Big Data Algorithm Midterm Problems
Part I

2017-10-28

1 Requirement

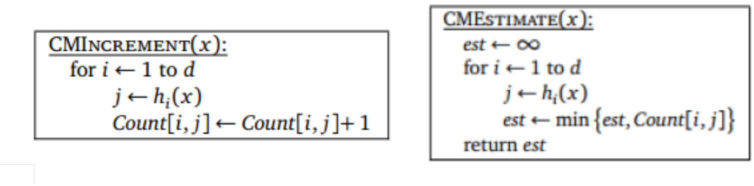
Choose and finish 5 problems, email your answers to TA Tong Yin (yintong@sjtu.edu.cn) before 9:59 am, Nev. 1st. And you may be asked to finish 3 of these problems, as well as 2 other problems in the class on Nov. 1st.

2 Problem 1

Morris Counting (Proposed by Robert Morris 1977) algorithm allows the counting of a large number of elements using memory of which size is small. The algorithm make the incrementing of counts become a probabilistic event. Assuming the current counts is X_n (have read n elements), the probability of incrementing is 2^{X_n}, and make $\hat{n} = 2^{X_n} - 1$. We had calculated the expected value $\mathbb{E}(\hat{n}) = n$ in the class, what is the variance of $\hat{n}$? And how to reduce the error?

3 Problem 2

The data structure consists of a $w \times d$ array of counters (all initially zero) and d hash functions, which remarkably only need to be 2-universal, not ideal random. If we set $w := [\frac{e}{\epsilon}]$ and $d := [ln(\frac{1}{\delta})]$, let f_x be the true frequency (number of occurrences) of x, and let $\hat{f}_x$ be the value returned by $CME_{STIMATE}$ shown in figure 1. It is easy to see that $f \leq \hat{f}_x$; we can never return a value smaller than the actual number of occurrences of x. Question: Please claim that $Pr[\hat{f}_x > f_x + \epsilon N] < \delta$ (where N is the total number of calls to $CMI_{NCREMENT}$). In other words, our estimate is never too small, and with high probability, it is not a significant overestimate either. (Notice that the error here is additive; the estimates or truly infrequent items may be much larger than their true frequencies.



```
CMINCREMENT($x$):
  for $i \leftarrow 1$ to $d$
      $j \leftarrow h_i(x)$
      $Count[i,j] \leftarrow Count[i,j]+1$
```

```
CMESTIMATE($x$):
  $est \leftarrow \infty$
  for $i \leftarrow 1$ to $d$
      $j \leftarrow h_i(x)$
      $est \leftarrow \min\{est, Count[i,j]\}$
  return $est$
```

Figure 1: $CME_{STIMATE}$ and $CMI_{NCREMENT}$

4 Problem 3

Recall that Morris's algorithm uses $\log\log N$ bits counting N elements. During the lecture, professor has already proved $2^{X_N} - 1$ is an unbiased estimate of N. However, in practice, an unbiased estimate is not enough. For example, if the output has a distribution as shown in Figure 2, although the expectation is unbiased, the probability for x to be around the true value is small. So what we actually want is the output to be around the true value with high probability, which means in the Morris's algorithm $p(|2^{X_N} - 1 - N| > \epsilon N)$ should be small.

