Randomized Algorithms: Assignment 1

January 28, 2018

Finding Median

The median is the value separating the higher half of a data sample from the lower half. The simplest algorithm for finding it sorts all elements in the input array and returns the element from the middle. However, there are many other algorithms available. In this assignment, you will compare the trivial algorithm mentioned above, numpy implementation, and randomized median search algorithm.

Randomized Median Algorithm

This is a pseudocode for the randomized median search algorithm.

Input: A set of n elements over a totally ordered universe.
Output: The median element of S, denoted by m.

1. Pick a multiset R of $\lceil n^{3/4} \rceil$ elements in S choosen independently and uniformly at random with replacement.
2. Sort the set R
3. let d be the $\left(\lfloor \frac{1}{2}n^{3/4} - \sqrt{n} \rfloor\right)$th smallest element in the sorted set R.
4. let u be the $\left(\lceil \frac{1}{2}n^{3/4} + \sqrt{n} \rceil\right)$th smallest element in the sorted set R.
5. By comparing every element in S to d and u compute the set $C = \{x \in S : d \leq x \leq u\}$ and the numbers $l_d = |\{x \in S : x < d\}|$ and $l_u = |\{x \in S : x > u\}|$
6. If $l_d > n/2$ or $l_u > n/2$ then FAIL.
7. If $|C| \leq 4n^{3/4}$ then sort the set C, otherwise FAIL.
8. Output the $(\lceil n/2 \rceil - l_u + 1)$th element in the sorted order of C.

Files description

In the provided template you will find three source files:

- median.py - contains algorithms for finding the median value of an array. You have to complete the **median_r** function here.
- misc.py - contains helper functions, no need to change them.
- experiments.py - It has all experiments for analysis of median algorithms. You should fill in the blanks in this file.
- run.py - this file runs all the experiments.

Versions

Please, use Python 3.6.4, numpy version 1.13.3, matplotlib version 2.1.1 and don't change the seed. This will help us to reproduce the specific behavior of the random number generator and is crucial for grading process.

Tasks

In this assignment, you should implement the randomized median algorithm **(20 points)**, do experiments and prepare the report with results and answers to the following questions.

1. **(10 points)** Make asymptotical estimations of time required for finding median value using sorting and randomized median algorithm. Provide both the words and the average cases.
2. **(15 points)** Compare results of the randomized median algorithm, sorting median algorithm and numpy median. Does it always give the same results? Why?
3. **(15 points)** Analyze the number of fails for the randomized median algorithm. What is the relation between the probability of failure and the size? Does it comply with the theoretical upper bound?
4. **(15 points)** Compare the average runtime of three algorithms. What is the observed relation between the runtime and input data size? Is this what you would expect? Why?
5. **(15 points)** Which strategy would you suggest if randomized algorithm returns fail? Analyze each suggestion.
6. **(10 points)** If we don't know the seed, is it possible to give an input in which randomized algorithm always fails? What if we know the seed? If it's possible, how do we construct data that fails the algorithm?