DATA MINING FINAL PROJECT REPORT

ANALYSIS ON IMDB DATABASE

Group 16

Christian Argueta

Ashwani Braj

Kyle Brost

Aishwarya Malepati

Literature Review

Hit or Flop: Box Office Prediction for Feature Films[1]

This project builds a system that can analyze historical movie data and can predict average user rating and degree of profitability of a particular movie. Since there is a solid connection between a film's budget and the gross earnings, predicting raw gross earnings does not particularly indicate a film's success. Therefore as a more meaningful indicator of a film's success, the gross earnings of a film is changed to a multiple of its budget. For this project, data access and preparation is done with the help sqlite and imdbpy. The methods used here for prediction are Naïve Bayes and SVM. The discussion reveals that the naïve Bayes model is superior for the given prediction tasks. We derived the idea of using imdbpy from this paper. Also this was one of the projects that highlighted the effectiveness of using Naïve Bayes classification. This project uses sqlite for processing and formatting the data whereas as we used MySQL for the same purpose. Also, the format of data differs.

The Netflix Project[2]

Given a user and a movie, this project predicts the rating the user gives to the movie. It uses a mapping from Netflix to features such as the director and genre gathered from IMDB. The algorithm uses k-means clustering to cluster the users together by the IMDB features each movie has. The predictions are primarily made through three methods: using Naive Bayes alone on the IMDB features, the average rating that a cluster gives to a movie, and through combining Naive Bayes with clustering. Along with other sources and class individual project, this project gave us a good idea about the usage of k-means for clustering. This another project that uses Naïve Bayes for classification. The questions being answered in this project are different from the ones we try to answer in ours.

Movie Review Analysis[3]

This is a study on the popularity of movies based on their genre. Also a study of the evolution of animated movies over the decades has been done. Packages plyr and ggplot2 have been used here. A modified version of the movie information and user ratings for IMDB datasets has been used. ggplot along with geom_bar is used here for plotting frequency of movies over decades. The study on the distribution of ratings for various genres has been performed with the use of ggplot along with geom_boxplot. This gave us an idea about various plotting methods and helped us choose one.

Brief Statistics of our Dataset:

The primary CSV file contains 9 attributes, which are listed below, that are pulled from the IMDB website. Two other CSV files were derived from this one and used for more efficient analysis.

- Movie_ID
- Title
- Year
- Genre
- MPAA
- Rating
- Name_ID
- Name
- Role

The CSV file 'use_this_dataset.csv' has approximately 6 million entries across approximately 13 thousand different movie titles, 25 different genres, 17 different MPAA ratings, 11 different cast roles, and 514,386 different names. The years for the movies range from 1927 to 2015. The ratings range from 1.1 to 9.4. The files 'imdb_arm.csv' and 'imdb_clustering.csv' were both derived from this CSV.

Methods and Materials

Dataset Creation:

The datasets for this project were created using MySQL and the imdb2py python library. The initial IMDb files were downloaded from IMDb's website in the form of *.list.gz files. These were collectively parsed using the python script imdbpy2sql.py, which formed a database containing automatically-generated tables of movie information. The tables generated included movie titles and information, person names and role in a given movie, and movie and person information; additionally, tables were generated to link those tables together and to define id values for attributes.

Once the dataset was imported into MySQL, we first noticed that some movies contained odd names for genres, and so we excluded these results from future tables by creating a table containing 32 primary genres (including action, drama, comedy, etc.). We then pulled the movie_id, title, genre, name_id, and name from the database on the condition that the genre was in our list of genres and that IDs matched between tables. We then added a year attribute to our table by parsing the earliest release date's string. One query at a time, we ultimately added attributes for the movie's rating, its MPAA rating (G, PG, etc.), and each person's role in a given movie. These queries generated a table of over 6 million lines across over 13 thousand movies – due to the nature of the dataset, each movie was listed multiple times based on the number of cast members and each of their roles (one cast member could be both an actor and a producer, for example), as well the number of genres for a movie. This meant that one movie with three genres and 20 cast members would be listed in the database over 60 times.

Given this dataset, we realized that we had too many items to effectively run our algorithms in R, and so we created subsets of this data. One dataset was created for our clustering approach, including only distinct movie IDs and each movie's genre, MPAA rating, and rating. This reduced our dataset from over 6 million lines to about 13 thousand, and also allowed the Naïve Bayes algorithm to run more efficiently. We also generated a new dataset for our association rule mining approach using the apriori algorithm. It was decided that the best way to relate cast members was to arrange the table structure such that each movie had columns containing an actor, an actress, a

director, an editor, a producer, and a writer, among some others that weren't used in the association mining. These columns of cast members were ran through the apriori algorithm to generate association rules for the cast of who frequently worked together. We made the decision to only include one of each distinct cast member role in the table, each of which were chosen randomly form the main dataset, because the alternative was to create a table with too many columns and a lot of null values for movies with fewer cast members than the average or maximum. This decision may bias the results of that section, but the alternative may have been computationally improbable or might also lead to incomplete results.

