Lab 3: Association Rules Mining

Due date: Tuesday, April 24, midnight.

Lab Assignment

In this assignment you will analyze a collection of market baskets and will determine *frequent itemsets* and *association rules* present in the collection.

Assignment Preparation

This is a pair programming assignment. Each student teams up with a partner. You get to select your partner at the beginning of the Tuesday, April 17 lab.

Main Dataset

The assignment is based on the `Extended BAKERY` dataset. The dataset is a modified version of the CSC 365 `BAKERY` dataset. The `Extended BAKERY` dataset describes the work of a *chain* of bakery shops that sell a variety of pastries and drinks to customers.

The data provided to you for this assignment is the information about purchases made by the bakery chain customers in various locations. The four sub-datasets contain information about 1000, 5000, 20,000 and 75,000 purchases.

For each sub-dataset we provide three files representing the same set of receipts. For simplicity, each file represents the exact purchases: i.e., which items were purchased on which receipt, but **omits other information from the dataset:** the store location, the employee who rang the purchase, the date of the purchase. Additionally, the *quantity* of the purchased item is omitted in two representations of the three listed.

The full description of the dataset is below.

Access to the dataset. All CSV files can be downloaded from the datasets wiki:

http://wiki.csc.calpoly.edu/datasets/wiki

The wiki page for the EXTENDED BAKERY dataset is:

http://wiki.csc.calpoly.edu/datasets/wiki/ExtendedBakery

Individual .CSV files are available from wiki pages devoted to sub-datasets:

http://wiki.csc.calpoly.edu/datasets/wiki/ExtendedBakery1000
http://wiki.csc.calpoly.edu/datasets/wiki/ExtendedBakery5000
http://wiki.csc.calpoly.edu/datasets/wiki/ExtendedBakery20K
http://wiki.csc.calpoly.edu/datasets/wiki/ExtendedBakery75K

Additionally, the zip file containing all CSV files can be downloaded from

http://wiki.csc.calpoly.edu/datasets/wiki/apriori

The list of **market baskets** for each dataset size is available in **three formats**:

1. Sparse Vector format. Files `XXXX-out1.csv`. Each line of the file has the following format:
   - First column is the receipt Id.
   - Subsequent columns store a list of goods purchased from the bakery ordered by `Goods.Id`.

   Example:

   ```
   1, 7, 15, 44, 49
   ```

   Receipt 1 contained purchases of a Coffee Eclair, a Blackberry Tart, Bottled Water and a Single Espresso.

2. Full Binary Vector format. Files `XXXX-out2.csv`. Each line of the file has the following format:
   - First column is the receipt id.
   - 50 columns follow, with 0s or 1s as values. A 1 in column $i+1$ means that a good with `Goods.Id` of i was purchased on the receipt.

   Example:

   ```
   1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1
   ```

3. Items Table. Files `XXXXi.csv`. Each line of the file represents a single tuple from the `Items` table. The columns are:

   - Receipt(number), Quantity, Item

   Example:

   ```
   1,3,7
   1,4,15
   1,2,49
   1,5,44
   ```

   (note, that the item IDs may be out of order)

Additional Dataset

Your frequent itemset mining algorithm shall work with any market basket data provided to it in one of the three formats described above (full vector, sparse vector, or items table).

In addition to the `EXTENDED BAKERY` dataset, we will make one more dataset, called `TRANSCRIPTION FACTORS` available to you. The dataset is a collection of information about the participation of specific proteins (called *transcription factors*) in the experssion of certain genes in the mammary gland tissues of different animals.

In short, a gene plays a role of a market basket (or, to be more exact - a role of a label of a market basket, the same way the receipt number playes the role of the label of a market basket in the `EXTENDED BAKERY` dataset). With each gene, a collection of transcription factor names is associated.

The data will be made available to you on Thursday, April 19. The data will be available in the *Items Table* format. Time permitting, the data may also get released in the other two formats.

This is an additional dataset for this assignment. The reason why this dataset is present is because the owner of the data, Dr. Dan Pterson, from the Cal Poly's Animal Science program, is interested in finding out *exactly the information* that can be discovered by applying the version of the Apriori algorithm that you will develop for this lab. Working with this dataset is an exploratory part of the assignment.

Mining Frequent Itemsets and Association Rules

Your task is to discover the association rules that exceed specific given values of *minimum support* and *minimum confidence*.

As discussed in class, mining association rules is a two-step process. On step one, the goal is to discover *frequent itemsets* with support exceeding *minsup*. On step two, the goal is to discover specific *association rules* found within the discovered frequent itemsets.

Algorithms for discovery of both frequent itemsets (Apriori) and association rules (genRules) have been discussed in class together with implementation strategies.

Skyline (Maximal) Frequent Itemsets and Association Rules

Association rules mining tends to discover **a lot** of rules in any given dataset. This is due to permutation properties of the rules (e.g., if $A, B \rightarrow C, D$ is an association rule, then so are $A, B, D \rightarrow C$, $A, B, C \rightarrow D$) and due to the large number of items in a typical dataset.

To make results of your work *observable*, we will be interested only in so-called **skyline** or **maximal** frequent itemsets and association rules.

