MovieLens Project- Harvard

Mary Lampmann

10/06/2020

Introduction

MovieLens is a web-based movie recommender system and virtual community that suggests movies for member users to watch based on their film preferences.
This report documents my creation of a Movie Recommendation System specific to the 10M version of the MovieLens dataset.

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

The goal of this project is the training of a machine learning algorithm which will use inputs from a subset of the data to predict ratings in a separate subset of that same data, and do so with the lowest residual mean squared error(RMSE) possible. For the grading on this specific project, the target is an RMSE below .8649.

Project Key steps

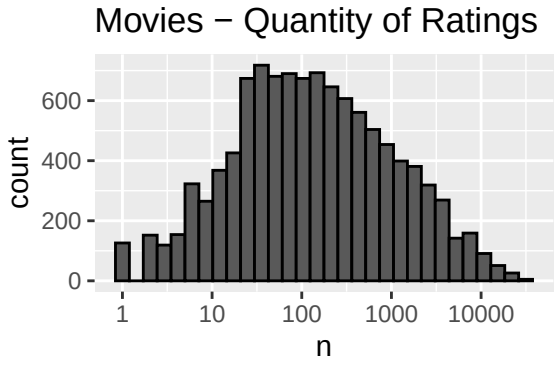
1. Partition the MovieLens 10M dataset to create a training subset(**edx**) and a final hold-out test set(**validation**) for use in assessing the residual mean squared error(RMSE) on the recommendation system proposed
2. Conduct exploratory data analysis (EDA) on the training set **edx** for use in machine learning(ML) model development
3. After EDA, partition the **edx** training dataset to allow separate training (*train_edx*) and test sets(*test_edx*) for use in evaluation of ML models , thus wholly preserving the hold-out **validation** data set for a final evaluation on the proposed Movie Recommendation ML algorithm
4. Iteratively train different machine learning algorithms on the dataset to elicit the lowest RMSE results, and generate a final Movie Recommendation Model
5. Test the final Movie Recommendation Model on the **validation** hold out test dataset

Methods/Analysis

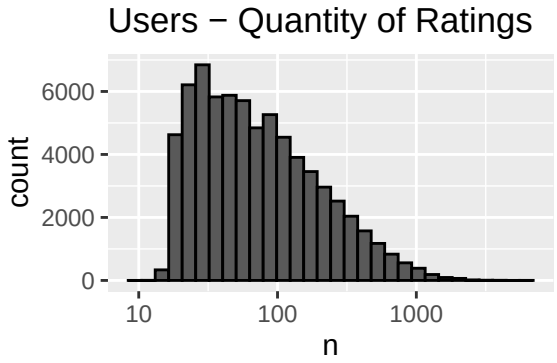
Exploratory Data Analysis The initial training dataset, **edx**, is comprised of 9000055 rows and 6 columns, and includes ratings of 10677 distinct movies, 69878 distinct users, and 797 distinct genres, with those distinct genres being a compilation of genres present in a specific movie.

A movie in this dataset, **edx**, has, on average, 843 reviews. A user has rated, on average, 129 movies. The genres have been rated, on average, 11292 times.

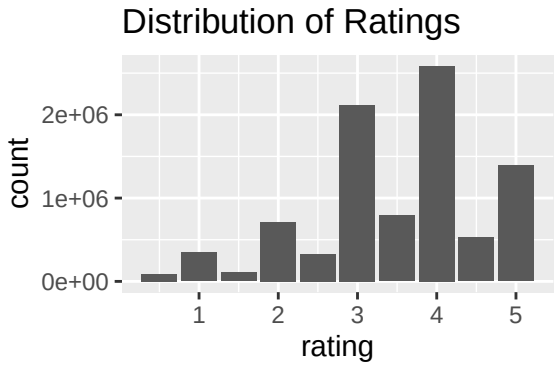
Plots of Movie, User and Rating Counts There were some movies that were rated only once. Movies with few ratings (<=10) are excluded in some of the following tables for purposes of illustration.



In general, users rated a series of movies, with the majority rating less than 100 movies.



