean(rating - mu))
predicted_ratings <- mu + validation%>%
  left_join(movie_avgs, by='movieId') %>%
  pull(b_i)
model_1_rmse <- RMSE(predicted_ratings, validation$rating)
paste("Model with one effect based on the film =",round(model_1_rmse,digits
= 3))
```

```
## [1] "Model with one effect based on the film = 0.944"
```

Different user has different pattern to the rating of the films, some make better some lower ratings

```
edx%>%group_by(userId)%>%summarise(avgRate=mean(rating))%>%arrange(avgRate)%
>%select(avgRate)%>%ggplot(aes(avgRate))+geom_histogram(color="white",bins=2
5)
```



Let us add the user bias in rating to the model

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

predicted_ratings <- validation %>%
  left_join(movie_avgs, by='movieId') %>%
  left_join(user_avgs, by='userId') %>%
  mutate(pred = mu + b_i + b_u) %>%
  pull(pred)
model_2_rmse <- RMSE(predicted_ratings, validation$rating)
print(paste("RMSE with 2 effects of user and Movie bias",round(model_2_rmse,
digits = 3)))
```

```
## [1] "RMSE with 2 effects of user and Movie bias 0.866"
```

Conclusion:

With the model based on mean rating corrected to the biases of users and movies we have reached the RMSE equal to 0.866. The accuracy target of the task is achieved

```
RMSE1<-data.frame(method="Naive",RMSE=RMSE_naive)
RMSE2<-data.frame(method="Movie bias effects Model",RMSE=model_1_rmse)
RMSE3<-data.frame(method="Movie and User bias effects Model",RMSE=model_2_rm
se)
ResultP<-bind_rows(RMSE1,RMSE2,RMSE3)
print(as_tibble(ResultP))
```

```
## # A tibble: 3 x 2
##   method                             RMSE
##   <chr>                             <dbl>
## 1 Naive                             1.06
## 2 Movie bias effects Model          0.944
## 3 Movie and User bias effects Model 0.866
```