A Network Analysis of Movie Popularity

Timothée Borget Dit Vorgeat, Yassine Zouaghi, Icíar Lloréns Jover and Pol Boudou Pérez

A Network Tour of Data Science, January 2019

1 Introduction

What can we say about the popularity of a movie based on its characteristics? Can certain actors, directors or story lines attract automatically the attention of the public? Popularity is defined as the state of being liked, admired or supported by a high number of people. When talking about a movie, it can be measured in several ways, namely how many people went to see the movie on the theaters, the critics' opinion, the movie fans' reviews or the Cinema Authorities' (such as Cannes or the Academy) judgment.

The aim of this project is to analyze several movie features by creating a film network in order to observe which characteristics lead to popularity. First, the movie dataset is used to conduct different regressions using all of the movies' features and different popularity targets. The goal of this regression is to obtain the most significant features using Machine Learning. Then, the relevant features are used to build a network, whose structure is studied to detect the popular movies' characteristics.

In order to reflect as close as possible the presented definition of popularity, four main metrics are used: the grades assigned by movie critics (Metacritic and Rotten Tomatoes), the users' opinions (IMDb grades), the film's box office and its awards (wins and nominations).

2 Data acquisition and cleaning

The OMDb API is used to acquire the data. It is a RESTful web service to obtain movie information [1]. This database allows access to the movie's actors, directors, writers and production company, the movie's genre, its ratings (on IMDb, Metacritic and Rotten Tomatoes), its box office, its awards (wins and nominations), its runtime, languages, country, release date, rated category and plot. This data is coupled with Kaggle's TMDB 5000 Movie Dataset in order to add the movie's budget [2].

The majority of the data is encoded in JSON structure. The cleaning process consists first in removing information not directly related to the movies themselves. Among the removed columns are URLs, columns related to the API, all-NaN columns and redundant columns. Since the project's goal is to explain any movie's popularity, all columns related to a date (release date of DVD, movie release date and year of release) are dropped. The remaining columns are formatted so as to make them exploitable. This formatting process involves converting all numerical features to int or float and to homogenize the actors, writers, directors and producers' names. After removing all the NaN columns, we realized that the runtime attribute only contained feature films (60 minutes or more). It was hence removed as well.

After the cleaning process, the remaining features are: actors, directors, writers, production companies, language, country, runtime, genre, rated category, plot and budget. The selected labels are the wins, nominations, box office and grades from Metacritic, Rotten Tomatoes and IMDb.

3 Exploration

3.1 Feature selection by regression

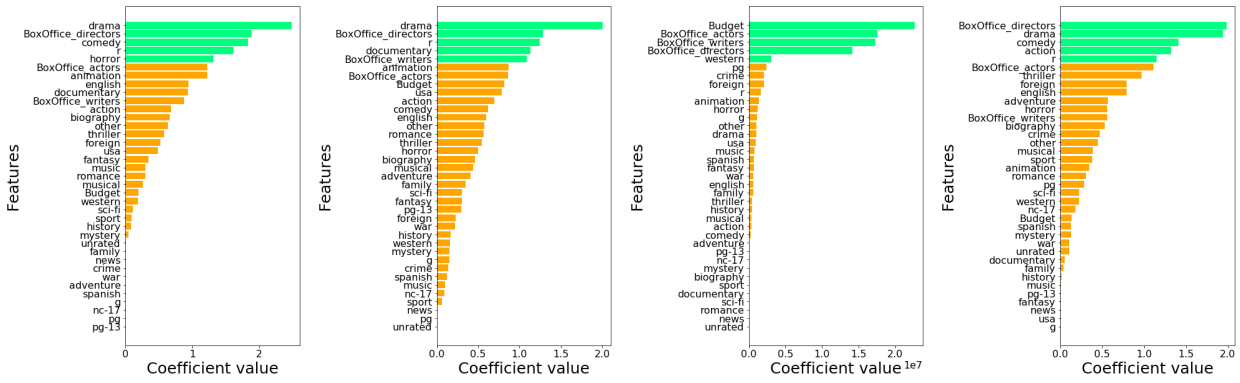
Our aim is to reduce the number of input features for the graph creation. PCA is not a good option as its interpretability about the input space is limited. Another option is to use a suitable machine learning algorithm coupled with Lasso regression that associates a coefficient to each feature proportional to its importance. The analysis is conducted by tuning the regularization term α and then keeping only the features with the highest coefficients. The tuning is performed using cross-validation across large ranges of α values and keeping the one yielding the lowest RMSE values.

Our metrics for popularity are the ones mentioned in the Section 2. All these targets have continuous numerical values for which no reliable way was found to divide them in classes. To learn which features are important to predict the labels, relying on regression is imperative.

