Data Science and Economics

Universit`a degli Studi di Milano

Unsupervised learning

(Association rules – Apriori algorithm)

Student: Vojimir Ranitovic 963780

Email: vojimir.ranitovic@studenti.unimi.it

Module: Statistical Learning, Deep Learning and Artificial Intelligence

Abstract

In this project, the Apriori algorithm is used to find association rules in Netflix's movies database. The goal is to use them for future recommendation purposes. Apriori managed to find around 65 thousand rules in the dataset of 16 million different ratings and threshold set at 20% of transactions.

Introduction

Association Rules are used to discover relationships between features in a massive dataset by establishing rules that are based on how frequently the features occur together in the dataset. A famous example of association rules and data mining is Market basket analysis, as the name said it is usually used to analyze store transactions. By analyzing the transactions and finding frequent itemsets and rules, store managers can make new pricing and marketing decisions. Usually, there are trivial rules such as a well-known bread/milk combination that is often bought together, or some unexplainable rules that are not worth action. On the other side, unfortunately rarely, there are actionable rules like the famous example of beer/diapers. Association rules are not used only in retail, they could also be used in medicine, User experience (UX) design, and recommendation system in websites or platforms like Netflix etc. In this project, association rules will be used to find patterns in the Netflix movie database, with the goal to make appropriate recommendations to customer, based on choices made by previous customers.

Algorithm and dataset

In this project, the Apriori algorithm is used to find frequent itemsets and association rules. The Apriori algorithm operates on a straightforward premise. When the support value of an itemset exceeds a certain threshold, it will be considered a frequent itemset.

.

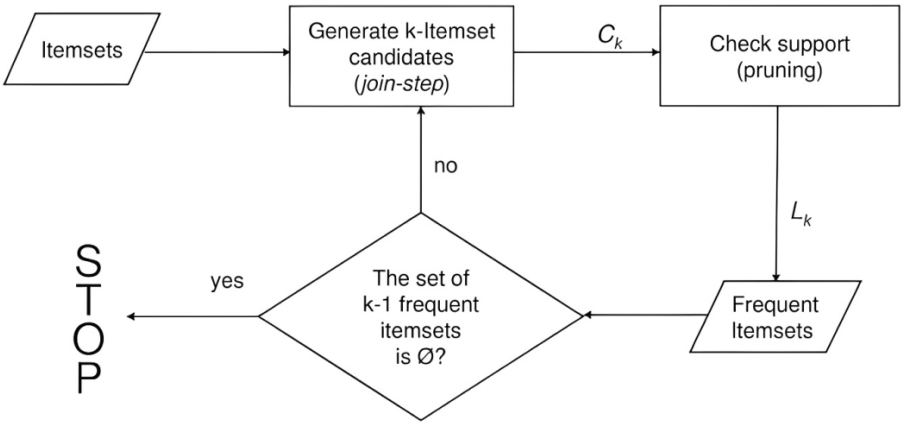

Figure 1: Flowchart of Apriori algorithm

In the picture above we can see how frequent itemsets are found. The procedure will stop if no itemset has a support level higher than a set threshold. An important property of itemsets is that if a set of items is frequent, then so are all its subsets. We exploit this property to eliminate the need to count certain itemsets by using its contrapositive: if an itemset is not frequent, then neither are its supersets. Some important evaluation metrics are support, confidence, and lift. Support measure is about how popular an itemset is, as measured by the proportion of transactions in which an itemset appears.

$$\text{Support}(\{X\} \rightarrow \{Y\}) = \frac{\text{Transactions containing both } X \text{ and } Y}{\text{Total number of transactions}}$$

Confidence measures how often items in Y appear in transactions that contain X. For example, that is to answer the question — of all the transactions containing {Lord of the Rings: The Fellowship of the Ring}, how many also had {Lord of the Rings: Two Towers} on them? Usually, confidence in this rule should be high, because people like also to watch the sequels of the movie (if they are fans).

$$\text{Confidence}\ (\{X\} \rightarrow \{Y\}) = \frac{\text{Transactions containing both } X \text{ and } Y}{\text{Transactions containing } X}$$