After tables were constructed, they were exported from MySQL using the mysqldump tool. This allowed tables from our database to be exported as .sql and .csv files; the .csv files were read into R for use in the different algorithms.

Genre Prediction:

Options for classification:

In the dataset we have created, movieID, year and movie title appear to be redundant for genre prediction. The values for director, producer, writer, actor, actresses and MPAA ratings seem to be suggestive factors for the prediction. The thought process behind this was that some directors might have a preference for particular genres and same might be the case for other cast and crew members. The MPAA ratings can also influence the genre. For example, most animated movies will be rated G.

Evaluation of classification technique used:

Naïve Bayes: We have used this because under the naïve Bayes assumption, each and every factor of the movie is considered independently to predict the genre of the movie.

Measures taken to improve the results:

From our original dataset we created an alternate table that better fit the project. This meant taking the former dataset which biased the number of genres for a given movie by the amount of cast

members (each movie's genre would be listed once for each unique cast member) and generating a new dataset from it which created columns for cast information. This allowed the dataset to list each movie's genre only once along with the MPAA and rating. The new table included extra information not needed for classifying the genre, and so several attributes were excluded – obvious attributes such as movie_id, year, and title trivially shouldn't have a correlation with genre, in addition to rating which we found to give insignificant results for genre prediction.

The usual casts:

Who works together?

We get 506 rules for predicting people working together. After removing redundancy, we derive 175 unique rules. These rules provide association information for various cast members. There are five roles in the dataset namely Director, Producer, Writer, Actor and Actress.

LHS of the rules:

We did not specify the attribute to be put on either left or right hand side of the rule. This is because for people working together, presence of one implies that of the other and vice versa. However, we curtailed the dataset to only those factors which have cast members' names.

Automatically checking all possibilities:

We used apriori function of the arules package to mine the association rules. We didn't provide it with arguments for lhs and rhs to make sure it mines all the possible combinations.

Findings aligned with genre prediction results:

What we are achieving from the naïve bayes algorithm is similar to what we achieve from apriori for genre prediction. From naïve bayes, we see that there are many movies in drama, comedy and thriller genres. It seems obvious that many directors and film makers will want to work on these genres. After doing the association rule mining using apriori we find this to be correct as multiple rules show up with a decent confidence of around 50% and a support of nearly 1%

Similar Movies:

Why K-Means?

K-means is one of the simplest algorithms which uses unsupervised learning method for solving known clustering issues. It works well with large datasets too. Most other methods are more expensive and memory-intensive. An alternative such as DBSCAN cannot effectively cluster data sets with large differences in density.

Would clustering the movies into k' clusters where k' > k, help in better categorization?

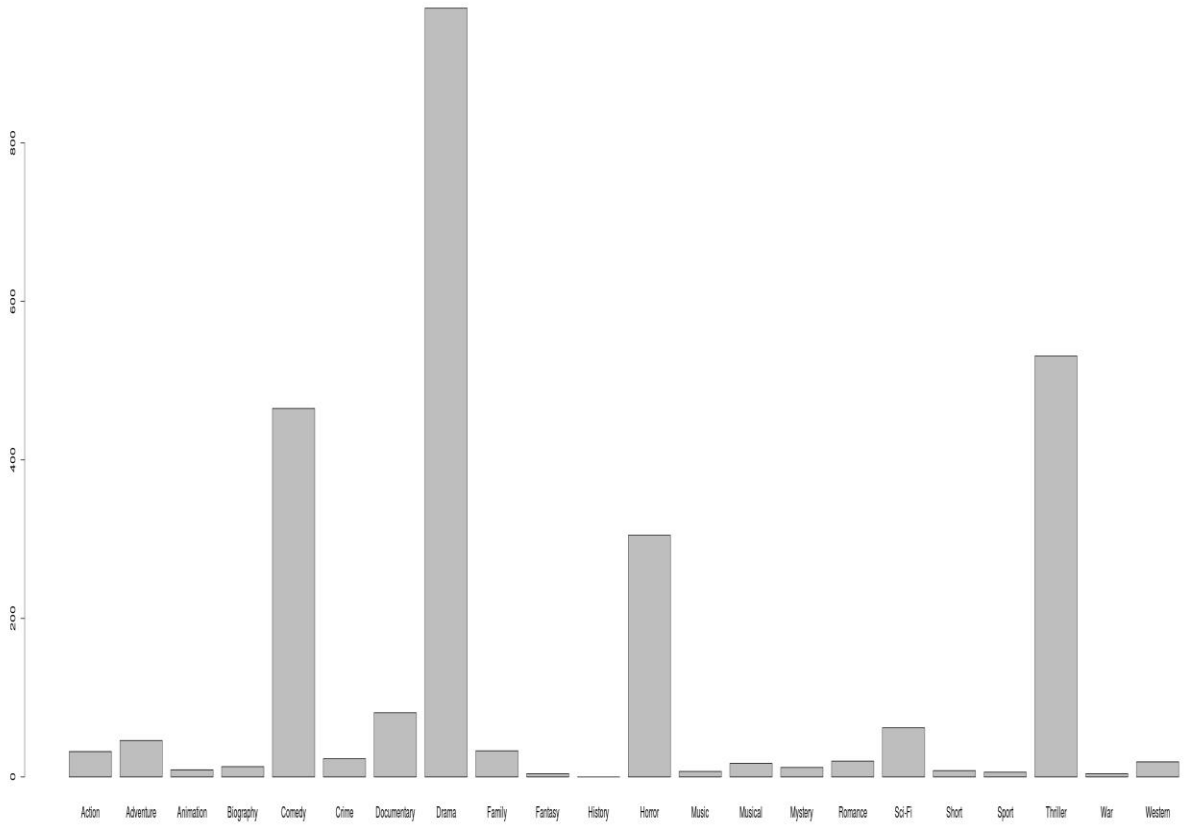
We found that clustering movies into k' clusters where k'>k does not help in better categorization. When k' is greater than k by just a small margin, the categorization does not show considerable difference. But when a much higher value of k' is taken, the noise points are considered as separate clusters too which is undesirable.