The budget can be used already as is but most of the other features need transforming for the regression. The country, the language, the rating and the genre are one-hot-encoded before being used. Such a solution cannot be applied on the actors, directors, writers and production companies. Indeed, there are so many of them that the number of features would exceed the number of samples. Moreover, because of the curse of dimensionality [3] there would be no way to obtain any meaningful prediction and therefore feature selection.

For this reason, in order to quantify the crew's success, their generated box office is considered. Ideally, every member would be associated with the sum of all box office generated in their previous movies. However, the dataset size is limited and there is no guarantee of having all the crew members' previous work if any. Therefore, any temporal dependency is dropped and all the box office generated in the dataset is considered.

Figure 1 shows the features that most influence the label prediction. The features are selected by coefficient value. For each label, the highest 5 are retained. The total selected features are: actors, directors, writers, budget, rating and genres.



(a) IMDb grades

(b) Combination of Rotten tomatoes and Metacritic grades as targets

(c) BoxOffice generated

(d) Combination of nominations and wins as targets

Figure 1: The coefficient value associated with each feature after Lasso regularization using different targets. *In order to combine two targets, their coefficients were normalized independently and then added together.*

3.2 Graph creation

The regression is useful to select which features have an impact on the selected labels. However, some of those features were engineered to fit the regression model. This is true especially for the actors, directors and writers, where the list of their names is replaced by their generated box office. To eliminate the bias on the previously made assumptions, the list of names is restored and used for the network construction.

Several adjacency matrices were computed and added to compute the graph's adjacency matrix. Below is how each of the adjacency matrices are created. All matrices are normalized to have a maximum of 1. Next, a sparsified matrix is computed where only the 250 strongest edges for each node are kept. This allows for faster computation even though it leads to data loss.

- **Actors / directors / writers**: edge weights represent how many common actors / directors / writers two films have.

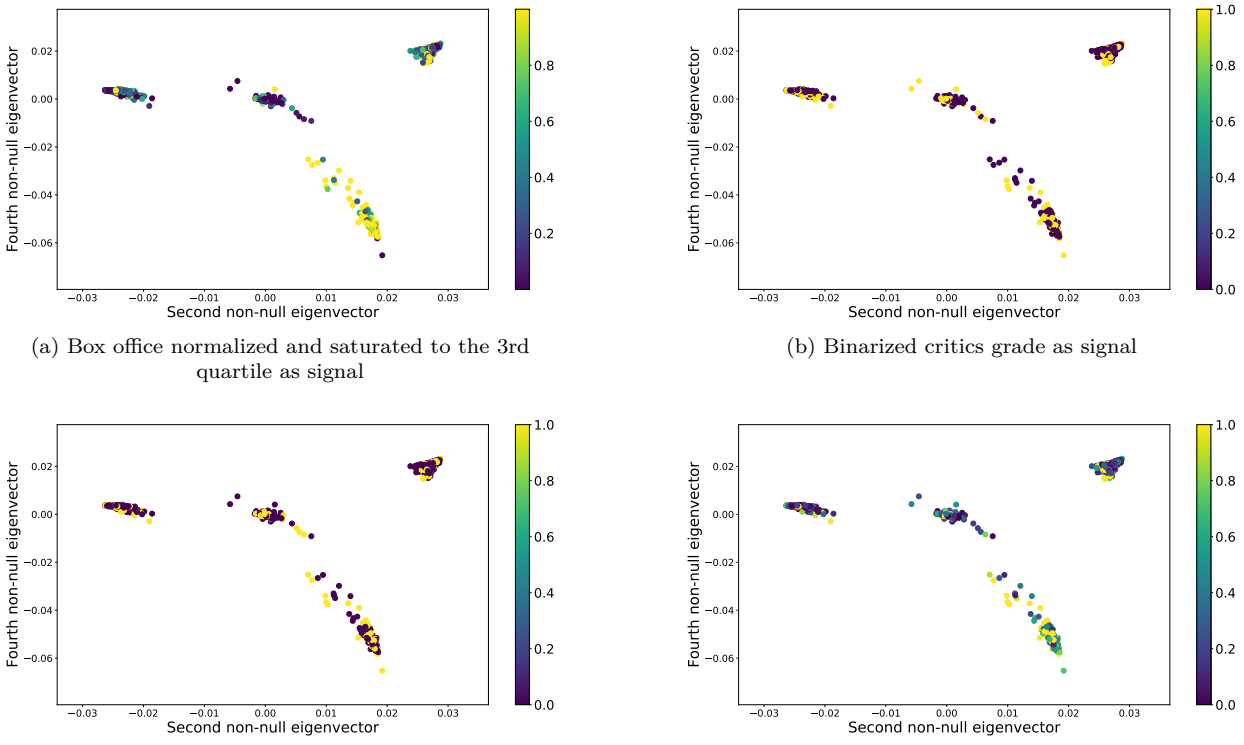
- **Budget**: the budget is categorized in bins. Each of them is coded with an integer representing the following intervals: no budget (less than 100 000 $), independent (100 000 to 10 million $), low budget (10

to 40 million $), medium budget (41 to 100 million $) and high budget (101 million $ and up). Two films are linked if and only if they belong to the same budget category.

