GENERATING HIERARCHY IN HOLLYWOOD ACCORDING TO CONSUMER PREFERENCES

Marlin Figgins
University of Washington
mfiggins@uw.edu

Xingzi Xu
Stanford University
xingzix@stanford.edu

Shashank Dalmia
University of California Berkeley
shashankdalmia24@berkeley.edu

Aditya Nair
University of California Berkeley
adityanair@berkeley.edu

October 4, 2020

EXECUTIVE SUMMARY

Hollywood and the larger film industry is highly profitable and commercial films have been embedded in American culture, but Hollywood still to some extent maintains it own unique culture. Though Americans spend millions and millions of dollars annually on this industry, it can be difficult to ascertain the extent to which film viewers influence the future and culture of the industry itself. To begin approaching this, we ask, "To what extent do consumer preferences and evaluations affect the social network of director-actor collaborations in Hollywood?".

In order to take advantage of the relationships between both actors and directors, we develop a social network describing the collaborations between various actors and directors. We identify various factors statistically associated with the probability of directors and actors collaborating with one another, then using a network dynamic model to infer which factors determine hierarchy between actors and directors. This allows us to determine a ranking of individuals in Hollywood, which we determine to be dominated primarily by actors. Further, analysis shows that Academy Awards carry significant weight in determining one's future collaborators, but additionally that one's future collaborators are likely to be similarly rated to oneself according to community ratings metrics on online communities like IMDb. This suggests that though prestigious awards like the Oscars awards can carry weight in determining one's career trajectory, it can prove to be difficult to climb the social ladder in Hollywood without being well-received by consumers.

Technical Summary

Data exploration.

In order to explore the relationship between consumer preferences and directors/actors and to potentially predict the prestige of a director/actor based on these preferences, we use the movie_industry and the_oscar_award data sets. The first data set contains relevant information of the most popular movies (approximately 220) in each year from 1986 to 2016 such as each film's director and its star, which we use to extract information about collaborations between directors and actors. We also use columns including gross, genre, rating, score, and votes to learn more information of each director and star. The latter data set contains Oscar Awards nominations and wins beginning in 1927, which we use as a proxy for measuring prestige and/or success. Since the data set is the full list of Oscar Awards nominations across all categories, we narrow our focus to the subset of the data which features actor, actress, and directing awards.

From exploratory data analysis on movie_industry, we note that category features such as genre and rating have levels with very small samples, which needs further bucketing for one-hot encoding. It is also interesting to see positive correlations between score and number of votes as well as score and gross of the movie. The variance of score is big for movies with small number of votes, yet once a movie has more than $\sim$750,000 votes, its score is very high. With time series analysis, we find that budget, gross, score, and votes all increase over time except a big drop for number of votes in 2015. The increasing gross might be a result of inflation. However, since we are interested in the interactions between directors and actors on a yearly basis, inflation would not present a big impact in our context.

The data cleaning process involves cleaning of the strings in the_oscar_award, especially in the category and name variables. We cross check the names in the_oscar_award and movie_industry and make sure that the same individual has exactly the same name strings in both data sets. We also re-encode rating in movie_industry to unify "UNRATED", "NOT RATED", and "Not specified". After initial data cleaning and filtering out nominations of movies before 1986, we have 852 nomination entries with 171 wins. Of directors and stars represented in movie_industry, 12.5% have been nominated for an Oscar.

We also note some limitations of our data sets. First, we only have around 220 movie records per year in the movie_industry. There are 82 nominees which are represented in the_oscar_award which are not represented in movie_industry. Second, movie_industry data set only features one star for each movie, either the leading actor or the leading actress, and therefore, we are limited in our ability to capture the relationship between directors and supporting actors.

Determining covariates.

In order to determine how a network of director-actor collaborations evolves, we attempt to discern what features in our data likely influence the probability of future collaborations. We conduct a series of hypothesis tests with the continuous features and categorical features in movie_industry as well as counts of Oscar nominations and wins in order to check whether they have significantly different distributions for director-actor pairs which have or have not collaborated.

There are 6,908,536 possible combinations of director-actor combinations in total, out of which 6,306 director-actor pairs have collaboration histories. Accordingly, the data is divided into two parts: pairs that have worked together and pairs that have not. Our objective is to determine features that help distinguish director-actor pairs that have worked together versus those who haven't for assessing the likelihood of a collaboration in the future.