The ratings ranged from 1/2 of a star up to 5 stars. In general, users were more likely to award whole number ratings than 1/2, with 4 stars being the most common rating.



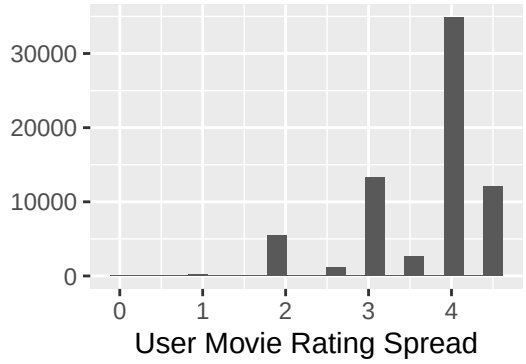
The mean value of the star ratings of the movies is 3.51, and standard deviation is 1.06. 95.21% of the ratings fall within 2 standard deviations of the mean.

Movie User and Genre Rating Examples The best rated movies (10+ ratings), based on rating value(stars), are consistent with popular and critically acclaimed movies, but the ratings count vary widely within the movies with the highest average star ratings. The Shawshank Redemption, at the #1 position with an average rating of 4.46 based on 28,015 ratings, had a volume of rating almost 18x that of #10 position Paths of Glory (1,571 ratings).

```
## # A tibble: 10 x 4
##    movieId title                                          count avg_rating
##      <dbl> <chr>                                          <int>      <dbl>
##  1     318 Shawshank Redemption, The (1994)              28015       4.46
##  2     858 Godfather, The (1972)                         17747       4.42
##  3      50 Usual Suspects, The (1995)                    21648       4.37
##  4     527 Schindler's List (1993)                       23193       4.36
##  5     912 Casablanca (1942)                             11232       4.32
##  6     904 Rear Window (1954)                             7935       4.32
##  7     922 Sunset Blvd. (a.k.a. Sunset Boulevard) (1950)  2922       4.32
##  8    1212 Third Man, The (1949)                         2967       4.31
##  9    3435 Double Indemnity (1944)                       2154       4.31
## 10    1178 Paths of Glory (1957)                         1571       4.31
```

There is low (22.2)% correlation between the number of ratings given and the average star rating in this subset(10+ ratings per movie) of the data.

Another important observation on the **edx** dataset: it is most common to see a significant spread between *the lowest rating* awarded by a specific user to a movie and *the highest*. On a star rating scale that starts at 0.5 and ends at 5.0 (an absolute spread of 4.5), the spread of ratings for a majority of users reside in the 4 and 4.5 values. This suggests that users are rating both movies that they liked and movies that they absolutely did not. The wide range of these ratings suggest value in incorporating user bias into the ML model.



A review of the genres (1000+ user ratings) shows significant variance between user rating counts and average ratings : genres like Drama | Film-Noir| Romance earned average ratings of 4.30 with almost 3000 user submissions, but the Drama and Comedy genres, with lower average star ratings, were each rated by over 700,000 times.

```
## # A tibble: 6 x 3
##   genres                          count avg_rating
##   <chr>                           <int>      <dbl>
## 1 Drama|Film-Noir|Romance          2989       4.30
## 2 Action|Crime|Drama|IMAX          2353       4.30
## 3 Animation|Children|Comedy|Crime  7167       4.28
## 4 Film-Noir|Mystery                5988       4.24
## 5 Crime|Film-Noir|Mystery          4029       4.22
## 6 Film-Noir|Romance|Thriller       2453       4.22
```

```
## # A tibble: 6 x 3
##   genres                count avg_rating
##   <chr>                 <int>      <dbl>
## 1 Drama                733296       3.71
## 2 Comedy               700889       3.24
## 3 Comedy|Romance       365468       3.41
## 4 Comedy|Drama         323637       3.60
## 5 Comedy|Drama|Romance 261425       3.65
## 6 Drama|Romance        259355       3.61
```

Core Genre Types - Ratings The 797 distinct genres in the dataset are composed of 19 base genre types, reaggregated in the following tables based on ratings of each movie that contained that genre (i.e. Drama). Users provided 700,000 + ratings of movies with Drama **as the only genre**(preceding table) , as compared to 3,900,000 + ratings on movies that contained Drama *as one genre of one or more genres*(following table).

