EE232 Project 3
Imdb Mining

Team members:
Sonali Garg (104944076)
Shweta Sood(905029230)
Karan Sanwal(205028682)
Ashish Shah(804946005)

Question 1: Perform the preprocessing on the two text files and report the total number of actors and actresses and total number of unique movies that these actors and actresses have acted in.

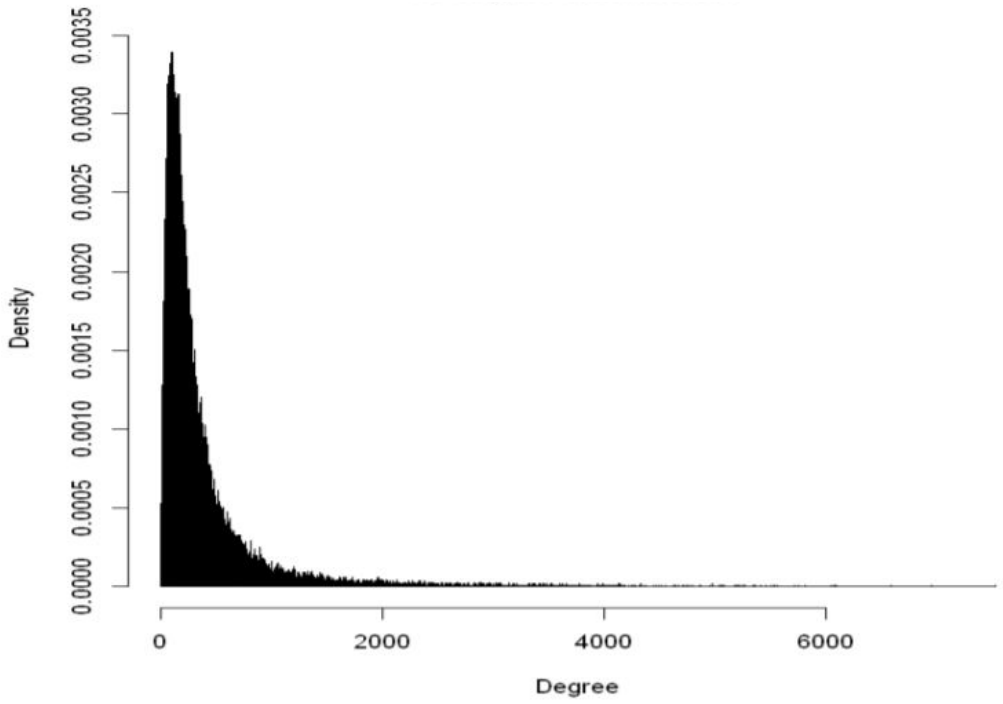
Solution:
In this question, we first merged the 2 text files, actor_movies and actress_movies into one and removed all those actors and actresses who acted in less than 10 movies.
Further, we cleaned the text file by removing all garbage characters(if any) and kept the movies and its year.

We got the following results.

Total number of actors: 74589
Total number of actresses: 38527
Total number of actors and actresses: 113116

Unique number of actor movies: 428851
Unique number of actress movies: 328532
Total unique movies: 468186

Question 2: Create a weighted directed actor/actress network using the 2 processed text file and equation 1. Plot the in-degree distribution of the actor/actress network. Briefly comment on the in-degree distribution.



From the above figure, we see that the the number of nodes with high in-degree are very less and the number of nodes with very low in-degree are very high.
In the graph all the nodes are actors and there is an edge between two actors only if they have acted in a movie together. This is uncommon in our graph and data file as most actors would not have acted together. Only long standing actors who have acted over several decades or extremely talented actors would have acted in a lot of movies with several other actors. There are very few such actors and we can see this from our distribution.

Question 3: Design a simple algorithm to find the actor pairings. To be specific, your algorithm should take as input one of the actors listed above and should return the name of the actor with whom the input actor prefers to work the most. Run your algorithm for the actors listed above and report the actor names returned by your algorithm. Also for each pair, report the (input actor, output actor) edge weight. Does all the actor pairing make sense?

Solution:

Input Actor	Output Actor	Edge Weight
Tom Cruise	Nicole Kidman	0.1746
Emma Watson (II)	Daniel Radcliffe	0.52
George Clooney	Matt Damon	0.1194

Tom Hanks	Tim Allen	0.1012
Dwayne Johnson (I)	Steve Austin Mark Calaway Paul Levesque	0.2051
Johnny Depp	Helena Bonham Carter	0.0816
Will Smith (I)	Darrell Foster	0.1224
Meryl Streep	Robert De Niro Kevin Klein	0.0618
Leonardo Dicaprio	Martin Scorsese	0.102
Brad Pitt	George Clooney	0.0986

The actor pairing in the above table makes complete sense. We can see from the below explanations.

1. Tom Cruise and Nicole Kidman were earlier married and acted together in many movies.
2. Emma Watson and Daniel Radcliffe acted in all 8 Harry Potter movies together (Last book split into 2 movies).
3. George Clooney and Matt Damon are real life great friends and have acted together in the Ocean's series as well as other movies like Confessions of a Dangerous Mind, The Monuments Men etc.
4. Tom Hanks and Tim Allen are known for their voicing talent and were the voices for the Toy Story and Cars series of animated movies.
5. Dwayne Johnson and Steve Austin were wrestlers together in WWE and have acted in many action movies together. It has the same edge weight with other actors as well. He acted with Mark Calaway as 'The Rock' and 'Undertaker'. Paul Levesque is 'Triple-H', another famous WWE actor.
6. Johnny Depp and Helena Bonham Carter are director Tim Burton's most beloved actors. They are able to portray a little off-the-beat roles with surprising ease, most of Allens movies are based in a fictitious world. They acted in movies including Sweeney Todd, Charlie and the Chocolate Factory, Alice in Wonderland together.
7. Will Smith and Darrell Foster have acted together in many action comedies like Men in Black series, I am Legend, Hancock.
8. Meryl Streep and Robert de niro have acted in many movies such as The deer hunter, Falling in love etc. She acted with Kevin Klein on Ricki and the Flash etc.
9. Leonardo Dicaprio and director Martin Scorsese have worked together in many movies such as Shutter Island and The departed.
10. Brad Pitt and George Clooney have acted together in the Ocean's series and movies like His Way, Confessions of a Dangerous Mind, Burn After Reading, etc.

From the above list, we see that the preferred actor of a given actor is the one who they shared a **series of movies** with or if they are the **preference of a particular director**. We see that the Jaccard weight gives a very good idea of importance of one node with respect to another.

Question 4: Use the google's pagerank algorithm to find the top 10 actor/actress in the network. Report the top 10 actor/actress and also the number of movies and the in-degree of each of the actor/actress in the top 10 list. Does the top 10 list have any actor/actress listed in the previous section? If it does not have any of the actor/actress listed in the previous section, please provide an explanation for this phenomenon.

Solution:

Top Actors	Pagerank Score	Number of movies	In-degree
Bess Flowers	0.0002352	828	7537
Fred Tatasciore	0.0001989	353	3954
Sam Harris (II)	0.0001972	600	6960
Steve Blum (IX)	0.0001955	373	3316
Harold Miller (I)	0.0001727	561	6587
Ron Jeremy	0.0001585	637	2905
Lee Phelps (I)	0.0001573	647	5563
Yuri Lowenthal	0.0001567	317	2662
Robin Atkin Downes	0.0001517	267	2953
Frank O'Connor	0.0001469	623	5502

From the above table, we see that there is no common actor who is present in both the previous question and this question. According to page rank scores, there is not even 1 famous actor(of today's age) in the pagerank list of top 10 actors. This can be reasoned by the big difference in the number of movies acted by these actors(such as bess flowers) and the number of movies acted by the famous celebrities. Actors in q4(pagerank celebrities) have acted in over 500 movies each but the famous actors in q3 have acted only in 50 movies.