We have three expectations and calculate the statistics correspondingly as followed:

1. A pair is more likely to work together if they have similar number of votes/score/gross.
   - Absolute difference of the log of total gross for a director and the log of total gross for an actor.
   - Absolute difference of the log of mean votes for a director and the log of mean votes for an actor.
   - Absolute difference of the log of mean score for a director and the log of mean score for an actor.
2. A pair is more likely to work together if they have similar number of Oscar nominations and wins.
   - Absolute difference of total nominations for a director and total nominations for an actor.
   - Absolute difference of total wins for a director and total wins for an actor.
3. A pair is more likely to work together if they have both worked in movies of similar genres and ratings.

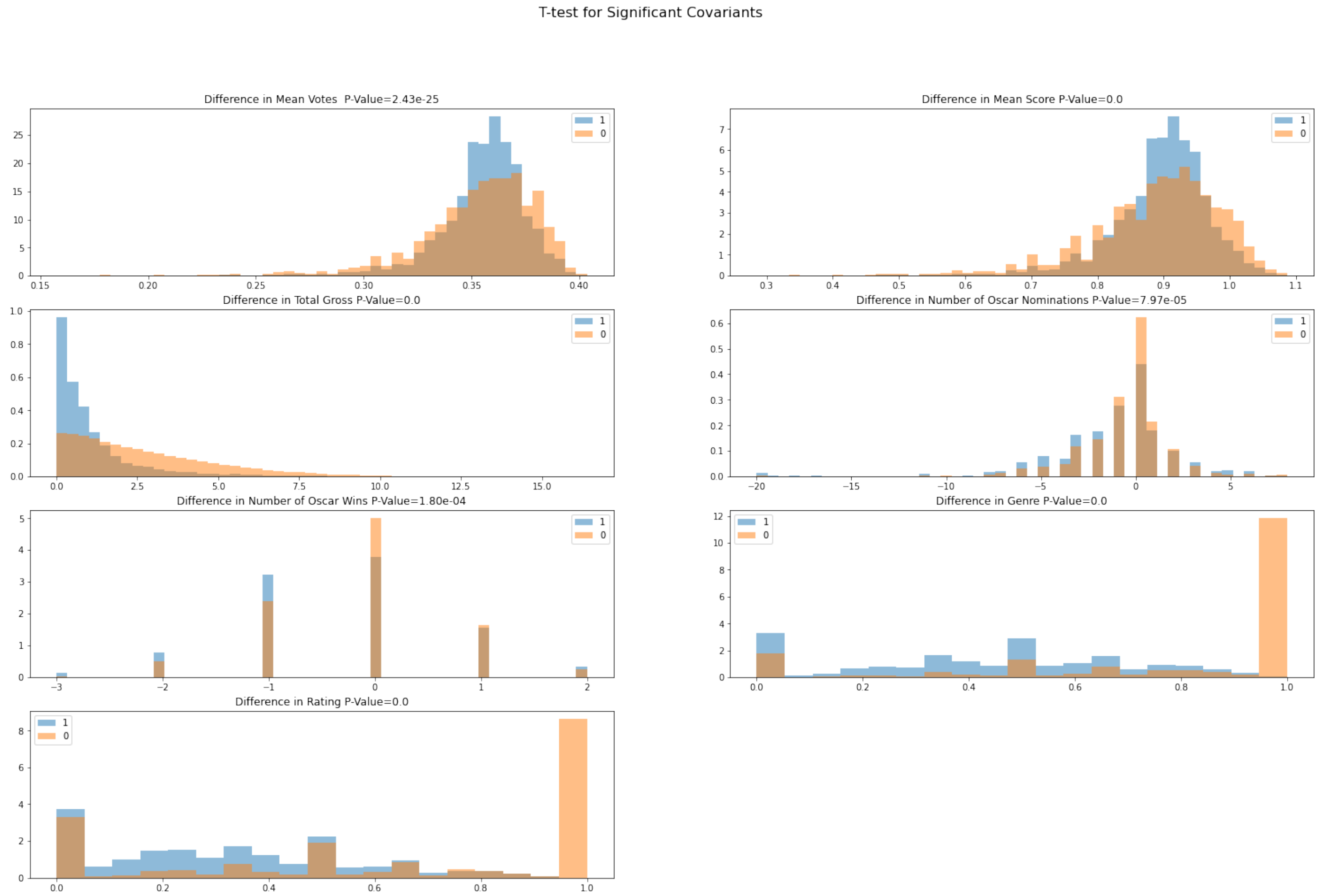

Figure 1: Difference in Distributions for Significant Covariates. Blue corresponds to individuals who have worked together. Orange corresponds to those who have not.