Results:

Genre Prediction results:

Using Naïve Bayes:

Predictions Plot:



Prediction table:

	Action	Adventure	Animation	Biography	Comedy	Crime
Action	3	0	0	0	6	0
Adventure	1	2	0	0	6	0
Animation	0	1	0	0	3	0
Biography	0	0	0	0	0	0
Comedy	4	4	1	1	137	0
Crime	2	0	0	0	1	0
Documentary	0	1	0	0	12	0
Drama	12	11	1	0	126	3
Family	0	1	0	0	11	0
Fantasy	0	0	0	0	0	0
History	0	0	0	0	0	0
Horror	3	3	0	1	42	4
Music	0	1	0	0	0	0
Musical	0	0	1	0	5	0
Mystery	0	0	0	0	3	0
Romance	0	0	0	0	5	0
Sci-Fi	0	3	0	0	12	1
Short	0	0	0	0	1	0
Sport	0	0	0	0	2	0
Thriller	5	8	0	0	74	3
War	0	0	0	0	0	0
Western	0	0	0	0	6	0

	Documentary	Drama	Family	Fantasy	History	Horror
Action	0	14	0	0	0	9
Adventure	0	21	1	0	0	5
Animation	0	3	0	0	0	1
Biography	0	2	0	0	0	0
Comedy	14	190	9	1	0	18
Crime	0	5	0	0	0	7
Documentary	27	20	0	0	0	1
Drama	57	335	10	2	0	135
Family	0	9	6	0	1	1
Fantasy	0	2	0	0	0	0
History	0	0	0	0	0	0
Horror	1	131	0	0	0	50
Music	0	0	0	0	0	0
Musical	1	5	1	0	0	0
Mystery	0	5	1	0	0	0
Romance	0	15	0	0	0	3
Sci-Fi	0	21	0	1	0	10
Short	0	1	0	0	0	0
Sport	0	0	0	0	0	1
Thriller	8	218	1	1	0	88
War	0	0	0	0	0	0
Western	0	4	0	0	0	2

	Music	Musical	Mystery	Romance	Sci-Fi	Short	Sport
Action	0	0	0	1	1	0	1
Adventure	0	0	0	0	0	0	0
Animation	0	0	0	0	0	1	0
Biography	0	0	0	0	0	0	0
Comedy	0	1	2	3	3	1	1

Crime	0	0	0	0	1	0	0
Documentary	0	0	0	0	3	0	0
Drama	0	0	3	3	17	4	1
Family	0	0	0	0	1	0	0
Fantasy	0	0	0	0	0	0	0
History	0	0	0	0	0	0	0
Horror	0	0	2	3	2	0	0
Music	0	0	0	0	0	0	0
Musical	0	0	0	0	1	0	0
Mystery	0	0	0	0	0	0	0
Romance	0	0	1	0	1	0	0
Sci-Fi	1	0	0	0	5	0	1
Short	0	0	0	0	0	1	0
Sport	0	0	0	0	0	0	0
Thriller	0	0	0	0	4	0	2
War	0	0	0	0	0	0	0
Western	0	0	0	1	0	0	0

	Thriller	War	Western
Action	8	0	1
Adventure	14	0	1
Animation	1	0	0
Biography	2	0	0
Comedy	61	0	0
Crime	9	0	1
Documentary	6	1	0
Drama	169	4	7
Family	2	0	0
Fantasy	1	0	0
History	0	0	0
Horror	93	1	0
Music	0	0	0
Musical	6	0	0
Mystery	3	0	0
Romance	4	0	0
Sci-Fi	17	1	2
Short	1	0	0
Sport	1	0	0
Thriller	163	0	0
War	0	0	0
Western	6	0	0