The in-degree of these actors is very high as compared to the famous actor list. Hence the pagerank actors have acted with many other/different actors of their time. Some of them have played supporting roles or voice roles as well.

Pagerank scores are higher for those nodes which share edges with nodes which have a high in-degree. Supporting actors usually share the movie with other supporting actors which have a higher in-degree. Hence these actors listed by pagerank have high indegree and edges with high in degree nodes.

Question 5: Report the pagerank scores of the actor/actress listed in the previous section. Also, report the number of movies each of these actor/actress have acted in and also their in-degree.

Solution:

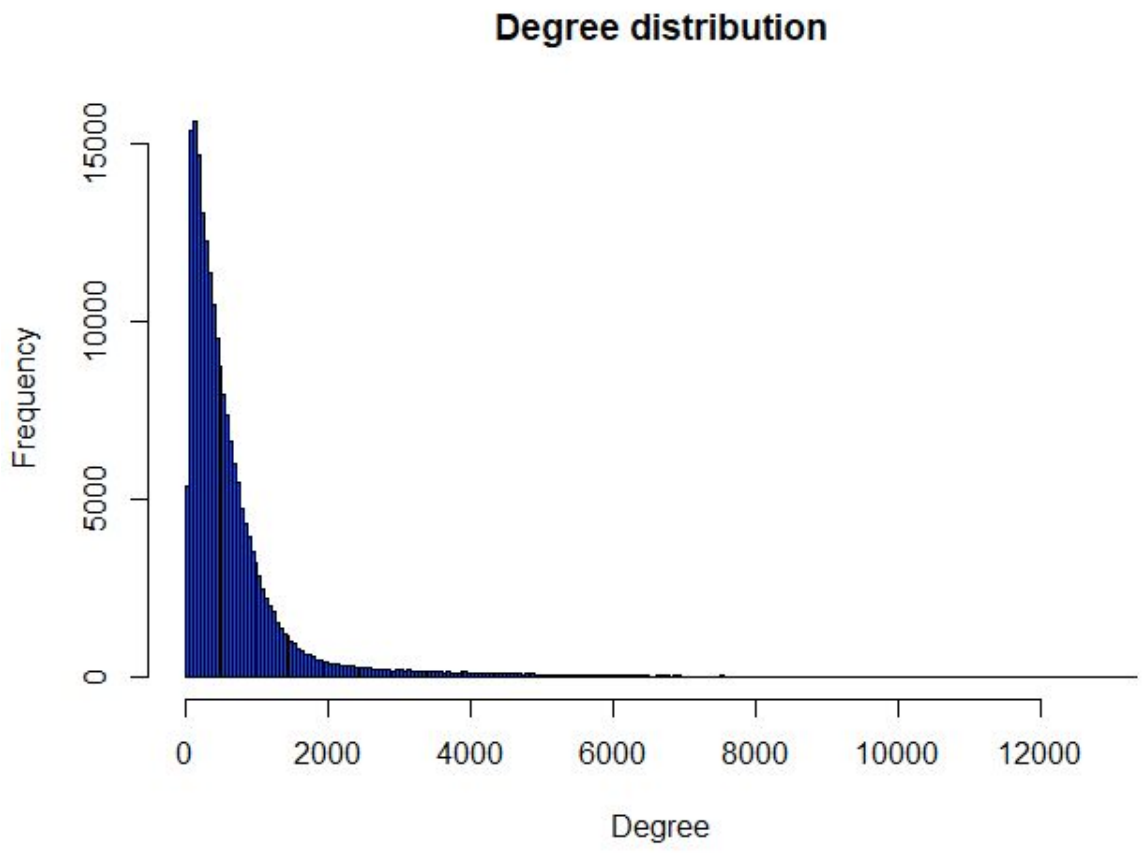
Top Actors	Pagerank Score	Number of movies	In-degree
Tom Cruise	3.9755e-05	63	1651
Emma Watson (II)	1.7489e-05	25	453
George Clooney	4.0039e-05	67	1573
Tom Hanks	5.1060e-05	79	2064
Dwayne Johnson (I)	4.2027e-05	78	1357
Johnny Depp	5.3826e-05	98	2144
Will Smith (I)	3.2023e-05	49	1319
Meryl Streep	3.9619e-05	97	1594
Leonardo Dicaprio	3.1688e-05	49	1301
Brad Pitt	4.2987e-05	71	1739

From the above table, we see that the pagerank scores for the famous celebrities are very low when compared with the pagerank scores of actors listed in q4. This can be reasoned by the low number of movies acted by them and the low in-degree.

These actors have acted in very less number of movies as leading roles and that's why they have shared movies with very few actors. They have acted with popular actors which also have shared a movie with very less actors with less in-degree. The probability of the random walker arriving on this node is very less when compared with the probability of arriving on the actors listed in the previous question.

Question 6: Create a weighted undirected movie network using equation 2. Plot the degree distribution of the movie network. Briefly comment on the degree distribution.

Solution:
We use the equation 2 given in the project statement to construct a weighted-undirected network. We used the processed text files from the previous section to create the movie network and further, only consider movies with more than 5 actors.
In our network, we get 203,567 nodes(unique movies) and 66,520,989 edges. We plot histogram for degree distribution. The degree distribution is as follows:



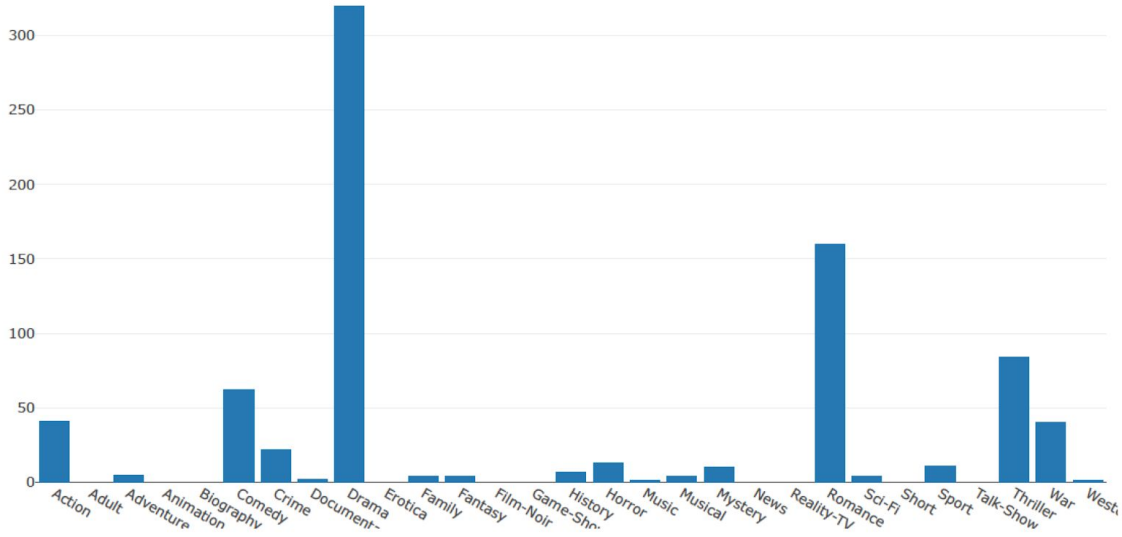
Observation:
We notice that the distribution seems to follow power law. We notice that most nodes have less degree while very few have very high degree. As the degree increases, the frequency of nodes that have that degree seems to fall exponentially. We know that for there to be an edge between 2 movies, they must have at least one actor in common. Hence, this distribution makes sense. Only few actors would have been a part of many movies(This would probably be the case for actors with small roles(supporting or extra) in multiple movies, actors with long careers, or

perhaps some pretty prolific actors). Others generally tend to be a part of a small number(not very high) of movies in their career and this distribution reflects that.

