MovieLens Project

1. Introduction

This report is an analysis of the *MovieLens* data set which contains 10 million records of the reviews of more than 10,000 movie titles by approximately 70,000 users. The analysis attempts to predict the rating that a user would give for a movie given a movie title's average rating and the average rating of all movies submitted by a given user.

Three different **linear regression** models are used to train and predict movie ratings. The first model compares actual ratings to the overall average movie rating (*mu_hat*) of approximately 3.5. The second model, in addition to the overall average, incorporates the average rating of each movie title and compares the predicted (fitted) results to the actual results. The third model augments the second model by adding the average movie ratings by user.

The models are tested using validation (training) sets containing 10% and 25% of the records of the original data set. The third model incorporating average ratings by users, included only those users submitting 100 ratings or more. The RMSE, Root Mean Squared Error, yielded smaller values with each successive model, suggesting greater model accuracy as a result.

The third model, based on movieID and userID and using a validation set of 10% of the original data, yields an RMSE of **0.86440**.

The source of the MovieLens data set may be extracted from the following links:

https://grouplens.org/datasets/movielens/10m/
http://files.grouplens.org/datasets/movielens/ml-10m.zip

The Github link for the PDF, R and RMD files can be found here:

https://github.com/yu138538/ML-Project

2. Analysis

2.1 The *MovieLens* Data set

The data set used for the analysis consists of 10 million rows, one row per movie review, and six variables. The analysis attempts to predict **rating** of each movie review.

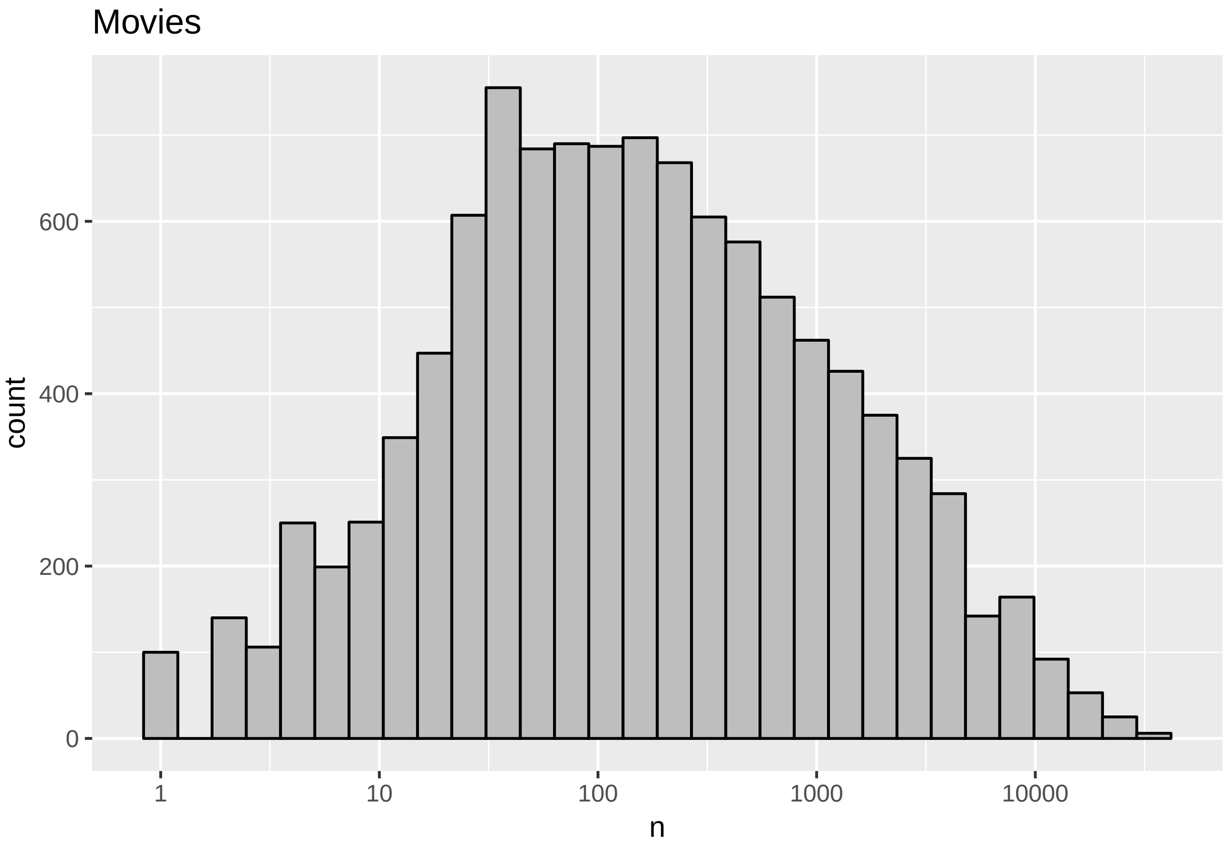
The structure of the *MovieLens* data

```
## Observations: 10,000,054
## Variables: 6
## $ userId    <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ movieId   <dbl> 122, 185, 231, 292, 316, 329, 355, 356, 362, 364, 37...
## $ rating    <dbl> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5...
## $ timestamp <int> 838985046, 838983525, 838983392, 838983421, 83898339...
## $ title     <chr> "Boomerang (1992)", "Net, The (1995)", "Dumb & Dumbe...
## $ genres    <chr> "Comedy|Romance", "Action|Crime|Thriller", "Comedy",...
```

```
##   n_users n_movies
## 1   69878    10677
```

