Sampling Bias in Early Movie Reviews
ECO1400 Term Paper

Jordan Long
ID: 1002541902

December 2021

Contents

1 Executive Summary

This paper explores how reviews on online platforms left by early adopters of new products differ from reviews left by later adopters, in the context of movie ratings. One reason to expect a difference is that a person watching a newly released movie likely had a high pre-existing interest in the movie that brought them to the theatre, and this pre-existing interest makes them more likely to enjoy the film. This is important because late adopters often use the signals from online review aggregators to make purchasing and consumption decisions. Therefore, being able to anticipate this early ratings bias can help them to make more informed decisions about new products.

To investigate this phenomenon, a dataset of reviews scraped from the Internet Movie Database (IMDb) is explored. The dataset contains ratings out of ten left by users who write written reviews for movies on the IMDb platform, along with the user and movie IDs, and the date of the rating. To the extent that these written reviews approximate the IMDb movie rating score that is computed from ratings by all users (not just those that leave reviews) and displayed on the site's main page, we can quantify the early ratings bias in a commonly used metric for deciding whether or not to watch a movie. The paper uses this dataset to construct a panel of average movie ratings in the periods following their release. Then, using standard panel data estimation techniques while controlling for seasonality and movie-specific effects, the paper estimates the extent to which the IMDb score of an average movie declines in each month following the movie's release.

It is found that the average movie's IMDb score declines by 0.5 points out of 10 in the first four months following the movie's release, with the long term amount of decline roughly stabilizing at 0.76 points out of 10 by three years after the movie's release. Remarkably, although the estimation techniques did not restrict the way that the IMDb score could evolve, it happens to follow a very clearly identifiable pattern to the human eye.

2 Introduction

Anecdotally, in the months after a movie is released, its rating on IMDb declines substantially. One plausible explanation for this is that the first people to watch and review the movie are the movie's fans, who are more inclined than the typical movie watcher to rate the movie favourably. This implies that there is systematic bias in a movie's early ratings, making it difficult to accurately assess the underlying quality of a movie. The problem that this analysis tries to solve is to verify and quantify how ratings decline in the months following a movie's release.

In order to answer this question, this paper builds a panel from a dataset of 1.4 million IMDb movie reviews from different users over time, encompassing over 3,000 popular movies. An assumption implicit in this question that the analysis will explore before attempting to answer the question is whether movie ratings can indeed be said to eventually stabilize. This is addressed by estimating a nonparametric regression of ratings in the months following a movie's release, which provides insight into how ratings change over time.

These results are immediately useful for using early IMDb ratings to assess whether or not to watch a movie, as understanding how ratings decline allows one to very tractably make predictions as to a movie's long-term rating. Further, this research makes clear the time frames that should be used in framing a predictive model for IMDb movie ratings. Such a predictive model could be implemented in a browser extension that adjusts IMDb pages to display predicted movie ratings.

As well, this problem is similar to that tackled by recommender systems, where the typical problem is to predict the rating that a given user will assign to a given movie. The difference is that in this paper the prediction is on a more aggregate level, predicting the average rating of a movie across the subset of the population that leave a rating rather than for an individual. These aggregate ratings could be valuable from a cultural lens, providing a representation of cultural tastes in the same way that an individual's movie ratings are a representation of their tastes. Further, understanding how these aggregate ratings are influenced by selection to leave a rating can provide more insight into the extent to which they are a reliable representation of culture.

Finally, there are numerous organizations that aggregate crowd ratings of product quality, such as GoodReads for books, Amazon for general products, or Yelp for restaurants, and similar early selection-to-rate effects could obscure underlying product quality in each of these scenarios.

3 Literature Review

Ramos et al. (2015) investigate patterns in IMDb movie ratings and suggest a model for the way that people become aware of movies that is independent of movie-specific features. They use three cross sections of movie ratings for the same movies, collected in March 2013, December 2014, and January 2015. They find that the distribution of number of votes across movies follows a scale-free power-law behaviour, even when subdividing their data by genre, year of release or positive/negative ratings. Because of this stable pattern, they suggest that their model for the evolution of the number of votes of a film need not incorporate movie-specific features, or even the ratings of the movie. Thus, they model the evolution of the number of votes a film receives using a network of social connections where interest in a movie spreads across edges in the network, and show that this model is consistent with the distribution of votes seen. However, they note that this model is not representative for high-budget films.

Lorenz (2009) also documents facts about movie rating distributions. Here, the focus is on finding distributions for individual ratings that are consistent with observed aggregate rating distributions.

Li and Hitt (2008) uses Amazon book ratings to document decreasing product reviews over time and assess the extent to which later purchasers account for this bias. They document decreasing product reviews using the specification $AvgRating_{it} = \alpha + \beta(T^{\lambda} - 1)/\lambda + u_i + e_{it}$ where $AvgRating_{it}$ is the average review of book i at time t, T denotes the time since a book's release, and u_i is a book fixed effect. They find evidence that product reviews decline significantly over time. Further, they find that consumers do not account for this bias completely, and they provide a framework for measuring the loss in consumer welfare that results from this bias.

Luca (2016) uses Yelp reviews and Washington State restaurant data to identify the causal impact of Yelp ratings on demand, finding that a one-star increase on Yelp causes a 5-9 percent increase in restaurant revenue. He also finds that consumers respond more to ratings with more reviews.

In the spirit of Ramos et al. (2015), the current paper documents empirical trends in IMDb movie rating data. The trend documented here for movies is similar to that of Li and Hitt (2008) for books, evidencing a declining trend among movie ratings over time.

