MovieLens Project

Dylan Henderson

3/12/2021

INTRODUCTION

The relevant dataset in this project includes 10 million observations of movie ratings representing more than 70,000 unique users and 10,000 movies. In addition to a rating, other important variables were included such as movie title, year of release, and genres. The dataset was partitioned into a training set ("edx") and a final hold-out test set ("validation").

```
str(validation)
```

```
## Classes 'data.table' and 'data.frame':   999999 obs. of  6 variables:
##  $ userId   : int  1 1 1 2 2 2 3 3 4 4 ...
##  $ movieId  : num  231 480 586 151 858 ...
##  $ rating   : num  5 5 5 3 2 3 3.5 4.5 5 3 ...
##  $ timestamp: int  838983392 838983653 838984068 868246450 868245645 868245920 1136075494 1133571200 8444
16936 844417070 ...
##  $ title    : chr  "Dumb & Dumber (1994)" "Jurassic Park (1993)" "Home Alone (1990)" "Rob Roy (1995)" ...
##  $ genres   : chr  "Comedy" "Action|Adventure|Sci-Fi|Thriller" "Children|Comedy" "Action|Drama|Romance|Wa
r" ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

The goal of the project was to develop a model using the edx dataset in order to make predictions of movie ratings in the validation set. Accordingly, the validation set was used only to evaluate the accuracy of these predictions based on the residual mean squared error ("RMSE").

Once the data was loaded into RStudio, it was cleaned to facilitate data exploration. This produced meaningful insights which then informed a framework for an initial model. Potential improvements to this model were explored and tested iteratively until a final model with a satisfactory RMSE was reached.

METHODS

Several important adjustments were made to the dataframes before conducting any data exploration. First, because the title years were embedded in the title column entries, this information needed to be extracted and stored in its own column. Similarly, a new boolean column was created for each genre stored in the genres column.

```
library(lubridate)

validation <- validation %>% mutate(
  rating_year = year(as_datetime(timestamp)),
  title_year = str_extract(title, "\\(\\d+\\)$"),
  title_year = as.numeric(str_remove_all(title_year, "\\(|\\)")),
  comedy = ifelse(str_detect(genres, "Comedy"), 1, 0),
  action = ifelse(str_detect(genres, "Action"), 1, 0),
  children = ifelse(str_detect(genres, "Children"), 1, 0),
  adventure = ifelse(str_detect(genres, "Adventure"), 1, 0),
  animation = ifelse(str_detect(genres, "Animation"), 1, 0),
  drama = ifelse(str_detect(genres, "Drama"), 1, 0),
  crime = ifelse(str_detect(genres, "Crime"), 1, 0),
  scifi = ifelse(str_detect(genres, "Sci-Fi"), 1, 0),
  horror = ifelse(str_detect(genres, "Horror"), 1, 0),
  thriller = ifelse(str_detect(genres, "Thriller"), 1, 0),
  mystery = ifelse(str_detect(genres, "Mystery"), 1, 0),
  romance = ifelse(str_detect(genres, "Romance"), 1, 0),
  fantasy = ifelse(str_detect(genres, "Fantasy"), 1, 0),
  musical = ifelse(str_detect(genres, "Musical"), 1, 0),
  war = ifelse(str_detect(genres, "War"), 1, 0),) %>%
  select(-genres, -timestamp)

sample_n(validation, 1)
```

```
##    userId movieId rating                 title rating_year title_year comedy
## 1:  50798    1208      3 Apocalypse Now (1979)        2002       1979      0
##    action children adventure animation drama crime scifi horror thriller
## 1:      1        0         0         0     1     0     0      0        0
##    mystery romance fantasy musical war
## 1:       0       0       0       0   1
```

The same adjustments were made to the edx set, which, unlike the validation set, was then further partitioned into edx_train and edx_test sets.

To determine a baseline RMSE for my predictions, I first created a naive prediction model which simply guessed the average movie rating from the edx_train set.