Confusion matrix:

Prediction	Reference Action	Adventure	Animation	Biography	Comedy	Crime
Action	3	0	0	0	6	0
Adventure	1	2	0	0	6	0
Animation	0	1	0	0	3	0
Biography	0	0	0	0	0	0
Comedy	4	4	1	1	137	0
Crime	2	0	0	0	1	0
Documentary	0	1	0	0	12	0
Drama	12	11	1	0	126	3
Family	0	1	0	0	11	0
Fantasy	0	0	0	0	0	0
History	0	0	0	0	0	0
Horror	3	3	0	1	42	4
Music	0	1	0	0	0	0
Musical	0	0	1	0	5	0
Mystery	0	0	0	0	3	0
Romance	0	0	0	0	5	0
Sci-Fi	0	3	0	0	12	1
Short	0	0	0	0	1	0
Sport	0	0	0	0	2	0
Thriller	5	8	0	0	74	3
War	0	0	0	0	0	0
Western	0	0	0	0	6	0

Prediction	Reference Documentary	Drama	Family	Fantasy	History	Horror
Action	0	14	0	0	0	9
Adventure	0	21	1	0	0	5
Animation	0	3	0	0	0	1
Biography	0	2	0	0	0	0
Comedy	14	190	9	1	0	18
Crime	0	5	0	0	0	7
Documentary	27	20	0	0	0	1
Drama	57	335	10	2	0	135
Family	0	9	6	0	1	1
Fantasy	0	2	0	0	0	0
History	0	0	0	0	0	0
Horror	1	131	0	0	0	50
Music	0	0	0	0	0	0
Musical	1	5	1	0	0	0
Mystery	0	5	1	0	0	0
Romance	0	15	0	0	0	3
Sci-Fi	0	21	0	1	0	10
Short	0	1	0	0	0	0
Sport	0	0	0	0	0	1
Thriller	8	218	1	1	0	88
War	0	0	0	0	0	0
Western	0	4	0	0	0	2

Prediction	Reference Music	Musical	Mystery	Romance	Sci-Fi	Short	Sport
Action	0	0	0	1	1	0	1
Adventure	0	0	0	0	0	0	0
Animation	0	0	0	0	0	1	0

```
  Biography        0       0       0       0       0      0      0
  Comedy           0       1       2       3       3      1      1
  Crime            0       0       0       0       1      0      0
  Documentary      0       0       0       0       3      0      0
  Drama            0       0       3       3      17      4      1
  Family           0       0       0       0       1      0      0
  Fantasy          0       0       0       0       0      0      0
  History          0       0       0       0       0      0      0
  Horror           0       0       2       3       2      0      0
  Music            0       0       0       0       0      0      0
  Musical          0       0       0       0       1      0      0
  Mystery          0       0       0       0       0      0      0
  Romance          0       0       1       0       1      0      0
  Sci-Fi           1       0       0       0       5      0      1
  Short            0       0       0       0       0      1      0
  Sport            0       0       0       0       0      0      0
  Thriller         0       0       0       0       4      0      2
  War              0       0       0       0       0      0      0
  Western          0       0       0       1       0      0      0
             Reference
Prediction    Thriller War Western
  Action             8   0       1
  Adventure         14   0       1
  Animation          1   0       0
  Biography          2   0       0
  Comedy            61   0       0
  Crime              9   0       1
  Documentary        6   1       0
  Drama            169   4       7
  Family             2   0       0
  Fantasy            1   0       0
  History            0   0       0
  Horror            93   1       0
  Music              0   0       0
  Musical            6   0       0
  Mystery            3   0       0
  Romance            4   0       0
  Sci-Fi            17   1       2
  Short              1   0       0
  Sport              1   0       0
  Thriller         163   0       0
  War                0   0       0
  Western            6   0       0
```