4 Data Source and External Validity

The data used for this analysis comes from IEEEDataport (Baghi, December 2020). The original data source contains the IMDb userID, movieID, rating, and review date for 4,669,820 ratings on a 10 point scale from 1,499,238 users to 351,109 movies. However, to improve external validity we only use 1.4 million reviews of 3,530 movies by 600,000 users. This selection process is described in the next section. This section discusses two critical aspects relating to external validity: the relationship between ratings left with written reviews and ratings left more generally; and the possible influence of bots on ratings. As the aim of this analysis is to understand how IMDb ratings evolve over time, which is more interesting if IMDb ratings are reliable indicators of the broad cultural appeal of various movies, these two aspects of external validity are of great importance.

Notably, each of the movie ratings come from written reviews on the IMDb website, rather than from users simply clicking to rate a film. This means that ratings in the dataset are not necessarily representative of the ratings that are displayed on IMDb movie pages, since there is likely a different selection mechanism for writing a review than leaving a rating. The question of the external validity of this analysis to actual movie ratings is relevant because IMDb displays an unspecified weighted average of ratings rather than average reviews on a movie page. Thus, people probably make their judgement of a movie quality based on a function of its ratings, rather than average review. To address this element of external validity, the dataset is further augmented with data on average ratings and number of votes downloaded from the IMDb interface at `https://www.imdb.com/interfaces/` (2021). Taking the difference between the average review and average rating for each movie reveals that the median movie has an average review 0.5 points higher than its average rating. As well, ratings and reviews do track each other somewhat closely, with over 50 percent of movies having average ratings within 2 points of their average review. The full distribution of

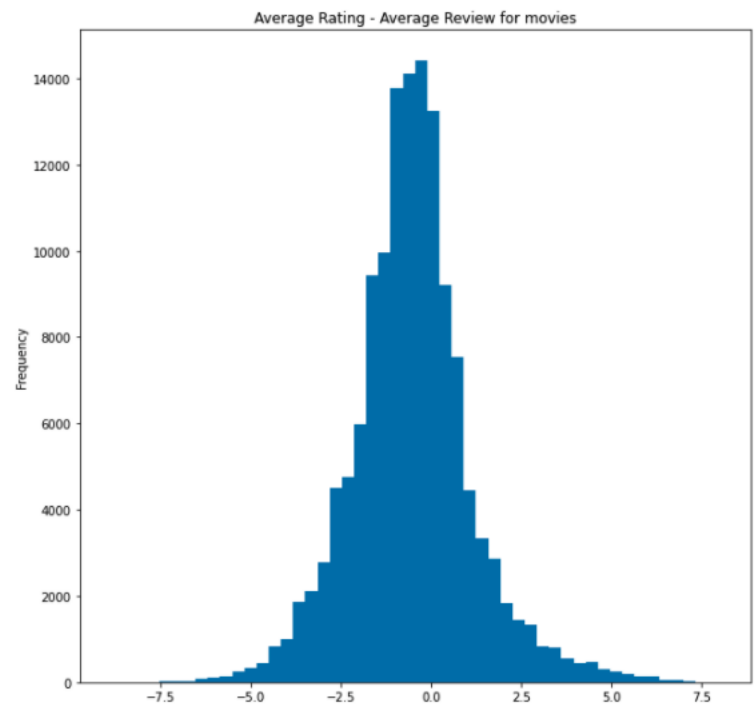

Figure 1: Ratings are generally higher than reviews

the difference in mean ratings and reviews is presented in Figure 1. This is consistent with people providing a review over just a rating when they are moved by the high quality of a movie. Another relevant difference between ratings and reviews is their volume. The median movie has 62 times more votes than ratings, implying that those leaving a review are indeed a very select subset of people rating a movie, who are themselves a select subset of people watching a movie.

However, one advantage of using review data as opposed to rating data is that it is likely subject to higher quality control, as it is harder for bots to write reviews than simply leave ratings. If bots had a sizeable influence on reviews, one might expect to observe users that have left an implausibly high number of reviews. Indeed, the user with the most reviews in this dataset has reviewed 24,145 films since June 2003, amounting to an average of over 4 films a day for 18 years! While this may seem implausible, by visiting the user's profile (`https://www.imdb.com/user/ur2467618/`) one can see that their reviews seem considered and that the user seems to be employed as a writer. This should be taken to indicate that the data is of generally high quality. Further, the dataset contains all of the reviews of this user, indicating that the dataset contains all reviews rather than just a sample. Further evidence that the dataset is not subject to the influence of bots is that the most reviews left by a single user on a single movie is less than 7, and that only 1 percent of users have left multiple reviews for the same movie.

5 Data Exploration, Augmentation and Transformation

The data source contains no information regarding how the data was sampled so the data was explored to better understand where it came from and what insights it might contain. A summary of this exploration follows, and a more detailed exploration can be found in the attached jupyter notebook.

The distribution of users by number of reviews is very skewed. While there are nearly 1.5 million users, there are only 50,000 users with over 10 reviews, and only 3,000 users with over 100 reviews. By taking a mean of each user's ratings and comparing them, we find that the median user gives a mean rating of 8.0. However, among users that leave more than 10 reviews, the median user gives a mean rating of 6.8, indicating that many of the reviews from people who rate infrequently are high. This is consistent with most people choosing to leave a rating only if they believe a film is particularly good. The dataset spans reviews from July 1998 through to December 2020, with three main noticeably different time periods. The first, from 1998 through 2004 have a growing but small number of ratings. The years 2005 through 2018 have a stable but twice as large number of ratings over time. Then, it appears that there were no ratings in the first three months of 2018, followed by another doubling in the number of ratings over time for 2019 and 2020. Two dates in 2019 and 2020 have an exceptional number of ratings, corresponding with the airing of the last Game of Thrones episode, and the release of the Indian film Dil Bechara (a remake of The Fault in Our Stars), respectively. All of this can be seen in Figure 2. This last point makes it clear that not every entry in the dataset is in fact a movie, and so further data is found to select only movies, as described below. The distribution of films by number of reviews is also very skewed. While there are 350,000 films in the dataset, only 52,000 have more than 10 reviews, and only 8,000 have more than 100 reviews.