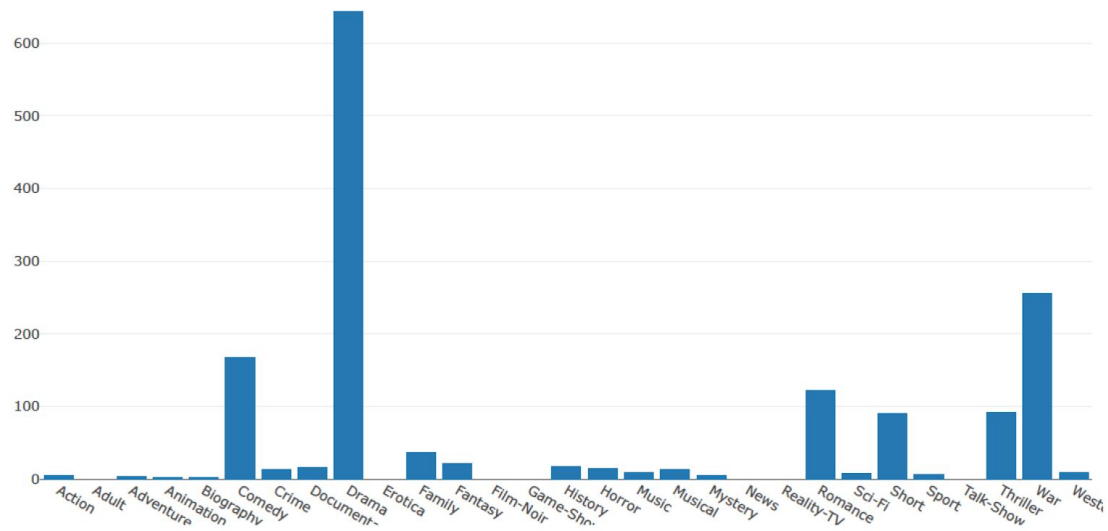
Question 7: Use the Fast Greedy community detection algorithm to find the communities in the movie network. Pick 10 communities and for each community plot the distribution of the genres of the movies in the community.

Solution: On running the fast greedy community detection algorithm, 30 communities were found. The plots for distribution of genre for 10 communities have been shown below:

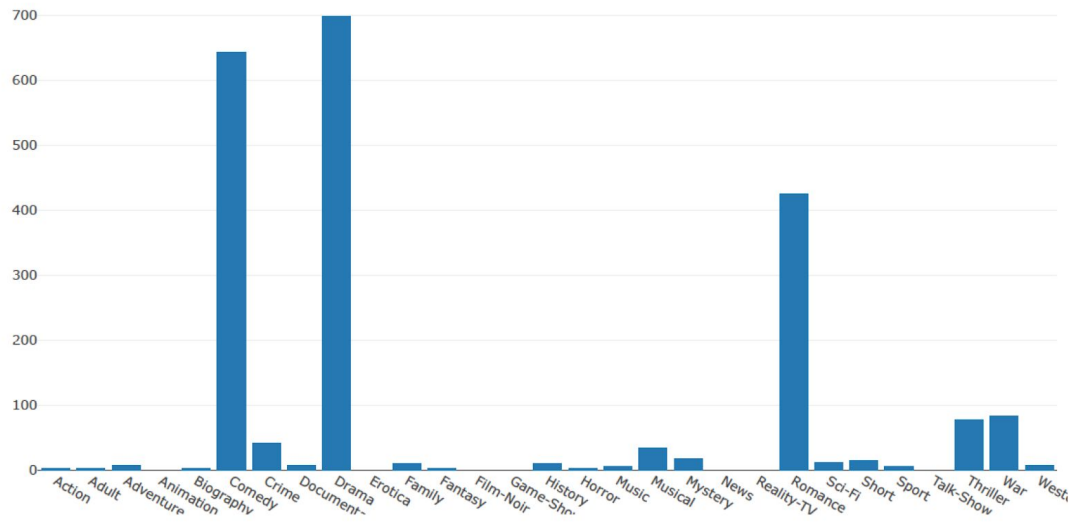
Community #10 (847 nodes):



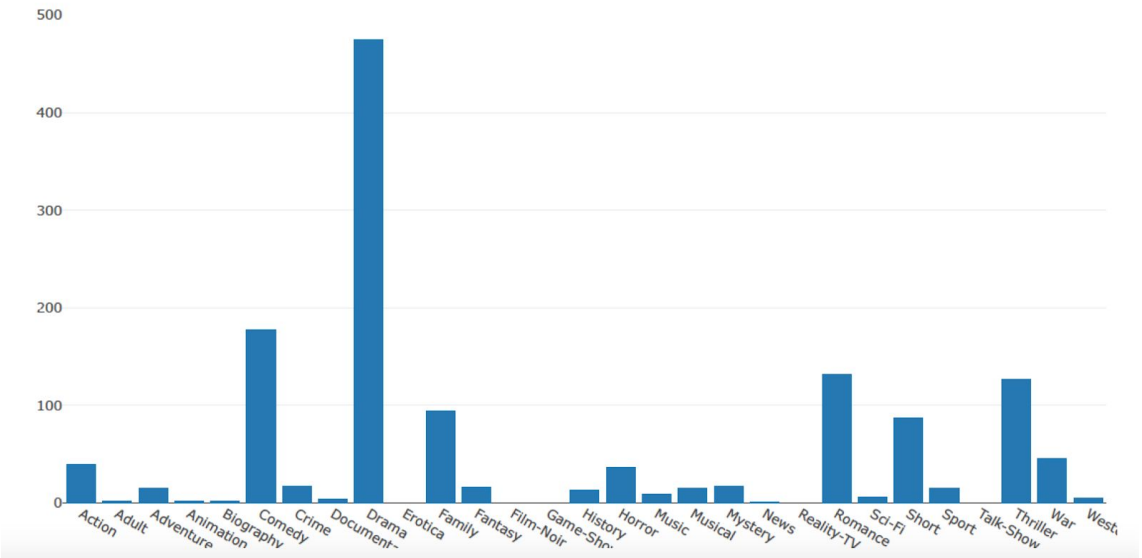
Community #12(1673 nodes):



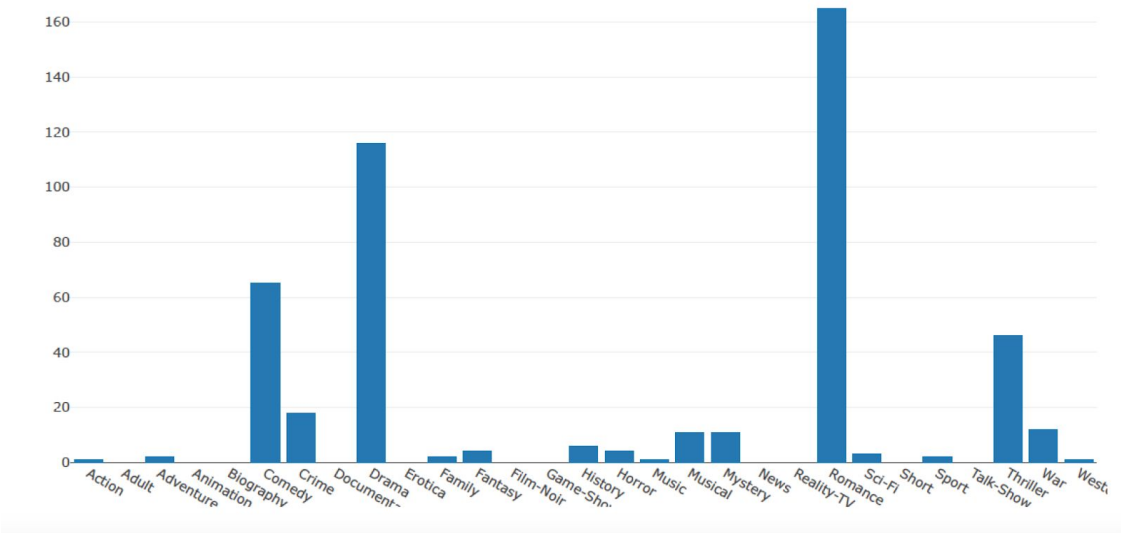
Community #17(2124 nodes):



