University of Ioannina,
Department of Computer Science & Engineering

Data Mining – CSE 012/CS 059
Fall 2015

Assignment 1

This is the first part of Assignment 1. The deadline for this part of the assignment is November 5, 1:00 pm, at the beginning of the class. You should turn in the code for question 1, and submit the remaining questions either electronically, or on paper. For late submissions the late policy on the page of the course will be applied. Details for the turn-in, and how to write reports are on the Assignments web page of the course.

Question 1 (Weighted Reservoir Sampling)

In class we described the Reservoir Sampling algorithm for sampling a single item from a stream of items. In this question you are required to modify the algorithm to do **weighted sampling**. We assume that each item i has a weight w_i. You will modify the sampling algorithm so that given a stream of weighted items it samples one item with probability proportional to its weight. That is, if the stream has N items with total weight $W = \sum_{i=1}^{N} w_i$, item i is should have probability w_i/W to be selected, for all $1 \leq i \leq N$.. Similar to the standard Reservoir Sampling algorithm, the length of the stream N is not known in advance, the algorithm should work with constant amount of memory, independent of N. Prove the correctness of your algorithm

Question 2 (Reservoir Sampling)

In this question you are required to modify the simple (no weights) Reservoir Sampling algorithm to sample $\boldsymbol{K}$ items from a stream of N items.

1. Describe the algorithm for sampling K items uniformly at random from a stream of N items. The algorithm should work in a single pass over the data, reading the items one by one, without prior knowledge of the size of the stream N, and using $O(K)$ of memory (assume the size of an item is fixed). (**Hint**: In a random sample each element should have probability K/N to appear in the sample).
2. Prove that your algorithm produces a uniform sample, that is, for every $i, 1 \leq i \leq N$, the i-th element has probability K/N to appear in the sample.
3. Write a program **in Python** that implements the sampling algorithm. Your program should sample K lines from a text document. It should be possible to use the program from command line, it should take as command line argument the value of K, read lines from the standard input, and output the sample in the standard output. For example the following command should print a random sample of 10 lines from the file input.txt:
   "`sample.py 10 < input.txt`".