HarvardX PH125.9x - Data Science: Capstone

Rodrigo Lange

2022-03-01

Contents

1 Introduction

This project is related to the HarvardX Data Science Course PH125.9x. The Capstone in this course requires the creation of a movie recommendation system using the MovieLens dataset, using the 10M version of this dataset (http://grouplens.org/datasets/movielens/10m/).

To train a machine learning algorithm, will be used the inputs in one subset to predict movie ratings in the validation set.

1.1 Dataset

This is the code to create Train and Final Hold-out Test Sets. I need to develop my algorithm using the edx set and predict movie ratings in the validation set (the final hold-out test set) as if they were unknown. RMSE will be used to evaluate how close the predictions are to the true values in the validation set (the final hold-out test set). I changed the code so it will check if the data set exists so it will not download again.

```
############################################################
# Create edx set, validation set (final hold-out test set)
############################################################

# Note: this process could take a couple of minutes

if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
if(!require(data.table)) install.packages("data.table", repos = "http://cran.us.r-project.org")

library(tidyverse)
library(caret)
library(data.table)

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

# Check if the file exists
datafile <- "MovieLens.RData"
if(!file.exists(datafile))
{
  print("Download")
  dl <- tempfile()
  download.file("http://files.grouplens.org/datasets/movielens/ml-10m.zip", dl)

  ratings <- fread(text = gsub("::", "\t", readLines(unzip(dl, "ml-10M100K/ratings.dat"))),
                   col.names = c("userId", "movieId", "rating", "timestamp"))

  movies <- str_split_fixed(readLines(unzip(dl, "ml-10M100K/movies.dat")), "\\::", 3)
  colnames(movies) <- c("movieId", "title", "genres")

  # if using R 4.0 or later:
  movies <- as.data.frame(movies) %>% mutate(movieId = as.numeric(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, sample.kind="Rounding") # if using R 3.5 or earlier, use `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)

  save(edx, validation, movielens, file = datafile)
  rm(dl, ratings, movies, test_index, temp, movielens, removed)
} else {
  load(datafile)
}
```

The edx and validation dataset contain 6 columns: “userId”, “movieId”, “rating”, “timestamp”, “title” and “genres”. Each row represents a single rating for a single movie.

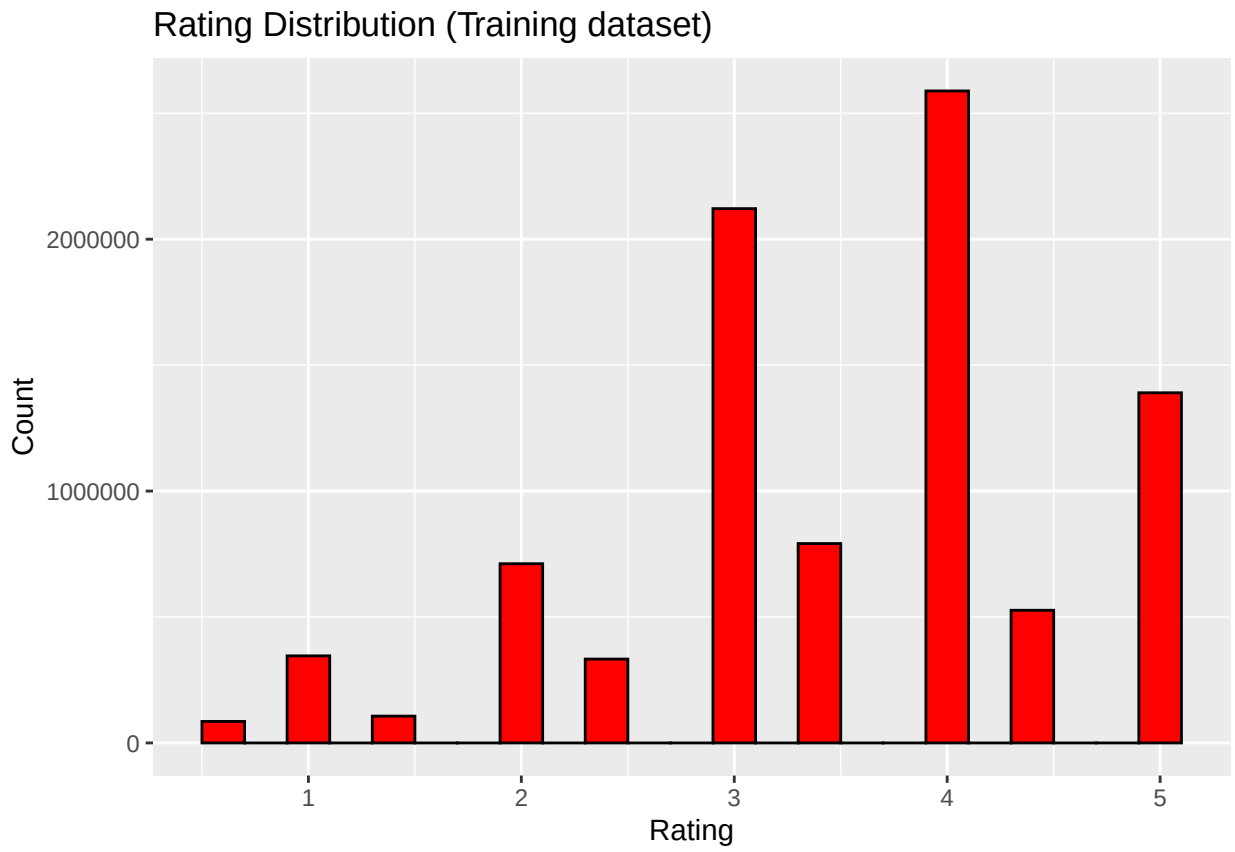
```
# Summarise Data
head(edx)
```

```
##    userId movieId rating timestamp                         title
## 1:      1     122      5 838985046              Boomerang (1992)
## 2:      1     185      5 838983525               Net, The (1995)
## 3:      1     292      5 838983421               Outbreak (1995)
## 4:      1     316      5 838983392               Stargate (1994)
## 5:      1     329      5 838983392 Star Trek: Generations (1994)
## 6:      1     355      5 838984474       Flintstones, The (1994)
##                            genres
## 1:                Comedy|Romance
## 2:         Action|Crime|Thriller
## 3:  Action|Drama|Sci-Fi|Thriller
## 4:       Action|Adventure|Sci-Fi
## 5: Action|Adventure|Drama|Sci-Fi
## 6:       Children|Comedy|Fantasy
```

