MovieLens rating prediction model - HarvardX Data science Capstone assignment #1

edX learner DBel_17

March 13th, 2023

INTRODUCTION

MovieLens is an extensive data set containing movie ratings submitted by users of the MovieLens website and curated by the GroupLens research lab, available through http://grouplens.org. MovieLens data is a popular resource for training machine learning algorithms to devise recommendation systems. In fact, participants of the MovieLens platform can receive movie recommendations based on the ratings they have provided. Recommendation systems are important tools for streaming and video rental services to keep their users engaged and committed to continued subscription. Thus, company profits depend on robustly performing algorithms which provide users with movie suggestions according to their preferences.

In the present project, we aimed to develop a model that can predict ratings for any given movie by any individual user. MovieLens contains several attributes that can help us predict ratings. In addition to the individual movie and user Ids, we have the movie title, movie release year, movie genre and the date of rating at our disposition:

Table 1: *Example of entries in the MovieLens database*

userId	movieId	rating	timestamp	title	genres
55604	367	3	848214234	Mask, The (1994)	Action|Comedy|Crime|Fantasy
67803	1219	5	958759032	Psycho (1960)	Horror|Thriller
50532	1197	4	993008971	Princess Bride, The (1987)	Action|Adventure|Comedy|Fantasy|Romance
23493	1584	2	1115405813	Contact (1997)	Drama|Sci-Fi
66183	3578	4	982739444	Gladiator (2000)	Action|Adventure|Drama

The outcomes, *i.e.* ratings, take discrete values and are characterized by:

Range of the rating scores: from 0.5 to 5

Rating score options: 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5 .

In order to build a linear prediction model, we implemented a strategy which begins by assessing the data distribution of individual features in relation to the ratings and then estimates individual effects incrementally. We started by accounting for movie- and user-specific effects and extended this basic model by reducing residual errors through the consideration of further predictors, such as genre, year of release or the rating frequency for each movie. Finally, the elaborated model was tested using the provided validation data set.

METHODS

Data partitioning

To independently validate our model, we partitioned the initial "edx" training data set into a further training and test set, using a 80/20 partition.

Data cleaning and processing

The *title* column contains both the movie title and the year of release. The year was extracted using the regex pattern "\(\d{4}\)$" in conjunction with parse_number(). This ensures that only four digit numbers in parentheses and at the end of the string were extracted, since by experience, there are movie titles like "1984" or "2001: A Space Odyssey". The title was tidied by using the pattern "\s\(\d{4}\)$" and str_replace().

A list of unique genres was extracted by splitting the *genres* column using the pattern "\|" and identifying all unique terms with unique().

The *timestamp* column was converted into POSIXct format with as_datetime() and further rounded to weeks via round_date(., unit="week") function.

A new *ratings_pa* feature representing the annual rating rate for a given movie was derived by dividing the amount of ratings a movie has received in the train data set by the years of its existence until the year 2009 and further by the number of rows in the train set. To scale this parameter into a reasonable range, a final multiplication with 10^6 was carried out:

$$f_i = \frac{S_i}{(2009 - y_i)} * \frac{10^6}{\sum_{i=1}^{N} S_i},$$

with f_i the annual rating frequency for movie i, S_i the amount of ratings movie i has received in the training data set, y_i the year of release for movie i and N the number of movies in the data set.

Data exploration and visualization

Graphs were generated using the *ggplot2* or *base* R packages. Histograms, box plots, scatter plots and smooth curves were preferred tools to examine the relationship between features and outcomes as well as features and residuals.

Modeling approach

A linear prediction model was built by starting with one predictor, delineating its effect and then taking the remaining residual error ϵ and using the next predictor to explain a share of it. As a figure of reference, first, a global mean estimate $\hat{mu}$ was calculated by averaging all ratings for a given movie and then averaging the average ratings of all movies:

$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^{N} \bar{R}_i$$

with $\bar{R}_i$ being the average rating of a movie i and N the number of movies contained in the training data.

Hence, our first premise for the outcome is

$$R_{i,j} = \mu + \epsilon_{i,j},$$

with $R_{i,j}$ being the true rating and $\epsilon_{i,j}$ the residual error for movie i and user j.

By sequentially computing movie, user, rating rate per anno, genre, year of release, review date and user's genre preference effects, the residuals were minimized.

Movie and user effects were calculated by averaging residual errors per movie and per user, respectively. The movie effect $\hat{m}_i$ for movie i, as the first effect estimate, was thus distilled by following formula:

$$\hat{m}_i = \frac{1}{S_i} \sum_{s=1}^{S_i} (R_i - \hat{\mu}),$$

with S being the number of ratings for movie i in the training data set.

User effect estimates $\hat{u}_j$ were computed analogously, but after subtracting the already calculated $\hat{m}_i$ effect.

Rate per anno, year of release and review data effects were modeled with the help of the loess() (locally estimated scatterplot smoothing) function after optimizing span.

