Movielens Project Report

Puneeth Nikin Krishnan

30/04/2020

Executive Summary

The objective of this project is to develop a recommendation system based on the movielens Dataset. To achieve this ratings need to be predicted for a user based on past ratings. The goal here is to accurately predict ratings. To be able to gauge the performance of different models RMSE (Root Mean Square Error) will be used. The dataset has been split into two for training(edx) and testing(validation) our model. The validation set will not be used except to validate the best performing model. The training set will be further split into a train set and a test set to identify the best model. The best model will be one which has the least RMSE. The steps followed to build this model are first importing the dataset, exploring or analysing and preparing the dataset, building the models, choosing the best model and finally validating the results on the bvalidation set.

Importing the Dataset

The code to import the dataset has been proveded by edx. https://courses.edx.org/login?next=/courses/course-v1%3AHarvardX%2BPH125.9x%2B1T2020/courseware/dd9a048b16ca477a8f0aaf1d888f0734/e8800e37aa444297a3a2f35bf84ce452/%3Fchild%3Dlast

Running the code will give us two dataframes namely edx and validation. The edx dataframe will be used to build our model.

Exploring and Preparing the Dataset

```
dim(edx)
```

```
## [1] 9000055       6
```

We notice that the dataset is pretty huge with 9,000,055 rows and 6 columns

```
names(edx)
```

```
## [1] "userId"    "movieId"   "rating"    "timestamp" "title"     "genres"
```

```
edx%>%head()
```

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

The edx dataframe comprises of 5 columns. ‘userId’ represents the unique user ID, ‘movieId’ represents the unique ID for each movie, timestamp denotes the time of the rating, title represents title and genres the combination of genres.

We can notice that the title section contains the year the movie was released in. We will use regex to extract the year released from this column. The regex pattern used to extract the data is “(\d{4})\)$” . This can be evidenced by the fact that the year is at the end of the string in column title. We seperate the

Table 1: Year extracted from column ‘title’

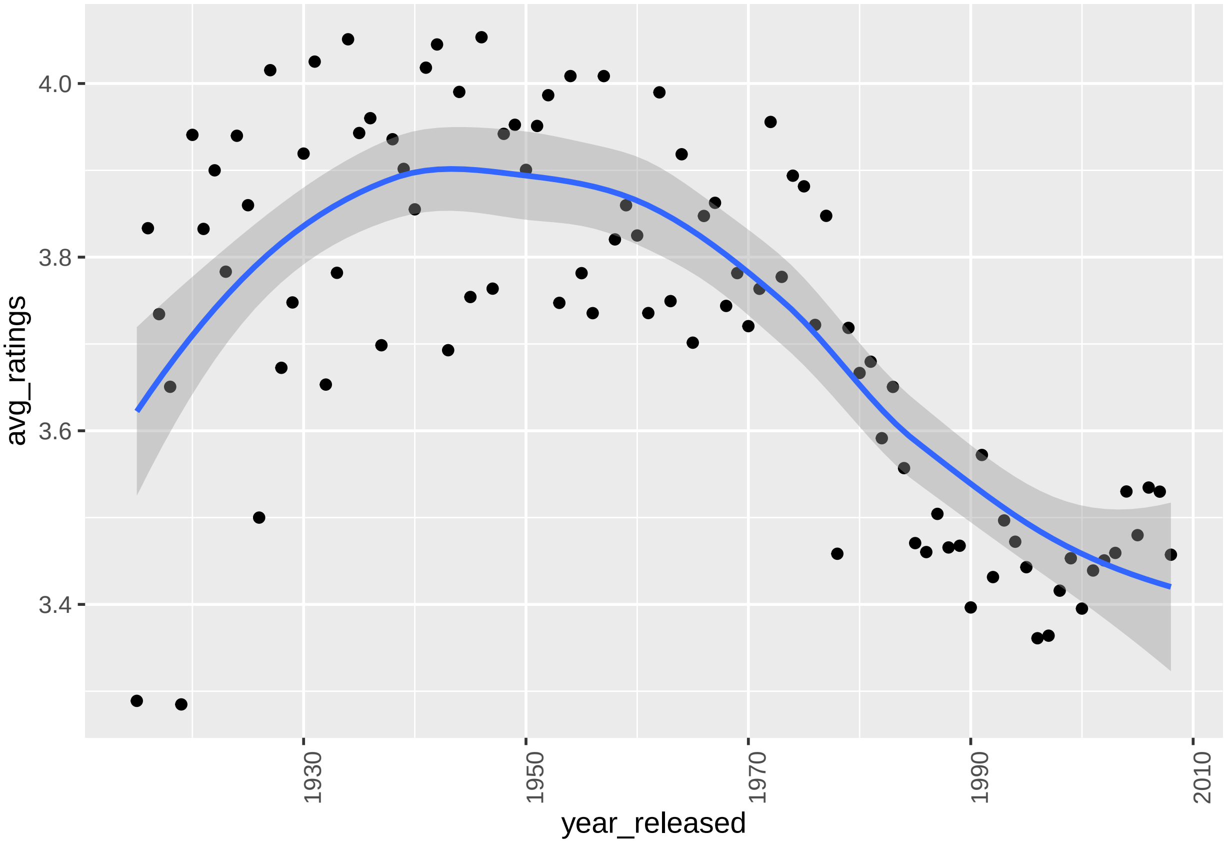
userId	movieId	rating	timestamp	title	genres	year_released
1	122	5	838985046	Boomerang (1992)	Comedy|Romance	1992
1	185	5	838983525	Net, The (1995)	Action|Crime|Thriller	1995
1	292	5	838983421	Outbreak (1995)	Action|Drama|Sci-Fi|Thriller	1995
1	316	5	838983392	Stargate (1994)	Action|Adventure|Sci-Fi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