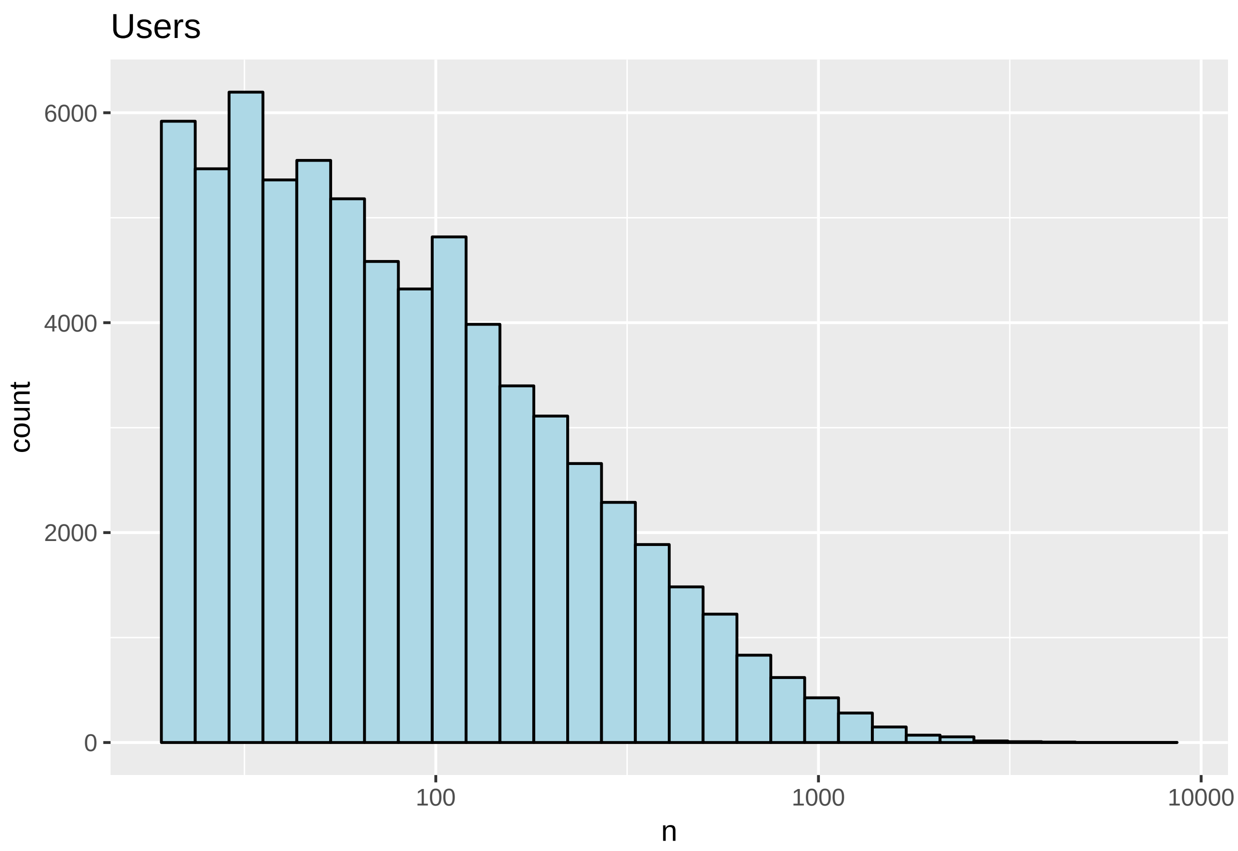
Histogram of the counts of movies rated

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Histogram of the echo of user ratings