- Total Variation Distance between proportion of movies in each genre. (We only calculated values for the top 8 genres which made up 98% of the data set.)
- Total Variation Distance between proportion of movies in each rating category. (We only calculated values for the top 5 rating categories which made up 99% of the data set.)

Therefore, for each director-actor pair, we calculate the difference between the pair, visualize and compare the distribution of the difference between the collaboration group and non-collaboration group, and calculate the p-values under T-Test and Kolmogorov-Smirnov Test. The null hypothesis is that there is no significant difference between the two groups. Figure 1 shows features that we identified that are statistically significant in inferring whether a director-actor are likely to work together. We could see that from the result, the distributions of collaboration group are very different from those of the non-collaboration group and the p-values are all very small, pointing towards the alternative. Through this, we can identify significant covariates for further modeling.

Generating covariates.

There are 2759 unique directors and 2504 unique actors across the 31 years of the movie industry data set. There are 171 people in the data set who have worked as both an actor as director and thus feature in both actor and director categories . For our analysis we consider them separately. Therefore, the covariates for "Clint Eastwood" as an actor and director are calculated separately. We will refer to these unique actors and directors as agents from here on. We calculate the covariates at every time step. Thus the granularity of our data set is every agent at a particular time step.

p-values of significant covariates under T-test and Kolmogorov-Smirnov Test		
	T-Test	Kolmogorov-Smirnov Test
Mean Votes	6.09e-17	9.08e-56
Mean Score	3.19e-13	7.48e-56
Total Gross	0.0	0.0
Nominations	2.05e-09	9.51e-10
Wins	8.33e-09	2.11e-07
Genre	0.0	0.0
Rating	0.0	0.0

Table 1: p-values of significant covariates under T-test and Kolmogorov-Smirnov Test

There are a total of 163,153 observations (31 time steps × 5263 agents). For each observation, we calculate 18 features which we believe to be covariates. They are as follows:

- `log_total_gross` : calculated as $\ln$(total gross+1) of all movies of the particular agent until that time step.
- `mean_score` : mean votes calculated as the mean of scores of all movies of the particular agent until that time step; 0 if no movie done until that time step.
- `mean_votes` : mean votes calculated as the mean of votes received of all movies of the particular agent until that time step; 0 if no movie done until that time step.
- `Comedy`, `Drama`, `Action`, `Crime`, `Adventure`, `Biography`, `Horror`, and `Animation` : Number of movies under each of those genre categories for each agent until that time step.Note that there are more genre classifications in the movie industry data set. However, these genres make up 98% of the data set. To reduce complexity, we do not create covariates for the remaining categories.
- `R`, `PG-13`, `PG`, `NOT RATED`, and `G` : Number of movies under each of those rating categories for each agent until that time step. 'Not specified', 'UNRATED' and 'NOT RATED' rating categories in the movie industry data set have been reclassified as 'NOT RATED'. Note that there are more rating classifications in `movie_industry`. However, the aforementioned categories make up approximately 99.6% of the data set. To reduce complexity, we do not include remaining categories as covariates.
- `nominations` and `win_counts` : Number of Oscar nominations and wins of the particular agent until that time step.

Director-actor collaboration networks.

Choosing to use a graph. For our director-actor collaboration networks, we find a model that integrates well with the rest of our pipeline and will be easy to transfer after creation and storage so our team can interact with the model on their own time. Logically, we decide on using a graph structure, which will be bipartite by nature of the restrictions we placed on the definition of collaboration, that can be easily transferred using some representation of an adjacency matrix.

Creating the model. Mechanically, we iterate through `movie_industry` for every year *t*, isolate all director-actor relationships from that year, and begin constructing a network by connecting the director and actor present in each relationship. For each actor and director pair present in the `movie_industry` data set, we need to represent a way to represent their collaboration or lack thereof. Mathematically, we present the complete network as a set $\{\Delta(t)\}$ indexed by the year. Each element $\Delta(t)$ is a square matrix with elements

$$\Delta_{ij}(t) = \begin{cases} 1 & \text{if agents } i \text{ and } j \text{ have collaborated in year } t, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Loading the graph into a visualization also allows us to see some interesting relationships between groups of agent pairs. For example, in Figure 2, we see that actors that work on multiple movies in a year work with smaller directors. In Figure 3, we can see DiCaprio's evolution over time (taking the log of total gross stops his node from blotting out the entire graph, but makes it quite difficult to see how successful his colleagues are)