In order to capture more information about which films in the dataset are in fact movies, the ratings dataset is augmented with data downloaded from the IMDb interface. This data includes film title, film type (movie, tvEpisode, short, tvSeries, tvMovie, etc.), release year, runtime, and genres. By keeping only movies, this reduces our number of movies from 351,109 to 139,631.

In order to study the way that movie ratings change in the months following their release, we would like a more precise estimate of the release date of a movie than the release year included in this data. We make the assumption that a movie's release date coincides with the date of the movie's first review. This assumption is more likely to be true for more popular movies, as the more people that watch a movie on its release date, the more likely one is to leave a review. Thus we further restrict our dataset to movies with over 100 reviews. Another artefact in the data that makes this assumption false is the fact that we do not have any reviews over the first few months of 2018. Hence, all movies that were released in these months are considered to be released in May 2018 under this strategy. For this reason, we remove movies classified as released in May 2018 from

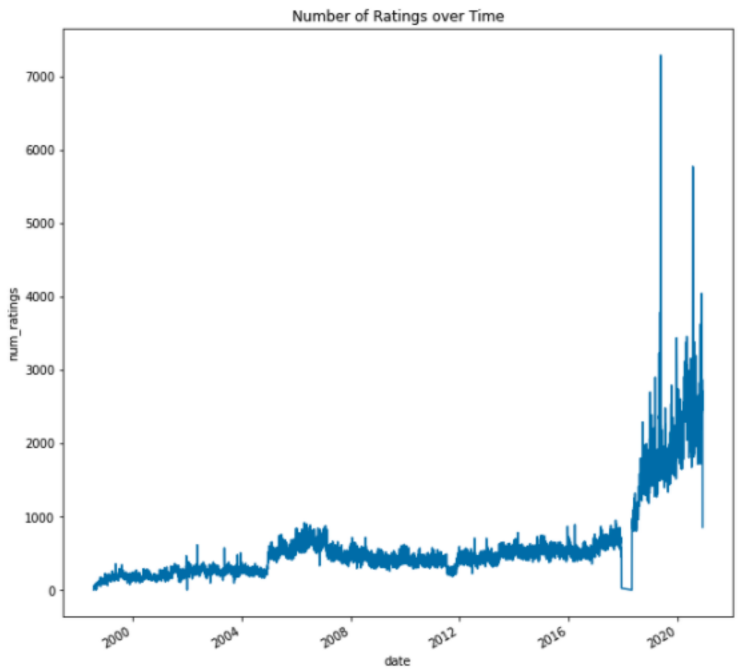

Figure 2: Number of reviews over time

our dataset. A third way that this assumption can be false is that it forces movies released before 1998 to be classified as released after 1998. However, we wish to exclude these movies from our analysis, since we don't have IMDb ratings for them around their release. Thus we filter on the release year provided by IMDb being after 2005 to exclude such movies. 2005 is chosen because January 2005 coincides with what appears to be a large change in the way users use IMDb, as evidenced by the sharp increase in daily numbers of reviews in Figure 2.

This leaves us with a final review dataset of 1.4 million reviews of 3,530 movies by 600,000 users. We are using almost 100 times fewer movies than we started with, but still more than one quarter of the original reviews, and more than one third of the original users.

A panel structure, common in economics, consists of multiple observations of the same subjects, typically over time. From this review dataset, we create a panel structure consisting of movies and their evolving average ratings on an aggregated monthly basis, with the aim of identifying an underlying time trend in average ratings.

6 Model

To isolate the effect of time since release on a movie's average rating, we estimate the following regression of the average rating of a movie, y_i:

$$y_{ir} = \sum_r \mu_r + \alpha_i + \lambda_i + \gamma_{month} + \beta_r + \epsilon_{ir}$$

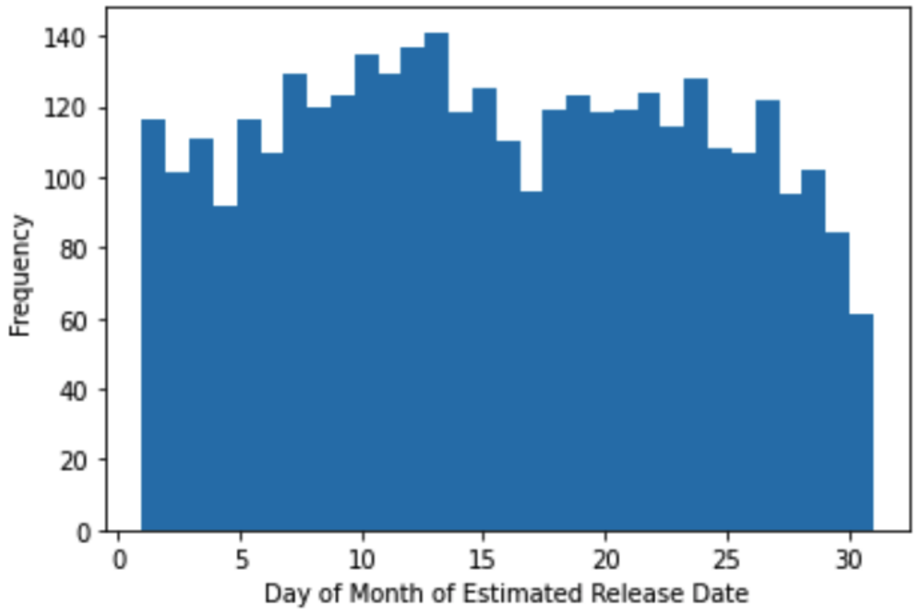

Figure 3: Distribution of movie releases over days of the month

Here μ_r is an indicator for whether we are observing the average rating r months after a movie's release, and these are the parameters of interest. Our data contains reviews as far as 191 months after a movie's initial release, and so r ranges up to this value. While not all movies are observed at all values of r, these missing data points are handled by the estimation procedure. Including a potentially different rating offset for each month after a movie's release allows us to see trends in ratings over time that are more flexible than enforcing a linear or other parametric relationship. One cause for concern in the use of this model is that in aggregating reviews to a monthly level, the average review in the first month of a movie's release may be an average of very few ratings if the movie was released late in the month. While this may add noise to our results, it is unlikely to bias our coefficients either upwards or downwards. Figure 3 shows that using our estimate of a movie's release date, movie releases follow an almost uniform distribution across days in the month.