The lift measures how likely item Y is purchased when item X is purchased while controlling for how popular item Y is. A lift value greater than 1 means that item Y is likely to be bought if item X is bought, while a value less than 1 means that item Y is unlikely to be bought if item X is bought.

$$\text{Lift}(\{X\} \rightarrow \{Y\}) = \frac{(\text{Transactions containing both } X \text{ and } Y)/(\text{Transactions containing } X)}{\text{Fraction of transactions containing } Y}$$

The dataset used for analysis can be downloaded from Kaggle. It is about 1342 Netflix movies that are rated by 143.458 users, and there are over 17 million ratings. Users rated each movie with a mark from 1 to 5. As it is shown in Figure 2., there are more scores that are 3 and above. This could be due fact that people are usually not interested in rating movies that they do not like.

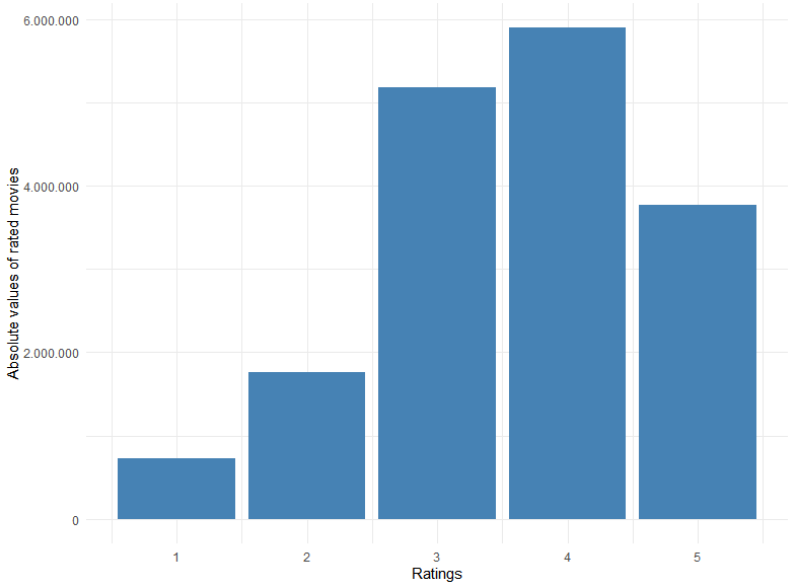

Figure 2: Absolute values of rated movies for each rating

There were movies with an average score of less than 3, that were deleted. I wanted to ensure that on average users will get recommended movies with scores of 3 (neutral) or above (very good or best rated). After removing these poorly rated movies, there were 1165 movies and 16 million reviews. In Figure 3., we can see the top 20 rated movies. This leads us to the conclusion that these most-rated movies would have high support and confidence in inspection association rules. It is not new to see something like this as popular movies would be ranked all the time with or without other popular/unpopular movies. Because of this, we introduced a lift measure to understand which movies are watched and ranked together excluding their popularity.