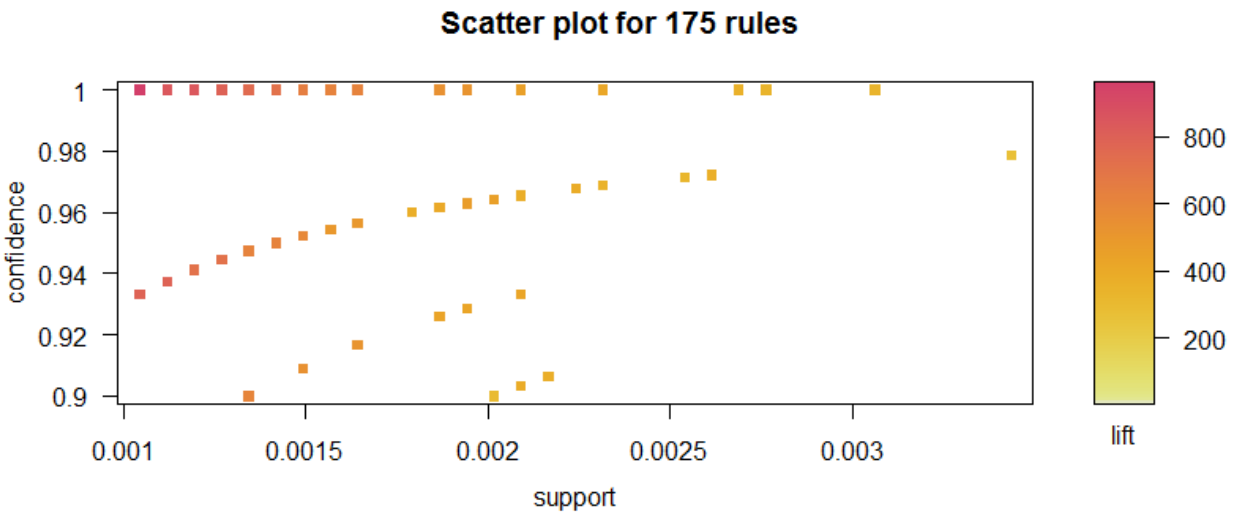
Overall Statistics

```
               Accuracy : 0.2733
                 95% CI : (0.2565, 0.2907)
    No Information Rate : 0.3753
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0696
 Mcnemar's Test P-Value : NA
```

The usual casts (using Apriori):

Rules Plot:



Final set of unique rules sorted based on Confidence:

```
     lhs                                  rhs
support     confidence lift

1   {Producer=Aylward, John}            => {Writer=Aylward, John}            0.001
045791 1.0000000  836.6875
2   {Producer=Allen, Tim}               => {Writer=Allen, Tim}               0.001
045791 1.0000000  956.2143
4   {Producer=Allen, Tim}               => {Director=Allen, Tim}             0.001
045791 1.0000000  892.4667
6   {Writer=Borghese, Paul}             => {Director=Borghese, Paul}         0.001
045791 1.0000000  787.4706
7   {Writer=Allen, Tim}                 => {Director=Allen, Tim}             0.001
045791 1.0000000  892.4667
9   {Actor=Alazraqui, Carlos}           => {Producer=Alazraqui, Carlos}      0.001
045791 1.0000000  743.7222
10  {Actor=Alazraqui, Carlos}           => {Director=Alazraqui, Carlos}      0.001
045791 1.0000000  743.7222
11  {Actor=Alazraqui, Carlos}           => {Writer=Alazraqui, Carlos}        0.001
045791 1.0000000  743.7222
```

```
12  {Producer=Armisen, Fred}            => {Director=Armisen, Fred}            0.001
045791 1.0000000  956.2143
14  {Producer=Armisen, Fred}            => {Writer=Armisen, Fred}              0.001
045791 1.0000000  956.2143
16  {Director=Armisen, Fred}            => {Writer=Armisen, Fred}              0.001
045791 1.0000000  956.2143
18  {Actor=Angarano, Michael}           => {Producer=Angarano, Michael}        0.001
045791 1.0000000  669.3500
19  {Actor=Angarano, Michael}           => {Director=Angarano, Michael}        0.001
045791 1.0000000  608.5000
20  {Actor=Angarano, Michael}           => {Writer=Angarano, Michael}          0.001
045791 1.0000000  582.0435
21  {Producer=Bernsen, Corbin}          => {Writer=Bernsen, Corbin}            0.001
045791 1.0000000  787.4706
22  {Producer=Bernsen, Corbin}          => {Director=Bernsen, Corbin}          0.001
045791 1.0000000  637.4762
23  {Actor=Baldwin, Stephen}            => {Producer=Baldwin, Stephen}         0.001
045791 1.0000000  514.8846
24  {Actor=Baldwin, Stephen}            => {Director=Baldwin, Stephen}         0.001
045791 1.0000000  418.3438
25  {Writer=Andreiu, Dan}               => {Producer=Andreiu, Dan}             0.001
120490 1.0000000  836.6875
27  {Writer=Andreiu, Dan}               => {Director=Andreiu, Dan}             0.001
120490 1.0000000  836.6875
```