Most movies have several genres attributed to them. We assumed that the effect of each assigned genre contributes equally to the final genre bias effect for each movie. Therefore, genre effects on residual errors were estimated by, first, averaging movie ratings in each available genre category separately, to obtain the effect $\hat{g}_k$ for each genre k. Then, a joint genre effect for each movie with genre assignments $k...K$ was computed by averaging the effects $\hat{g}_k$ to obtain a movie-individual genre effect $\hat{g}_i$.

$$\hat{g}_i = \frac{1}{K_i} \sum_{k=1}^{K_i} \hat{g}_k, \text{ with } K_i \text{ being the number of genres movie } i \text{ is assigned to.}$$

The effect of each user's preferences for genres was estimated by averaging the residuals in the ratings of each individual user for all the available 18 genres (set to 0 if any genre was not rated at all by a particular user), yielding a user/genre interaction effect estimate $\hat{g}_{j,k}$ for user j and genre k. Then the user/genre effect was extrapolated to a particular movie categorized into genres k by averaging all applicable $\hat{g}_{j,k}$ to yield the final user/genre estimate for movie i – $\hat{g}_{j,i}$:

$$\hat{g}_{j,k} = \frac{1}{T_k} \sum_{t_k=1}^{T_k} \epsilon_{j,ik}, \text{ with } T_k \text{ being the amount of ratings user } j \text{ has submitted for movies in genre } k \text{ and } \epsilon_{j,ik} \text{ the residual for user's } j \text{ rating of movie } i \text{ categorized in genre } k$$

and

$$\hat{g}_{j,i} = \frac{1}{K_i} \sum_{k=1}^{K_i} \hat{g}_{j,k}$$

Regularization attempts were applied to the estimations of all effects, by testing a range of tuning parameter values on the test set. Optimal tuning parameters served to derive final effect estimates. For the movie effects, for instance, this was accomplished using the formula:

$$\hat{m}_i = \frac{1}{S_i + \lambda} \sum_{s=1}^{S_i} (R_i - \hat{\mu}),$$

with tuning parameter λ.

The final model is a linear model with several loess-estimated effects:

$$R_{i,j} = \mu + m_i + u_j + a_i + g_i + y_i + d_w + g_{j,i} + \epsilon_{i,j},$$

with u_j the user-specific effect for user j, a_i the loess estimate for the effect of the annual rating rate for movie i, y_i the loess estimate of the release year effect for movie i, d_w the loess estimate of the rating date effect for week w, $g_{j,i}$ the composite effect of user's j preference for movie's i genre categorization, and $\epsilon_{i,j}$ the remaining residual error for movie i and user j.

Assessing model performance

We used the root mean squared error (RMSE) as the loss function to quantify the remaining residuals and assess the performance of the estimates on the test set, and at very last on the final holdout validation test set.

Data clipping

Ratings take discrete values in *0.5* intervals. The numeric range of our continuous rating predictions is not restrained, however. Therefore, final prediction values were capped and the optimal capping floor and ceiling values in the range of the observed rating scores were computed using the loss function.

RESULTS

Estimation of movie effect

Since the quality of a movie should have the largest effect on its rating, we first decided to derive a measure of how a movie, the subject of the rating, influences the review score. Naïvely put, are there movie effects? We examined the ratings distribution after averaging the ratings for each movie.

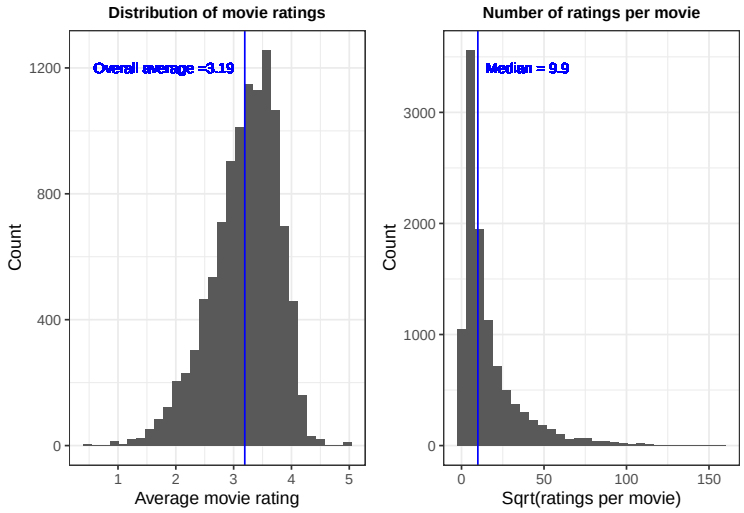

Figure 1: Movie statistics

Movie ratings show a slightly skewed, right-sided distribution, with 55.1% of all movie ratings scoring above overall average rating (Fig. 1, left). It is also obvious that most of the movies received less than 10 ratings (Fig. 1, right). Therefore estimating their average rating from the opinion of only a few users might not be very robust. Regularization was applied to introduce a penalty for low amount of ratings. We started building our model by accounting for the movie effect m_i for movie i:

$$\bar{R}_i = \mu + m_i + \epsilon_{i,j}, \text{ with average movie rating } \bar{R}_i \text{ and global mean } \mu,$$

and calculated $\hat{m}_i$ by simply subtracting $\hat{\mu}$ from $\bar{R}_i$. We also defined a tuning parameter λ to penalize predictions of rarely reviewed movies and regularized the effect by the amount of reviews that a movie has received to avoid over- or underestimation of rarely rated movies. We probed a range of λ and examined which value of λ minimizes the RMSE of the prediction in the test set.