The other parameters are as follows:

- α_i is a movie specific effect on ratings. Because of the large number of movies in the data, it is computationally infeasible to estimate each of these coefficients, and so we rely on efficient fixed effects and random effects estimation procedures to avoid needing to estimate these parameters.

- λ_i is an indicator for the month in which a movie was released, which may help capture movie-specific attributes. For example, indie films are probably first reviewed in months with big film festivals, large blockbusters may be more common in the summer, and horror movies are probably released close to Halloween. As a time-invariant covariate, this parameter will not be estimated when using the within-transformation for the fixed-effect model.

- γ_r is an indicator for the month of observation, which is included to capture any seasonality in ratings.
- β_i is an indicator for whether the observation is after 2018, which coincides with a large increase in the volume of IMDb ratings and could reflect a different sample of people leaving ratings with potentially different aggregate preferences.

Note that this specification masks a lot of heterogeneity in the way that movie ratings evolve over time, as movie specific effects only have the potential to affect levels of ratings rather than the way that these ratings evolve. Thus the main purpose of this specification is to capture aggregate trends in the data. μ_{12} has an easy interpretation as the mean difference in a movie's average rating one year following its initial release compared to the month of its release, assuming either that both observations are before 2018 or both after 2018. Other interpretations of μr parameters are complicated by seasonality coefficients, but still possible. For example, the difference in mean average ratings between the first and second months following a movie's release for movies released in January 2019 is $\mu_2 - \mu_1 + \gamma_{March} - \gamma_{February}$.

To estimate this model both a fixed effects and a random effects estimator are tried and a Hausman test is used to assess which one to use. Both of these estimation approaches assume that there is no endogeneity between covariates and the error terms ϵ_{ir}.

7 Empirical Results

Both a fixed effects model and a random effects model were fit using the specification above, and a Hausman test was performed to choose between them. The Hausman test has a p-value of 1, indicating that there are no grounds to reject the null hypothesis that covariates and movie-specific effects are uncorrelated. Thus, since both fixed effects and random effects are consistent estimators, but random effects is more efficient, we use its results.

See Table 1 for a listing of all of the coefficient estimates from the model. The most important finding is that the coefficients for month from release (first review month offset) range in value from -0.32 to -0.76 over the 10 years after a movie is released, with coefficients reaching -0.7 by the end of the first year. Further, all of these estimates are statistically different from zero at the 1 percent level, and all estimates in the first 10 years have standard errors of less than 0.02. From this we interpret that the mean drop in average rating for a movie one year after its release is 0.7 points out of 10. Further, each of the coefficients for month of observation (month) are relatively small (less than 0.02), and many are not statistically significantly distinguishable from 0 at the 5 percent level. On the other hand, each of the coefficients for month from release in the first six

