

MovieLens

Daniel Teodoro

9/22/2019

Project 2 - MovieLens

Daniel Teodoro

Generates predicted movie ratings and calculates RMSE.

```
## Joining, by = c("userId", "movieId", "rating", "timestamp", "title", "genres")
```

Quiz

Q1) How many rows and columns are there in the edx dataset?

```
## [1] "The edx dataset has 7500096 rows and 6 columns."
```

Q2) How many zeros and threes were given in the edx dataset?

```
## [1] "0 ratings with 0 were given and 1768483 ratings with 3"
```

Q3) How many different movies are in the edx dataset?

```
##      n_movies
## 1         10677
```

Q4) How many different users are in the edx dataset?

```
##      n_users
## 1         69878
```

Q5) How many movie ratings are in each of the following genres in the edx dataset?

```
## [1] "Drama has 3259188 movies"
```

```
## [1] "Comedy has 2950958 movies"
```

```
## [1] "Thriller has 1938557 movies"
```

```
## [1] "Romance has 1426745 movies"
```

Q6) Which movie has the greatest number of ratings?

```
## # A tibble: 10,676 x 2
##   title                                     number
##   <fct>                                     <int>
## 1 Pulp Fiction (1994)                       26106
## 2 Forrest Gump (1994)                       25943
## 3 Silence of the Lambs, The (1991)          25306
## 4 Jurassic Park (1993)                     24385
## 5 Shawshank Redemption, The (1994)          23420
## 6 Braveheart (1995)                         21930
## 7 Terminator 2: Judgment Day (1991)         21773
## 8 Fugitive, The (1993)                     21671
## 9 Star Wars: Episode IV - A New Hope (a.k.a. Star Wars) (1977) 21427
## 10 Apollo 13 (1995)                        20250
## # ... with 10,666 more rows
```

Q7) What are the five most given ratings in order from most to least?

```
##
##           4           3           5           3.5           2
## -2157173 -1768483 -1158680 -659558 -592641
```

Q8) True or False:

In general, half star ratings are less common than whole star ratings (e.g., there are fewer ratings of 3.5 than there are ratings of 3 or 4, etc.).

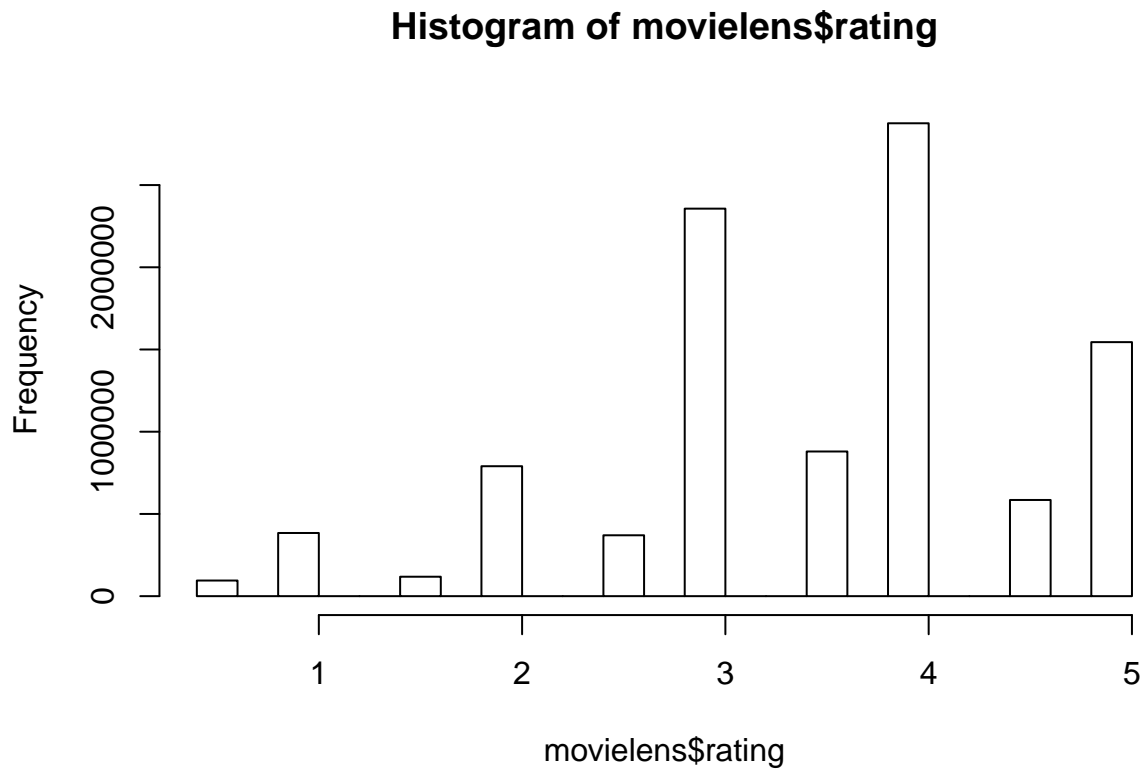
```
## [1] TRUE
```

Data Analysis

More than 10 million ratings.