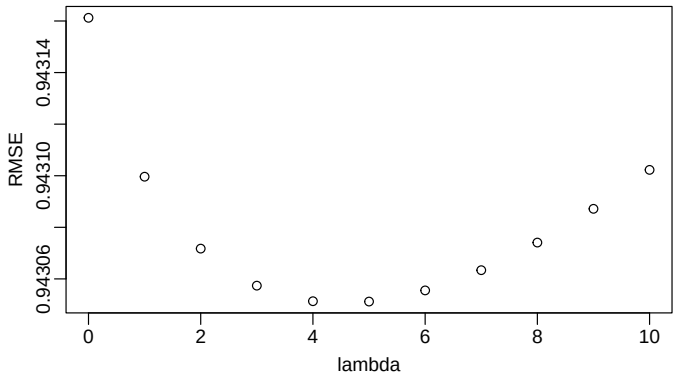

Figure 2: Movie effect regularization

A λ value of 5 was found to be optimal (Fig. 2). The RMSE obtained when using μ as the sole predictor dropped from 1.107 to 0.94305 by 0.164 when accounting for movie-specific effects.

Estimation of user effect

Next, we considered the effects of subjective rating based on user identity. How does the perception of the individual influence their judgement? We obtained the distributions of the users' average ratings and the amount of ratings each user has submitted.

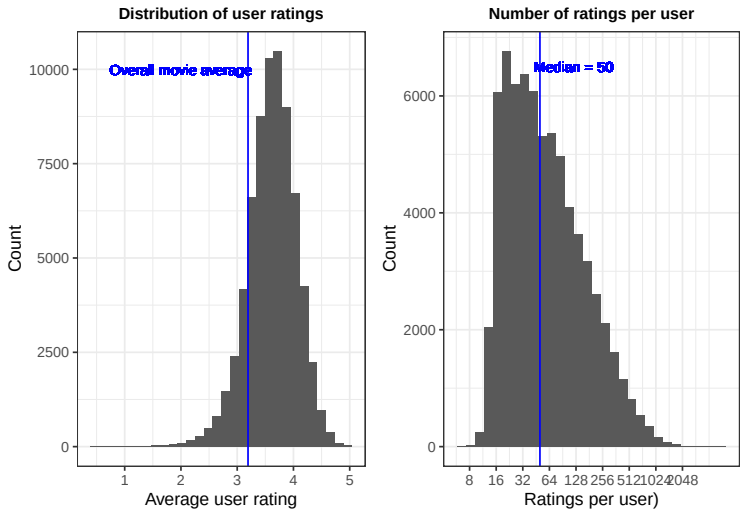

Figure 3: User statistics

Interestingly, the average ratings per user are very favorable, with 85.0% of users rating above μ on average (Fig. 3, left). We might therefore be able to identify the smaller subset of users that are more critical of movies in general and account for their demanding taste, but also those who particularly enjoy movies on average. Half of the users have submitted more than 50 ratings (Fig. 3, right). This implies that we might be able to make robust predictions for most of individual user's tendencies.

At last, we looked at how users' average rating relates to the amount of ratings they have delivered at the top and bottom 15% of the rating distribution

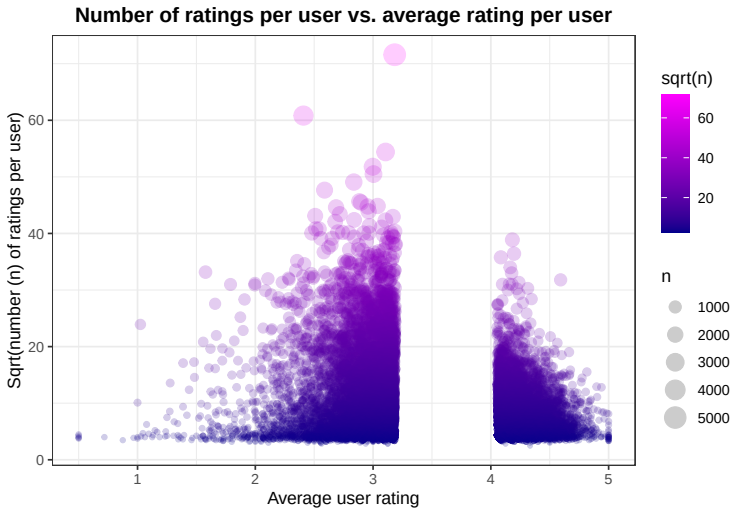

Figure 4: User activity vs. average user rating

More extreme raters usually submitted only relatively few reviews and the frequent raters have average ratings much closer to the global mean (Fig. 4). Therefore we also needed to penalize low rating activity by regularization of the user effect, as otherwise we might under- or overestimate the predicted rating for a movie. We are expanding our model by a user effect u_j for user j:

$$R_{i,j} = \mu + m_i + u_j + \epsilon_{i,j}$$

Our estimate for u_j is therefore:

$$\hat{u}_j = R_{i,j} - \hat{\mu} - \hat{m}_i$$

Further we regularize the effect by the amount of reviews that this user has contributed to the train set.

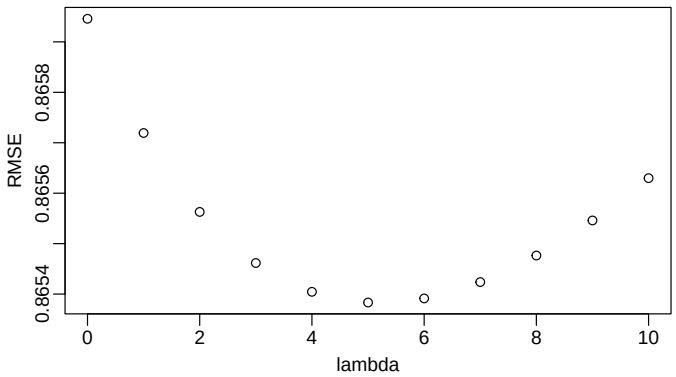

Figure 5: User effect regularization

A λ value of 5 was found to be optimal (Fig. 5). The RMSE dropped from 0.94305 to 0.86538 by 0.0777 when adding user- to the movie-specific effects.