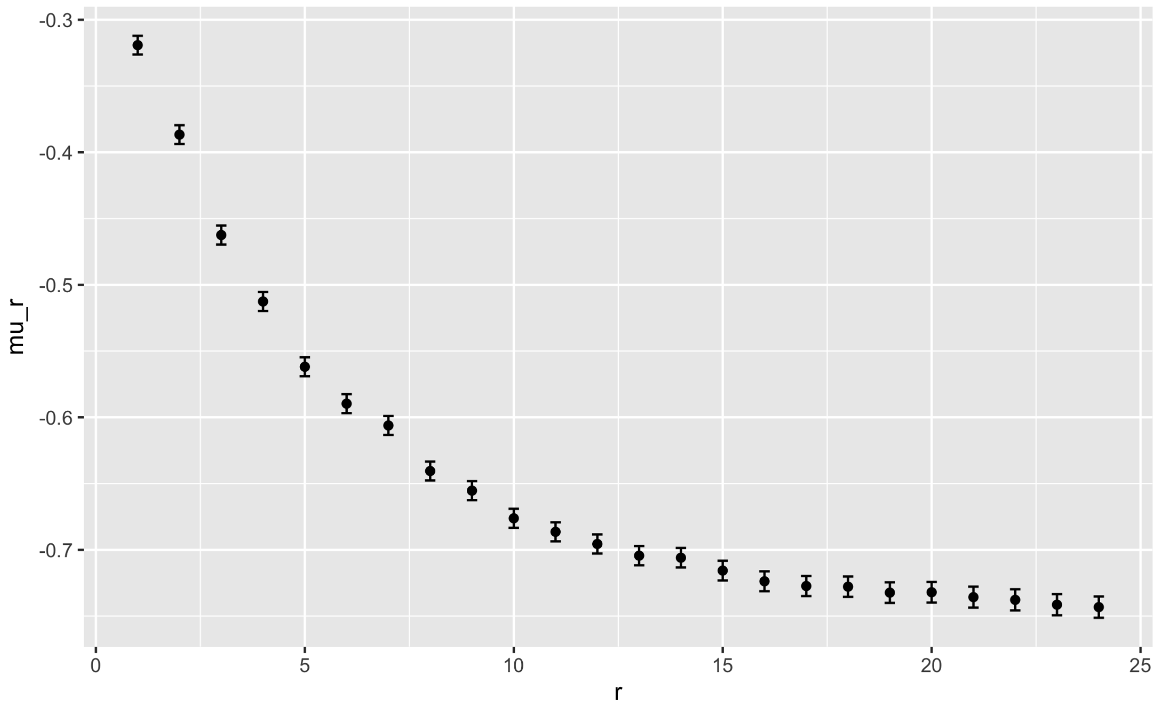

Figure 4: μ_r estimates for r ranging from 1 to 24

months vary by much more than this, and so the difference in mean average rating is dominated by the coefficients for μ_r. Most of the release month coefficients (first review month 1 through 12) are statistically indistinguishable from zero at the 5 percent level, though movies released in September have statistically significant 0.32 point higher ratings than movies released in January. Note that September is the month in which the Toronto International Film Festival occurs, which sees the release of many critically acclaimed indie films all at once. Finally, the coefficient for whether a film is being rated after 2018 is statistically significant at the 1 percent level, associated with a modest mean increase in ratings of 0.04.

Figure 4 displays the coefficient estimates for the first 24 months following a film's release. Again, while the difference in consecutive estimates technically do not represent the difference in mean ratings associated with an additional month after a movie's release, they practically do thanks to the very small month effects. The graph shows that there is a clear decrease in average reviews, which seems to follow a clear parametric trend.

Bringing this back to the original question about predicting where a movie's rating may end up, it appears that movie ratings reliably stabilize within three years at an average rating 0.76 points lower than in their month of release. This evidence favours the idea that early moviegoers, or the early moviegoing experience, is markedly different from that later on.

8 Conclusions

Using IMDb movie review data, we have constructed a panel of average movie ratings for movies in the months following their release. We have used this to explore the way that movie ratings change over time through a nonparametric random effects model including month of rating and month of movie release as covariates. It is found that movie ratings decline substantially, 0.7 out of 10 points, in the year following a movie's release.

One area for further research includes exploring heterogeneity in movie rating trajectories. Another is building a predictive model that includes more movie-specific features. The finding of this paper that ratings largely stabilize after three years makes the problem of predicting long-term ratings more well-defined. A final area for further research is collecting and using data on IMDb movie ratings rather than just reviews.

9 References

Baghi, Vahid. December 2020. "IMDb Users' Ratings Dataset." *IEEEDataport* https://doi.org/10.21227/br41-bd49.

IMDb. December 2021. https://www.imdb.com/interfaces/

Li, Xinxin, and Hitt, Lorin. 2008. "Self-selection and Information Role of Online Product Reviews." *Information Systems Research* 19,no.4 456-474. https://doi.org/10.1287/isre.1070.0154

Lorenz J. 2009. "Universality in movie rating distributions." *The European Physics Journal B.* 71: 251-258. https://doi.org/10.1140/epjb/e2009-00283-3.

Luca M. 2016. "Reviews, Reputation, and Revenue: The Case of Yelp.Com". *Harvard Business School NOM Unit Working Paper* No. 12-016 http://doi.org/10.2139/ssrn.1928601

Ramos M, Calvão AM, Anteneodo C. 2015. "Statistical Patterns in Movie Rating Behavior." *PLoS One* 10,no. 8. https://doi.org/10.1371/journal.pone.0136083

10 Attachments

Table 1: Fixed Effects and Random Effects Regression Coefficients

	Dependent variable:	
	cumavg_review	
	FE	RE
	(1)	(2)

as.factor(first_review_month_offset)1	-0.319^{***}	-0.319^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)2	-0.387^{***}	-0.387^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)3	-0.462^{***}	-0.462^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)4	-0.513^{***}	-0.513^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)5	-0.562^{***}	-0.562^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)6	-0.590^{***}	-0.590^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)7	-0.606^{***}	-0.606^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)8	-0.641^{***}	-0.641^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)9	-0.656^{***}	-0.655^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)10	-0.676^{***}	-0.676^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)11	-0.687^{***}	-0.686^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)12	-0.696^{***}	-0.696^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)13	-0.705^{***}	-0.704^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)14	-0.706^{***}	-0.706^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)15	-0.716^{***}	-0.716^{***}
	(0.007)	(0.007)
as.factor(first_review_month_offset)16	-0.724^{***}	-0.724^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)17	-0.728^{***}	-0.727^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)18	-0.728^{***}	-0.728^{***}

	(0.008)	(0.008)
as.factor(first_review_month_offset)19	-0.733^{***}	-0.732^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)20	-0.732^{***}	-0.732^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)21	-0.736^{***}	-0.736^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)22	-0.738^{***}	-0.738^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)23	-0.742^{***}	-0.741^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)24	-0.744^{***}	-0.743^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)25	-0.746^{***}	-0.746^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)26	-0.747^{***}	-0.746^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)27	-0.747^{***}	-0.747^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)28	-0.748^{***}	-0.747^{***}
	(0.008)	(0.008)
as.factor(first_review_month_offset)29	-0.748^{***}	-0.748^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)30	-0.752^{***}	-0.752^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)31	-0.754^{***}	-0.754^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)32	-0.758^{***}	-0.757^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)33	-0.756^{***}	-0.756^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)34	-0.755^{***}	-0.755^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)35	-0.758^{***}	-0.757^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)36	-0.757^{***}	-0.756^{***}
	(0.009)	(0.009)

