Ying-Hsi Liu

SI 601 Project Proposal

Visualizing and analyzing Netflix’s ratings and IMDB’s ratings on TV shows

Motivation

Netflix is a growing provider of internet streaming media that mainly based in the United States. Netflix is expanding its business and starting to provide services throughout the world. Netflix provides wide variety of selections of TV shows and movies, users only have to pay a small amount of money per month ($7.99/month) to subscribe to Netflix, and they will be able to watch the contents Netflix provides. In order to create better user experience, Netflix provides ratings for every TV shows and movies. These ratings are based on average user ratings. I want to explore that if the platform of watching a TV show will have an effect on users’ ratings. My motivation on doing this data analysis is because since internet streaming is changing the media market, I would like to know that if users will change their preference towards the shows they are watching if they are using different platforms to watch the show. By knowing that, I will be able to design better platforms that fit the users’ tastes. I will compare IMDB’s average user rating on a TV show to the rating of Netflix on the same show. IMDB’s users are watching TV show from various platforms, whereas Netflix’s users watch the same show only on Netflix. By comparing the correlation between these two datasets, I will be able to find out that if the watching platform will have an effect on user’s ratings.

Data Sources

I used two datasets. One is the rating charts of top 250 TV shows on IMDB, and one is the ratings of the same TV shows on Netflix.

I used API provided by Flix Roulette to get the ratings of the TV shows in Netflix.( <http://netflixroulette.net/api/)> The data returned in the format of JSON lines, containing information of tv show titles, show id, category, cast, and rating. Basically everything on the description page of Netflix.

For IMDB ratings, I used BeautifulSoup to fetch the html document on IMDB and used urllib2 to read the data from <http://www.imdb.com/chart/toptv/?ref_=nv_tp_tv250_2> and parse it to python. It was returned in a html doc and be decoded in utf-8. I only use the overall rating on the show, not focusing on the ratings on individual episode. The ratings on Netflix is 5-star base, whereas IMDB’s ratings are 10-star base.

To avoid having bad data like the TV show with only one rating, I first sort the TV shows by popularity, making sure that the ratings are made by multiple users. I fetched 250 ratings of the TV shows. I used the 250 TV shows to retrieve data from Netflix, and I found 72 TV shows that is available on Netflix, and these two data will be my dataset.

Data Manipulation

For ratings on IMDB, I used BeautfifulSoup to fetch the data. I used soup.find to find the lines that contained rating and tv show titles. If a rating is missing, I used if rating\_row to avoid having missing rating data. Then I stored the data in a list containing data of rating, id, and tv show titles. I also stored the titles in a list because I would use the titles for the web API. I outwrote a tsv file to store the data for next step.

The Netflix API will use the tv show titles to get the ratings. Sometimes the web API will return error, so I used try, except urllib2.HTTPError to avoid the situation. The API needs title that the blank space was replaced by %20, and I did it. The data was returned in a JSON string, I output the data.

Now I have the data from Netflix, I needed to get the ratings. I stored the data in a dictionary, and the keys are the tv show titles. I used JSON loads to load the lines, and found the row with rating and TV show titles. Again, to avoid situation where no rating is available, I used if rating to make sure that I could get all the correct data. I put the rating in dictionary with the keys as the tv show titles.

Since that rating have not been encoded, I encoded it with utf-8, and multiply it by 2. I also convert the rating from string to float.

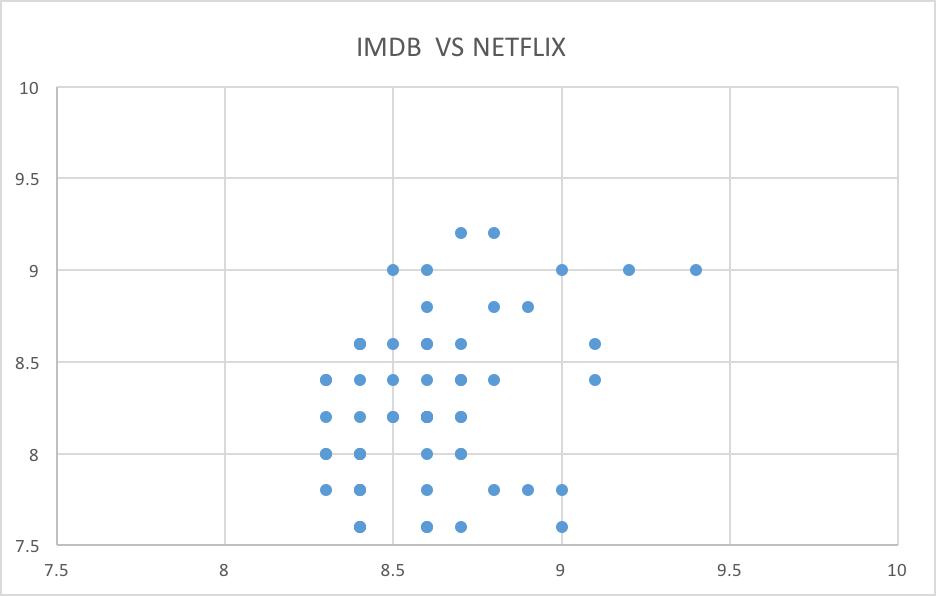
For my final result, I created a list to store data from the IMDB data list. I encoded the IMDB data list with utf-8. I used the tuple of l to store the data from IMDB list, and check if the IMDB title is in the Netflix dictionary list, if is, then append the data in the tuple. I output the result in a csv file, and I used it to graph out the plot to see the correlation relationship.

In my file, I will be able to get the TV show with two ratings, one from Netflix and one from IMDB. Thus, I will use these two sets of data to explore the correlation between ratings from Netflix and IMDB.

In the process of manipulating the data, the first challenge I encountered was unable to fetch the data from IMDB file. I tried a few time and found out that the error occurred because one TV show was missing the rating, so the whole line broke. I wrote an if line to avoid this. The second challenge was that I did not use the utf-8 encode to encode my JSON string, so it was unable to be manipulated. The third challenge was that the API needs the title with %20 instead of blank. It took me a few times to get the correct lines of all the TV show titles. The web API sometimes will return error, so I looked up online and found out that I should use except urllib2.HTTPError: to avoid bad error return.

Analysis and Visualization

I graph out the ratings from IMDB and Netflix to see if there is a correlation between these two sets of data on the same TV shows. By analyzing 74 TV shows, I will be able to see the relationship between these two ratings. I drew a scatter plot to examine if there is any correlation between these two ratings.



As we can see, there is a slightly correlation between these two data sets. There are about 7 data where the IMDB ratings are clearly lower than the Netflix ratings, but generally speaking, the two ratings are close to each other.

I would make a conclusion that users tend to make higher ratings on Netflix but not all the users will have this tendency. In general, the platform where users watch the TV shows does not dramatically affect user’s ratings.