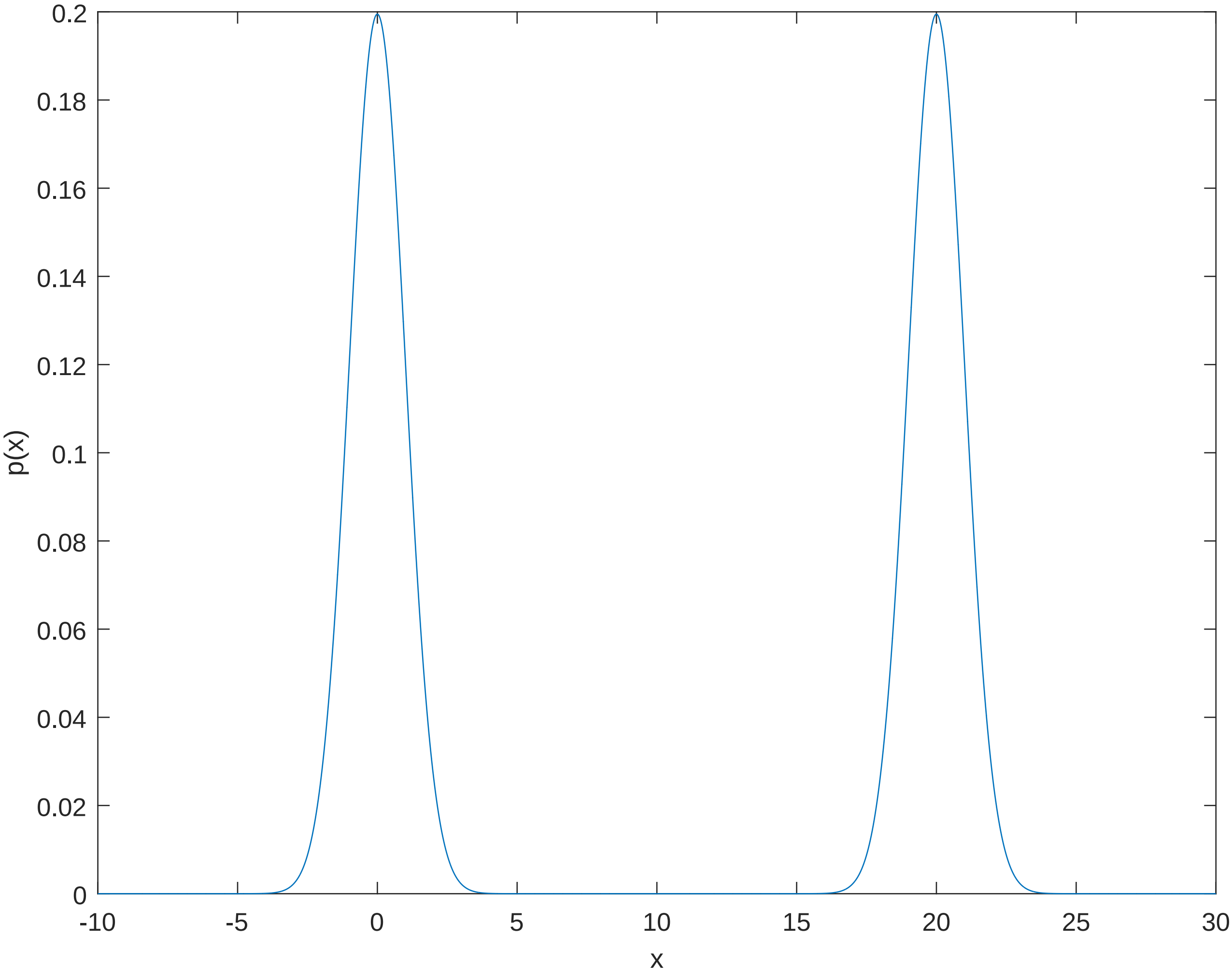


Figure 2: $p(x) \sim (N(0,1) + N(20,1))/2$

1. Prove that $E(2^{2X_n}) = \frac{3}{2}n^2 + \frac{3}{2}n + 1$.

2. Prove that $p(|2^{X_n} - 1 - n| > \epsilon n) \leq \frac{1}{2\epsilon^2}$.

3. (Bonus) Is the upper bound of $p(|2^{X_n} - 1 - n| > \epsilon n)$ small enough? Can you improve the Morris's algorithm to make it smaller? Please derive or prove a smaller upper bound of the improved algorithm. (Hint: Do you remember the tool we used to reduce the false-positive rate in membership test?)

5 Problem 4

Give n points set $\mathbf{S}$, find a subset $\mathbf{S}'$ of $\mathbf{S}$ to make that the number of points outside $conv(\mathbf{S}')$ a and the number inside $conv(\mathbf{S}')$ b satisfies $|a-b| \leq 1$.