as.factor(first_review_month_offset)37	-0.759^{***}	-0.758^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)38	-0.760^{***}	-0.759^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)39	-0.760^{***}	-0.760^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)40	-0.762^{***}	-0.762^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)41	-0.763^{***}	-0.762^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)42	-0.760^{***}	-0.760^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)43	-0.762^{***}	-0.761^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)44	-0.757^{***}	-0.757^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)45	-0.762^{***}	-0.761^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)46	-0.762^{***}	-0.762^{***}
	(0.009)	(0.009)
as.factor(first_review_month_offset)47	-0.761^{***}	-0.760^{***}
	(0.010)	(0.010)
as.factor(first_review_month_offset)48	-0.760^{***}	-0.760^{***}
	(0.009)	(0.010)
as.factor(first_review_month_offset)49	-0.764^{***}	-0.763^{***}
	(0.010)	(0.010)
as.factor(first_review_month_offset)50	-0.760^{***}	-0.760^{***}
	(0.010)	(0.010)
as.factor(first_review_month_offset)51	-0.764^{***}	-0.763^{***}
	(0.010)	(0.010)
as.factor(first_review_month_offset)52	-0.766^{***}	-0.766^{***}
	(0.010)	(0.010)
as.factor(first_review_month_offset)53	-0.768^{***}	-0.768^{***}
	(0.010)	(0.010)
as.factor(first_review_month_offset)54	-0.765^{***}	-0.764^{***}
	(0.010)	(0.010)
as.factor(first_review_month_offset)55	-0.761^{***}	-0.760^{***}

	(0.010)	(0.010)
as.factor(first_review_month_offset)56	−0.767***	−0.766***
	(0.010)	(0.010)
as.factor(first_review_month_offset)57	−0.768***	−0.767***
	(0.010)	(0.010)
as.factor(first_review_month_offset)58	−0.761***	−0.760***
	(0.010)	(0.010)
as.factor(first_review_month_offset)59	−0.764***	−0.764***
	(0.010)	(0.010)
as.factor(first_review_month_offset)60	−0.764***	−0.763***
	(0.010)	(0.010)
as.factor(first_review_month_offset)61	−0.769***	−0.768***
	(0.010)	(0.010)
as.factor(first_review_month_offset)62	−0.766***	−0.765***
	(0.010)	(0.010)
as.factor(first_review_month_offset)63	−0.765***	−0.764***
	(0.011)	(0.011)
as.factor(first_review_month_offset)64	−0.766***	−0.765***
	(0.011)	(0.011)
as.factor(first_review_month_offset)65	−0.764***	−0.764***
	(0.011)	(0.011)
as.factor(first_review_month_offset)66	−0.761***	−0.761***
	(0.011)	(0.011)
as.factor(first_review_month_offset)67	−0.763***	−0.762***
	(0.011)	(0.011)
as.factor(first_review_month_offset)68	−0.764***	−0.764***
	(0.011)	(0.011)
as.factor(first_review_month_offset)69	−0.768***	−0.767***
	(0.011)	(0.011)
as.factor(first_review_month_offset)70	−0.765***	−0.764***
	(0.011)	(0.011)
as.factor(first_review_month_offset)71	−0.767***	−0.766***
	(0.011)	(0.011)
as.factor(first_review_month_offset)72	−0.769***	−0.769***
	(0.011)	(0.011)
as.factor(first_review_month_offset)73	−0.771***	−0.770***
	(0.011)	(0.011)

as.factor(first_review_month_offset)74	-0.770^{***}	-0.769^{***}
	(0.011)	(0.011)
as.factor(first_review_month_offset)75	-0.769^{***}	-0.768^{***}
	(0.011)	(0.011)
as.factor(first_review_month_offset)76	-0.772^{***}	-0.771^{***}
	(0.011)	(0.011)
as.factor(first_review_month_offset)77	-0.771^{***}	-0.770^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)78	-0.767^{***}	-0.766^{***}
	(0.011)	(0.011)
as.factor(first_review_month_offset)79	-0.768^{***}	-0.768^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)80	-0.772^{***}	-0.771^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)81	-0.767^{***}	-0.766^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)82	-0.767^{***}	-0.766^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)83	-0.767^{***}	-0.766^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)84	-0.765^{***}	-0.764^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)85	-0.764^{***}	-0.764^{***}
	(0.011)	(0.012)
as.factor(first_review_month_offset)86	-0.767^{***}	-0.766^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)87	-0.766^{***}	-0.765^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)88	-0.768^{***}	-0.768^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)89	-0.770^{***}	-0.769^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)90	-0.764^{***}	-0.763^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)91	-0.770^{***}	-0.769^{***}
	(0.012)	(0.012)
as.factor(first_review_month_offset)92	-0.772^{***}	-0.771^{***}

	(0.012)	(0.012)
as.factor(first_review_month_offset)93	−0.767***	−0.766***
	(0.012)	(0.012)
as.factor(first_review_month_offset)94	−0.769***	−0.769***
	(0.012)	(0.012)
as.factor(first_review_month_offset)95	−0.768***	−0.768***
	(0.012)	(0.012)
as.factor(first_review_month_offset)96	−0.768***	−0.768***
	(0.012)	(0.012)
as.factor(first_review_month_offset)97	−0.770***	−0.769***
	(0.012)	(0.012)
as.factor(first_review_month_offset)98	−0.765***	−0.764***
	(0.012)	(0.012)
as.factor(first_review_month_offset)99	−0.768***	−0.767***
	(0.013)	(0.013)
as.factor(first_review_month_offset)100	−0.761***	−0.760***
	(0.013)	(0.013)
as.factor(first_review_month_offset)101	−0.769***	−0.768***
	(0.013)	(0.013)
as.factor(first_review_month_offset)102	−0.768***	−0.767***
	(0.013)	(0.013)
as.factor(first_review_month_offset)103	−0.772***	−0.771***
	(0.013)	(0.013)
as.factor(first_review_month_offset)104	−0.768***	−0.767***
	(0.013)	(0.013)
as.factor(first_review_month_offset)105	−0.770***	−0.769***
	(0.013)	(0.013)
as.factor(first_review_month_offset)106	−0.771***	−0.770***
	(0.013)	(0.013)
as.factor(first_review_month_offset)107	−0.764***	−0.764***
	(0.013)	(0.013)
as.factor(first_review_month_offset)108	−0.771***	−0.770***
	(0.013)	(0.013)
as.factor(first_review_month_offset)109	−0.770***	−0.769***
	(0.013)	(0.013)
as.factor(first_review_month_offset)110	−0.768***	−0.767***
	(0.013)	(0.013)