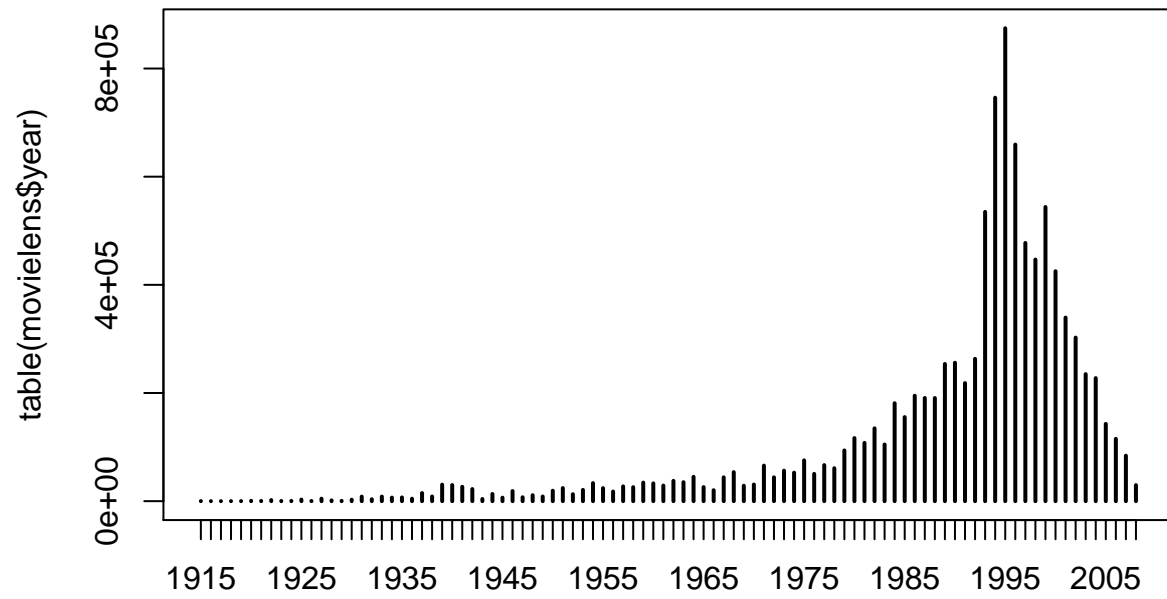
```
## 'data.frame':   10000054 obs. of  6 variables:
## $ userId   : int  1 1 1 1 1 1 1 1 1 1 ...
## $ movieId  : int  122 185 231 292 316 329 355 356 362 364 ...
## $ rating   : num  5 5 5 5 5 5 5 5 5 5 ...
## $ timestamp: int   838985046 838983525 838983392 838983421 838983392 838983392 838984474 838983653 838983653 838983653 ...
## $ title    : Factor w/ 10676 levels "...All the Marbles (a.k.a. The California Dolls) (1981)",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ genres   : Factor w/ 797 levels "(no genres listed)",...: 577 187 489 210 98 71 460 542 309 274 ..
```

The plot shows that 5 ratings are more common than 0.5.