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The Validation set used for testing is 10% of the original data or approximately 2.5 million records.

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

A preview of the Training Data Set

```
## Observations: 9,000,064
## Variables: 6
## $ userId    <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ movieId   <dbl> 122, 185, 231, 292, 316, 329, 355, 356, 362, 364, 37...
## $ rating    <dbl> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5...
## $ timestamp <int> 838985046, 838983525, 838983392, 838983421, 83898339...
## $ title     <chr> "Boomerang (1992)", "Net, The (1995)", "Dumb & Dumbe...
## $ genres    <chr> "Comedy|Romance", "Action|Crime|Thriller", "Comedy",...
```

The training set used to build the regression model, contains the remaining 75% of the data, or 7.5 million records.

A preview of the Testing Data Set

```
## Observations: 999,990
## Variables: 6
## $ userId    <int> 2, 2, 3, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5...
## $ movieId   <dbl> 648, 780, 110, 1252, 1408, 4995, 6287, 39, 440, 589,...
## $ rating    <dbl> 2.0, 3.0, 4.5, 4.0, 3.5, 4.5, 3.0, 3.0, 3.0, 5.0, 5....
## $ timestamp <int> 868244699, 868244698, 1136075500, 1133571071, 113357...
## $ title     <chr> "Mission: Impossible (1996)", "Independence Day (a.k...
## $ genres    <chr> "Action|Adventure|Mystery|Thriller", "Action|Adventu...
```

Both the validation and training data sets are filtered to remove blank movie and user ID numbers as demonstrated by the semi-joins in the original R code.

The accuracy of models used in the analysis are measured using the *RMSE*, *Root Mean Square Error* defined as

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

where

$$\hat{y}_{u,i}$$

is the predicted rating as fitted by the regression model.

And

$$y_{u,i}$$

is the actual value. The objective of the analysis is to minimise the value of *RMSE*.

The analysis of the *MovieLens* data is done using three different Linear Regression models.

2.2 The Overall Average Rating for all Movie Titles

The average rating for all movies submitted by all users is simply the arithmetic mean of all ratings.

$$Y_{u,i} = \mu + \varepsilon_{u,i}$$

where

$$\varepsilon_{u,i}$$

is the randomly distributed error.

The average rating across all movies in the data set is

```
## [1] 3.512384
```

The RMSE using only the overall average rating yields

```
## [1] 1.059465
```

2.3 The *Movie* Effect Model

In addition to the overall average, this model incorporates the average rating of each movie title.

$$Y_{u,i} = \mu + b_i + \varepsilon_{u,i}$$

where

$$b_i$$

is the factor of each movie *i*.

Typically, a model of this type would be built using a linear regression model such as

