Frequent Patterns

1. Frequent patterns: getting started

   (a) Get ready to use the same dataset as for the recommendation exercises: `http://cazabetremy.fr/Teaching/DSIA/ratings_clean_names.csv`

   (b) Pivot the table to have a dataframe such as users correspond to ”transactions” (rows), and movies correspond to items (columns)

   (c) Transform the datatable into a binary dataset such as 0 corresponds to not-watched, and 1 correspond to watched. You can use something like `pivoted=pivoted.notnull().astype('int')`. But it can be better to filter out low scores first, such as 1 means ”users liked movie”.

   (d) We will use library `mlxtend` to extract the frequent patterns. Use `mlxtend.frequent_patterns.apriori` to discover frequent itemsets. For instance, choose `min_support=0.05`, `max_len=3`, and the `use_colnames=True` option to get easily understandable results fast. (It takes less than a minute to compute).

   (e) Sort values by support to observe the most frequent itemsets.

   You can use `pd.set_option('display.max_colwidth', None)` to force displaying the columns in full.

   (f) Use `df['itemsets'].apply(lambda x: len(x))` to count the size of the itemsets

   (g) You can now filter to show the most frequent 2-itemsets.

2. Association rules

   (a) Use `mlxtend.frequent_patterns.association_rules` to compute the typical scores on frequent patterns

   (b) You can now sort association rules by their `lift`. Interpret the observations.

   (c) You can search for the *antecedents* that explain the best a given *consequents*. Be careful, objects inside columns are `frozenset`

   (d) Explore the results with different thresholds for support, maximum length, and association rule scores to find relevant explanations on why some people liking some movies tend to like some other movies.

1 Going further

3. From frequent patterns to graphs

   (a) We can use frequent pattern information to build a graph: using your favorite associate rule metric, you can filter only the most relevant rule of the form $movie1 \rightarrow movie2$, and represent them by an edge between the two corresponding nodes in a graph.

   (b) You can compare the results with those obtained in the previous class using distance scores.