```
mu <- mean(edx_train$rating)
RMSE(edx_test$rating, rep(mu, nrow(edx_test)))
```

```
## [1] 1.060122
```

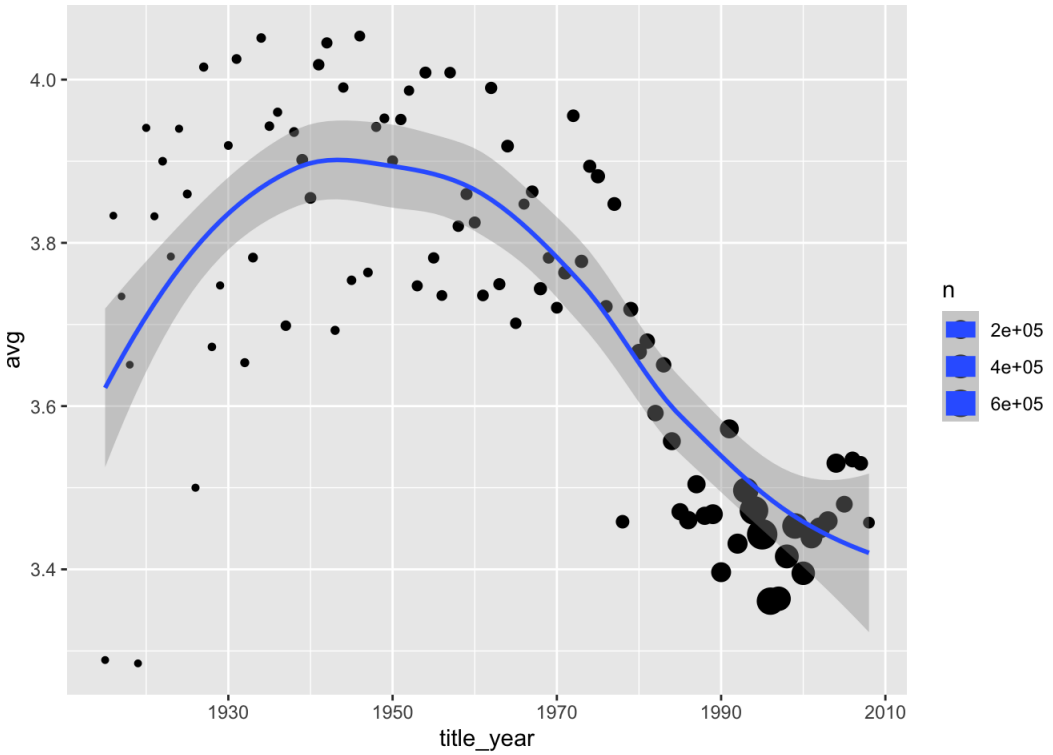
To develop a more sophisticated model, I wanted to determine whether there were any clear biases among the different genres. For simplicity, I ignored the possible effect of genre combinations (e.g. romantic comedy or sci-fi mystery) and instead focused on genres in isolation, which indicated dramas had the most significant effect on the conditional mean rating.

