University of Wrocław: Algorithms for Big Data (Fall'19) 07/10/2019

Lecture 1: Approximate Counting, Distinct Elements

Lecturer: *Przemysław Uznański* Scribes: *Mikołaj Słupiński*

1 Introduction

1.1 Topics during the course

- Streaming (counting, heavy hitters, norm estimation, sampling) ($\sim$ 4 Lectures)
- Dimensionality reduction and sparse linear algebra (eg. JL, approx matrix mul, compressed sensing) ($\sim$ 4 Lectures)
- Applications (geometry algo, coresets, graph algorithms, ANN, sliding window) ($\sim$ 4 Lectures)

1.2 Motivation

Linear time/space algorithms are not good enough with modern datasets and their volume. Typical problem we are dealing with in this course: here is a stream of data, process it in a small space to compute output X. Usually there is a lower-bound preventing us to do it in a very small space *exactly*. Hence we need to relax our problem to achieve very efficient (in space and time) algorithms. Examples:

- Think of any recommendation system, where each user has assigned highly dimensional vector of preferences. We want to test similarity/dissimilarity of user profiles.
- Database with approximate index (Approx Membership Queries), to quickly eliminate queries for elements that are not in the DB, except for few false positives.
- Lossy compression of audio or images selects heavy hitters in the frequency domain. How to find them without computing FFT explicitly?
- Count distinct elements in a stream, or maintain statistics in a continuous stream of updates (router + number of unique IP).

1.3 Techniques

- Probabilistic tools - few probabilistic bounds are good enough 90% of the time, sometimes we will need to go a little bit deeper (fancy distributions),
- relaxing problem: $1 \pm \varepsilon$ approximation and $1 - \delta$ correctness guarantee,
- linear algebra,
- trace amounts of combinatorics and "typical" A&DS - that's why it might be tricky for CS students.

2 Approximate counting

The problem is to maintain a counter that supports following operations:

$$\begin{aligned} \text{reset}(),\ &[n \leftarrow 0] \\ \text{inc}(),\ &[n \leftarrow n+1] \\ \text{query}(),\ &[\text{output n}] \end{aligned}$$

Simple lowerbound of $\log(n)$ bits for exact (information-theoretic lowerbound).

Goal: algorithm that queried outputs n' such that $\mathbf{Pr}(|n - n'| > \varepsilon n) < \delta$.

2.1 Morris' algorithm [Sr.78]

Local state: X [int], represents $n \sim 2^X$. The crucial part of algorithm is to design how we increase X.

Inc: $X \leftarrow X + 1$ with some small probability ($\sim 2^{-X}$), with intuition being that the ppb of n being exactly $2^{X+1} - 1$ is 2^{-X}.

Let us analyze increment probability $= 2^{-X}$. Let X_n be random variable denoting state of algorithm after n increases.

Theorem 1.

$$\mathbf{E}[2^{X_0}] = 2^{X_0} = 1 \tag{1}$$
$$\mathbf{E}[2^{X_n}] = n + 1 \textit{ by induction} \tag{2}$$

Proof.

$$\begin{aligned} \mathbf{E}2^{X_{n+1}} &= \sum_{j=0}^{\infty} \mathbf{Pr}\left(X_n = j\right) \cdot \mathbf{E}\left(2^{X_{n+1}} | X_n = j\right) \\ &= \sum_{j=0}^{\infty} \mathbf{Pr}\left(X_n = j\right) \cdot \left(2^j\left(1 - \frac{1}{2^j}\right) + \frac{1}{2^j} \cdot 2^{j+1}\right) \\ &= \sum_{j=0}^{\infty} \mathbf{Pr}\left(X_n = j\right) 2^j + \sum_j \mathbf{Pr}\left(X_n = j\right) \\ &= \mathbf{E}2^{X_n} + 1 \\ &= (n+1) + 1 \end{aligned}$$

□

Morris algorithm output: $Z = 2^{X_n} - 1 \leftarrow$, which is an unbiased estimator of n (that is $\mathbf{E}[Z] = n$).