as.factor(first_review_month_offset)111	-0.770^{***}	-0.769^{***}
	(0.013)	(0.013)
as.factor(first_review_month_offset)112	-0.769^{***}	-0.768^{***}
	(0.013)	(0.013)
as.factor(first_review_month_offset)113	-0.772^{***}	-0.771^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)114	-0.768^{***}	-0.767^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)115	-0.772^{***}	-0.771^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)116	-0.772^{***}	-0.771^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)117	-0.770^{***}	-0.769^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)118	-0.766^{***}	-0.765^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)119	-0.773^{***}	-0.772^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)120	-0.763^{***}	-0.762^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)121	-0.775^{***}	-0.774^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)122	-0.767^{***}	-0.767^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)123	-0.775^{***}	-0.774^{***}
	(0.014)	(0.014)
as.factor(first_review_month_offset)124	-0.767^{***}	-0.766^{***}
	(0.015)	(0.015)
as.factor(first_review_month_offset)125	-0.769^{***}	-0.768^{***}
	(0.015)	(0.015)
as.factor(first_review_month_offset)126	-0.772^{***}	-0.771^{***}
	(0.015)	(0.015)
as.factor(first_review_month_offset)127	-0.777^{***}	-0.776^{***}
	(0.015)	(0.015)
as.factor(first_review_month_offset)128	-0.779^{***}	-0.778^{***}
	(0.015)	(0.015)
as.factor(first_review_month_offset)129	-0.772^{***}	-0.771^{***}

	(0.015)	(0.015)
as.factor(first_review_month_offset)130	−0.771***	−0.770***
	(0.016)	(0.016)
as.factor(first_review_month_offset)131	−0.781***	−0.780***
	(0.015)	(0.016)
as.factor(first_review_month_offset)132	−0.783***	−0.782***
	(0.015)	(0.015)
as.factor(first_review_month_offset)133	−0.786***	−0.785***
	(0.015)	(0.015)
as.factor(first_review_month_offset)134	−0.781***	−0.780***
	(0.016)	(0.016)
as.factor(first_review_month_offset)135	−0.782***	−0.781***
	(0.016)	(0.016)
as.factor(first_review_month_offset)136	−0.788***	−0.787***
	(0.016)	(0.016)
as.factor(first_review_month_offset)137	−0.781***	−0.780***
	(0.016)	(0.016)
as.factor(first_review_month_offset)138	−0.792***	−0.791***
	(0.016)	(0.016)
as.factor(first_review_month_offset)139	−0.784***	−0.783***
	(0.017)	(0.017)
as.factor(first_review_month_offset)140	−0.778***	−0.777***
	(0.017)	(0.017)
as.factor(first_review_month_offset)141	−0.789***	−0.788***
	(0.017)	(0.017)
as.factor(first_review_month_offset)142	−0.796***	−0.795***
	(0.017)	(0.017)
as.factor(first_review_month_offset)143	−0.796***	−0.795***
	(0.017)	(0.017)
as.factor(first_review_month_offset)144	−0.795***	−0.794***
	(0.017)	(0.017)
as.factor(first_review_month_offset)145	−0.793***	−0.792***
	(0.017)	(0.017)
as.factor(first_review_month_offset)146	−0.796***	−0.795***
	(0.017)	(0.017)
as.factor(first_review_month_offset)147	−0.795***	−0.793***
	(0.018)	(0.018)

as.factor(first_review_month_offset)148	-0.792^{***}	-0.790^{***}
	(0.017)	(0.017)
as.factor(first_review_month_offset)149	-0.795^{***}	-0.794^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)150	-0.797^{***}	-0.796^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)151	-0.794^{***}	-0.793^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)152	-0.789^{***}	-0.788^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)153	-0.780^{***}	-0.779^{***}
	(0.019)	(0.019)
as.factor(first_review_month_offset)154	-0.788^{***}	-0.787^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)155	-0.804^{***}	-0.802^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)156	-0.803^{***}	-0.802^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)157	-0.806^{***}	-0.804^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)158	-0.804^{***}	-0.803^{***}
	(0.019)	(0.019)
as.factor(first_review_month_offset)159	-0.803^{***}	-0.802^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)160	-0.801^{***}	-0.800^{***}
	(0.018)	(0.018)
as.factor(first_review_month_offset)161	-0.805^{***}	-0.803^{***}
	(0.019)	(0.019)
as.factor(first_review_month_offset)162	-0.802^{***}	-0.801^{***}
	(0.019)	(0.019)
as.factor(first_review_month_offset)163	-0.811^{***}	-0.810^{***}
	(0.019)	(0.019)
as.factor(first_review_month_offset)164	-0.816^{***}	-0.815^{***}
	(0.019)	(0.020)
as.factor(first_review_month_offset)165	-0.810^{***}	-0.808^{***}
	(0.021)	(0.021)
as.factor(first_review_month_offset)166	-0.823^{***}	-0.822^{***}