Annual rating rate effect

To explain residual error after estimating movie and user effects, we computed how many ratings each movie has received per year (year after last rating date minus release year), and divided the resulting annual rating rate by the size of the training database, i.e. the total amount of reviews (we called this feature f_i, see Methods/Data cleaning section). The measure of how often a movie is rated allows the extrapolation to how often it is being watched and therefore how popular it is, as well as how reliable the ratings are (considering the central limit theorem).

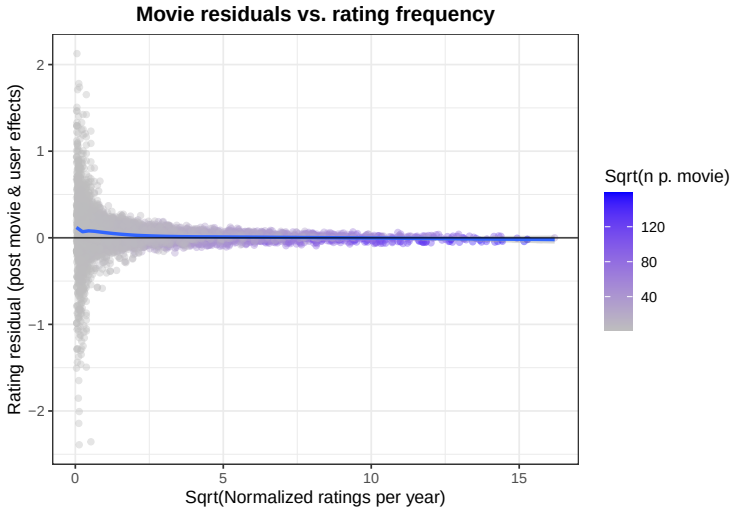

Figure 6: Residuals vs. annual rating rate

We took 2009 as the cutoff year as the years of release range from 1915 to 2008. We plotted the remaining residuals against f_i (Fig. 6) observing that movies with low rating frequencies tend to have positive residuals (Fig. 6), suggesting that we underestimated their ratings. In contrast, the residuals for very frequently rated movies were very low, showing that the present state of the model was able to make a robust prediction for these items (Fig. 6). We might still improve our estimate by accounting for low values of the f_i feature.

We applied the loess() function to estimate the effect a_i for the nearest integer of f_i, using an optimal span of 0.2. Since low rating rates correlated with positive residuals, regularization was not effective here. Assigning the effect estimate $\hat{a}_i$ to each movie i resulted in a RMSE of 0.8652, a further improvement by 0.00018 over the movie-user model.

Genre effect

We continued by examining if the updated residuals (after accounting for movie, user and rating frequency effects) could be explained by genre. At first, we filtered for those movies that have only one genre assigned. This allowed us to avoid the genre assignment ambiguities of many of the entries and explore if unambiguous, singular genre assignments have an influence on the residual errors.

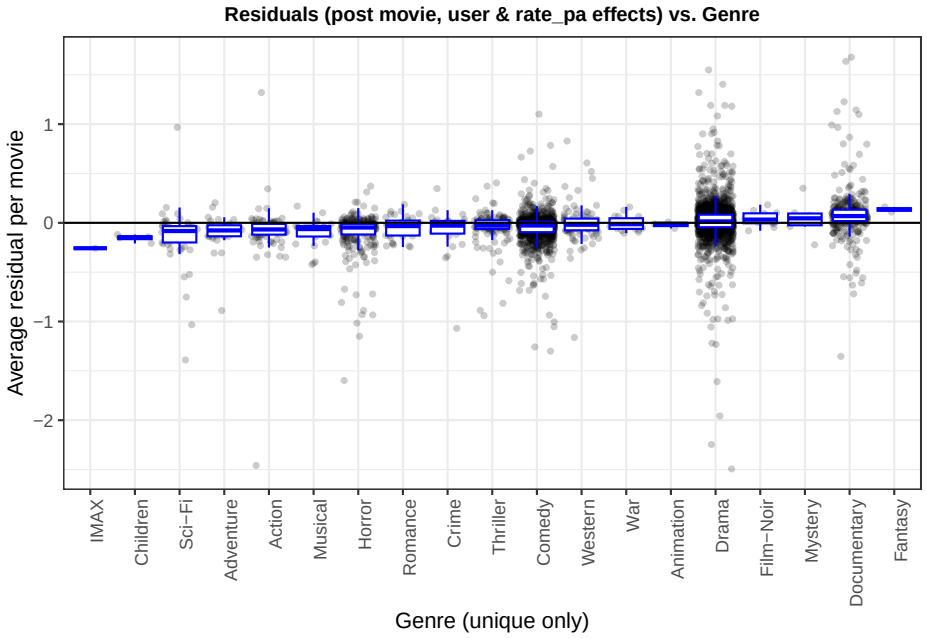

Figure 7: Residuals vs. definite genre assignments

When analyzing the distribution of average movie rating residuals across categories (Fig. 7), it was evident that genres like "Children" or "Fantasy" are made up of only a few movies, hence they are likely to be rarely the only genre attributes. Therefore, it wouldn't be possible to reliably estimate the effects of a singular genre for most genres. However, to make a general case, we explored if there were any significant differences between categories that are sufficiently crowded (n>50), i.e. "Horror", "Sci-Fi", "Action", Comedy", "Thriller", "Western", "Drama" and "Documentary" (Fig. 8). After asserting that the data is not normally distributed, we applied a pairwise Wilcoxon-test with multiple testing correction by the Benjamini-Hochberg method.