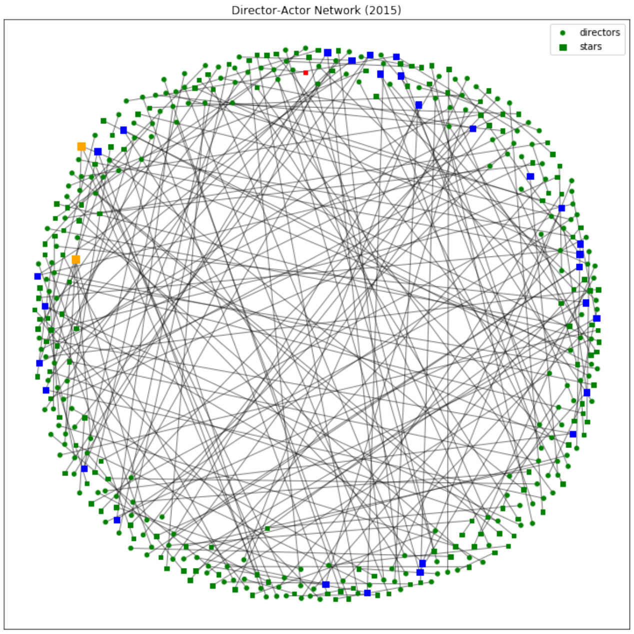

Figure 2: A densely populated graph with node colors delineating how active the corresponding agent is. Green, blue and yellow represent the agent working on 1, 2, and 3 movies respectively. Leonardo DiCaprio is highlighted in red for your viewing pleasure.

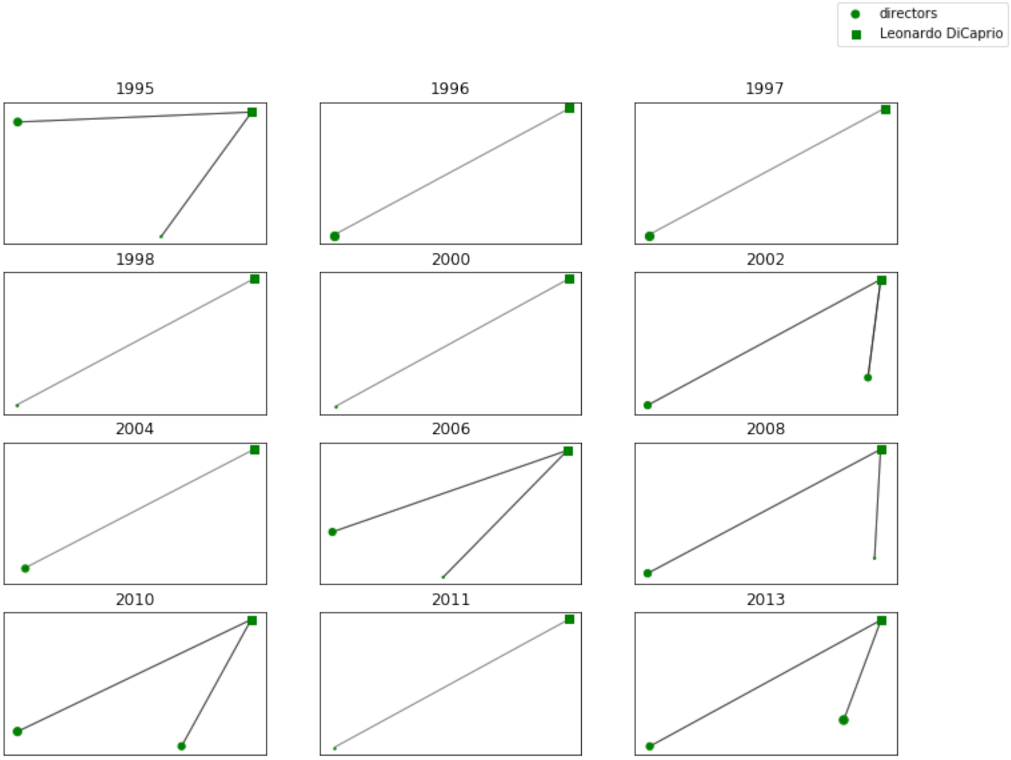

Figure 3: A longitudinal analysis of everyone's favorite actor, Leonardo DiCaprio, with node size scaled by the log of the total gross. Note that though he becomes increasingly successful, he often works with directors who make nowhere near as much as he does

Computational Burden. With over 5000 unique agents, constructing and storing these as networks as adjacency matrices yielded extremely large files . To assuage this, we stored the matrices in a sparse matrix format, which brought the full storage cost down by 99.7%. We realized this would be possible because our graph is bipartite in nature (directors are connected only to actors and vice versa), so two quadrants of the adjacency matrix are entirely filled with zeros.