We can convert the timestamp column to a readable datetime object.

Table 2: timestamp column to a readable datetime object

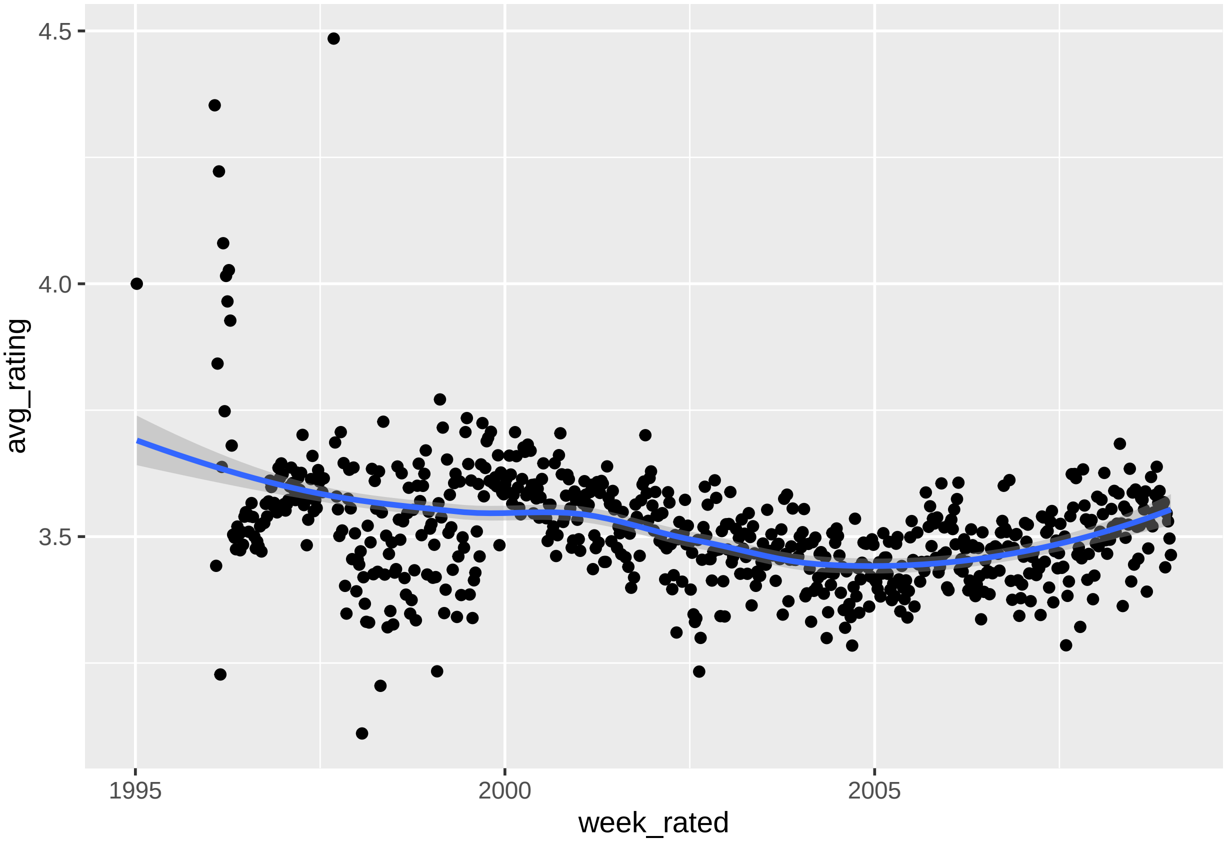
userId	movieId	rating	timestamp	title	genres	year_released
1	122	5	838985046	Boomerang (1992)	Comedy|Romance	1992
1	185	5	838983525	Net, The (1995)	Action|Crime|Thriller	1995
1	292	5	838983421	Outbreak (1995)	Action|Drama|Sci-Fi|Thriller	1995
1	316	5	838983392	Stargate (1994)	Action|Adventure|Sci-Fi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

