Association mining with MATLAB

Objective: The objective of this exercise is to understand association mining, how frequent itemsets can be extracted by the Apriori algorithm and be able to calculate and interpret association rules in terms of support and confidence.

Material: Lecture notes *"Introduction to Machine Learning and Data Mining"* as well as the files in the exercise 12 folder available from Campusnet.

Part 1: Group discussion (max 15 min)

For the group discussion, each group should have selected a *discussion leader* at the previous exercise session. The purpose of the discussion leader is to ensure all team members understands the answers to the following two questions:

Multiple-Choice question: Solve and discuss **problem 19.1** from chapter 19 of the lecture notes. Ensure all group members understand the reason why one of the options is true and why the other options can be ruled out. (After today's exercises make sure to complete the remaining multiple-choice problems listed as part of the preparation for week 12 on the course homepage).

Discussion question: Discuss the following question in the group

- Consider the market-basket problem with 10 customers and 10 items from the slides. There are 1024 possible itemsets, why could we so quickly rule out most of them? Re-do the solution in the group. Can you find an association rule of the form $\{\text{beer}\} \rightarrow ?$ with confidence greater than .5?

Part 2: Programming exercises

Piazza discussion forum: You can get help by asking questions on Piazza: https://piazza.com/dtu.dk/fall2018/02450

Software installation: Extract the Matlab toolbox from DTU Inside. Start Matlab and go to the `<base-dir>/02450Toolbox_Matlab/` directory using the command `cd('<base-dir>/02450Toolbox_Matlab/')` and run `setup.m`. Remember the purpose of the exercises is not to re-write the code from scratch but to work with the scripts provided in the directory `<base-dir>/02450Toolbox_Matlab/Scripts/`

Representation of data in Matlab:

	Matlab var.	Type	Size	Description
	`X`	Numeric	$N \times M$	Data matrix: The rows correspond to N data objects, each of which contains M attributes.
	`attributeNames`	Cell array	$M \times 1$	Attribute names: Name (string) for each of the M attributes.
	`N`	Numeric	Scalar	Number of data objects.
	`M`	Numeric	Scalar	Number of attributes.
Regression	`y`	Numeric	$N \times 1$	Dependent variable (output): For each data object, `y` contains an output value that we wish to predict.
Classification	`y`	Numeric	$N \times 1$	Class index: For each data object, `y` contains a class index, $\texttt{y}_n \in \{0, 1, \ldots, C-1\}$, where C is the total number of classes.
	`classNames`	Cell array	$C \times 1$	Class names: Name (string) for each of the C classes.
	`C`	Numeric	Scalar	Number of classes.
Cross-validation				All variables mentioned above appended with `_train` or `_test` represent the corresponding variable for the training or test set.
	`⋆_train`	—	—	Training data.
	`⋆_test`	—	—	Test data.

12.1 Association Analysis

In this last exercise we will focus on association analysis. Association analysis is widely used in data mining in order to identify important co-occurrence relationships. We will use the following definition of association rule discovery:

> **Association Rule Discovery.** Given a set of transactions T, find all the rules having support $\geq minsup$ and confidence $\geq minconf$, where $minsup$ and $minconf$ are the corresponding support and confidence thresholds.

We have summarized the most important terms in table 1. We will use the Apriori algorithm to find all itemsets with support greater than $\geq$ minsup. The Apriori algorithm is based on the following principle:

> **Apriori principle.** If an itemset is frequent, then all of its subsets must also be frequent.

As a result of the Apriori principle we can start looking at frequent 1-itemsets. The frequent 2-itemsets can then only contain the items in the extracted 1-itemsets and so on and so forth. This greatly reduces the number of itemsets to check to find all frequent itemsets.