2.1.1 Analysis of variance to extract guarantees:

Theorem 2. *We show inductively that* $\mathbf{E}[2^{2X_n}] = 3/2n^2 + 3/2n + 1$.

Proof. see exercise □

Since

$$\begin{aligned}\mathbf{Var}[Z] &= \mathbf{Var}[2^{X_n}] \\ &= \mathbf{E}[2^{2X_n}] - (\mathbf{E}[2^{X_n}])^2 \\ &= \frac{3}{2}n^2 + 3/2n + 1 - (n+1)^2 \\ &= \frac{1}{2}n^2 - \frac{1}{2}n,\end{aligned}$$

by Chebyshev's inequality $\mathbf{Pr}(|Z - n| > \varepsilon n) \leq 1/(2\varepsilon^2)$.

This only gives failure probability $\delta < \frac{1}{2}$ for $\varepsilon > 1$, which is not very informative: (large) constant approximation with constant probability. But that was to be expected: our algorithm only outputs powers of two, so it cannot do much better job.

2.2 Morris+

Repeat k times independently, take average of estimations. Since variance is additive: $\mathbf{Var}(Z') = \frac{1}{k^2}(\mathbf{Var}(Z_1) + \mathbf{Var}(Z_2) + \cdots + \mathbf{Var}(Z_k)) = 1/k\mathbf{Var}(Z)$ so number of iterations necessary becomes: $k = \mathcal{O}(\frac{1}{\varepsilon^2\delta})$ (ok for 9/10 ppb of correctness, bad for whp correctness).

2.3 Morris++

Run t copies of Morris+ algorithm, each with $\delta = \frac{1}{3}$ and take median of estimations as a final estimation. Each estimation is ok with probability $\geq \frac{2}{3}$, so for the median to fail at least $\frac{1}{6}$ fraction of estimations need to fail (all too large or all too small) Chernoff bound gives us:

$$\mathbf{Pr}\left(\sum_{i=1}^{t} Y_i \leq \frac{t}{2}\right) \leq \mathbf{Pr}\left(\left|\sum_{i=1}^{t} Y_i - \mathbf{E}\sum_{i=1}^{t} Y_i\right| \geq \frac{t}{6}\right) \leq 2e^{-t/3} < \delta \tag{3}$$

for $t = \Theta(\lg(1/\delta))$. Final **bit** complexity $\mathcal{O}(\log\log(n/(\varepsilon\delta))\frac{1}{\varepsilon^2}\log(\frac{1}{\delta}))$.

Lowerbound: $\Omega(\log\log_{1+\varepsilon} n) = \Omega(\log(1/\varepsilon) + \log\log n)$ (for $\delta = 0$, its trickier to prove lowerbound involving δ)

3 Distinct elements

Input: Stream of values $i_1, i_2, \ldots, i_m$ from $[n]$ query() $\leftarrow$ number of distinct elements

Trivial solution: remember the stream, bitvector

3.1 Flajolet Martin [FM85]

Pick a hash function $h : [n] \to [0, 1]$ (for a moment let us assume ideal real numbers, and perfectly random hash function).

1. initially $Z = 1$
2. input X: $Z = \min(Z, h(X))$

3. estimator: $Y = 1/Z - 1$

Observation 3. *Repeats do not affect Z.*

If t is the number of distinct elements, then $Z = \min(r_1, r_2, \ldots, r_t)$ where r_i are all independent and from $[0,1]$.