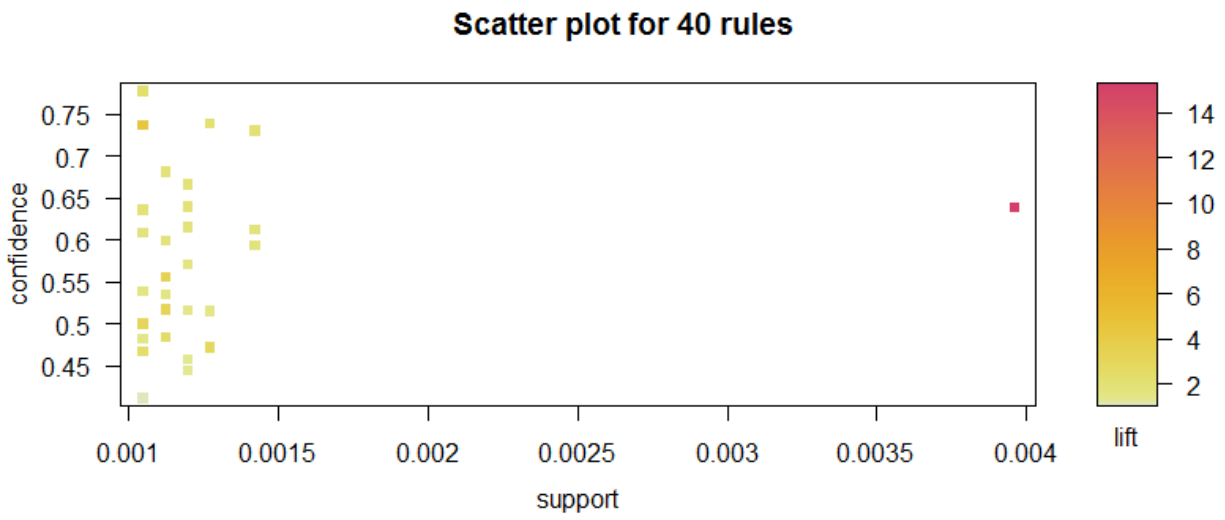
Final set of unique rules sorted based on Support:

```
lhs                                     rhs
support     confidence lift
1   {Producer=Aylward, John}             => {Writer=Aylward, John}
0.001045791 1.0000000  836.6875
2   {Producer=Allen, Tim}                => {Writer=Allen, Tim}
0.001045791 1.0000000  956.2143
4   {Producer=Allen, Tim}                => {Director=Allen, Tim}
0.001045791 1.0000000  892.4667
6   {Writer=Borghese, Paul}              => {Director=Borghese, Paul}
0.001045791 1.0000000  787.4706
7   {Writer=Allen, Tim}                  => {Director=Allen, Tim}
0.001045791 1.0000000  892.4667
9   {Actor=Alazraqui, Carlos}            => {Producer=Alazraqui, Carlos}
0.001045791 1.0000000  743.7222
10  {Actor=Alazraqui, Carlos}            => {Director=Alazraqui, Carlos}
0.001045791 1.0000000  743.7222
11  {Actor=Alazraqui, Carlos}            => {Writer=Alazraqui, Carlos}
0.001045791 1.0000000  743.7222
12  {Producer=Armisen, Fred}             => {Director=Armisen, Fred}
0.001045791 1.0000000  956.2143
14  {Producer=Armisen, Fred}             => {Writer=Armisen, Fred}
0.001045791 1.0000000  956.2143
16  {Director=Armisen, Fred}             => {Writer=Armisen, Fred}
0.001045791 1.0000000  956.2143
18  {Actor=Angarano, Michael}            => {Producer=Angarano, Michael}
0.001045791 1.0000000  669.3500
19  {Actor=Angarano, Michael}            => {Director=Angarano, Michael}
0.001045791 1.0000000  608.5000
```

```
20  {Actor=Angarano, Michael}            => {Writer=Angarano, Michael}
0.001045791 1.0000000  582.0435
21  {Producer=Bernsen, Corbin}           => {Writer=Bernsen, Corbin}
0.001045791 1.0000000  787.4706
22  {Producer=Bernsen, Corbin}           => {Director=Bernsen, Corbin}
0.001045791 1.0000000  637.4762
23  {Actor=Baldwin, Stephen}             => {Producer=Baldwin, Stephen}
0.001045791 1.0000000  514.8846
24  {Actor=Baldwin, Stephen}             => {Director=Baldwin, Stephen}
0.001045791 1.0000000  418.3438
25  {Writer=Andreiu, Dan}                => {Producer=Andreiu, Dan}
0.001120490 1.0000000  836.6875
27  {Writer=Andreiu, Dan}                => {Director=Andreiu, Dan}
0.001120490 1.0000000  836.6875
```

Relation between genre and cast using association rule mining:

Plot for genre prediction based on cast:



Final set of unique rules sorted based on confidence:

```
lhs                              rhs                support     confidence li
ft
135 {Producer=Bower, Tom}         => {Genre=Drama}       0.001045791 0.7777778
2.093307
278 {Director=Basile, Joseph}     => {Genre=Drama}       0.001269889 0.7391304
1.989292
176 {Producer=Basile, Joseph}     => {Genre=Drama}       0.001045791 0.7368421
1.983133
```

```
179 {Writer=Basile, Joseph}      => {Genre=Drama}       0.001045791 0.7368421
1.983133
180 {Producer=Allen, Woody}     => {Genre=Comedy}      0.001045791 0.7368421
4.305589
181 {Writer=Arnett, Will}       => {Genre=Comedy}      0.001045791 0.7368421
4.305589
324 {Actress=Adams, Jane}       => {Genre=Drama}       0.001419287 0.7307692
1.966789
248 {Actress=Allen, Joan}       => {Genre=Drama}       0.001120490 0.6818182
1.835042
284 {Writer=Buscemi, Steve}     => {Genre=Drama}       0.001195189 0.6666667
1.794264
287 {Producer=Bale, Christian}  => {Genre=Drama}       0.001195189 0.6666667
1.794264
292 {Writer=Arkin, Alan}        => {Genre=Drama}       0.001195189 0.6400000
1.722493
420 {Actress=\\N}               => {Genre=Documentary} 0.003959065 0.6385542
15.210543
271 {Director=Angarano, Michael} => {Genre=Drama}      0.001045791 0.6363636
1.712706
314 {Writer=Aiello, Danny}      => {Genre=Drama}       0.001195189 0.6153846
1.656243
323 {Producer=Arkin, Alan}      => {Genre=Drama}       0.001195189 0.6153846
1.656243
325 {Producer=Aiello, Danny}    => {Genre=Drama}       0.001195189 0.6153846
1.656243
360 {Director=Buscemi, Steve}   => {Genre=Drama}       0.001419287 0.6129032
1.649565
279 {Writer=Angarano, Michael}  => {Genre=Drama}       0.001045791 0.6086957
1.638241
311 {Writer=Broadbent, Jim}     => {Genre=Drama}       0.001120490 0.6000000
1.614837
374 {Producer=Atkins, Lasco}    => {Genre=Drama}       0.001419287 0.5937500
1.598016
```

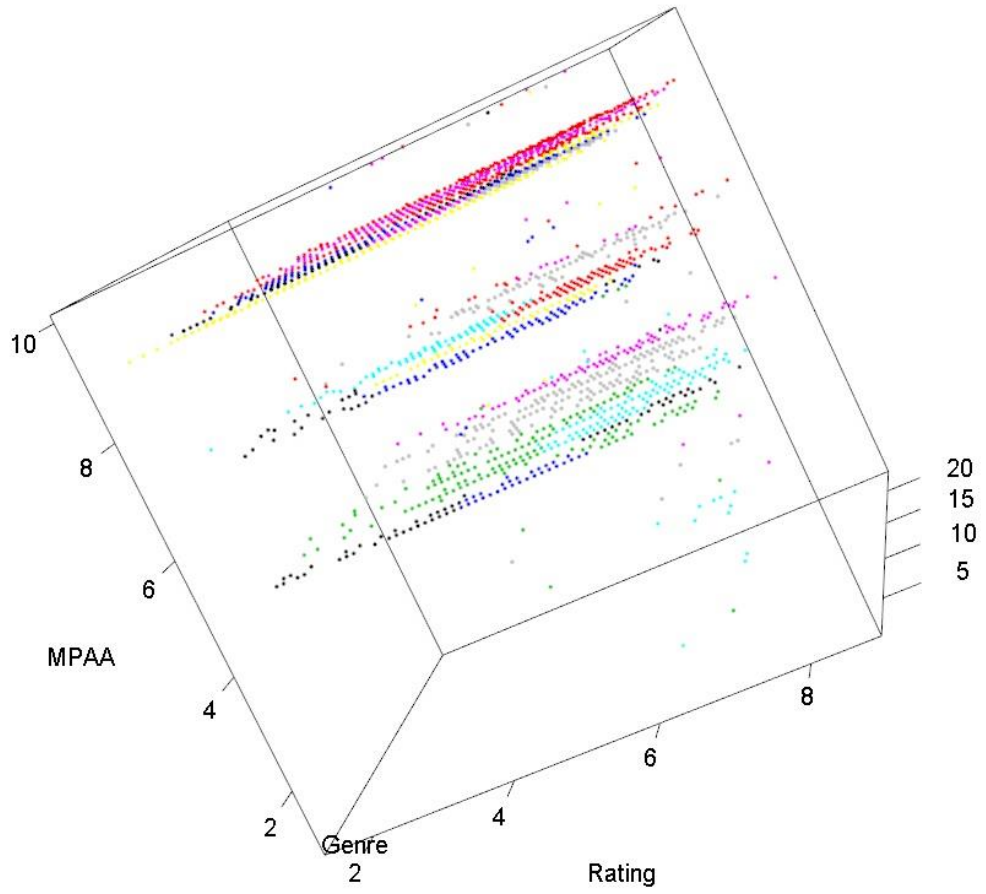
Final set of unique rules sorted based on support:

```
lhs                            rhs                    support     confidence li
ft
420 {Actress=\\N}               => {Genre=Documentary} 0.003959065 0.6385542
15.210543
324 {Actress=Adams, Jane}       => {Genre=Drama}       0.001419287 0.7307692
1.966789
360 {Director=Buscemi, Steve}   => {Genre=Drama}       0.001419287 0.6129032
1.649565
374 {Producer=Atkins, Lasco}    => {Genre=Drama}       0.001419287 0.5937500
1.598016
278 {Director=Basile, Joseph}   => {Genre=Drama}       0.001269889 0.7391304
1.989292
377 {Director=Atkins, Lasco}    => {Genre=Drama}       0.001269889 0.5151515
1.386476
386 {Actress=Aaron, Caroline}   => {Genre=Comedy}      0.001269889 0.4722222
2.759336
284 {Writer=Buscemi, Steve}     => {Genre=Drama}       0.001195189 0.6666667
1.794264
```

```
287 {Producer=Bale, Christian}   => {Genre=Drama}       0.001195189 0.6666667
1.794264
292 {Writer=Arkin, Alan}         => {Genre=Drama}       0.001195189 0.6400000
1.722493
314 {Writer=Aiello, Danny}       => {Genre=Drama}       0.001195189 0.6153846
1.656243
323 {Producer=Arkin, Alan}       => {Genre=Drama}       0.001195189 0.6153846
1.656243
325 {Producer=Aiello, Danny}     => {Genre=Drama}       0.001195189 0.6153846
1.656243
335 {Director=Arkin, Alan}       => {Genre=Drama}       0.001195189 0.5714286
1.537940
340 {Producer=Broadbent, Jim}    => {Genre=Drama}       0.001195189 0.5714286
1.537940
361 {Director=Baker, Dylan}      => {Genre=Drama}       0.001195189 0.5161290
1.389107
381 {Director=Baldwin, Alec}     => {Genre=Drama}       0.001195189 0.4571429
1.230352
387 {Actress=Aaron, Caroline}    => {Genre=Drama}       0.001195189 0.4444444
1.196176
248 {Actress=Allen, Joan}        => {Genre=Drama}       0.001120490 0.6818182
1.835042
311 {Writer=Broadbent, Jim}      => {Genre=Drama}       0.001120490 0.6000000
1.614837
```