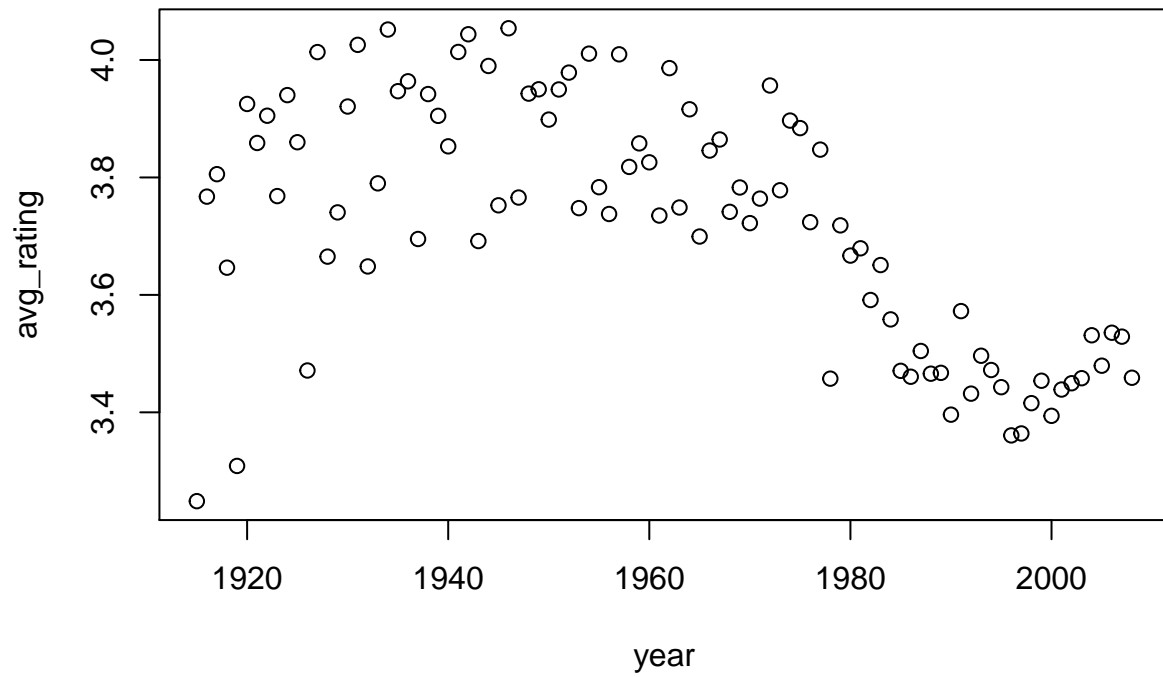


##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	0.500	3.000	4.000	3.512	4.000	5.000

Recent movies get more ratings.



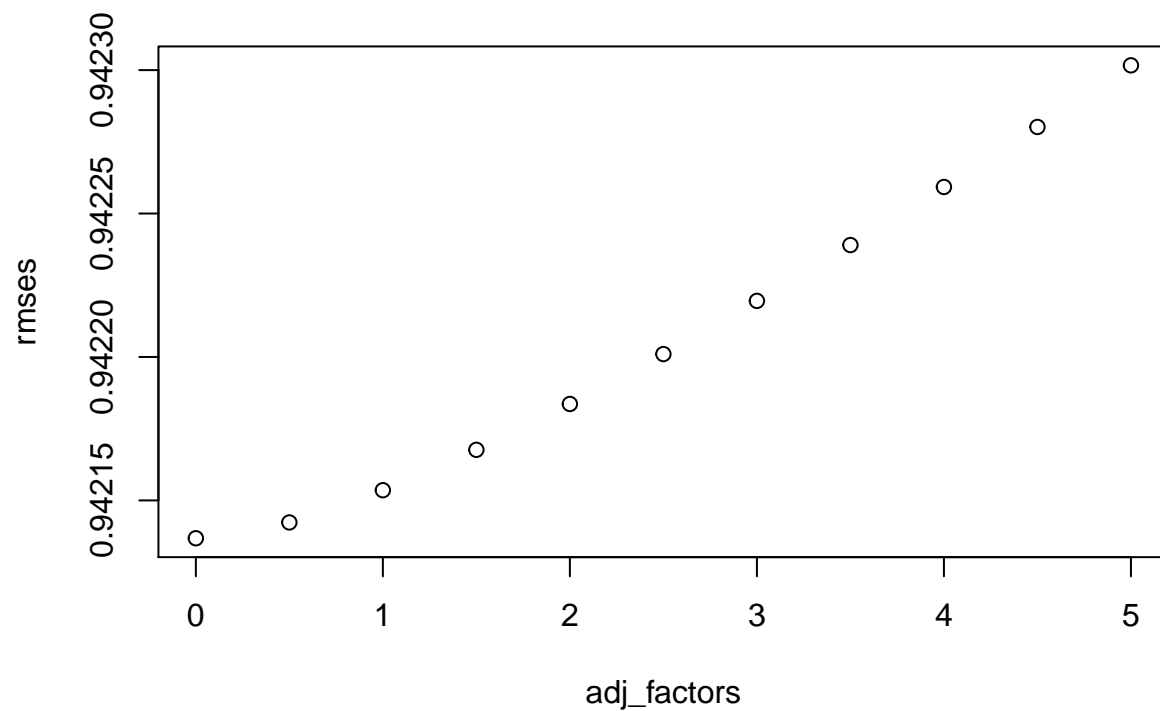
Last decades, ratings average get lower.



Results

RMSE

$$\sqrt{(TrueRatings - PredictedRatings)^2}$$



```
## [1] "Best RMSE: 0.942136809739054 is achieved with Adjustment Factor: 0"
```

Predictions

CSV file <predictions.csv>

```
predictions <- sapply(adj_factor,function(l){

  # The mean of training set
  mts <- mean(edx$rating)

  # Get movie effect with best adjustment factor
  me <- edx %>%
    group_by(movieId) %>%
    summarize(me = sum(rating - mts)/(n()+1))

  # Predict on validation set
  predicted_ratings <-
    validation %>%
    left_join(me, by = "movieId") %>%
    mutate(pred = mts + me) %>%
    .$pred
```

```
return(predicted_ratings)

})

write.csv(validation %>% select(userId, movieId) %>% mutate(rating = predictions),
          "predictions.csv", na = "", row.names=FALSE)
```

Conclusion

Using 1/4 as part of the data for the MovieLens validation set and removing the low rating

records in the training model, it was possible to obtain an RMSE index of 0.94 and,

verifying it with actual data, the model was reasonably efficient.

References

<http://www.montana.edu/rotella/documents/502/MarkdownEqnExamples.Rmd> <https://www.calvin.edu/~rpruim/courses/s341/S17/from-class/MathinRmd.html> http://www.datainsight.at/report/Capstone_04_fin.html <https://r4ds.had.co.nz> <http://stackoverflow/>