- **Genre**: if two movies share more than one 'genre' label, the resulting weight is 1. Similarly, if a single 'genre' label is shared, the weight is 0.5.

4 Exploitation

4.1 Laplacian embedding



(a) Box office normalized and saturated to the 3rd quartile as signal

(b) Binarized critics grade as signal

(c) Binarized IMDb grade as signal

(d) Awards normalized and saturated to the 3rd quartile as signal

Figure 2: Laplacian embedding using different labels as signals

The Laplacian Eigenmaps [5] technique is used to project the data into a 2D vector space to visualize the graph. Two transforms are applied to the labels to better visualize those signals. The highest outlier values of the box office and awards labels are clipped to the third quartile. Then, each label value is divided by the same third quartile. The critics and IMDb grades label are binarized into the "good and bad" categories (1 or 0 respectively) defined by Rotten Tomatoes where the separation grade is 60%. The IMDb is separated at 68% (the mode of its grade distribution). From this figure, we observe that the bottom cluster has more awards and a higher box office. Moreover, the user popularity (IMDb, critics) is quite diverse in every cluster.

4.2 Clustering

The clustering process is performed using DBSCAN as it uses distance and is not susceptible to shapes, unlike namely K-means. Because the datapoints are tightly grouped and elongated, this clustering approach gives good results.

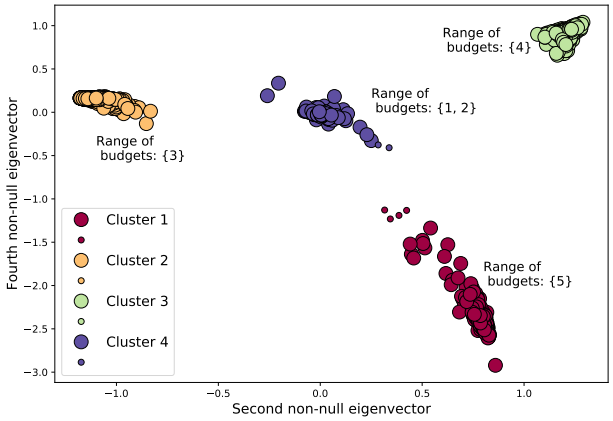
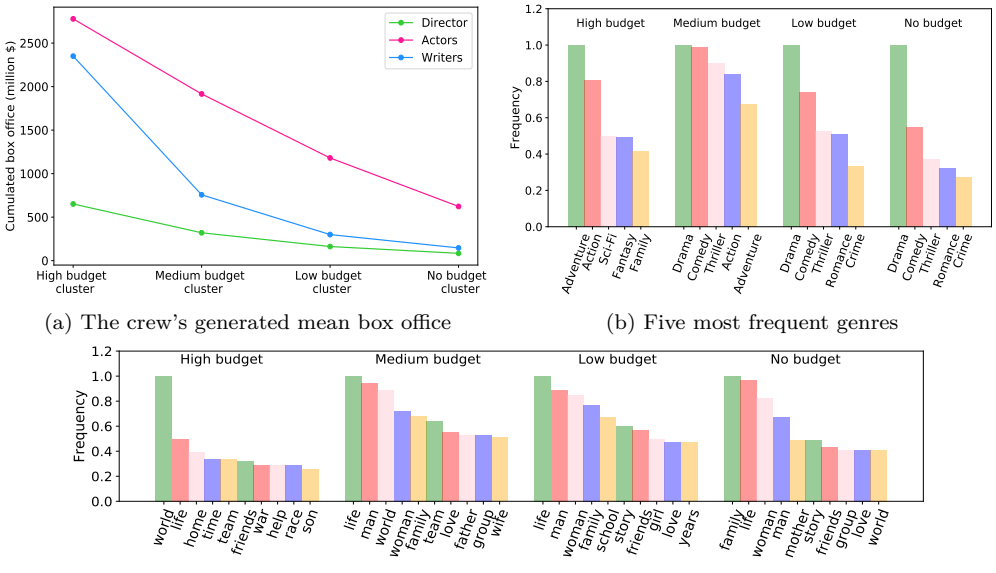


Figure 3: Graph formed of four clusters represented by different movie budgets

The points are grouped in four clusters and there are no outliers. A mean silhouette coefficient [4] of 0.941 (maximum at 1) indicates that each point is well matched to its own cluster and poorly matched to the others. The clusters' defining characteristic is found to be the budget. Cluster 4 regroups independent and no-budget films. Clusters 1, 2 and 3 are formed by the high budget, medium and low budget respectively.