```
edx %>% group_by(drama) %>% summarize(avg = mean(rating))
```

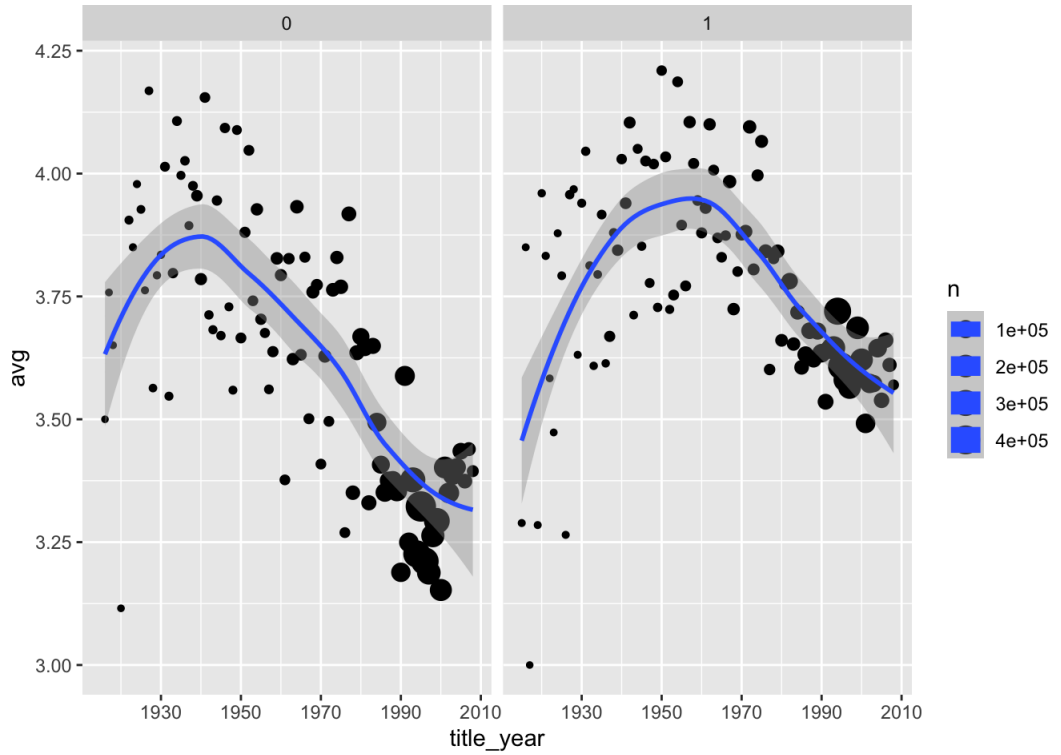
```
## # A tibble: 2 x 2
##   drama   avg
##   <dbl> <dbl>
## 1     0  3.39
## 2     1  3.67
```

In fact, dramas were rated on average nearly 0.3 stars higher than those without a drama genre designation. Already, the naive model could be improved upon by predicting these averages depending on whether a given movie was categorized as a drama, rather than a single average across all movies.

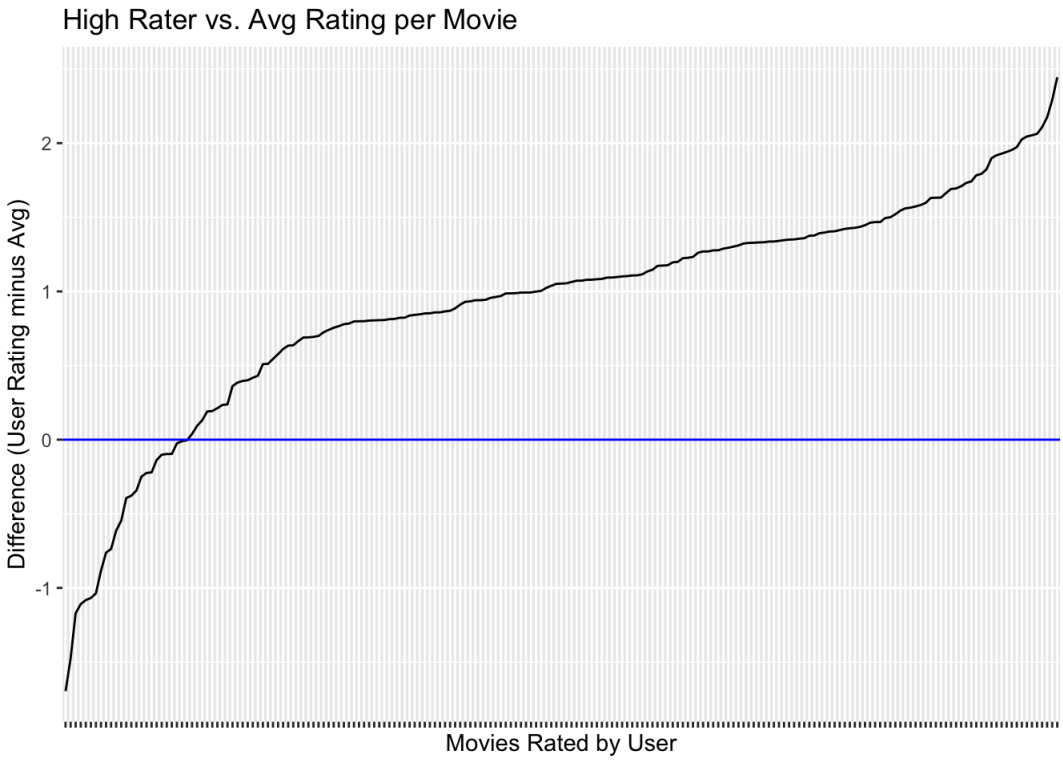
There was also a clear bias associated with a movie's year of release:

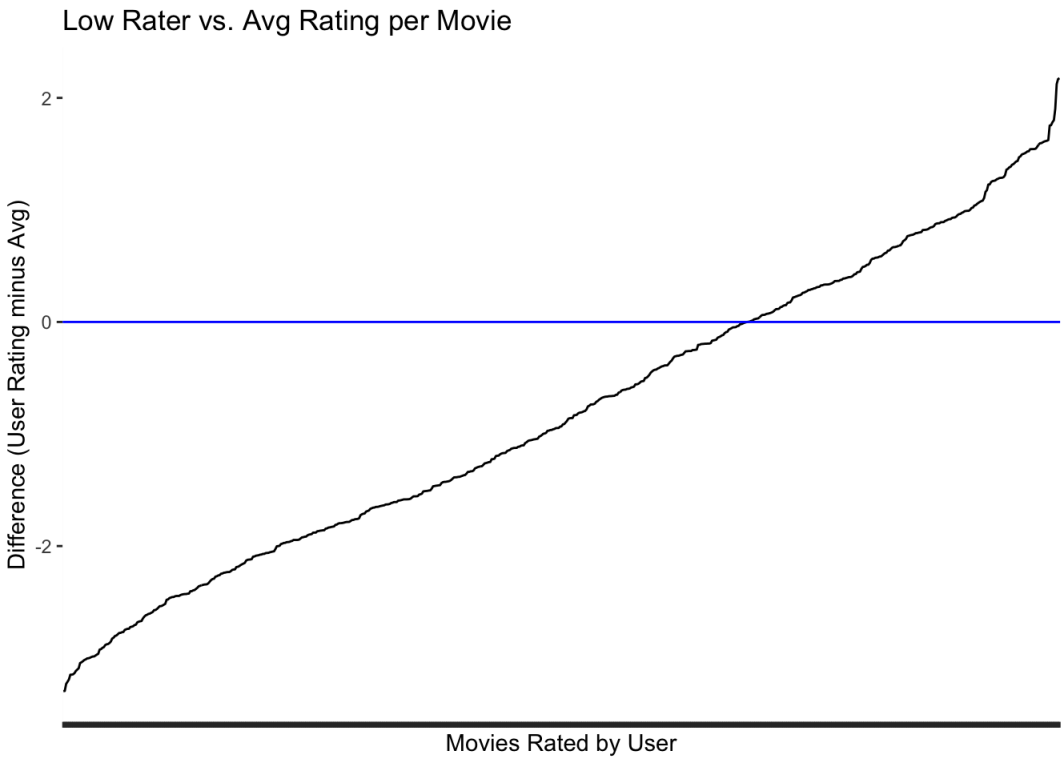


Most likely, this is due to a selection bias such that individuals only watch older movies that are known to be higher quality like The Wizard of Oz or Gone with the Wind. Whatever the reason may be, this tendency of older movies being rated higher than newer movies needed to be accounted for. Although the loess regression appears to model the effect decently, training on the full dataset will take prohibitively long. Instead, we can approximate the year effect directly. Notice that in the plots below, when faceting by the drama genre, the year effect is approximately the same shape. The drama curve on the right is essentially a vertical transformation equal to the difference in conditional means, 0.3. Also notice the difference in the size of the points, which denotes the number of ratings for each year. Regularization will therefore be used to weight years appropriately.



We also observe a clear bias associated with different users, meaning some rate movies higher on average and others rate movies lower on average. Below are two plots illustrating this. Each shows a user with more than 100 different movie ratings, one randomly sampled from the top quartile by average rating and the other from the bottom quartile. For each movie these individuals have rated, a difference of their rating versus the movie's average rating among all users is plotted. If these users were unbiased, we would expect approximately half of the differences to be above/below zero.





However, the differences are not centered at zero; a significant majority of the first user's ratings exceed the average rating for each movie he or she has watched, while the second user has the opposite bias. The bias for any given user is therefore defined as the average of all of these differences. To avoid over-fitting a bias estimate for users with relatively few observations (i.e. shrinking their bias estimate toward zero), we will also use regularization.

The same intuition applies to bias associated with individual movies. For a widely acclaimed movie such as The Godfather, this effect means that a user with higher ratings on average would rate the movie especially highly, while a "grumpy" user's bias may be completely counteracted. Again, we will use regularization to apply the appropriate weighting to movies with relatively few ratings.

RESULTS

In summary, in making predictions on an unknown set of observations, the model will first look to whether a given movie is categorized as a drama to determine a starting point for the prediction. Next, it will make bias estimates for title year, userId, and movieId based on the residuals observed sequentially after factoring in all the prior biases. For example, the first estimate among these three will take into account the fact that we have already adjusted the predictions based on genre. Since title year, userId, and movieId bias estimates will all be regularized, we will test out multiple different values of lambda and choose the one that minimizes the RMSE on the test set. Once a lambda is determined, we will repeat the process to train on the entire edx set and ultimately make our final predictions on the validation set.

The final model produces the following RMSE:

```
RMSE(validation$rating, predictions)
```

```
## [1] 0.8647818
```

One of the major challenges was determining the most effective way to incorporate a genre effect. Training a linear model inclusive of all 15 genres was not time-feasible. For the same reason, it was also not feasible to incorporate the effects of certain combinations of genres, which I suspect could have offered some predictive value. For example, thrillers and mysteries individually had similar ratings to the overall average, but in combination the genre effect is significant.

```
edx %>% group_by(thriller, mystery) %>% summarize(avg = mean(rating))
```

```
## # A tibble: 4 x 3
## # Groups:   thriller [2]
##   thriller mystery   avg
##      <dbl>   <dbl> <dbl>
## 1        0       0  3.51
## 2        0       1  3.59
## 3        1       0  3.47
## 4        1       1  3.71
```

Working off the assumption that choosing one genre was most efficient, I determined drama as the best candidate given its prevalence - approximately 43% of the training set - and the magnitude of the difference in conditional means. Indeed, no other genre was able to improve upon the naive prediction model in terms of RMSE as much as drama.

The other aspect of the model that required some trial-and-error was the sequencing of bias estimates. In the end, the order that minimized the RMSE was title year, movieId, and userId.

CONCLUSION

The objective of the project was to minimize the RMSE of the final predictions. Before this could even be attempted, several preliminary steps were required. First, data needed to be cleaned and reformatted in such a way that allowed for data exploration. The insights gained from this exercise would provide a framework for building a model. An important aspect to the modeling approach itself was beginning as simple as possible and gradually increasing complexity as accuracy improved. Testing potential changes often was also key.

The final model achieves a reasonably small RMSE without the use of advanced machine learning algorithms and only four variables, bias estimates for the drama genre, users, movies, and title years. The strength of simplicity may also be the model's greatest shortcoming - the majority of information provided in the dataset does not ultimately factor into predictions. Future work would explore the data further and find ways of taking into account other variables.