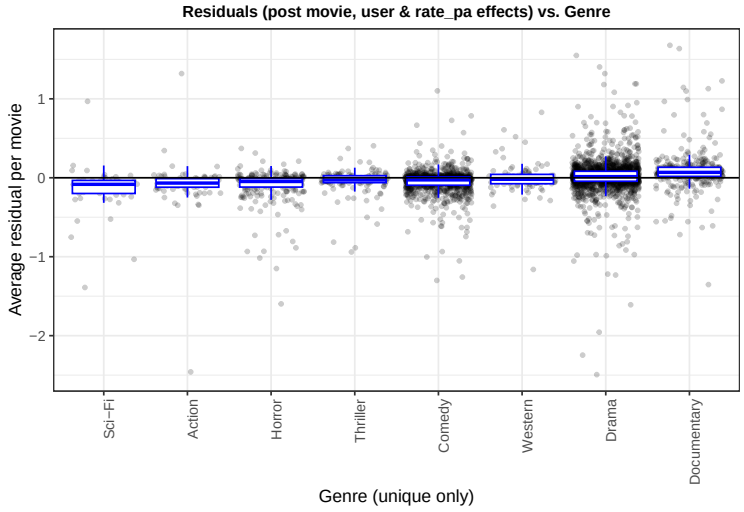

Figure 8: Residuals vs. definite genres (n>50)

Table 2: Pairwise Wilcoxon test p-values

	Action	Horror	Thriller	Comedy	Western	Drama	Documentary
Sci-Fi	0.0790875	0.0114932	0.0000384	0.0000291	0.0000291	0.0000000	0
Action	NA	0.4170620	0.0013342	0.0036606	0.0008606	0.0000000	0
Horror	NA	NA	0.0008950	0.0001124	0.0003325	0.0000000	0
Thriller	NA	NA	NA	0.3270976	0.4036444	0.0000012	0
Comedy	NA	NA	NA	NA	0.0511384	0.0000000	0
Western	NA	NA	NA	NA	NA	0.0087566	0
Drama	NA	NA	NA	NA	NA	NA	0

As can be seen in Table 2, the majority of comparisons, except for, e.g., "Action vs. Horror", "Sci-Fi vs Action" or "Comedy vs. Thriller" have a significant p-value < 0.05. We conclude that genre differences have significant effects on current model's movie rating residuals, with some genres, like Sci-Fi, having lower than 0 and some, such as Documentary, higher than 0 median residuals.

To compute a quantitative term g_k for how much any genre k influences the rating, we reiteratively filtered for genre assignments to contain each of the available genres (except for the rarely attributed non-genre term "IMAX") separately, and then grouped by movieId. I.e., if a movie is categorized as both "Fantasy" and "Action", it will contribute to the effect estimates within both "Fantasy" and "Action" genres. We then averaged the per movie rating residuals in the train data to obtain a genre effect estimate $\hat{g}_k$ for any genre k.

Figure 9: Average residuals per genre with 95% confidence intervals

As can be seen in Fig. 9, e.g. Horror and Action movies associated with negative residuals, implying overestimation by our current model, while Documentaries, War or Film-Noir movies have the most positive rating residuals, implying underestimation by our current model. It can also be seen that the ranking of the categories is quite well preserved when comparing to ranking of genres when we filtered for movies that had only one genre assigned to them (Fig. 8). Therefore, it is reasonable to assume that filtering single genres from combinations of genres to deduct the influence of any single genre is reliable enough.

Now we will incorporate the genre effects into our model:

$$R_{i,j} = \mu + m_i + u_j + a_i + g_i + \epsilon_{i,j}$$

We averaged the effect estimate for each genre that a movie is assigned to (as described in Methods/Modeling approach), in order to obtain $\hat{g}_i$, and examined if regularization for the size of the genre category in the training set had a beneficial effect.

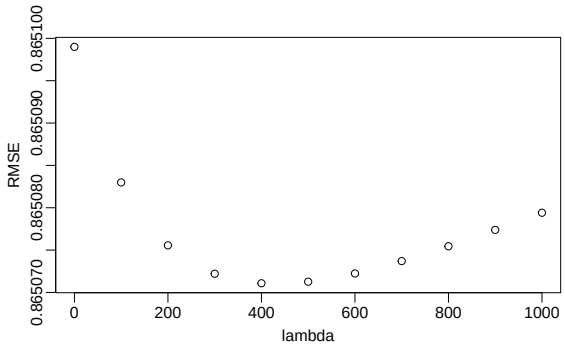

Figure 10: Regularization of genre effects

A λ value of 400 was found to be optimal (Fig. 10). The RMSE dropped from 0.8652 to 0.86507 by 0.00013 when accounting for movie, user, rating frequency and now movie genre effects.

Year of release effect

Next, we checked if there is any correlation between the year of release and the remaining average residuals per movie. We therefore plotted the residuals against year of release.