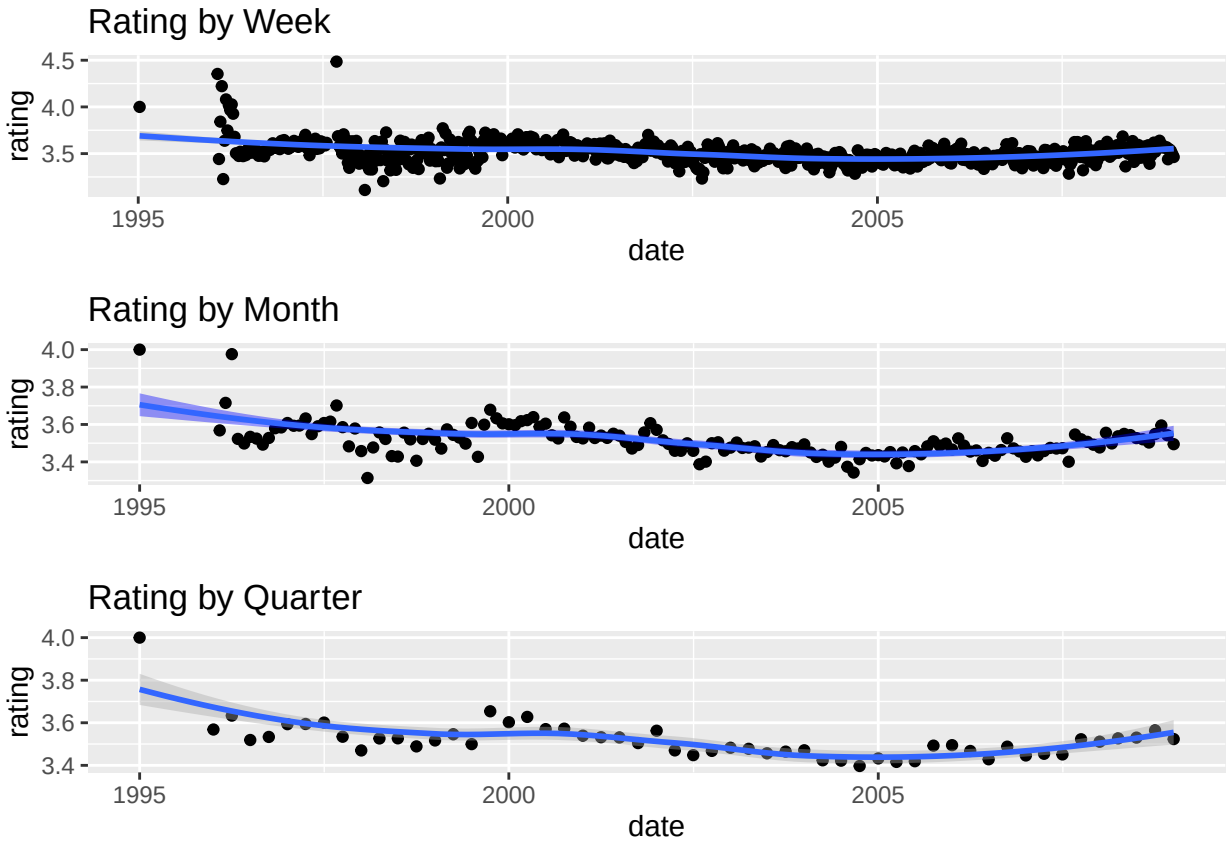
```
## # A tibble: 20 x 3
##    genres                count avg_rating
##    <chr>                 <int>      <dbl>
##  1 Drama                3910127       3.67
##  2 Comedy               3540930       3.44
##  3 Action               2560545       3.42
##  4 Thriller             2325899       3.51
##  5 Adventure            1908892       3.49
##  6 Romance              1712100       3.55
##  7 Sci-Fi               1341183       3.40
##  8 Crime                1327715       3.67
##  9 Fantasy               925637       3.50
## 10 Children              737994       3.42
## 11 Horror                691485       3.27
## 12 Mystery               568332       3.68
## 13 War                   511147       3.78
## 14 Animation             467168       3.60
## 15 Musical               433080       3.56
## 16 Western               189394       3.56
## 17 Film-Noir             118541       4.01
## 18 Documentary            93066       3.78
## 19 IMAX                    8181       3.77
## 20 (no genres listed)         7       3.64
```