6 Problem 5

Why Kolmogorov complexity is not computable?

7 Problem 6

Under the circumstance that the input size is much larger than the main memory we have to process a dataset, we usually couldnt afford to find an exact median but have to estimate as long as the error rate is tolerable. Now we consider sampling method to reduce the data size. Given a dataset of size N:

- divide the data into m groups, each with r elements
- sort each group and take x numbers from it such that the k-th chosen number would be at $k \cdot r/x$ locations of its group
- merge the $m \cdot x$ numbers into a sorted list D

1. Analyze the time complexity of this algorithm
2. Assuming we pick the median of D to estimate the median, denoted by M, and the location of M in the whole dataset would be obtained in an interval $[L, H]$, please give solution to L and H

8 Problem 7

1. We know Yaos principle is that the best randomized algorithm on the worst time complexity is equivalent to the worst distribution for the best average algorithmic solution.

$$min_{\mathcal{R}} max_{x \in \mathcal{I}} Cost(\mathcal{R}, x) = max_{\mathcal{D}} min_{A \in \mathcal{A}} Cost(A, \mathcal{D})$$

$Cost(\mathcal{R}, x)$: Expected cost of a randomized algorithm R on an input x;
$Cost(A, \mathcal{D})$: Expected cost of a deterministic algorithm A on a random input distribution D.

(a) Show that $max_{x \in \mathcal{I}} Cost(\mathcal{R}, x) \geq min_{A \in \mathcal{A}} Cost(A, \mathcal{D})$

(b) Since that $max_{x \in \mathcal{I}} Cost(\mathcal{R}, x) \geq min_{A \in \mathcal{A}} Cost(A, \mathcal{D})$ proved in (a), show that $min_{x \in \mathcal{I}} P(\mathcal{R}, x) \leq max_{A \in \mathcal{A}} P(A, \mathcal{D})$.
(Hint: let $Cost(\mathcal{R}, x) = -P(\mathcal{R}, x), Cost(A, \mathcal{D}) = -P(A, \mathcal{D})$)

(c) Consider a data stream $\{a_1, a_2, ..., a_n\}$, we want to get the maximum of the data stream. We will see a_i one by one, if we see a_i, we must decide whether to choose this a_i as maximum or not.

- If we choose a_i as maximum, then the procedure terminates and a_i is the one we believe the maximum.
- If we don't choose a_i as maximum, we will continue to see a_{i+1} and could not choose a_i later.

Prove that for any randomized selecting maximum algorithm, there is an input for which the randomized algorithm select maximum with probability less than or equal to $\frac{1}{n}$. (Hint: 1. Use the conclusion in (b) 2. consider the input data distribution $\{a_1 = 1, a_2 = 2, ..., a_{k-1} = k-1, a_k = k, a_{k+1} = 0, ...a_n = 0\}$ where k is drawn uniformly at random from 1,2...,n-1,n. 3. first consider what will happen if k = n and then consider k≠n)

9 Problem 8

Prove the following theorem:
For an n-vertices graph $G = (V, E)$ with m edges, two disjoint sets $C_0 \subseteq V$ and $V_0 \subseteq V$ can be computed in time $O(\sqrt{n} \cdot m)$, such that the following three properties hold.

1. Let $D \subseteq V_0$ be a vertex cover of the subgraph $G[V_0]$. Then $C := D \bigcup C_0$ is a vertex cover of G.
2. There is a minimum vertex cover S of G with $C_0 \subseteq S$.
3. The subgraph $G[V_0]$ has a minimum vertex cover of size at least $[V_0]/2$.

10 Problem 9

There is a real-only tape which stores large amount of raw integer data which known as an exponential distribution. And you can only read very small part of sample them, which means the medium of the sample can only be the medium of the distribution but not exactly the medium of the raw data. Now, you get the sample $X = x_1, x_2, x_3, \ldots, x_n$, please calculate the median of the raw data.