University of Vienna
Faculty of Computer Science
Prof. Claudia Plant
Prof. Torsten Möller
Prof. Marcus Hudec

Foundations of Data Analysis

WS 2017

General Remarks:

- The deadline for submission is **1:15 pm** on **26.01.2018**. No deadline extensions can be granted.
- Upload your solutions as a zip archive with the following naming scheme **registrationnumber_A6.zip**. The archive should contain a report with your assumptions, results and a description how to compile and run your apriori implementation.
- If you have problems do not hesitate to contact the tutor or post a question on the Moodle system.
- The solutions for this assignment will be discussed in the lecture on 26.01.2018.
- Mark external sources you are using in the code and report

Task 6-1 Apriori Algorithm (60P)

Task Definition:

In this programming assignment, you are required to implement the Apriori algorithm (do not use a built-in implementation) and apply it to mine frequent itemsets from a real-life data set.

The provided input file ("adult.txt") is a preprocessed version of the census data relating to income data from the United States available at 'https://archive.ics.uci.edu/ml/datasets/census+income'. Each line contains income data $> 50K$ or $<= 50K$ as well as attributes containing information on workclass, education, marital-status, occupation, relationship, race, sex and country of origin.

An example: Private;Masters;Never-married;Prof-specialty;Not-in-family;White;Female;United-States;>50K

In the example above, the subject earns an income greater 50 000 $ and the attributes tell us that she is employed privately, obtained a masters degree, was never married, and is a female, white, professional who was born in the United States.

Output: You need to implement the Apriori algorithm and use it to mine category sets that are frequent in the input data. When implementing the Apriori algorithm, you may use any programming language you like. You might be faster with scripting languages. After implementing the Apriori algorithm, please set the **relative minimum support** to **0.1** and run it on the 30161 instance of the adults.txt file. In other words, you need to extract all the category sets that have an absolute support larger than 3000.

(a) **Part 1 (15 points):** Output all the length-1 frequent items (itemsets containing only one element) with their absolute supports into a text file named "oneItems.txt" and place it in the root of your zip file. Every line corresponds to exactly one frequent items and should be in the following format: `Support: item` For example, suppose an item ($> 50K$) has an absolute support 3000, then the line corresponding to this frequent item set in "patterns.txt" should be: `3000:>50K` You can think of this as a dictionary with corresponding frequency counts of the items contained in the dictionary.

(b) **Part 2 (45 points):** Please write all the frequent item sets along with their absolute supports into a text file named "patterns.txt" and place it in the root of your zip file. Every line corresponds to exactly one frequent item set and should be in the following format: `support: item_1, item_2, ...` For example, suppose an item set ($> 50K$,Master) has an absolute support 3851, then the line corresponding to this frequent item set in patterns.txtßhould be: `3851:>50K,Master`

Task 6-2 Clustering in R (40 P)

Task Definition:

In this task you will apply different clustering techniques and evaluate the results on the iris dataset which is already known from the pen and paper exercise. The dataset is included in R so you don't need to download or import it. The description of the data can be found in the UCI Machine Learning Repository[1].

(a) plot the iris data

(b) apply k-Means Clustering

(c) apply DBSCAN

(d) apply EM-clustering

(e) evaluate your results using NMI

(f) discuss your results. Describe also the parametrization used and how you have chosen them.

To load the iris dataset just type `iris` in R. Use the following libraries `dbscan, EMCluster, NMI` to solve this task.

[1] https://archive.ics.uci.edu/ml/datasets/Iris