Definition. A frequent itemset is called a **skyline** (**maximal**) frequent itemset, if *it is NOT a subset of any other frequent itemset.* An association rule is called a **skyline** association rule if its right side and its left side form a **skyline** frequent itemset.

Informally, **skyline** or **maximal frequent itemsets** are those, that cannot be extended further to form larger frequent itemsets. To constrain the output of your work, you need only to report **skyline** frequent itemsets.

Furthermore, to simplify the process of mining association rules, you shall report **only skyline** association rules in which the right side of the rule contains *a single item.*

Deliverables

You should discover **skyline frequent itemsets** and **skyline association rules** in each of the four `EXTENDED BAKERY` datasets. Additionally, make an effort to discover **skyline frequent itemsets** in the `TRANSCRIPTION FACTORS` dataset (we do not need association rules for the `TRANSCRIPTION FACTORS` dataset).

Submit the following:

- A report containing the list of skyline frequent itemsets and the list of skyline association rules you discovered in each of the four datasets.

  For each *skyline frequent itemset* specify:

  1. All *items* in it. Use `Goods.Flavor` and `Goods.Food` attribute values to describe each item.
  2. The support of the itemset.

Notice, that `Goods.Flavor` and `Goods.Food` attributes are NOT present in the input market baskets (all the formats described above contain only the `Goods.Id` attribute). It is the job of your software to report these attributes given the good ids.

For each *skyline association rule* specify:

1. All items on the left side of the rule (`Goods.Food`+`Goods.Flavor`).
2. The item on the right side of the rule (`Goods.Food`+`Goods.Flavor`).
3. The support of the rule.
4. The confidence of the rule.

- Any software you have written to discover association rules.

  In general, a program for association rules discovery should take as input the following parameters:

  1. Filename. Name of the CSV file containing the dataset. Your program can use any of the formats made available to you.
  2. minSup. The minimum support number for frequent itemset and association rule discovery.
  3. minConf. The minimum confidence number for association rule discovery.

  Additionally, you may include any optional flags that specify whether:

  – all rules/frequent itemsets or skyline rules/frequent itemsets should be returned. (the default behavior is to print skylines).
  – only rules, only frequent itemsets or both rules and frequent itemsets shall be printed.

  Given the need to discover *only* the skyline frequent itemsets (but not the association rules) for `TRANSCRIPTION FACTORS` dataset, it makes sense to either have two executables, or to include the last flag from the list above.

  Generally speaking, you may elect to implement your software in any way you like: e.g., you can split reading/parsing data, frequent itemset search and association rules discovery into three separate pieces of code if this is more convenient for you.

- A `README` file which contains the following information (at least):

  – Names of all students in the pair/team.
  – Specification of which type(s) of input format your program(s) take(s).
  – Instructions on how your code should be run. This is especially important if you implemented association rules mining as a sequence of separate programs.

- A **separate report** regarding your discoveries for the `TRANSCRIPTION FACTORS` dataset. You are expected to use the program you develop to find the appropriate values of the `minsup` threshold that yield meaningful results (you are also expected to determine, as part of your investigations what "meaningful" is in this dataset). The report shall identify the different `minsup` values you used on each of the runs that brought you some meaningful information, and the frequent itemsets discovered for each value of `minsup` tested. The report can contain results of running your program, but it shall, in general, be readable for a non-computer scientist.

Note: Frequent itemset/association rule lists that you submit can be the output of your program(s), as long as your program output follows the guidelines specified above.

Note: Each `EXTENDED BAKERY` dataset incorporates a specific set of association rules (and frequent itemsets), that stand out. You may have to try your program with a number of minConf and minSup parameters until you discover all of them, but overall, the **separation between the frequent itemsets/association rules** and all other itemsets/candidate rules is **very robust**.

Training Dataset

To help you calibrate your discovery process, we are providing one more dataset for you. The dataset contains 1000 market baskets and has the following **association rules** seeded in it:

Lemon Cake $\longrightarrow$ Single Espresso
Blackberry Tart $\longrightarrow$ Apple Danish
Napoleon Cake $\longrightarrow$ Gongolais Cookie
Apple Tart and Berry Tart $\longrightarrow$ Blueberry Tart

All these rules have support of at least 10% and a confidence of at least 90%. Note that other rules (e.g., Berry Tart and Blueberry Tart $\longrightarrow$ Apple Tart will also exist in the dataset and would need to be reported).

The training dataset can be downloaded from the course web page. The `example.zip` file contains the following four files inside the `example` directory:

`out1.csv`	market baskets in **sparse vector** format
`out2.csv`	market baskets in **full binary vector** format
`lab2-example-output`	output of the TA's rule mining program
`Rules2.xml`	an XML file specifying the rules found in the dataset

Submission Instructions

Report submission. While I prefer hardcopies of the reports, reports in soft copy only will also be accepted. Both reports shall be word-processed, with the results of running your program included where necessary. Submit the reports by April 24, midnight. If you have hard copies of your report, bring them to class on **April 26**.

Code submission. You will use the handin tool to submit your . Each pair submits exactly one copy of all materials. Put all your files in a single archive (zip or gzipped tar), name it `lab03.zip` or `lab03.tar.gz` and submit as follows:

```
$ handin dekhtyar lab03-466 lab03.zip
```