Similar Movies

3d plot of clusters:



The above 3d plot was obtained using the plot3d method from the 'rgl' package. We used the kmeans function of the 'fpc' package to construct the graph. The three axes are MPAA, genre and rating. The plot clearly shows multiple clusters for genres as identified.

Discussion:

With our classification approach, we've found that the attributes used with the Naïve-Bayes algorithm correctly predicts some genres over others. Generally, comedies, dramas, horrors, and thrillers were predicted most accurately. We find these results to be relatively accurate given our attributes under the assumption that these genres host cast members who typically reside within specific genres; for example, Michael Bay typically directs action movies, Bruce Campbell

typically acts in horrors, etc. Additionally, it appears that some genres (comedy, drama) are rated much differently than others – a good horror might get a rating of 4.0 on Netflix, but an average drama might have a rating of roughly 7.0.

Our association rule mining results are relatively unreliable for frequent cast rules. Given the dataset used and the approach made to the algorithm, our rules simply list people who play more than one role, for example if 'Producer=Affleck, Ben', 'Writer=Affleck, Ben', and 'Actor=Affleck, Ben' then the rule generated will be 'Director=Affleck, Ben'. This data does not give us the results we were looking for; we're aware of some frequent casts (any Adam Sandler movie contains the same actors, Johnny Depp and Helene Bonham Carter frequently work under Tim Burton, etc.), but no relations of that sort are present in our rules. However, by observing the generated rules for genre based on cast information our results are relatively accurate.

Clustering with MPAA, rating, and genre appears to be accurate as well given our assumptions of certain genres belonging primarily to certain MPAA ratings and user ratings, similar to our justification for the accuracy of our classification. Our data might be improved by plotting our three attributes against one another separately in two dimensions, but we thought that data could be assumed from our 3d plot.

Conclusion:

For each of our data mining methods, we chose our algorithms primarily due to their complexity; given our large dataset, it seemed impractical to sacrifice a significant amount of time for an insignificant change (improvement or not) in accuracy. Using these methods, we've found significant relations between attributes such as cast, rating, and genre.

Using Naïve-Bayes for classification, we were able to semi-accurately classify some genres over others. We suspect this is because those particular genres employ a lesser variety of cast members and are typically rated similarly. Our approach with association rule mining led to insignificant data in that the generated rules only exist for cast members who consistently work as multiple roles for a film. We also mined association rules for genres which are observably accurate. Finally our

approach to clustering using kmeans generates obvious clusters that lend to significant results relating genre, rating, and MPAA.

References:

1. **http://cs229.stanford.edu/proj2013/cocuzzowu-hitorflop.pdf**

2. **http://cs229.stanford.edu/proj2008/KammHuangSathi-TheNetflixChallenge.pdf**

3. **https://rpubs.com/arun_infy13/97529**

4. **http://stats.stackexchange.com/questions/58855/why-do-we-use-k-means-instead-of-other-algorithms**