Modeling hierarchy in Hollywood collaborations.

In order to understand how consumer preferences contribute to the formation of hierarchy. We rely on a network dynamics model. This formulation has been used to recreate hierarchies in higher education such as those governing prestige in PhD-granting institutions in mathematics. [1].

Collaboration dynamics. We take in various covariates which affect individual actors' and directors' probability of working with another in a particular year as a matrix $\mathbf{X}$ with each column $\mathbf{x}_1, \ldots, \mathbf{x}_k$ corresponding to different predictors. The matrix X contains director-actor specific information so that x_{im} is the value of the m-th covariate corresponding to the i-th agent. Then we determine the utility of a collaboration between agents i and j as

$$u_{ij} = \sum_{m=1}^{k} \beta_m \left| x_{im} - x_{jm} \right| . \tag{2}$$

In this way, estimating the coefficient vector $\beta = (\beta_1, \beta_2, \ldots, \beta_k)$ acts as a way to compute relative weight of how differences in each factor contribute to predicting collaborations between individuals.

We then compute the probability of a collaboration occurring between individuals i and j in that year as

$$p_{ij} = \frac{e^{u_{ij}}}{\sum_{k=1}^{n} e^{u_{ik}}}. \tag{3}$$

Computing Hierarchy. Using the probabilities described above, we are able to compute the rankings of various individuals in terms of how likely it is for others to work with them. Therefore, we compute the rank of individual i as

$$\gamma_i = \sum_{j=1}^{n} p_{ij}. \tag{4}$$

Inference. Given our generated director-actor collaboration network $\{\Delta(t)\}$, we can compute the maximal likelihood estimates of β associated with our model. The likelihood for the parameters β is given implicitly through

$$\mathcal{L}(\beta \mid \{\Delta(t)\}) = \sum_{i,j,t} \Delta_{ij}(t) \log p_{ij}(t) + C \tag{5}$$

where C is a constant which does not depend on β.

We use this framework to generate hierarchies within our director-actor network. Our goal is to infer the maximum likelihood coefficient vector $\hat{\beta}$. We take advantage of the fact that this problem is convex in β and solve for β using convex optimization schemes implemented in SciPy [2]. This alongside our computation of rank allows us to infer which factors affect the probability of an actor becoming well-connected.

Implementation. Despite having access to covariates available for 31 years, we restrict this hierarchy model to the most recent 10 years of data (2007-2016) for several reasons:

1. **Individual history.** In order to properly account for the fact that some directors and actors begin their careers before the time at which our data begins, we restrict our inference of the hierarchical dynamics to the 10 most recent years, so that we can use the entire history of the agents active in that time period in our analysis.
2. **Computational burden.** By restricting the time frame in our analysis, we reduce the computation necessary for a single evaluation of the likelihood by reducing the number of agents active in this time period, which is expensive to evaluate as the number of operations necessary scales $O(n^2)$ with n agents. This issue is partially remedied by representing $\Delta(t)$ as a sparse matrix.

	Small Model	Full Model
log_total_gross	-7.41e-03	-9.09e-03
mean_score	3.66e-03	9.50e-03
mean_votes	-1.12e-07	-1.71e-07
Comedy		-3.78e-03
Drama		4.93e-02
Action		-8.45e-03
Crime		-2.41e-02
Adventure		-3.32e-02
Biography		2.39e-02
Horror		-2.01e-02
Animation		-1.18e-03
R		3.05e-03
PG-13		-1.07e-03
PG		-3.45e-02
NOT RATED		-1.61e-02
G		3.97e-03
nominations		5.99e-02
win_counts		1.07e-02

Table 2: Inferred weights $\hat{\beta}$ corresponding to variables above. Larger, positive parameters indicate preference for collaborations between agents of similar quantities e.g. we see a strong preference for Academy award nominated individuals to collaborate with one another as well as a preference for individuals with similarly consumer-rated filmographys to collaborate.