The variance between average ratings - - Film-Noir and Horror representing opposite ends of the average star ratings users awarded - - suggest value in building genre bias into the movie recommendation model.

```
## # A tibble: 20 x 3
##    genres                count avg_rating
##    <chr>                 <int>      <dbl>
##  1 Film-Noir             118541       4.01
##  2 Documentary            93066       3.78
##  3 War                   511147       3.78
##  4 IMAX                    8181       3.77
##  5 Mystery               568332       3.68
##  6 Drama                3910127       3.67
##  7 Crime                1327715       3.67
##  8 (no genres listed)         7       3.64
```

```
##  9 Animation           467168       3.60
## 10 Musical             433080       3.56
## 11 Western             189394       3.56
## 12 Romance            1712100       3.55
## 13 Thriller           2325899       3.51
## 14 Fantasy             925637       3.50
## 15 Adventure          1908892       3.49
## 16 Comedy             3540930       3.44
## 17 Action             2560545       3.42
## 18 Children            737994       3.42
## 19 Sci-Fi             1341183       3.40
## 20 Horror              691485       3.27
```

Time Effect on Ratings Different time periods (week, month, quarter) were evaluated to analyze the value of including time of a user rating as a component of the machine learning algorithm model. Although there was some time effect in play, the effect appeared to be minimal, and will not be incorporated into the ML model.



Model Construction Methods

Split edx into train and test sets The next step in the machine learning algorithm creation was the split of the **edx** data set into training and testing subsets, *train_edx* and *test_edx* respectively.

A residual mean squared error function(RMSE) was created, as was a data frame that would detail the RMSE values associated with different machine learning algorithms.

```
RMSE <- function(true_ratings, predicted_ratings){
  sqrt(mean((true_ratings - predicted_ratings)^2))
}
```

The iterative movie recommendation algorithm process: Next steps were the creation and training of different recommendation models, with the rationale for the model and RMSE results detailed here.

* Average As Predictor Model - compare the mean value of the *train_edx* rating to the *test_edx* rating values

```
## # A tibble: 1 x 2
##   Method                RMSE
##   <chr>                <dbl>
## 1 Average As Predictor  1.06
```

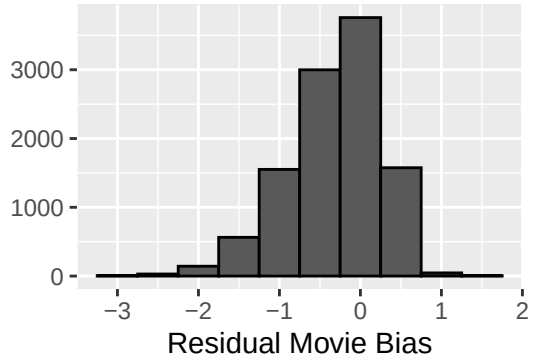
** Movie and User Model - group by distinct movie identification number and distinct user identification number to identify unique rating "bias" for that specific movie and user. Adjust each individual user/movie rating by these biases, generate predicted ratings on the *test_edx* dataset using the residual values.