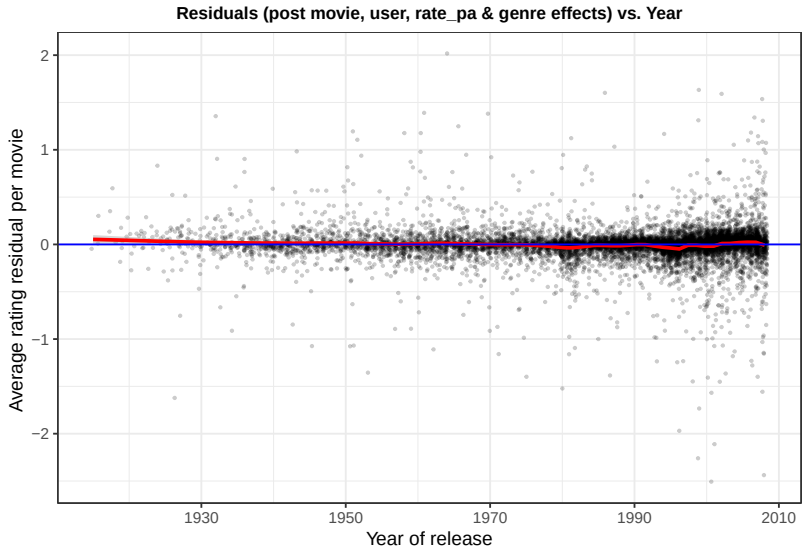

Figure 11: Residuals vs. year of release

We observed some time-dependency with movies created before the 1940s having slightly positive residuals. In addition, there were troughs around 1980 and 1995, resulting in slightly negative residuals (Fig. 11). We deemed it worth to account for these effects. Thus, we add the effect of the year y_i for a movie i to our model:

$$R_{i,j} = \mu + m_i + u_j + a_i + g_i + y_i + \epsilon_{i,j}$$

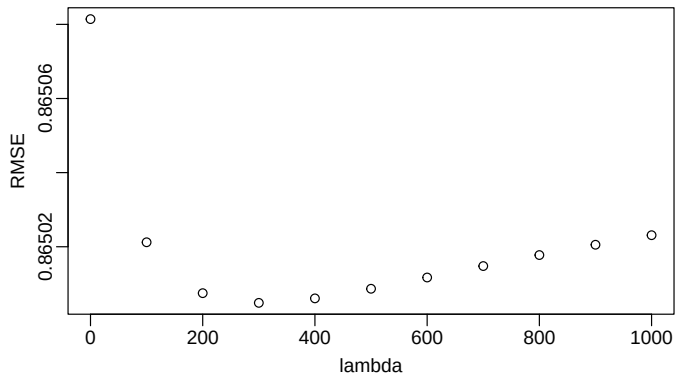

Figure 12: Year effect regularization

By estimating the year effect with the loess() function using a span of 0.16 and an optimal λ of 300 (Fig. 12), the updated model yielded a RMSE of 0.865, which is a further, modest improvement by 0.00007 over the previous model increment.

Date of rating

Next, we examined if the time of review affected the rating submitted. To explain such effects, one possibility would be that over time users became more critical of movies and overall ratings declined or that sentiment trend was influenced in some other way. We rounded the dates to weeks to limit the number of time points to a reasonable number.

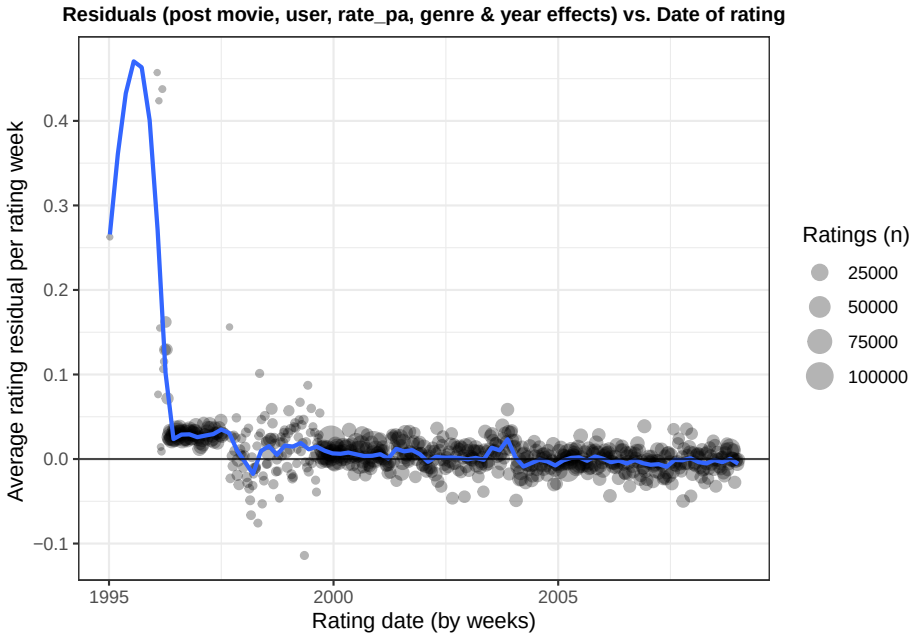

Figure 13: Residuals vs. rating date

As can be seen from Fig. 13, rating activity began in the 1990s. There is a downtrend in rating residuals between the end of 90s and early 2000s with a small spike around 2004. It seems as if in the 90s the rating residuals are more positive overall, maybe due to enthusiasm about the initial phase of the internet.
We modeled the date of review trend with the loess algorithm using a span of 0.025 to obtain the estimate $\hat{d}_w$. Regularization with the weekly rating batch size did not benefit the prediction accuracy.