	(0.020)	(0.020)
as.factor(first_review_month_offset)167	−0.826***	−0.825***
	(0.021)	(0.021)
as.factor(first_review_month_offset)168	−0.825***	−0.823***
	(0.020)	(0.021)
as.factor(first_review_month_offset)169	−0.820***	−0.818***
	(0.021)	(0.021)
as.factor(first_review_month_offset)170	−0.827***	−0.826***
	(0.021)	(0.021)
as.factor(first_review_month_offset)171	−0.835***	−0.833***
	(0.023)	(0.023)
as.factor(first_review_month_offset)172	−0.830***	−0.829***
	(0.024)	(0.024)
as.factor(first_review_month_offset)173	−0.832***	−0.831***
	(0.023)	(0.023)
as.factor(first_review_month_offset)174	−0.834***	−0.832***
	(0.024)	(0.024)
as.factor(first_review_month_offset)175	−0.829***	−0.827***
	(0.024)	(0.024)
as.factor(first_review_month_offset)176	−0.825***	−0.823***
	(0.026)	(0.026)
as.factor(first_review_month_offset)177	−0.831***	−0.830***
	(0.027)	(0.027)
as.factor(first_review_month_offset)178	−0.824***	−0.822***
	(0.027)	(0.027)
as.factor(first_review_month_offset)179	−0.807***	−0.805***
	(0.029)	(0.029)
as.factor(first_review_month_offset)180	−0.826***	−0.825***
	(0.031)	(0.031)
as.factor(first_review_month_offset)181	−0.827***	−0.825***
	(0.031)	(0.031)
as.factor(first_review_month_offset)182	−0.825***	−0.824***
	(0.031)	(0.032)
as.factor(first_review_month_offset)183	−0.825***	−0.824***
	(0.035)	(0.035)
as.factor(first_review_month_offset)184	−0.816***	−0.815***
	(0.037)	(0.038)

as.factor(first_review_month_offset)185	-0.801^{***}	-0.799^{***}
	(0.042)	(0.042)
as.factor(first_review_month_offset)186	-0.803^{***}	-0.801^{***}
	(0.046)	(0.046)
as.factor(first_review_month_offset)187	-0.785^{***}	-0.784^{***}
	(0.054)	(0.054)
as.factor(first_review_month_offset)188	-0.856^{***}	-0.854^{***}
	(0.061)	(0.062)
as.factor(first_review_month_offset)189	-0.825^{***}	-0.824^{***}
	(0.074)	(0.074)
as.factor(first_review_month_offset)190	-0.860^{***}	-0.858^{***}
	(0.102)	(0.102)
as.factor(first_review_month_offset)191	-1.129^{***}	-1.127^{***}
	(0.203)	(0.203)
X2018_or_laterTrue	0.041^{***}	0.040^{***}
	(0.002)	(0.002)
as.factor(first_review_month)2		-0.024
		(0.096)
as.factor(first_review_month)3		0.111
		(0.099)
as.factor(first_review_month)4		0.070
		(0.103)
as.factor(first_review_month)5		-0.028
		(0.098)
as.factor(first_review_month)6		0.033
		(0.096)
as.factor(first_review_month)7		0.144
		(0.096)
as.factor(first_review_month)8		-0.064
		(0.094)
as.factor(first_review_month)9		0.316^{***}
		(0.083)
as.factor(first_review_month)10		0.195^{**}
		(0.092)
as.factor(first_review_month)11		0.146
		(0.099)
as.factor(first_review_month)12		0.106

		(0.108)
as.factor(month)2	−0.003	−0.003
	(0.004)	(0.004)
as.factor(month)3	−0.003	−0.003
	(0.003)	(0.004)
as.factor(month)4	−0.004	−0.004
	(0.003)	(0.003)
as.factor(month)5	−0.001	−0.001
	(0.003)	(0.003)
as.factor(month)6	0.008**	0.007**
	(0.003)	(0.003)
as.factor(month)7	0.010***	0.010***
	(0.003)	(0.003)
as.factor(month)8	0.011***	0.011***
	(0.003)	(0.003)
as.factor(month)9	0.015***	0.015***
	(0.003)	(0.003)
as.factor(month)10	0.013***	0.013***
	(0.003)	(0.003)
as.factor(month)11	0.013***	0.013***
	(0.003)	(0.003)
as.factor(month)12	0.004	0.004
	(0.003)	(0.003)
Constant		6.911***
		(0.067)
Observations	176,516	176,516
R^2	0.168	0.204
Adjusted R^2	0.150	0.203
Note:	*p<0.1; **p<0.05; ***p<0.01	

11 Code

Dataset cleaning and creation largely performed in Python with Jupyter Notebooks. This code is attached in the Jupyter Notebook file CleanerNotebook.ipynb.

Estimation and testing R code is presented here:

```
if (!require("plm")) install.packages("plm")
library(plm) # function: plm
library(stargazer)
library(arm)
source("data directory path.R")
dat <- read.csv(file="211215clean_movie_panel.csv", header=TRUE, sep=",", na = ".")
attach(dat)
FE <- plm(cumavg_review ~ as.factor(first_review_month_offset) + X2018_or_later +
as.factor(month) + as.factor(first_review_month), data = dat,
index = c("movie","first_review_month_offset"), model = "within")
summary(FE)
RE <- plm(cumavg_review ~ as.factor(first_review_month_offset) + X2018_or_later +
as.factor(first_review_month) + as.factor(month), data = dat,
index = c("movie","first_review_month_offset"), model = "random")
summary(RE)
phtest(FE,RE)
stargazer(FE, RE, title="Panel Data", column.labels=c("FE","RE"), no.space = TRUE,
omit.stat=c("f", "ser"))
library(ggplot2)
y <- RE$coefficients[2:25]
sd <- summary(RE)$coefficients[2:25,2]
q <- qplot(1:24, RE$coefficients[2:25])+geom_errorbar(aes(x=1:24, ymin=y-sd, ymax=y+sd),
width=0.25)
q + xlab("r") + ylab(expression(mu_r))
ggsave("project_coefplot.png")
```