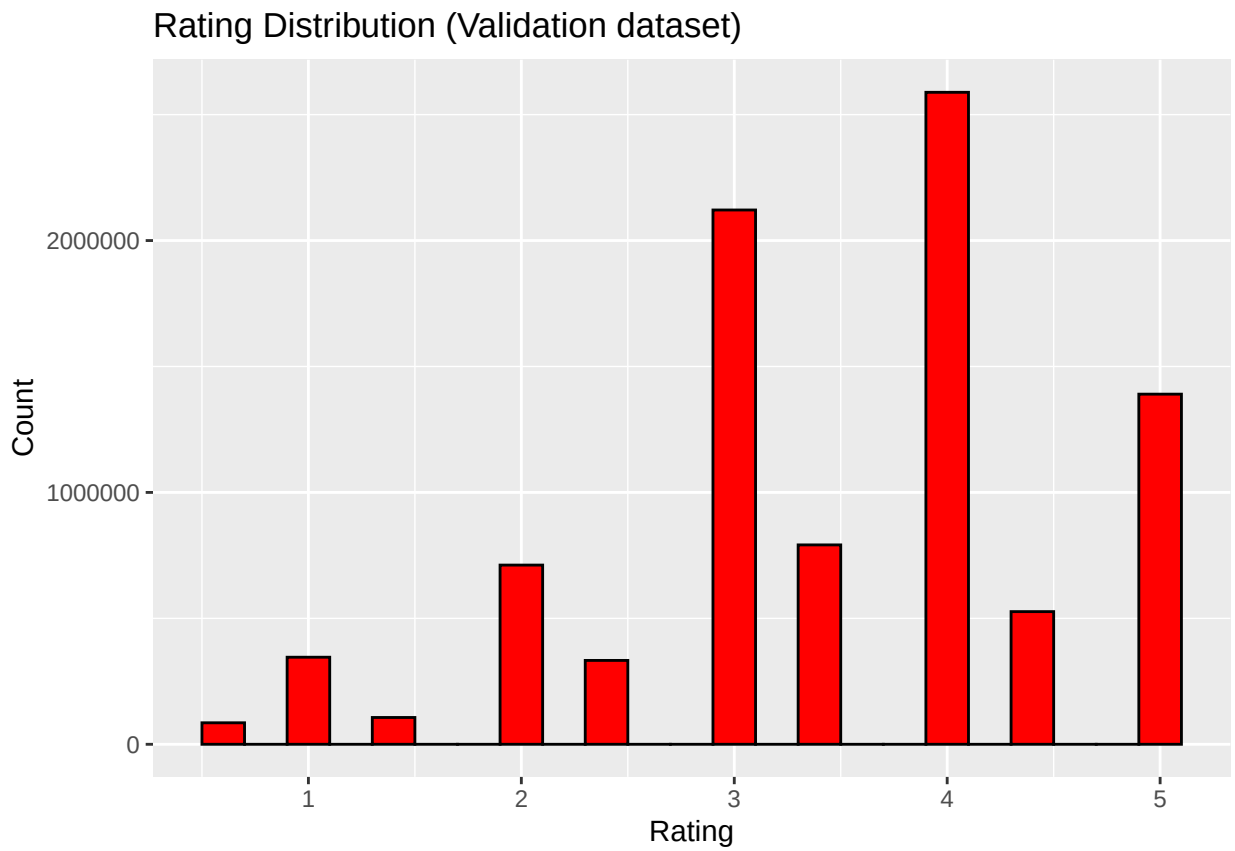
The lowest rating is 0.5 and the highest is 5 in edx and validation dataset.

```
# Review Training rating distribution
edx %>%
  ggplot(aes(x = rating)) +
  geom_histogram(binwidth=0.2, color="black", fill="red") +
  xlab("Rating") +
  ylab("Count") +
```

```
  scale_y_continuous(labels = function(x) format(x, scientific = FALSE)) +
  ggtitle("Rating Distribution (Training dataset)")
```




```
# Review Validation rating distribution
edx %>%
  ggplot(aes(x = rating)) +
  geom_histogram(binwidth=0.2, color="black", fill="red") +
  xlab("Rating") +
  ylab("Count") +
  scale_y_continuous(labels = function(x) format(x, scientific = FALSE)) +
  ggtitle("Rating Distribution (Validation dataset)")
```



2 Analysis

3 Results

4 Conclusion