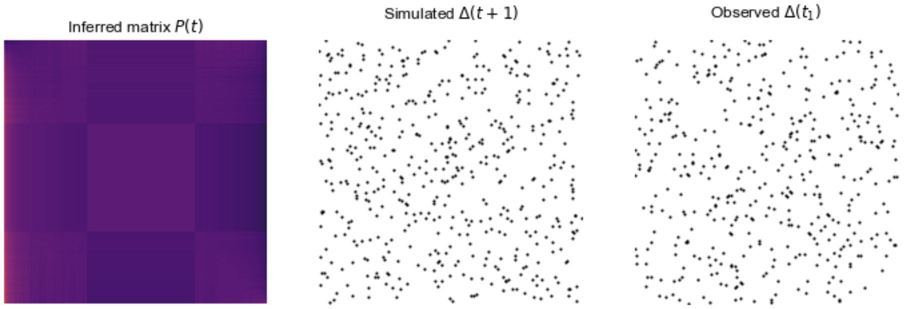

Figure 4: Visualizing the various matrices generated by our model. Using the probability matrix associated with the inferred $\hat{\beta}$, we can simulate hypothetical collaborations among agents in our network which resemble our observed collaboration network.

Findings. Using the inference scheme described previous, we inferred model parameters describing the relative weight of certain parameters in determining individual connections and thereby hierarchy. These values are presented in table 2. In both models, we can see that the mean score of an individual's past films as evaluated by consumers plays a significant role in determining hierarchy. In other words, individuals are more likely to collaborate with individuals of similar filmography quality as evaluated by consumers. We also see that the parameters corresponding the Academy Award nominations and wins are positive and relatively large indicating a preference for individuals who have been nominated to work with one another.

This inference also allows us to visualize the rankings of individuals. We present that 5 most highly ranked and 5 lowest ranked agents in 5.

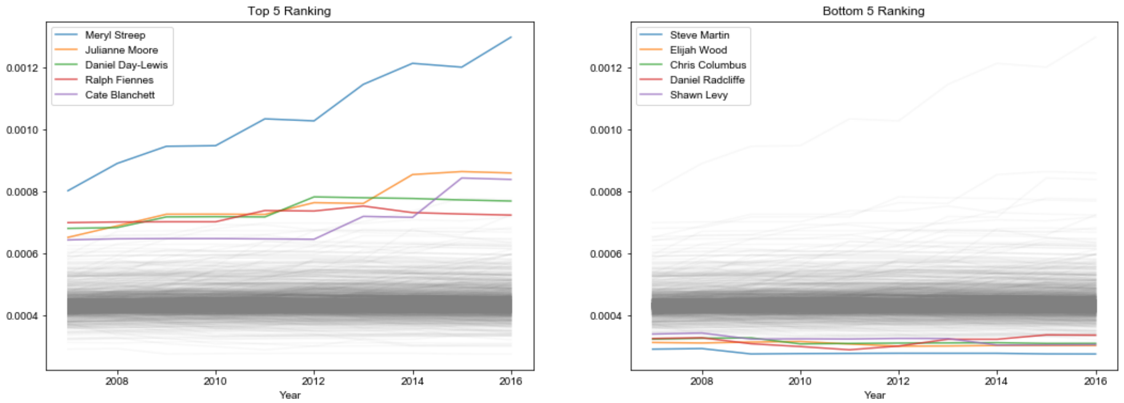

Figure 5: We visualize the top and bottom ranked agents according to our model. 'Top' and 'Bottom' were computed according actors sum total score across the years 2007 to 2016.

Looking at the rankings here, there appears to be a multitude ways of climbing the hierarchy. One such way is appearing in movies with large commercial success. Other ways is appearing with alongside highly-ranked agents. Alternatively, if you're Meryl Streep, you do it all! We also see a pattern among the top 5 ranked agents all have been nominated for Academy Awards previously and 4 are winners.

Possible extensions of this method. The method described above is extremely extensible. Though, we relied on covariates generated from `movie_industry` and `the_oscar_award` for our estimates in 2. This methodology can also be used to infer the importance of network metrics such as node degree, PageRank centrality as well as variables found in other data sets.

Another possible dimension for extension is in the role of various regularization techniques. For example, as regularization using the L^1 norm is known to promote sparsity, this method could be similarly expanded for sparse regression on the various covariates to identify features which contribute the most to observed collaboration and hierarchy within the larger network.

References

[1] M. Kawakatsu, P. S. Chodrow, N. Eikmeier, and D. B. Larremore, “Emergence of hierarchy in networked endorsement dynamics,” 2020.

[2] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python,” *Nature Methods*, vol. 17, pp. 261–272, 2020.