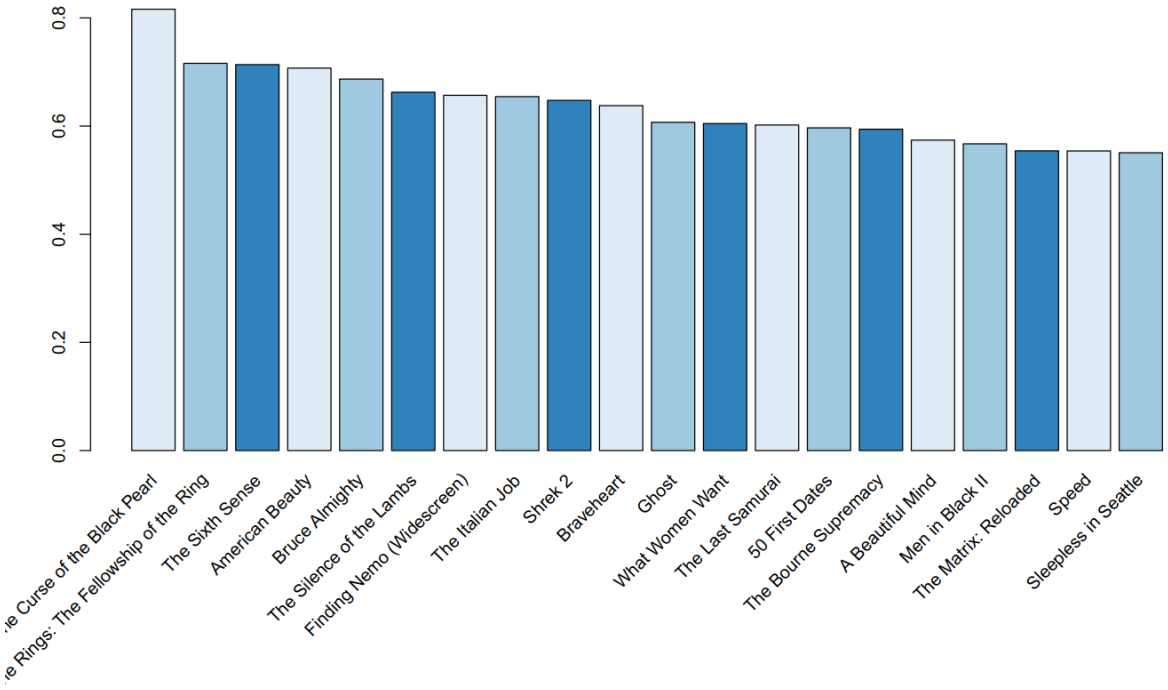

Figure 3: Relative Item Frequency Plot – Top 20 movies

To be able to use Apriori in R, the dataframe should be transformed into a suitable user-movie matrix. The columns will be all the movies, and the rows (transactions) would be users. This matrix has 143.458 rows and 1165 columns, and it is highly sparse.

Results and discussion

A threshold for an interesting frequency of rules is subjective, and depends a fair bit on the purpose. Often we will pick a support threshold that leaves us with a manageable number of rules to achieve that purpose. However, support is not the only measure of 'interestingness' available, and using another measure as well as support may allow us to have a lower support threshold without having too many rules to deal with. In R and "arules" library, we can set minimum support and confidence for Apriori algorithm. As it is expected if we increase minimum support the number of rules will drop down, and also the same happens when confidence value is increasing.