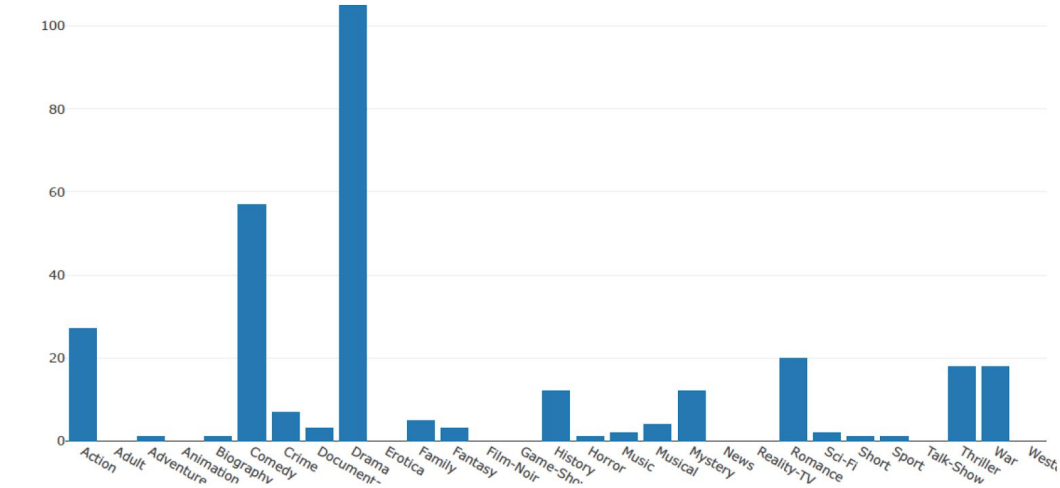
Community #19(1622 nodes):



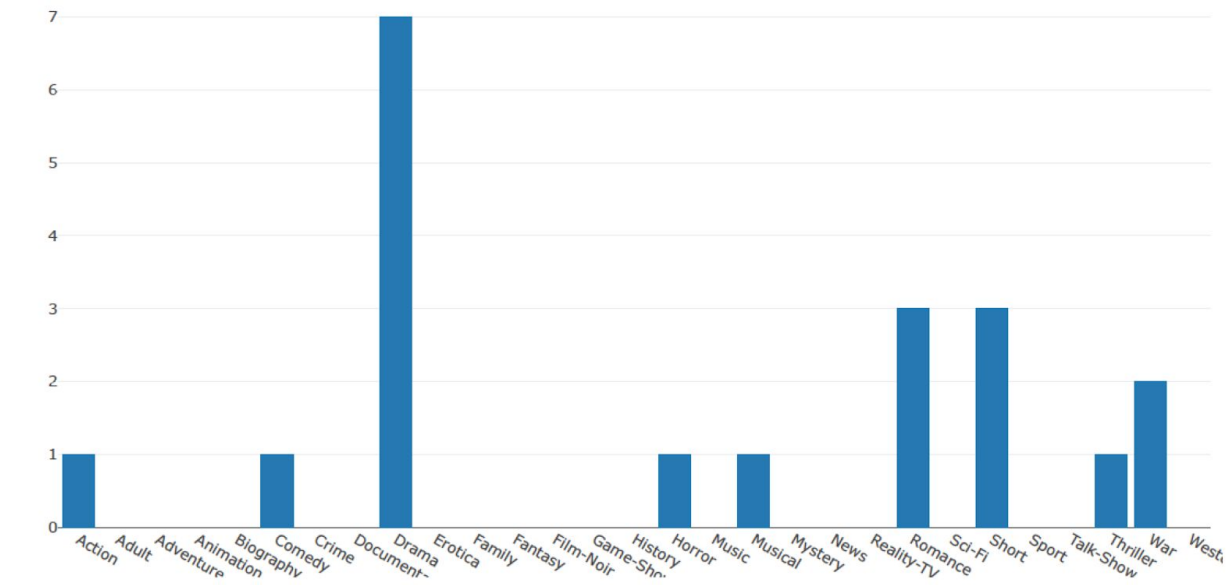
Community #21 (684 nodes):



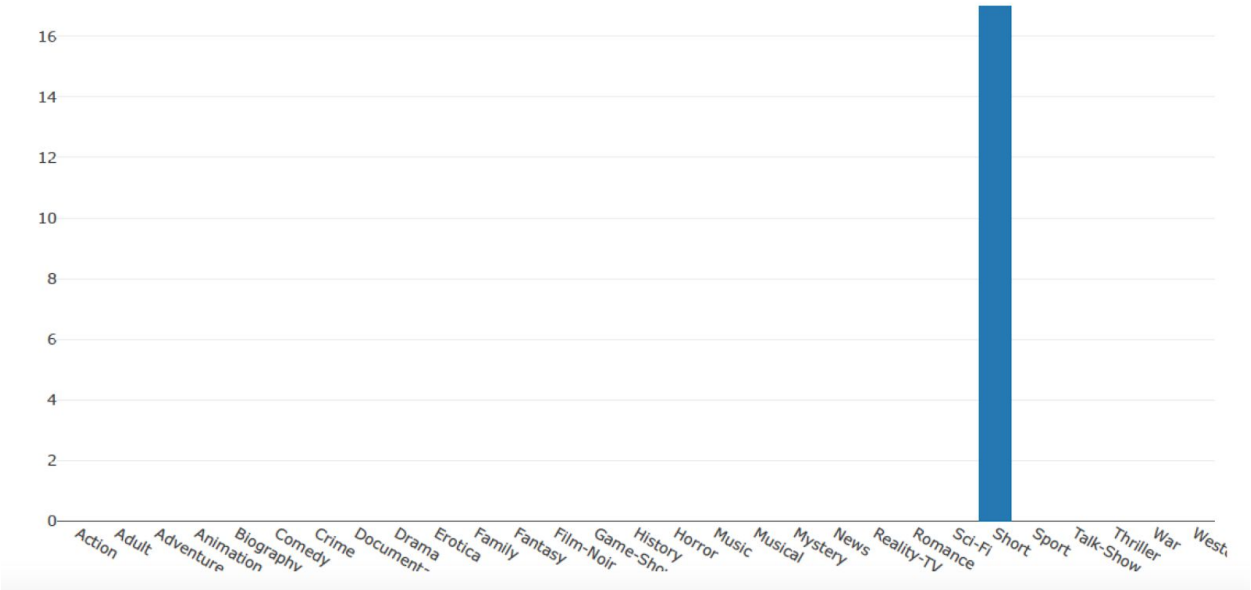
Community #23 (627 nodes):