Lemma 4.

$$\mathbf{E}[Z] = \frac{1}{t+1} \tag{4}$$

Proof. Pick fresh A at random from $[0,1]$. By symmetry,

$$\mathbf{E}[Z] = \mathbf{Pr}[A < Z] = \mathbf{Pr}[A \text{ is minimal among } A, r_1, \ldots, r_t] = \frac{1}{(t+1)}.$$

□

Lemma 5.

$$\mathbf{E}[Z^2] \leq \frac{2}{(t+1)(t+2)} \tag{5}$$

Proof. Pick fresh A, B at random from $[0,1]$. By symmetry, $\mathbf{E}[Z^2] = \mathbf{Pr}[A < Z \wedge B < Z] = \frac{2}{(t+1)(t+2)}$ □

Alternative proof.

$$\begin{aligned}
\mathbf{E}\left[Z^2\right] &= \int_0^\infty \mathbf{Pr}\left(Z^2 > \lambda\right) d\lambda \\
&= \int_0^\infty \mathbf{Pr}(Z > \sqrt{\lambda}) d\lambda \\
&= \int_0^1 (1-\sqrt{\lambda})^t d\lambda \\
&= 2\int_0^1 u^t(1-u)du \quad [u = 1-\sqrt{\lambda}] \quad = \frac{2}{(t+1)(t+2)}
\end{aligned}$$

□

$$\mathbf{Var}[Z] = \frac{2}{(t+1)(t+2)} - \frac{1}{(t+1)^2} = \frac{t}{(t+1)^2(t+2)} < (\mathbf{E}[Z])^2 \tag{6}$$

Remark 6. *Applying Chebyshev's inequality → results in a guarantee of a (large) constant approximation with lets say $\frac{9}{10}$ probability.*

Issue: $\mathbf{E}[\frac{1}{Z}] \neq \frac{1}{\mathbf{E}[Z]}$, but concentrating Z with $1+\varepsilon$ multiplicative error will give $1+\varepsilon$ multiplicative error for $\frac{1}{Z}$.

3.2 FM+

To reach better approximation guarantee, we need to concentrate our output around expected value.

Approach 1 copy approach from Morris' algorithm - "repeat k times and take average" to improve variance, set $k = \mathcal{O}(\frac{1}{\varepsilon^2})$ for $\frac{9}{10}$ probability of $1+\varepsilon$ approximation.

Approach 2 replace "take minimum" with "take k-th smallest value" (to be analyzed → exercise).

3.3 FM++

To improve probability of success, repeat FM+ algorithm $t = \mathcal{O}(\log \delta^{-1})$ times, and take median of answers. This boosts probability of success to $1-\delta$.

Total memory complexity is
$\mathcal{O}(\log n \frac{1}{\varepsilon^2} \log \delta^{-1})$ of **words** (each word is $\log n$ bits).

3.4 Issues

Recall "for a moment let us assume ideal real numbers".

We only care about relative order of hashes, and use actual value as an estimator. Using hash-functions of form $h : [n] \to \{\frac{0}{M}, \frac{1}{M}, \ldots, \frac{M-1}{M}, \frac{M}{M}\}$ for some $M = n^3$, as it only introduces small relative error (whp each hash is $\geq \frac{1}{n}$ thus relative error introduced is at most $(1+\frac{1}{n})$, and wlog $\varepsilon > \frac{1}{n}$), and whp there are no collisions of hashes.

Recall "and perfectly random hash function".

Randomness vs. pseudorandomness → c.f. exercises

4 Further reading

- hyperloglog algorithm, which very efficient in theory and practice, but has extremely nontrivial analysis [DF03] [HNH13]
- [Bla18] - optimal $\Theta(\log n + \frac{\log \delta^{-1}}{\varepsilon^2})$ bits.

References

[Bla18] Jaroslaw Blasiok. Optimal streaming and tracking distinct elements with high probability. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 2432–2448, 2018.

[DF03] Marianne Durand and Philippe Flajolet. Loglog counting of large cardinalities (extended abstract). In *Algorithms - ESA 2003, 11th Annual European Symposium, Budapest, Hungary, September 16-19, 2003, Proceedings*, pages 605–617, 2003.

[FM85] Philippe Flajolet and G. Nigel Martin. Probabilistic counting algorithms for data base applications. *J. Comput. Syst. Sci.*, 31(2):182–209, 1985.