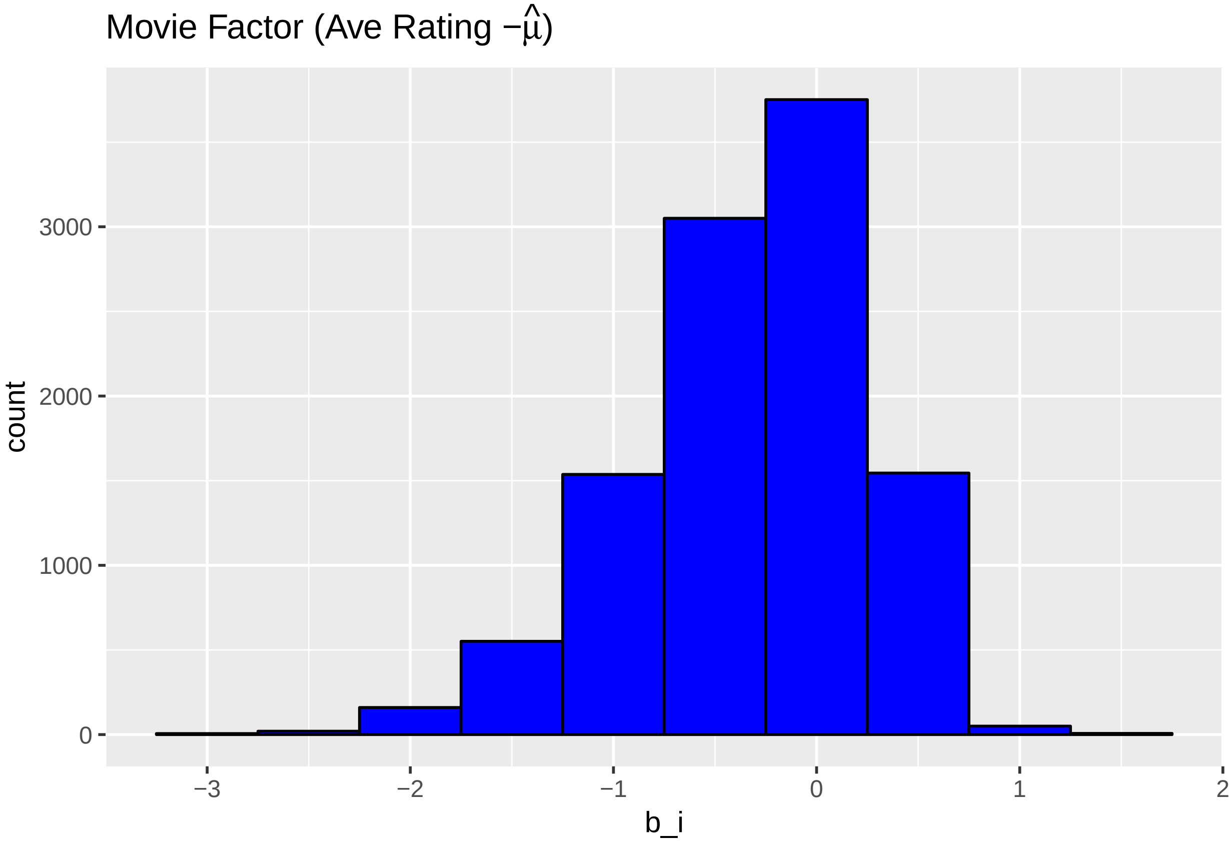
```
lm(rating ~ as.factor(movieID), data = train_set)
```

However given that **movieID** yields more than 10,000 factors, such a model would be computationally intensive and not be feasible. Instead, each movie factor is approximated as the average of

$$b_i = Y_{u,i} - \mu$$

This approach leads to much faster computation.

A histogram of the distribution of Movie Factor



The RMSE incorporating the movie factor yields

```
## [1] 0.9428505
```

The *Overall Average* RMSE compared to the *Movie Effect* RMSE

```
## # A tibble: 2 x 2
##   method                RMSE
##   <chr>                <dbl>
## 1 Overall Average    1.0595
## 2 Movie effect model 0.94285
```

The results suggest increased accuracy with the inclusion the movie factor

2.4 The User and Movie Model

The model is further refined to include the average rating by user.

$$Y_{u,i} = \mu + b_i + b_u + \varepsilon_{u,i}$$

where

$$b_u$$

is the effect of user u.

Once again a standard linear regression model could be generated using

```
lm(rating ~ as.factor(movieID) + as.factor(userID), data = train_set)
```

In addition to 10,000 movieID factors, the userID factor adds an additional 70,000 factors, which once again would not generate a feasible model.