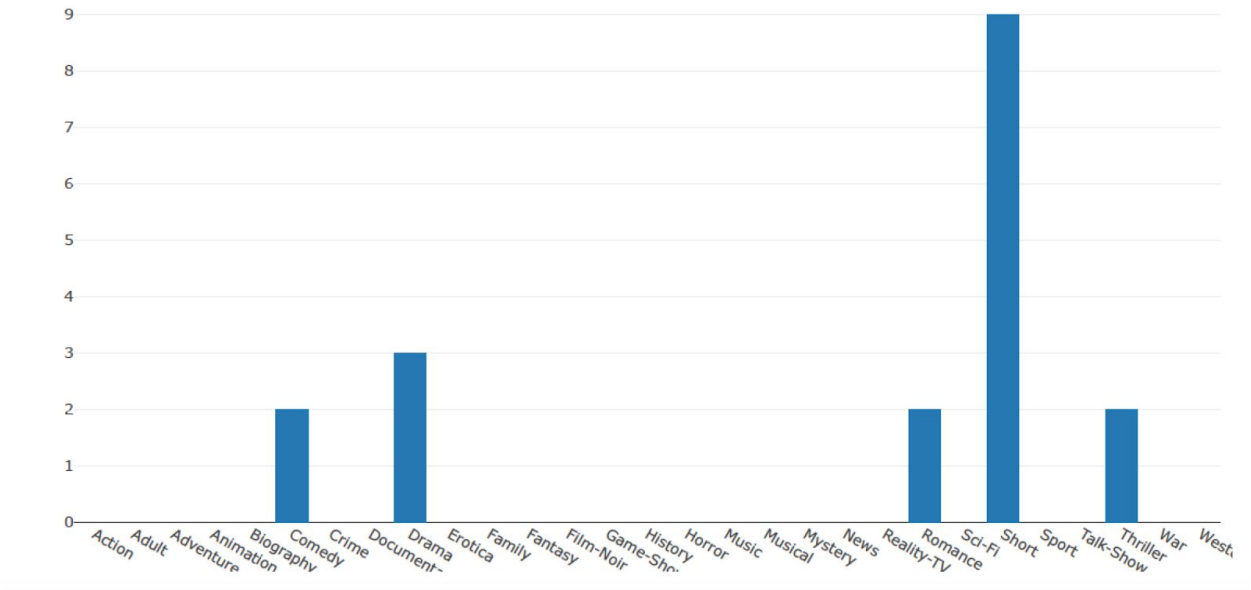
Community #24 (23 nodes):



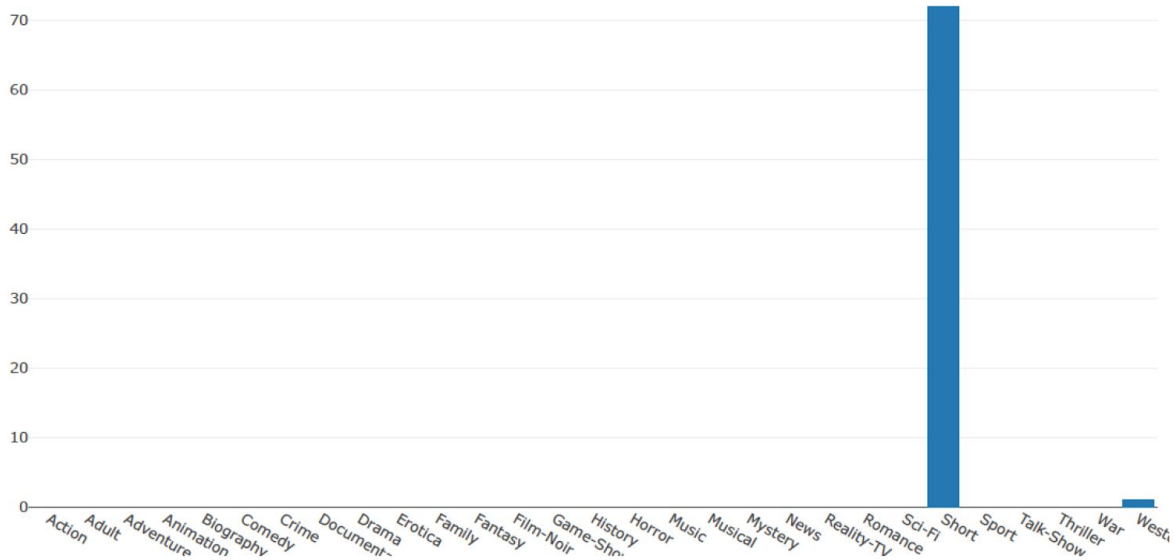
Community #26 (17 nodes):



Community #28 (18 nodes):



Community #30 (73 nodes):



Observation:
We observe that while the genres are distributed across communities, generally(not always) there is one genre that has considerably higher frequency than the others. This could be because the edges are based on common actors between movies and often, actors are associated with same/similar genres of movies. For example, Amy Poehler would often be associated with comedy movies. Another interesting observation seems to be be that the genre drama seems to dominate (specially when the community size is relatively large(not always but in many cases), while in smaller ones we see short dominating in our random subset(again, not always but in many cases)). This make sense as the genre drama has the highest frequency across the movies in our graph.

Question 8(a): In each community determine the most dominant genre based simply on frequency counts. Which genres tend to be the most frequent dominant ones across communities and why?

Solution: In this part, we determine the most dominant genre in communities based on frequency counts. The dominant genre per community is as given below:

Community #	Dominant genre	Community #	Dominant genre
1	Thriller	16	Drama
2	Adult	17	Drama
3	Drama	18	Short
4	Drama	19	Drama
5	Drama	20	Drama
6	Drama	21	Romance
7	Drama	22	Short
8	Drama	23	Drama
9	Drama	24	Drama
10	Drama	25	Thriller
11	Drama	26	Short
12	Drama	27	Adult
13	Drama	28	Short
14	Drama	29	Musical
15	Drama	30	Short

Observation: We observe that the genre Drama is the dominant one in 19 communities, short is dominant in 5 communities, thriller and adult are dominant in 2 communities, musical and romance are dominant in one community each. Drama being the dominant genre based on frequency makes a lot of sense as it has the highest frequency count across all our movies(46397 movies) followed by comedy and short with 21949 and 21505 movies each. The frequency of drama is greater than twice the next highest genre. Hence, the genre drama tends to dominate in most communities based on frequency counts. After that, short dominates in the most(mostly in communities with smaller number of movies). Frequency count as a metric for dominance is biased towards genres that have higher counts across the dataset(i.e. movies in the graph that we know genre of).

Question 8(b): In each community, for the i th genre assign a score of ln(c(i)) ∗ p(i) q(i) where: c(i) is the number of movies belonging to genre i in the community; p(i) is the fraction of genre i movies in the community, and q(i) is the fraction of genre i movies in the entire data set. Now determine the most dominant genre in each community based on the modified scores. What are your findings and how do they differ from the results in 8(a).

Solution:
Now based on the modified scores (as explained in the question), the dominant genre per community is as given below:

Community #	Dominant genre	Community #	Dominant genre
1	Documentary	16	Action
2	Adult	17	Comedy
3	Family	18	Fantasy
4	Mystery	19	Family
5	Comedy	20	Adventure
6	Musical	21	Romance
7	Animation	22	Short
8	History	23	Action
9	Adventure	24	Sport
10	Romance	25	Thriller
11	Musical	26	Short
12	War	27	Adult
13	Action	28	Short
14	Family	29	Musical
15	Western	30	Short

Observation:

It this part, we observe that short is the dominant genre in 4 communities, family, musical and action in 3 communities each, Adult, romance, adventure and comedy in 2 communities each, Sport, History, Animation, War, Western, Fantasy, Mystery, Thriller and documentary in 1 community each.

Hence, we get a variety of dominant genres, and there is no one that really mostly dominates across communities. This is different from part (a), where we mostly got drama as the dominant genre(due to its high frequency in movies list). That is possibly not a very good way to differentiate between the communities as drama becomes dominant simply due to its high frequency. In this part, we use the score metric to rectify that. In score, we first take natural log of the count(hence, its actual value matters less but still is important), also we multiply with [p(i)/q(i)], which is the ratio of “fraction of genre i movies in the community” to “the fraction of genre i movies in the entire data set”. Hence, the bias due to high frequency count in the dataset is removed due to the q(i) term. (Basically, dividing by “the fraction of genre i movies in the entire data set” penalizes movies with high count in dataset, hence compensating for the bias). Also, in part a) we saw 6 different dominant genres, while in this part we see a lot more.

Question 8(c): Find a community of movies that has size between 10 and 20. Determine all the actors who acted in these movies and plot the corresponding bipartite graph (i.e. restricted to these particular movies and actors). Determine three most important actors and explain how they help form the community. Is there a correlation between these actors and the dominant genres you found for this community in 8(a) and 8(b).