We could improve the RMSE to 0.86489 by 0.00011 .

User/genre interaction effect

Lastly, we considered if the remaining residuals could be reduced by accounting for individual user's preferences for certain genres. We obtained the residuals and calculated the average residuals per user and per genre. For demonstration purposes we plotted the residuals for five users, that have contributed over 1000 ratings to our training data set against genre categories (Fig. 14). User "58357", for instance, seems to have relative antipathy against Documentaries, Film-Noir or Western movies, while user "42791", in contrast, is more in favor of these genres. It therefore seemed reasonable to identify and factor in these biases.

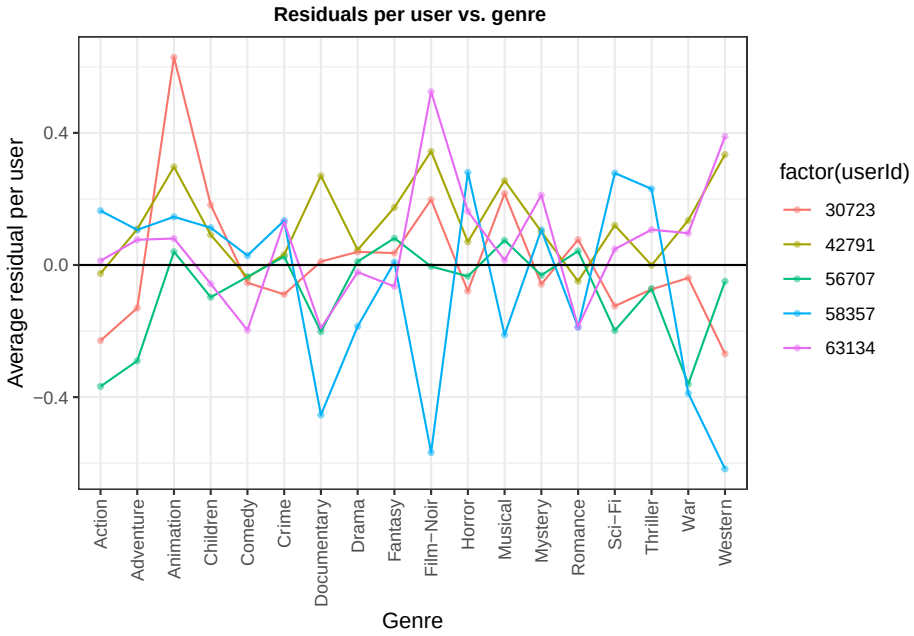

Figure 14: Residuals per user (frequent raters) vs. Genre

We took the averaged users' residuals $\hat{g}_{j,k}$ and then averaged these over the genres of each rated movie in the training set to obtain the effect estimate $\hat{g}_{j,i}$. Regularization by the amount of submitted ratings per user improved the RMSE at an optimal tuning parameter λ of 20 (Fig. 15).

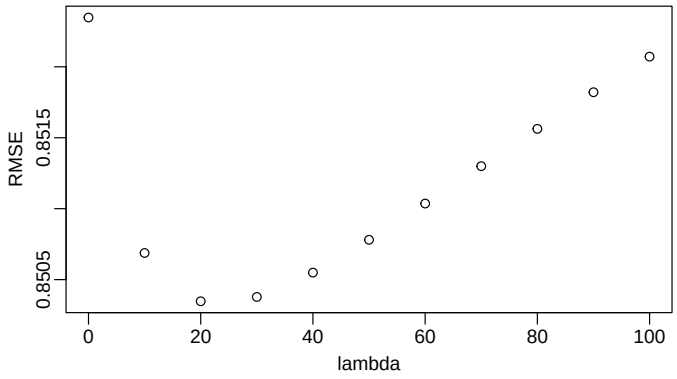

Figure 15: Regularization of genre-user effects

The RMSE now is 0.85035, an improvement by 0.01454.

Thus, our final model is:

$$R_{i,j} = \mu + m_i + u_j + a_i + g_i + y_i + d_w + g_{j,i} + \epsilon_{i,j}$$

Last model tuning

We then looked at the distribution of our finalized predictions against the true ratings to visually evaluate the utility of the current model (Fig. 16).

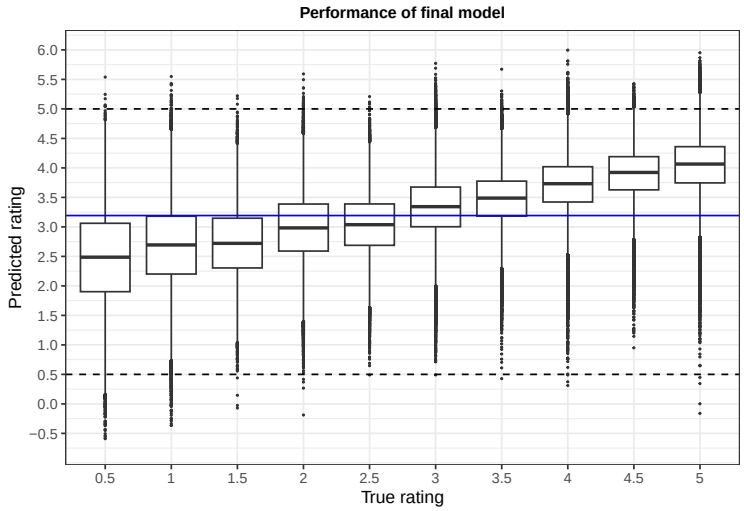

Figure 16: Predicted vs. true ratings

Our predictions per true rating stratum generally showed a good matching, ascending trend. The correlation between predicted and true ratings reached: $r = 0.5966$