Term	**Meaning**
$I = \{i_1, i_2, \ldots, i_d\}$	The set of all items
$T = \{t_1, t_2, \ldots, t_N\}$	The set of all transactions
Transaction, t_i	A subset of items: What was bought by a customer
Transaction width	Number of items in transaction
Itemset	A set of items from the set I of all items
k-itemset	An itemset having k items
Support count, $\sigma(X)$	Number of transactions that contain a particular itemset $\sigma(X) = \lvert\{t\}\rvert$
Association rule	Implication expression of the form $X \leftarrow Y$ where $X \bigcap Y = \varnothing$
Support, $s(Y \leftarrow X)$	Strength of association rule, $s(Y \leftarrow X) = \frac{\sigma(X \bigcup Y)}{N} = P(X, Y)$
Confidence, $c(Y \leftarrow X)$	Frequency items in Y appear in transactions containing X, $c(Y \leftarrow X) = \frac{\sigma(X \bigcup Y)}{\sigma(X)} = \frac{P(X,Y)}{P(X)} = P(Y\lvert X)$
Support-based pruning	Pruning strategy based on the Apriori principle (formed by the anti-monotone property)
Anti-monotone property	The support for an itemset never exceeds the support for its subsets (Apriori principle)
F_k	The set of frequent k-itemsets

Table 1: Association mining nomenclature.

12.1.1 In table 2 some of the courses that 6 students completed during their studies are given. Find all itemsets with minsup $\geq 80\%$.

12.1.2 What is the confidence of the rule 02457 $\leftarrow$ 02450?

We will use the Apriori algorithm to automatically mine for associations[1]. We have included a publically available implementation in the toolbox, see `https://se.mathworks.com/matlabcentral/fileexchange/52867-apriori-algorithm-for-association-rule-`

[1] A high-performing version of the Apriori algorithm is also available from: `http://www.borgelt.net/apriori.html`.

	02322	02450	02451	02453	02454	02457	02459	02582
student 1	0	1	0	0	1	1	1	1
student 2	1	1	1	0	0	1	1	1
student 3	0	1	0	1	0	1	0	1
student 4	0	0	1	0	0	1	1	0
student 5	0	1	0	0	0	1	1	0
student 6	0	1	1	0	0	1	1	1

Table 2: Students that upon completing their engineering degree had taken various of the courses 02322, 02450, 02451, 02453, 02454, 02457, 02459 and 02582.

12.1.3 Inspect the file `Data/courses.txt` and the script `ex12_1_3.m`. The script loads the course data file from table 2. Make sure you understand how the data in table 2 is stored in the file and how the script transforms it

12.1.4 Inspect and run the script `ex12_1_4.m`. The script transforms the binary matrix, plus label information, into a set of transactions. Make sure you understand how this transformation performs and relate it to the notation of the lecture notes. Finally note how the Apriori algorithm is invoked to find association rules with minsup $\geq 80\%$ and $\geq 100\%$ and print them. Inspect the print command to see how you can access each association rule programmatically. What are the generated association rules?

We will in this last part of the exercise mine for associations in the wine data [1](`http://archive.ics.uci.edu/ml/datasets/Wine+Quality`) considered in the previous exercises. However, as this data is not binary we will need to convert it to a format suitable for association mining. We will thus binarize the data by dividing each attribute into given percentiles.

12.1.5 Inspect and run the script `ex12_1_5.m`. The script load the `Data/wine2.mat` data into Matlab and divide each of the attributes in the data into percentiles using the function `binarize`.

The scripts convert the continuous attributes into a one-out-of-K coding based on percentiles. Note how the function also transforms the attribute names. Why do you think we don't just include the 50-100 percentiles of a variable? What are the benefits of including variables corresponding to the 0-50 percentile?

12.1.6 Inspect and run the script `ex12_1_6.m` to find association rules in the Wine dataset with minsup $\geq 30\%$ and $\geq 60\%$. Do these association rules make sense?

12.1.7 Often we are interested in rules with high confidence. Is it possible for itemsets to have very low support but still have a very high confidence?

12.1.8 (optional) Try find associations also in terms of the type of wine by adding two additional columns to the binary data corresponding to `1-y` and `y` (Note this is easiest done by adding new columns to the X-matrix and changing the `attributeNames` variable.)

12.2 Tasks for the report

12.3 Work on report three

For the remainder of this exercise work on report three due next week. In the association-mining section of the report investigate how attributes associate based on mining for association using the Apriori algorithm. Use the `binarize2` function as in the above scripts to convert your dataset to an appropriate format and play around with the and minsup variables to get a reasonable small, and therefore hopefully also interpretable, set of association rules. Include these rules in the report along with your interpretation in a reasonably easy-to-read format.

References

[1] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. Modeling wine preferences by data mining from physicochemical properties. *In Decision Support Systems, Elsevier*, 47(4):547–553, 2009.