Solution: We chose community#27, which had 11 movies. On plotting a bipartite graph with the movies and the actors that acted in them, we get:

The node for actors (14 actors) are:
A8240, A20201, A21944, A24479, A32881, A60394, A65188, A68266, A69501, A83734, A89718, A93355, A102090, A111631
The nodes for movies (11 nodes) are:
M112631, M112632, M112633, M112634, M112635, M112636, M112637, M112638, M112639, M165971, M165972
We have included the mappings to the actual movie names and actors for all these in the readme(in case it was required).

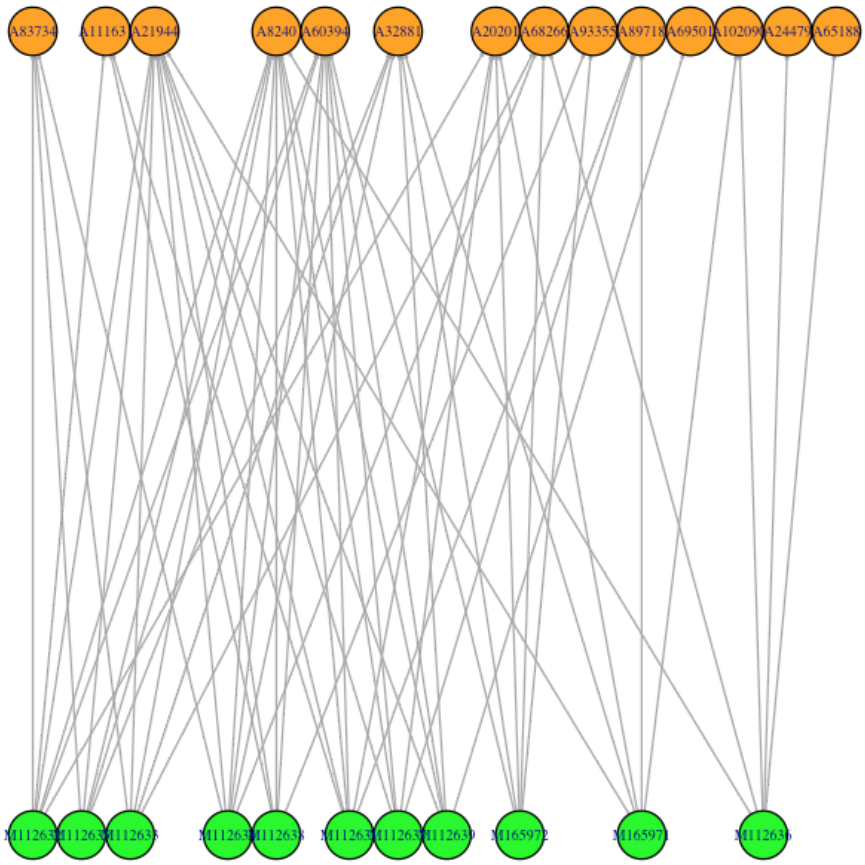


Fig: Bipartite graph

The 3 most important actors(by looking at the degree) are:
A8240 -> Brossman, James(degree 9)
A21944 -> Forte, Alex (degree 9)
A60394->Scott, Ian (III) (degree 8)

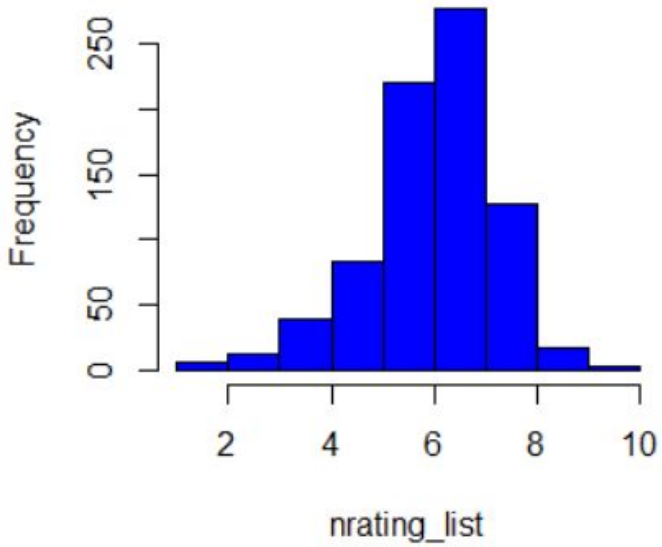
Observation:
We notice that these actors have pretty high degrees, i.e. they were a common factor in many of the movies(James Brossman was in 9 out of 11 movies of the community , and so on). Hence they led to edges between the movies. Also, we have 11 movies and just 14 actors, that means these movies had a lot of common actors, hence them being in a community together makes a lot of sense. The 3 top actors we see were common in 9, 9 and 8 movies respectively. This all led to high weight edges between the movies and hence, they formed 1 community.

The dominant genre found for this community in both 8(a) and 8(b) was Adult.
For Brossman, James, we see that he was in 23 movies(21 Adult, 1 romance, 1 short).
For Forte, Alex, we see that he was in 19 movies(18 Adult, 1 romance).
For Scott, Ian (III), we see that he was in 23 movies(21 Adult, 1 thriller, 1 documentary).

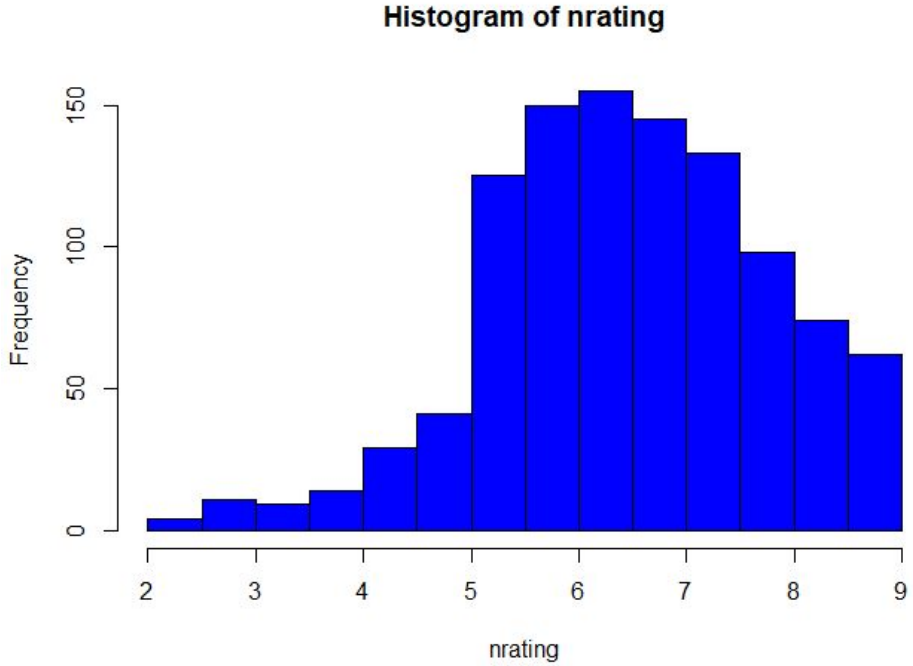
It can hence be seen that there is a direct link between the most frequent genres for these actors with the dominant genre of the community. The actors' most frequent genre is same as the dominant genre of the community.

Question 9: For each of the movies listed above, extract it's neighbors and plot the distribution of the available ratings of the movies in the neighborhood. Is the average rating of the movies in the neighborhood similar to the rating of the movie whose neighbors have been extracted? In this question, you should have 3 plots.