Exploring the relationship between ratings and the year the movie was released



The plot shows that there is a clear dip in ratings for movies that were released after the late 1980s. There could be a relationship between the year released and the ratings as there are two clear clusters.

Exploring the relationship between ratings and when the movie was rated



The relationship does not seem to be strong between the year as a predictor and ratings.

Exploring the relationship between ratings and different genres.

```
unique(unlist(str_split(as.vector(unique(edx$genres)),"\\|")))
```

```
##  [1] "Comedy"             "Romance"            "Action"
##  [4] "Crime"              "Thriller"           "Drama"
##  [7] "Sci-Fi"             "Adventure"          "Children"
## [10] "Fantasy"            "War"                "Animation"
## [13] "Musical"            "Western"            "Mystery"
## [16] "Film-Noir"          "Horror"             "Documentary"
## [19] "IMAX"               "(no genres listed)"
```

```
length(unique(edx$genres))
```

```
## [1] 797
```

There are 20 unique genres and 797 unique combination of these genres(including "(no genres listed)")

Table 3: Top 10 genres by rating

genres	avg_ratings
Action|Adventure|Comedy|Fantasy|Romance	4.195557
Action|Crime|Drama|IMAX	4.297068
Animation|Children|Comedy|Crime	4.275429
Animation|IMAX|Sci-Fi	4.714286

genres	avg_ratings
Crime|Film-Noir|Mystery	4.216803
Crime|Film-Noir|Thriller	4.210157
Crime|Mystery|Thriller	4.198981
Drama|Film-Noir|Romance	4.304115
Film-Noir|Mystery	4.239479
Film-Noir|Romance|Thriller	4.216470

Table 4: Top 10 genres by number of ratings

genres	avg_ratings	n_ratings
Action|Adventure|Sci-Fi	3.507407	219938
Action|Adventure|Thriller	3.434101	149091
Comedy	3.237858	700889
Comedy|Drama	3.598961	323637
Comedy|Drama|Romance	3.645824	261425
Comedy|Romance	3.414486	365468
Crime|Drama	3.947135	137387
Drama	3.712364	733296
Drama|Romance	3.605471	259355
Drama|Thriller	3.446345	145373