In this case the user factor can be approximated as the average of

$$b_i = \ Y_{u,i} - \mu - b_i$$

where

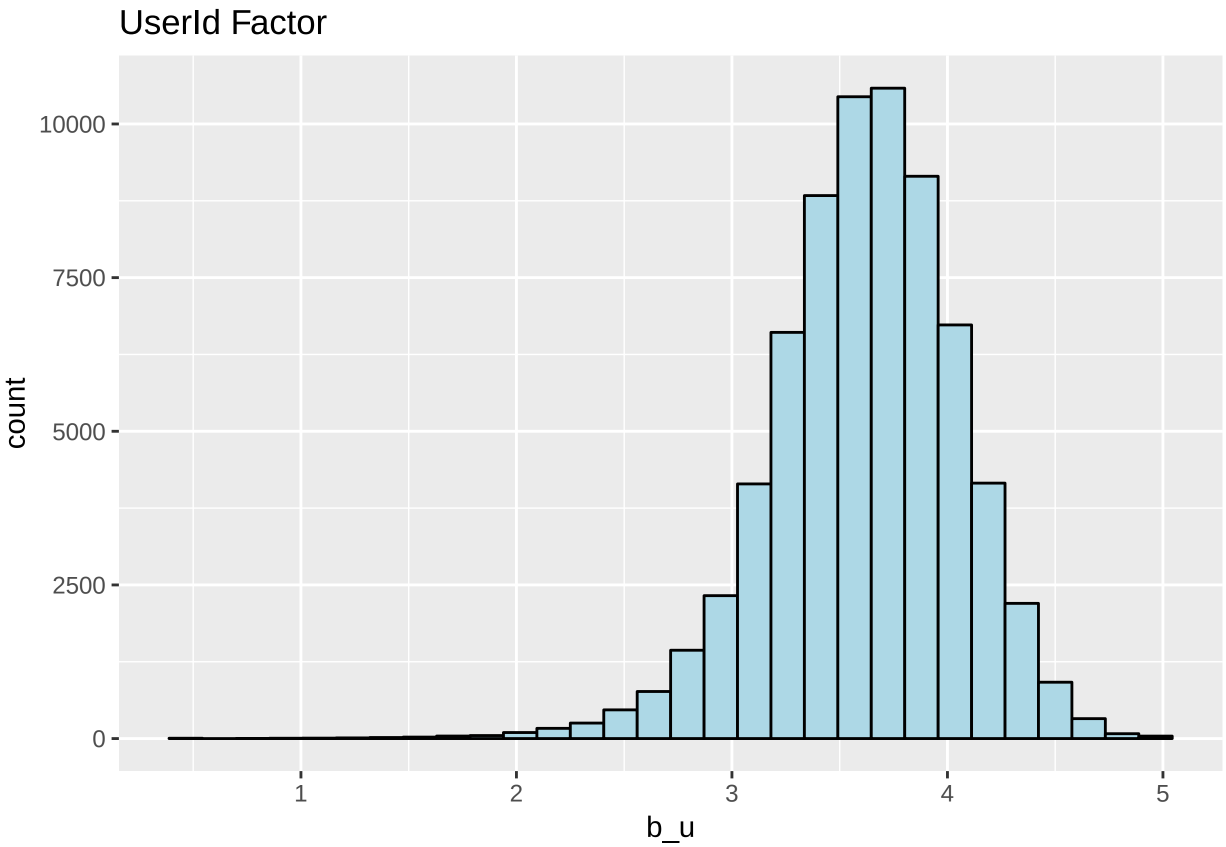
$$b_u$$

is the factor of user u.

The data used to approximate the user factor includes only users who submitted 100 or more ratings.

A histogram of the distribution of user ratings

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The RMSE including the user effect yields

```
## [1] 0.8643992
```

which suggests even greater accuracy than the movie effect model.

3. Results

The results of the regression models yield the following results.

A comparison of the RMSE of all three models

```
## # A tibble: 3 x 2
##   method                         RMSE
##   <chr>                         <dbl>
## 1 Overall Average             1.0595
## 2 Movie effect model          0.94285
## 3 User and Movie effect model 0.86440
```

The RMSE values suggest that the accuracy of the models increase with the addition of the movieID and userID factors.

An additional test using 25% of the original data set for validation yields

```
## # A tibble: 3 x 2
##   method                       RMSE
##   <chr>                       <dbl>
## 1 Overall Average             1.0595
## 2 Movie effect model          0.94321
## 3 User and Movie effect model 0.86532
```

Which is consistent with the results using 25% of the data for validation, but is slightly less accurate.

4. Conclusion

As demonstrated, the RMSE totals suggest that the accuracy of the modified linear regression model increases with the addition of movie and user rating factors. The analysis could have been further expanded to see the effect of adding a factor for **genre** , **timestamp** or **release date**, which was not included in the original data set.