Movie Name = Batman v Superman: Dawn of Justice (2016)
Rating of the Movie = 6.6
Average Rating of the Movies in the Neighborhood = 6.34

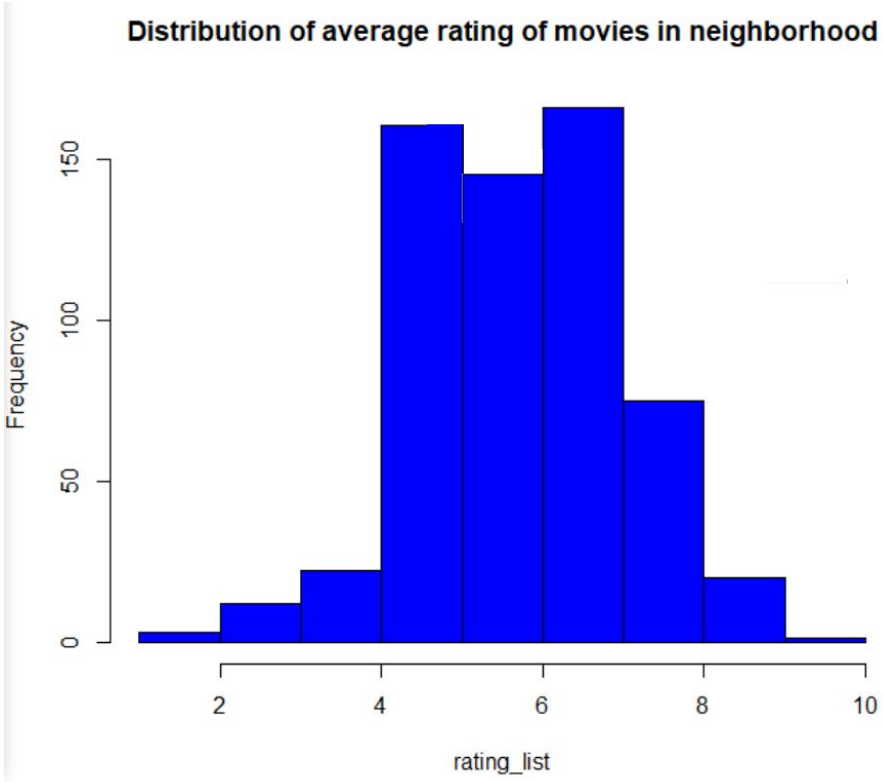


Movie Name = Mission: Impossible - Rogue Nation (2015)
Rating of the Movie = 7.4
Average Rating of the Movies in the Neighborhood = 6.25



Movie Name = Minions (2015)
Rating of the Movie = 6.4

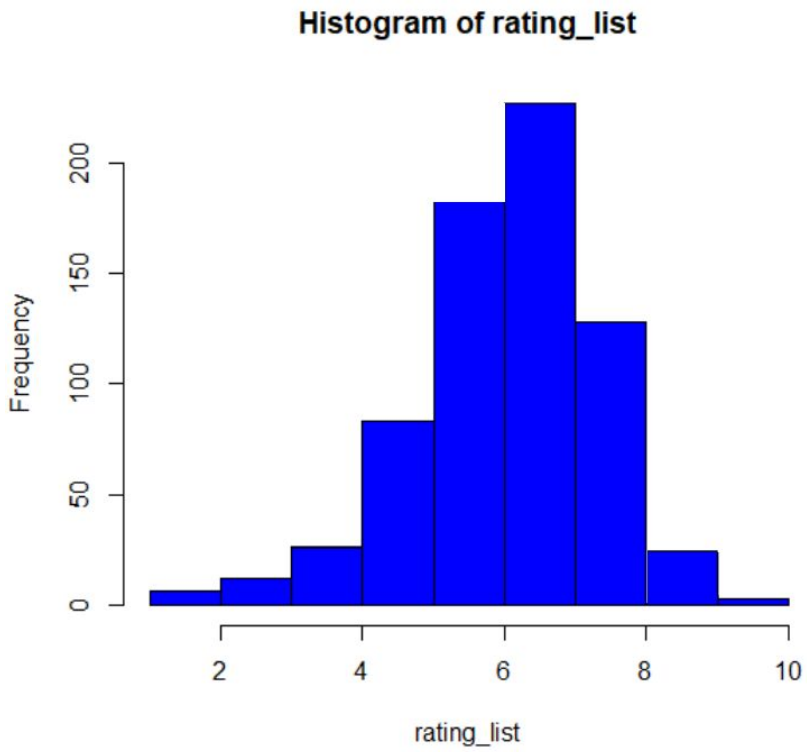
Average Rating of the Movies in the Neighborhood = 6.81



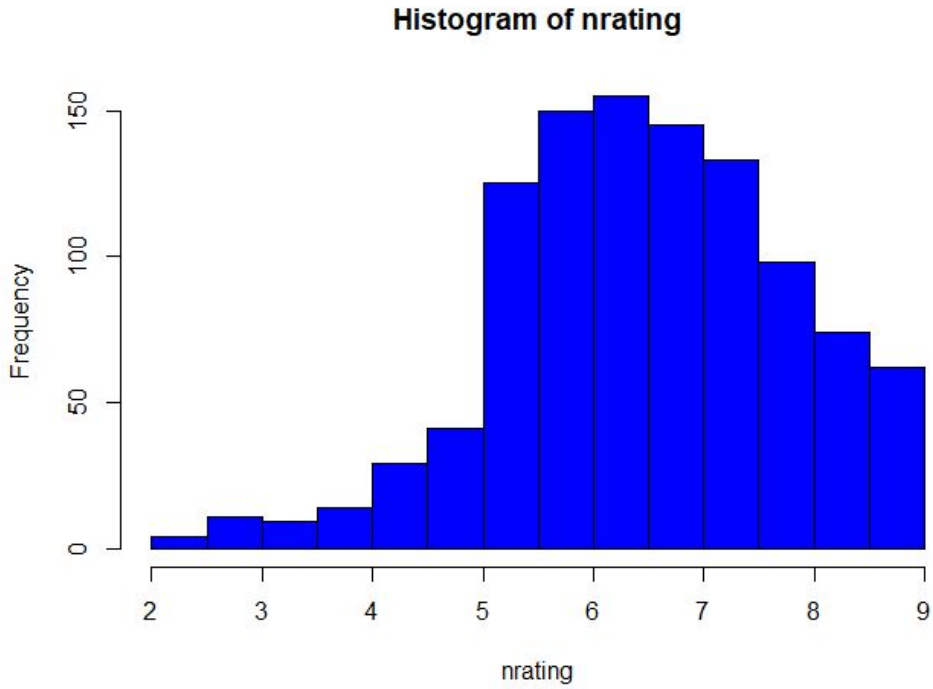
For all the movies listed, we see that the ratings of the movie are close to the average ratings of the movies in the neighborhood. However, for the second movie - Mission: Impossible - Rogue Nation (2015) this difference is still more than 1. So, it is important to search for a more accurate representation of the movie's ratings based on the neighborhood.

Question 10: Repeat question 10, but now restrict the neighborhood to consist of movies from the same community. Is there a better match between the average rating of the movies in the restricted neighborhood and the rating of the movie whose neighbors have been extracted. In this question, you should have 3 plots.