However, we can see that some of our predicted ratings extend over the natural range of the ratings. We, therefore decided to cap our predictions. While it is reasonable to cap them to 0.5 as minimum and 5.0 as maximum values, it is not immediately clear if this would be the optimum, since our failed estimates extend over a large range of true ratings. For this reason, we built a simple, RMSE minimization function to find the optimum cut-offs. The floor should be somewhere below the global mean and the ceiling above the global mean.

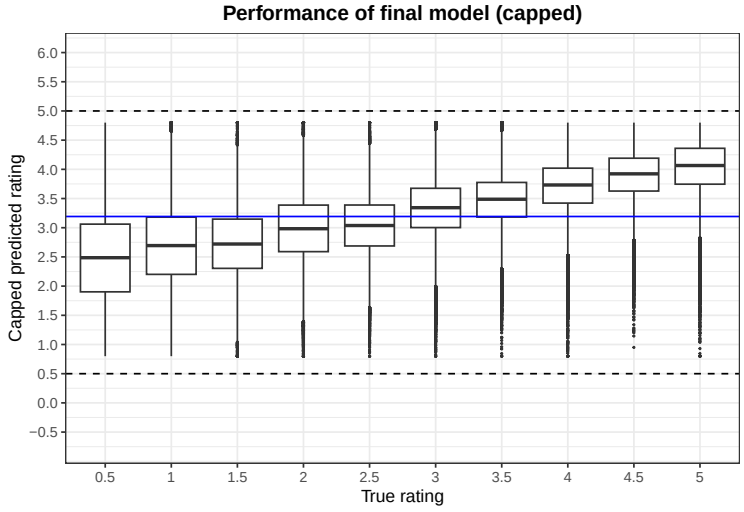

Figure 17: True vs. cap-predicted ratings

We obtained an optimum floor value of 0.8 and a ceiling value of 4.8. When applying this clipping to the model prediction on the test set, the RMSE improved to 0.84987 by another 0.00048. Capping, therefore, decreases the impact of our misses and makes our predictions more conservative. Capping affected 0.9% of values which appears reasonable and not too corrective (Fig. 17). The correlation coefficient r between true and capped predicted ratings slightly rose to 0.5972
In summary, incremental addition of feature effect estimates to reduce residuals led to a step-wise decrease in RMSE:

Table 3: *Step-wise development of a linear, rating prediction model and the according RMSEs*

model	RMSE	Improvement
Average	1.10700	0.00000
Movie	0.94305	0.16395
Movie User	0.86538	0.07767
Movie User Rating.pa	0.86520	0.00018
Movie User Rating.pa Genre	0.86507	0.00013
Movie User Rating.pa Genre Year	0.86500	0.00007
Movie User Rating.pa Genre Year Revdate	0.86489	0.00011
Movie User Rating.pa Genre Year Revdate UserxGenre	0.85035	0.01454
Movie User Rating.pa Genre Year Revdate UserxGenre CAP	0.84987	0.00048

Performance on the final holdout test set

At last, the performance of our prediction algorithm was assessed on the final holdout validation test after adding the computed feature effects.

RMSE when applying final model and prediction capping to the final holdout validation set: **0.84989**.

CONCLUSION

We have demonstrated that it is possible to devise a recommendation system based on *a priori* knowledge of users and their rating behavior. By generating a linear model through incremental reduction of residual errors, we achieved fairly high accuracy of our predicted ratings in a validation set, amounting to a RMSE of **0.84989**. Estimation of movie and user effects, followed by preferences of users for genres, had the largest share in reducing the loss, and these features might be preferentially considered when developing models on similar data sets. Accounting for variables such as year of release, movie genre, rating rate and rating date further improved our predictions, albeit rather modestly. By applying regularization, where warranted, we ensured to prevent overfitting of our model. Indeed, the RMSE when applying the model with regularization was virtually the same on our test set and the final holdout validation set.

It is conceivable that the recommendation system can be improved further. We have not taken into account individual user preferences for certain movie characteristics other than genre. A matrix factorization approach could have utility in explaining preferences of specific users for certain movie types, such as artsy as opposed to blockbuster movies, or for particular actors. In addition, our prediction system only allows us to extrapolate to movies and users that were contained in our training data set. To build a recommendation system that can predict ratings of novel movies, it might be warranted to explore the power of movie title effects. Some movie titles might contain additional information. For instance, we know from experience that sequels are likely to do worse than the original pilot movies. On the other hand, users that had rated a pilot and its sequel movie highly, might want to watch the second sequel. Furthermore, a text mining language analysis of movie titles, by for instance sentiment analysis, could be used to train an algorithm by assessing if certain terms associate with better or worse ratings. This would allow for building a true recommendation system that can make a judgement based on genre and title of movies that have not been part of a training data set, but also incorporate known genre preferences of particular users.