4.3 Cluster analysis

An analysis of clusters' features is presented in Figure 4.



(a) The crew's generated mean box office

(b) Five most frequent genres

(c) Ten most frequent plot words

Figure 4: Feature analysis by cluster

First, Figure 4 (a) shows that high budget films have crew members who have a record of highly profitable movies behind them. The generated box office versus cluster curves are especially steep for the director. This suggests that the disparity between clusters for the director's generated box office is greater than the disparity for actors or writers. Figures 4 (b) and (c) show how cluster 1 films tend to present different

scenarios than other clusters. On the one hand, medium, low and no-budget films are all dominated by human-interaction genres (drama, comedy, thriller, romance) and seem to present human-centered stories ('woman', 'man', 'family', 'girl'). On the other hand, high budget films tend to diversify into "grand-scale" genres (adventure, action, sci-fi, fantasy) and their plots contain more belligerent or adventurous terms like 'time', 'war', 'race' or 'help'.

Figure 5 presents the label characterization of each cluster. For academical, critical and fan popularity a U-shape is observable. The two ways to make a film popular among users and authorities are located at extreme budget situations. Financial success does not follow this law and is simply proportional to the film's budget. Indeed, independent movies rarely aim for mainstream popularity and their target is the recognition of their peers. All in all, the two general ways to make a popular movie appear to be the blockbuster way and the small independent route.

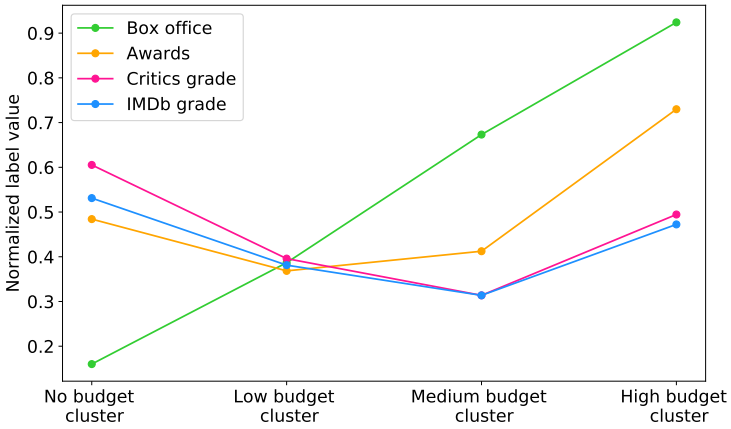


Figure 5: Labels' analysis by cluster

5 Discussion and conclusion

The conducted analysis was intended to determine the main reasons influencing a film's popularity. Popularity was measured as the collected box office, awards won and grades from users and critics.

We selected the features that best allowed us to predict the labels. They were used to create a movie network that showed four different clusters after Laplacian embedding and DBSCAN clustering (Figures 2 and 3). When investigated further, the most discriminating factor between the clusters seemed to be the budget. These clusters had interesting characteristics. The cluster having the largest budget showed grand-scale genres, adventurous storylines and a crew having generated a lot of box office in other movies. The clusters having medium, low and no budget have more human-centered genres and plots, and the profitability of their crew members decrease with the budget. As for the labels, the highest and lowest budget clusters showed higher popularity than clusters in between across user and authorities-based measures. Financial success was fully explained by the clusters' budget (Fig. 5). Our findings hence suggest that in order for a movie to be popular among users, critics and award-providers, movies need to be either blockbusters or independent movies. The amount of box office is proportional to the budget. Hence, movie popularity can be fully explained by budget.

These findings need to be put into perspective. After cleaning, only 2000 movies were left. This number may be not big enough to be representative. Moreover, using each crew member's generated box office could have resulted in an overestimation of the crew features' importance during Lasso. Another factor is the categories chosen for the budget and some labels. These were chosen by thresholding based on markers on the website or ones that create logarithmic classes. Different division may lead to variation in the results.

References

[1] Omdbapi.com. (n.d.). OMDb API - The Open Movie Database. [online] Available at: http://www.omdbapi.com.

[2] Dane, S. (2018). TMDB 5000 Movie Dataset. [online] Kaggle.com. Available at: https://www.kaggle.com/tmdb/tmdb-movie-metadata [Accessed 17 Jan. 2019].

[3] Computer vision for dummies. (2019). The Curse of Dimensionality in Classification. [online] Available at: http://www.visiondummy.com/2014/04/curse-dimensionality-affect-classification/ [Accessed 17 Jan. 2019].

[4] wikipedia.org. (2019). Silhouette (clustering). [online] Available at: https://en.wikipedia.org/wiki/Silhouette_(clustering) [Accessed 17 Jan. 2019].

[5] Belkin, M. and Niyogi, P. (2003). Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. Neural Computation, 15(6), pp.1373-1396.