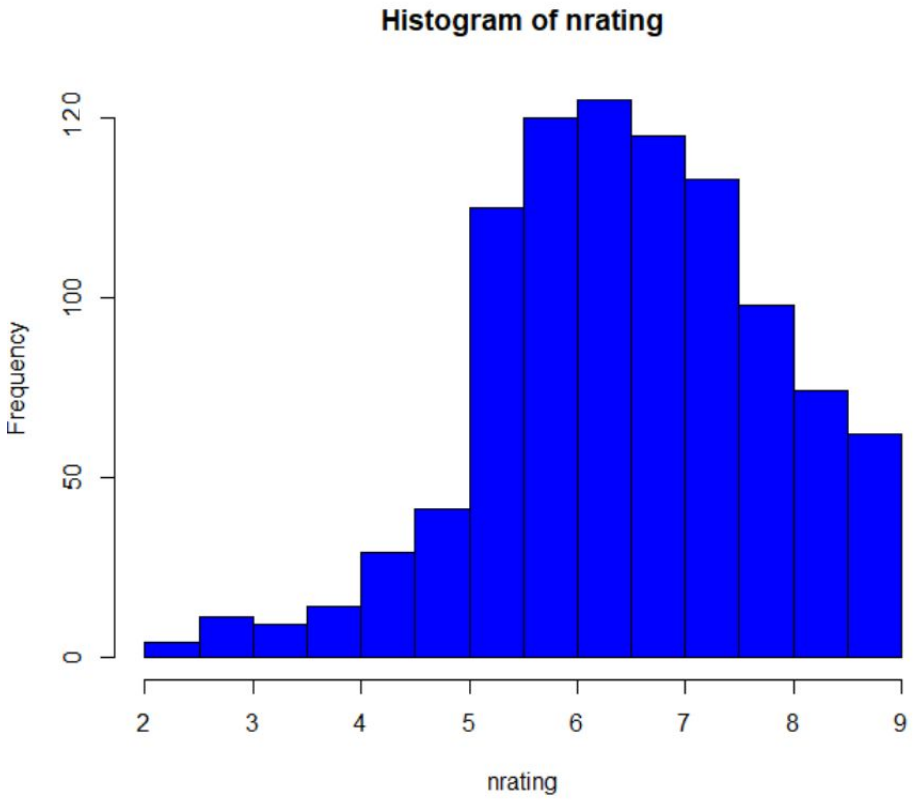
Movie Name = Batman v Superman: Dawn of Justice (2016)
Rating of the Movie = 6.6
Average Rating of the Movies in the Restricted Neighborhood consisting of movies from the same community = 6.36



Movie Name = Mission: Impossible - Rogue Nation (2015)
Rating of the Movie = 7.4
Average Rating of the Movies in the Restricted Neighborhood consisting of movies from the same community = 6.28



Movie Name = Minions (2015)
Rating of the Movie = 6.4
Average Rating of the Movies in the Restricted Neighborhood consisting of movies from the same community = 6.86



The average ratings of the movies in the neighborhood belonging to the same community are closer to the movie rating than the previous part. However, the difference can be accounted to the similarity in the nodes in neighborhood and community. We noticed that the neighborhood of the node had a large number of nodes in the same community. This explains the small improvement in scores. Also, all the movies and their neighbors mostly belonged to the first community.

Question 11: For each of the movies listed above, extract it's top 5 neighbors and also report the community membership of the top 5 neighbors. In this question, the sorting is done based on the edge weights.

Movie Name = Batman v Superman: Dawn of Justice (2016)

Top Movies	Edge Weights	Community Membership
Eloise (2015)	0.11250000	1
The Justice League Part One (2017)	0.07575758	1
Into the Storm (2014)	0.07291667	1

Love and Honor (2013)	0.06097561	1
Man of Steel (2013)	0.05982906	1

The table shows the top 5 movies associated with Batman v Superman: Dawn of Justice (2016), sorted by the edge weights. We notice that all these movies are closely associated with batman as they are superhero/action based movies. Also, all of them belong to the same community 1.

Movie Name = Mission: Impossible - Rogue Nation (2015)

Top Movies	Edge Weights	Community Membership
Fan (2015)	0.1585366	1
Phantom (2015)	0.1460674	1
Breaking the Bank (2014)	0.1028037	1
Suffragette (2015)	0.1022727	1
Now You See Me: The Second Act (2016)	0.1011236	1

The table shows the top 5 movies associated with Mission: Impossible - Rogue Nation, sorted by the edge weights. We notice that all these movies are closely associated with mission impossible - action/mystery/thriller. Also, all of them belong to the same community 1.

Movie Name = Minions (2015)

Top Movies	Edge Weights	Community Membership
The Lorax (2012)	0.2777778	1
Inside Out (2015)	0.2500000	1
Up (2009)	0.2424242	1
Surf's Up (2007)	0.2250000	1
Despicable Me 2 (2013)	0.2250000	1

The table shows the top 5 movies associated with Minions(2015), sorted by the edge weights. We notice that all these movies are closely associated with Minions - comedy/animation. Also, all of them belong to the same community 1.

So, overall we deduce that the most strongly correlated neighbor movies are in the same community as the movie and share the same genre and relate in terms of the ratings too. Also, there is a relation between movie ratings of neighborhood and the edge weights between a movie and its neighbors. That is, if the number of common actors between two movies are high, the ratings of the movies will be similar as well. This is commonly observed phenomenon as well. When popular actors come together for some movies, those movies have high ratings (making them similar). Similarly, unpopular ones come together for some movies, those movies have low ratings (making them similar).

Question 12: Train a regression model to predict the ratings of movies: for the training set you can pick any subset of movies with available ratings as the target variables; you have to specify the exact feature set that you use to train the regression model and report the root mean squared error (RMSE). Now use this trained model to predict the ratings of the 3 movies listed above (which obviously should not be included in your training data).

One of the founding principles of machine learning is feature engineering, and often feature engineering is seen as one of the most crucial part in deciding the goodness of a Machine Learning Algorithm.

One of the early intuitions in Feature Engineering work as follows : ‘If it does not make sense to a user, it will not make sense for a machine’. Think of it this way - If it is not possible for a human to predict the gender of a person by seeing their hair color, then developing a gender predicting machine learning algorithm that only takes in ‘hair-color’ as a feature would not be very useful as well.

So the challenge in this question hence was to effectively use genuine chosen features that would produce meaningful answers.

Since there is a wide range of rating within the same genre, Genre as a feature was dropped. Every actor was NOT assigned an integer as training on such a large model could take a lot of time , and the curse of dimensionality would get added on as we would not have equally biased data per actor.

So finally the following features were chosen :

1) Average IMDB rating of the movies of the actors and actresses acting in the movie.
2) Average IMDB rating of the movies of the director of the movie

Using this a model was trained using 20,000 films as the training set. Training was done using Gradient descent with a learning rate of 0.01 and in a batch size of 50.

The final test RMSE was 0.3766.

The rating produced were as follows :
• Batman v Superman: Dawn of Justice (2016); Rating: 6.6 , Predicted Rating : 6.62
• Mission: Impossible - Rogue Nation (2015); Rating: 7.4 , Predicted Rating : 6.88
• Minions (2015); Rating: 6.4 , Predicted Rating : 6.6

The ratings are quite to what they are supposed to be. The model however is trying to generalize to a trend. It is trying to figure out a formulae on how the director and actor's rating effect a movie's ratings. The formulae giving the least error in the training set produces a good result in the test set but not the best possible result (as we will see in the next question).

Question 13: Create a bipartite graph following the procedure described above. Determine and justify a metric for assigning a weight to each actor. Then, predict the ratings of the 3 movies using the weights of the actors in the bipartite graph. Report the RMSE. Is this rating mechanism better than the one in question 12? Justify your answer

After creating the Bipartite graph, the weight of the actor was assigned as re average rating of his/her movies in general. Using this, weighted voting technique , the rating of the movie was predicted as the average of the weights connected to it. For the given test-set this performed much better with a **RMSE of 0.17.**

The rating produced were as follows :
• Batman v Superman: Dawn of Justice (2016); Rating: 6.6 , Predicted Rating : 6.37
• Mission: Impossible - Rogue Nation (2015); Rating: 7.4 , Predicted Rating : 7.24
• Minions (2015); Rating: 6.4 , Predicted Rating : 6.88

This technique seemed to perform higher for the test-set compared to Linear Regression but we tested this in other test sets and there ,Linear Regression seemed to perform better. This technique is based on the assumption that the actors previous movie ratings would be the sole contributing factor to their other movie's rating but sometimes even actors with previously poor rated movies perform great under a good director, and the Linear regression model accounts for that where as this does not. But these movies had iconic actors like Ben Affleck and Tom Cruise which caused the movies rating to be influenced by the actor. Hence the method seem to perform good here.