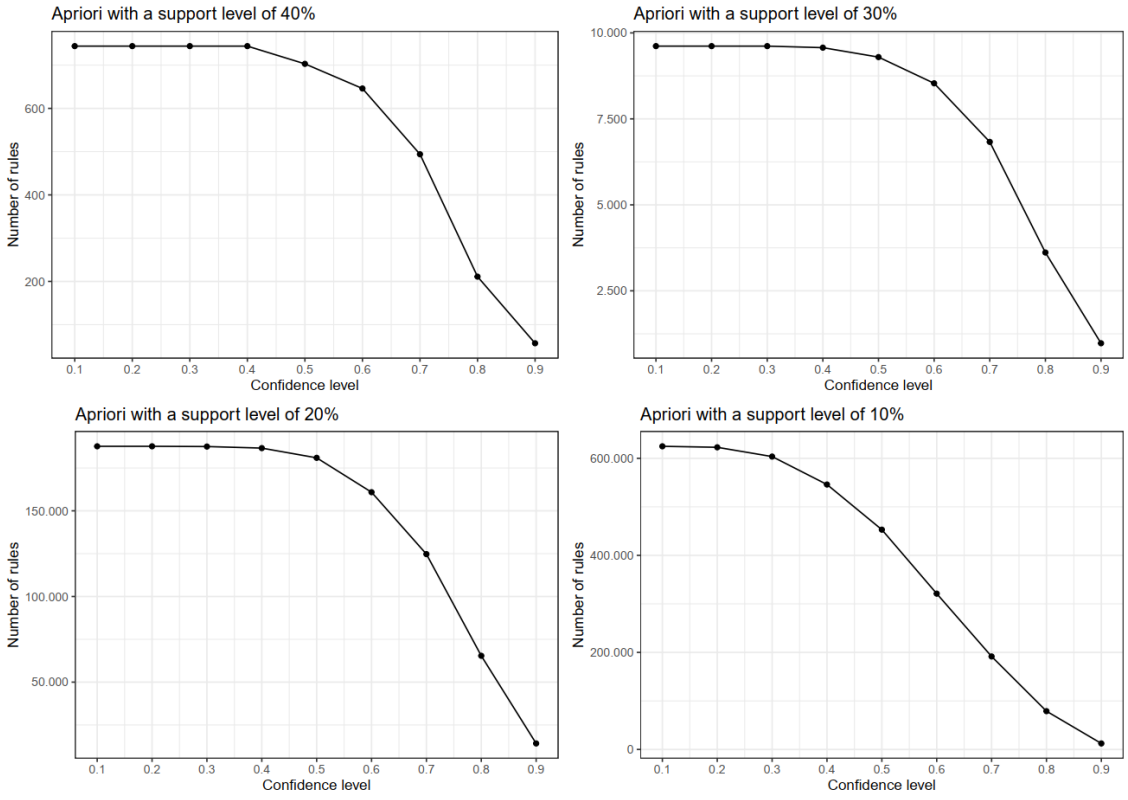

Figure 4: Different levels of minimum support and confidence level – effect on number of rules

On Figure 4., we can see four different minimum supports and different confidence levels. If we, for this dataset, choose a support level of 10%, we will have a high amount of rules for all confidence levels, and consequently a lot of them as useless one. On the other side, if we choose the support of 40%, there are a few rules and probably we will miss the interesting ones. For further analysis, the

threshold of 20% support and 80% of confidence level is used. This setup gives us 65 thousand rules. After filtering them by the highest confidence measure, we can see in Table 1., that "*Pirates of the Caribbean: The Cure of the Black Pearl"* is recommended all the time in the first 10 instances. This is because "*Pirates of the Caribbean: The Cure of the Black Pearl"* is the most-rated movie.

Table 1: Association rules sorted according highest confidence

Rules	Support	Confidence	Lift
{Finding Neverland,Lord of the Rings: The Fellowship of the Ring,X2: X-Men United} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2072453	0.96491627	1.1823614
{Hook,Shrek 2,X2: X-Men United} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2026238	0.96392094	1.1811418
{Finding Nemo (Widescreen),The Village,X2: X-Men United} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2027911	0.96327936	1.1803556
{Finding Nemo (Widescreen),Hook,The Last Samurai} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2087998	0.96324404	1.1803123
{Signs,The Last Samurai,X2: X-Men United} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2407325	0.96321191	1.1802729
{Road to Perdition,Shrek 2,X2: X-Men United} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2274603	0.96304932	1.1800737
{Finding Nemo (Widescreen),Road to Perdition,X2: X-Men United} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2312733	0.96282539	1.1797993
{Shrek 2,Signs,X2: X-Men United} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2468736	0.96260057	1.1795238
{Finding Nemo (Widescreen),The Last Samurai,X2: X-Men United} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2782696	0.96239151	1.1792677
{Finding Nemo (Widescreen),Hook,X2: X-Men United} => {Pirates of the Caribbean: The Curse of the Black Pearl}	0.2092808	0.96187486	1.1786346

As we said, a better measure, that will neutralize the effect of popularity is lift. In figure 5 we can see 10 rules with the highest lift measure.

Table 2: Association rules sorted according highest lift

Rules	Support	Confidence	Lift
{Bad Boys II,Pirates of the Caribbean: The Curse of the Black Pearl,The Recruit} => {S.W.A.T.}	0.2053702	0.8699067	1.8276691
{Bad Boys II,Man on Fire,The Italian Job} => {S.W.A.T.}	0.2035718	0.85975035	1.8063307
{Bad Boys II,The Recruit} => {S.W.A.T.}	0.221835	0.85957378	1.8059597
{Bad Boys II,Bruce Almighty,Man on Fire} => {S.W.A.T.}	0.2053772	0.85013129	1.7861211
{Bad Boys II,The Italian Job,The Last Samurai} => {S.W.A.T.}	0.204715	0.84832029	1.7823162
{Bad Boys II,Man on Fire,Pirates of the Caribbean: The Curse of the Black Pearl} => {S.W.A.T.}	0.2130728	0.84418239	1.7736225
{Bad Boys II,The Sum of All Fears} => {S.W.A.T.}	0.2148643	0.84200175	1.769041
{Bad Boys II,Bruce Almighty,The Last Samurai} => {S.W.A.T.}	0.2055584	0.83854181	1.7617717
{Bad Boys II,Bruce Almighty,The Italian Job} => {S.W.A.T.}	0.22302	0.83496007	1.7542465
{Bad Boys II,Man on Fire} => {S.W.A.T.}	0.2318867	0.83219092	1.7484285