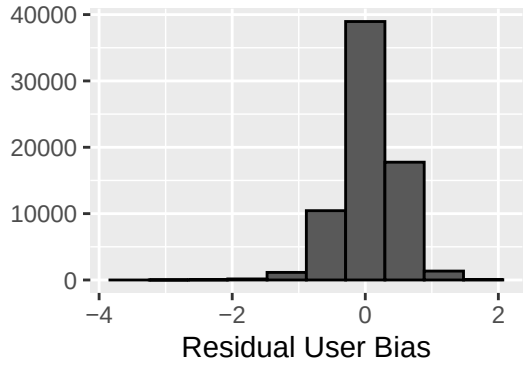
Method	RMSE
Average As Predictor	1.0601
Movie and User Model	0.8647

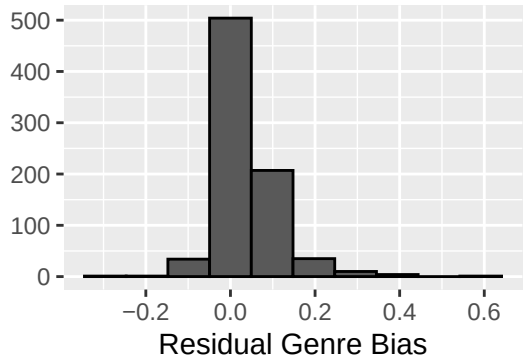
*** Movie User and Genre Model - same as Movie and User model above, adds group by distinct genre identifier, identification of genre biases, adjustments to reflect residual biases, generation of predicted ratings on the *test_edx* dataset.

Method	RMSE
Average As Predictor	1.0601
Movie and User Model	0.8647
Movie and User and Genre Model	0.8643

Residual Biases in Model Incorporating movie, user, and genre biases into the model reduced the RMSE to 0.8643, not a material change from the model considering movie and user biases. A review of the residual biases for each prior to a regularization process shows the residual biases to be applied to the model as both *a reduction to the average(movie rating bias)* and an *addition to the average (user rating bias, and to a smaller degree, genre rating bias)*,see below:



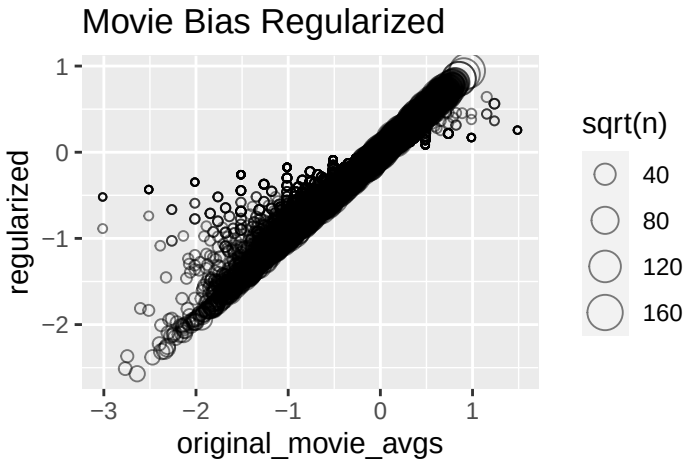


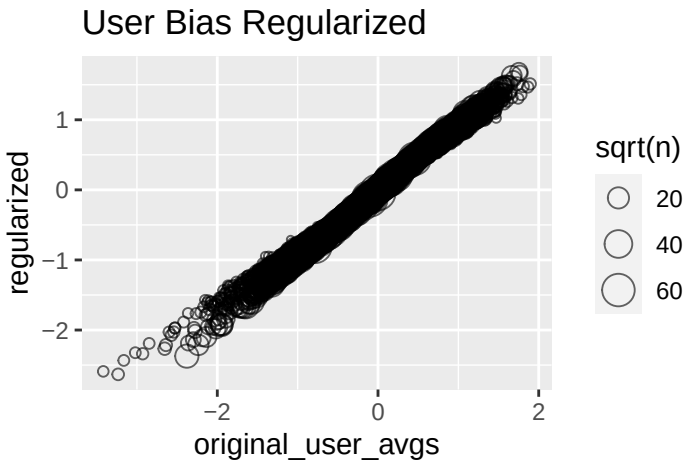


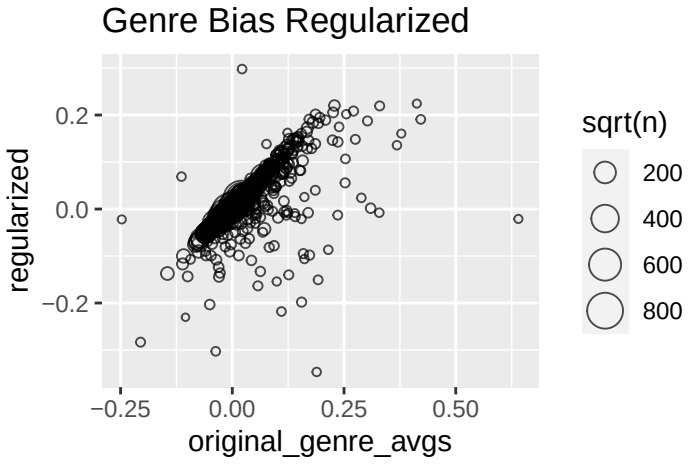
Regularization and Cross Validation The next step in the model building process was to control for the effect of outliers on our model prediction by adding an error term to our model. Cross validation was used to determine the error model value (lambda) that minimizes the RMSE for this dataset.

Method	RMSE
Average As Predictor	1.0601
Movie and User Model	0.8647
Movie and User and Genre Model	0.8643
Regularized Movie User Genre Bias Model	0.8638

Regularization Impact vs. Original Data The impact of the addition of the error term of 4.8 can be seen in the following charts, which demonstrate how the regularization process shrinks the values of the outliers in the data. In the chart, the size of the circle reflects the square root of the size of the original signal.







Final Model and Results

Final Model - Regularized Movie & User & Genre Bias Model - Full Edx dataset and Validation holdout The final Movie Recommendation System Model incorporates evaluation of residual bias in movie, user, and genres, and the use of a cross validated optimal lambda in a regularization process. The R code and corresponding RMSE results are as follows:

```
mu <- mean(edx$rating)

movie_reg_avgs <- edx %>%
  group_by(movieId) %>%
  summarize(b_i = sum(rating - mu)/(n()+tuned_lambda))
user_reg_avgs <- edx %>%
  left_join(movie_reg_avgs, by="movieId") %>%
  group_by(userId) %>%
  summarize(b_u = sum(rating - b_i - mu)/(n()+tuned_lambda))
genre_reg_avgs <- edx %>%
  left_join(movie_reg_avgs, by="movieId") %>%
  left_join(user_reg_avgs, by="userId") %>%
  group_by(genres) %>%
  summarize(b_g = sum(rating - b_i- b_u - mu)/(n()+tuned_lambda))

predicted_ratings <-
  validation %>%
  left_join(movie_reg_avgs, by = "movieId") %>%
  left_join(user_reg_avgs, by = "userId") %>%
  left_join(genre_reg_avgs, by = "genres") %>%
  mutate(pred = mu + b_i + b_u + b_g) %>%
  .$pred

model_final <- RMSE(validation$rating,predicted_ratings)
rmse_final_validation <- data_frame(Method="Regularized Movie User Genre Model(Full Edx & Validation)",

rmse_final_validation
```

```
## # A tibble: 1 x 2
##   Method                                                    RMSE
##   <chr>                                                    <dbl>
## 1 Regularized Movie User Genre Model(Full Edx & Validation) 0.864
```

The final Movie Recommendation Model above generated an RMSE on the Validation hold out dataset of .864, below the goal RMSE of .8649.

Conclusion

The final Movie Recommendation Model above, incorporating regularization (error term lambda = 4.8) on a dataset of residual biases in movie, user, and genre, does achieve our goal of predicting how many stars a user will give a movie, with RMSE results here below the target as well as very similar to levels that earned the Netflix Prize in 2007 and 2008, albeit on a much smaller database than that of the prize competition.

Limitations of the model The RMSE results here are tied to this model's fit to the specific subset of data that was partitioned in this process into the **edx** set(training set) and **validation** set (test set/hold out), and might not be replicated as favorably should the same model be used on future data partitions of the same MovieLens 10M dataset. Repeat of this model evaluation process will require iteratively assessing this same model on different seeds for the data partition, both at the point of partition of the initial MovieLens 10M set, and at the point of partition of the edx set into *train_edx* and *test_edx* subsets.