In all these rules, there is Bad Boys II, so people that watched/ranked that movie is likely to watch S.W.A.T. Another interesting way to show these associations is in Figure 5. For example, if someone ranked *"The Italian Job"*, *"Man on Fire"* and also *"Bad Boys II"* he is likely to rank also the *"S.W.A.T"*. Also worth mentioning that here we also can see that *"Bad Boys II"* is in every rule, it is a joint dot on graph for all other movies.

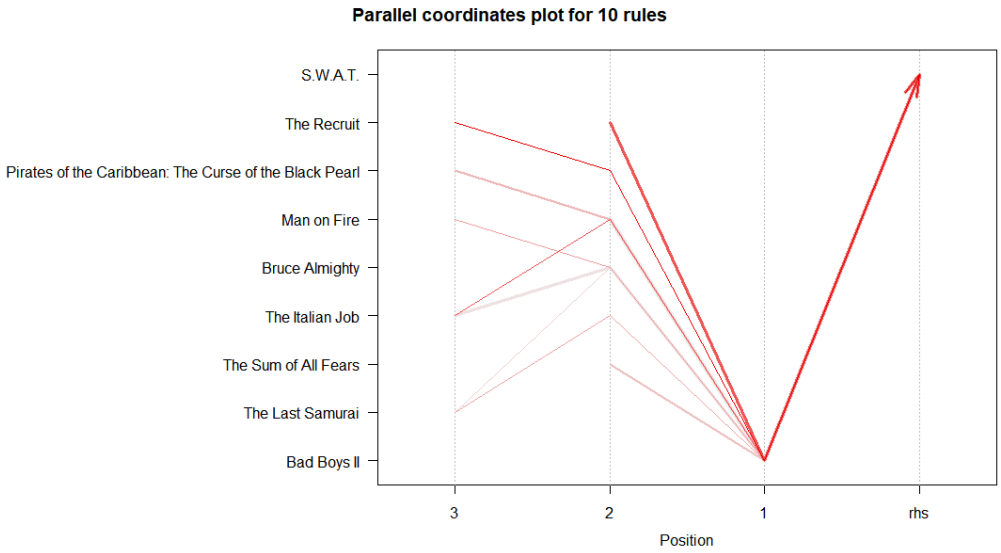

Figure 5: Another way of presenting association rules – Paracoord method

Another interesting opportunity is to answer the question to whom to recommend some movie. For example, we can investigate which movies users watched if our goal is to recommend *"Lord of the Rings"*.

Table 3: Association rules sorted according highest lift – "Lord of the Rings" as recommended movie

Rules	Support	Confidence	Lift
{Braveheart,Signs,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.2418199	0.941973498	1.3155405
{A Beautiful Mind,Signs,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.2179105	0.940265287	1.3131548
{Signs,The Mummy,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.2110862	0.938831189	1.311152
{Braveheart,Road to Perdition,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.2245884	0.937934849	1.3099002
{A Beautiful Mind,The Matrix: Reloaded,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.240377	0.937065217	1.3086857
{Signs,The Matrix: Reloaded,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.2567372	0.936913332	1.3084736
{Signs,The Silence of the Lambs,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.2415968	0.936501932	1.307899
{Braveheart,The Matrix: Reloaded,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.2773843	0.936151693	1.3074099
{Finding Nemo (Widescreen),Signs,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.2503172	0.936131387	1.3073815
{Lethal Weapon,Signs,X2: X-Men United} => {Lord of the Rings: The Fellowship of the Ring}	0.2096363	0.936097364	1.307334

Table 3., shows that users who watched *"Braveheart"*, *"X-Men"*, *"The Mummy"*, *"The Matrix"* etc. (as we can see this is some mixture of science fiction and historical fiction), would probably like *"Lord of the Rings"*. Also, one graphical way to represent similar movies that are watched and ranked together is the dendrogram. Here is used Jaccard similarity as a measure of the similarity of movies.

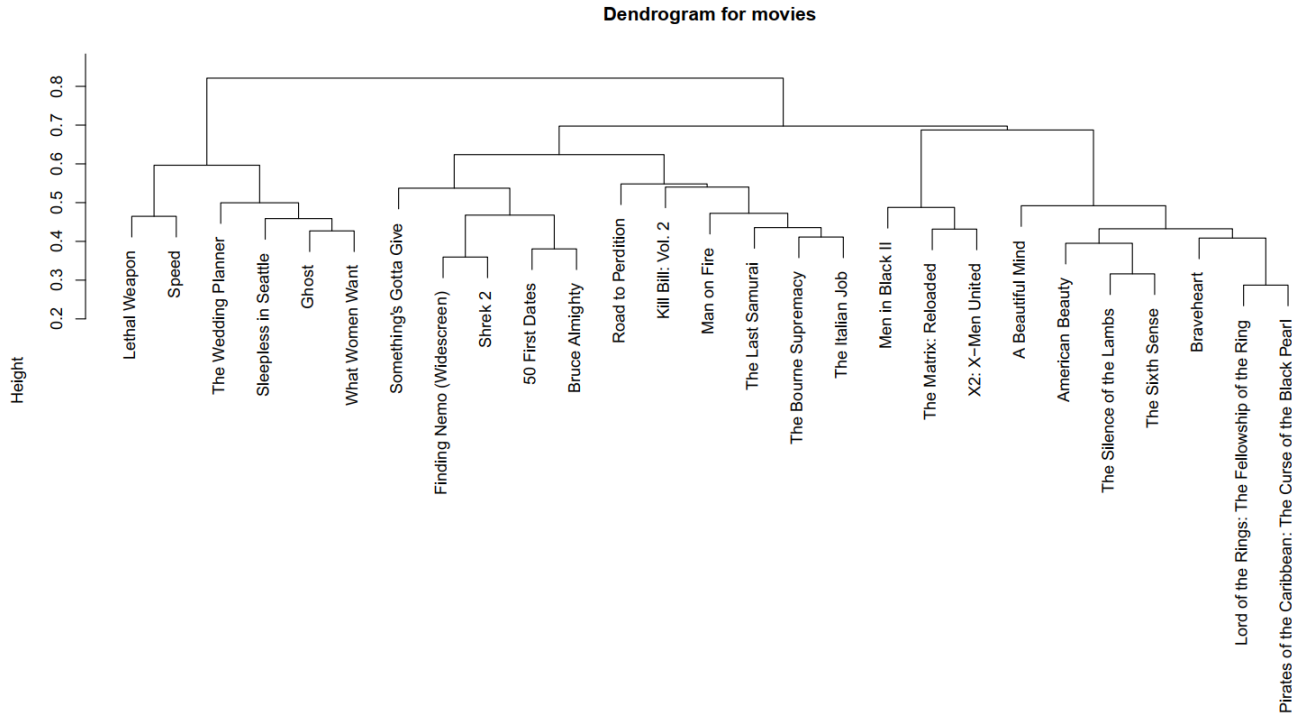

Figure 6: Dendrogram for movies – Jaccard similarity

As we can see from the dendrogram, some movies are more similar as *"Lord of the rings"* and *"Pirates of the Caribbean"* (adventure movies), or *"Man in Black"*, *"The Matrix"* and *"X-Men"* (mostly super-human, futuristic-action movies), or *"Shrek"* and *"Finding Nemo"* (cartoons)

Conclusion

As we could see, this is one way to mine the data, find association rules, and recommend movies that will probably be likable to users. This can be a way to show people on Netflix or any other movie platform that according to your history à "people also watch this movie". In the end, support was at 20% and confidence at 80%, as it is the middle ground between execution time and the number of rules, but we could choose some other values according to our goal. The dendrogram showed that these movies could be clustered and it looks like sorting according to the genre. Despite it is easy to understand and implement the Apriori algorithm, it is slow and memory storage could also be a problem (especially for low support values and a massive database). Solutions for that can be sampling, usage of different "state of the art" algorithms, or parallel programming. In future work, this slowness and inefficiency of Apriori could also be lowered or overcome by clustering the movies and then implementing the algorithm for each of the clusters.