[HNH13] Stefan Heule, Marc Nunkesser, and Alexander Hall. Hyperloglog in practice: algorithmic engineering of a state of the art cardinality estimation algorithm. In *Joint 2013 EDBT/ICDT Conferences, EDBT '13 Proceedings, Genoa, Italy, March 18-22, 2013*, pages 683–692, 2013.

[Sr.78] Robert H. Morris Sr. Counting large numbers of events in small registers. *Commun. ACM*, 21(10):840–842, 1978.

A Probability recap

Definition 7. *1. The empty set is an event, $\varnothing \in \mathcal{F}$*

2. Given a countable set of events $A_1, A_2, \dots$, its union is aso an event, $\bigcup_{i \in \mathbf{N}} A_i \in \mathcal{F}$

3. if A is an event, then so is the complementary set A^c

Definition 8. *1. $\mathbf{Pr}(\emptyset) = 0, \mathbf{Pr}(\Omega) = 1$*

2. if $A_1, A_2, \dots$ are mutually excluding events, then $\mathbf{Pr}\left(\cup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mathbf{Pr}\left(A_i\right)$
A $\mathbf{Pr} : \mathcal{F} \mapsto [0, 1]$ satysfying these is called a probability.
The triple $(\Omega, \mathcal{F}, \mathbf{Pr})$ is called a probability space.

Definition 9. *We define coditional probability as*

$$\mathbf{Pr}(A|B) = \frac{\mathbf{Pr}(A \cap B)}{\mathbf{Pr}(B)}$$

Theorem 10. *Let $B_1, \dots, B_n$ be a partition of Ω, then*

$$\mathbf{Pr}(A) = \sum_{i=1}^{n} \mathbf{Pr}\left(A|B_i\right) \mathbf{Pr}\left(B_i\right) \tag{7}$$

Definition 11. *Events A and B are called independent if*

$$\mathbf{Pr}(A \cap B) = \mathbf{Pr}(A)\mathbf{Pr}(B). \tag{8}$$

When$0 < \mathbf{Pr}(B) < 1$, this is the same as

$$\mathbf{Pr}(A|B) = \mathbf{Pr}(A) = \mathbf{Pr}\left(A|B^c\right) \tag{9}$$

A family $\{A_i : i \in I\}$ of events is called independent if

$$\mathbf{Pr}\left(\cap_{i \in J} A_i\right) = \prod_{i \in J} \mathbf{Pr}\left(A_i\right) \tag{10}$$

for any finite subset J of I.

Definition 12. *A random variable is Informally: A quantity which is assigned by a randomexperiment. Formally: A mapping $X : \Omega \to \mathbf{R}$.*

Definition 13. *The cumulated distribution function(cdf) is:*

$$F(x) = \mathbf{Pr}(X \leq x) \tag{11}$$

If satisfies following properties:

1. $\lim_{x \to -\infty} F(x) = 0, \lim_{x \to +\infty} F(x) = 1$

2. $x < y \Rightarrow F(x) \leq F(y)$

3. F is right-continuous, ie. $F(x + h) \to F(x)$ as $h \downarrow 0$

Definition 14. *The mean of a stochastic variable is*

$$\mathbf{E}X = \sum_{i\in\mathbb{Z}} i\mathbf{Pr}(X = i)$$

in the discrete case, and

$$\mathbf{E}X = \int_{-\infty}^{+\infty} f(x)dx$$

in the continuous case. In both cases we assume that the sum/integral exists absolutely. The variance of X *is*

$$\mathbf{Var}X = \mathbf{E}(X - \mathbf{E}x)^2 = \mathbf{E}X^2 - (\mathbf{E}X)^2$$

Definition 15. *The conditional expectation is the mean in the conditional distribution*

$$\mathbf{E}(Y|X = x) = \sum_y y f_{Y|X}(y|x) \tag{12}$$

It can be seen as a stochastic variable: Let $\psi(x) = \mathbf{E}(Y|X = x)$*, then* $\psi(X)$ *is the conditional expectation of* Y *given* X

$$\psi(X) = \mathbf{E}(Y|X) \tag{13}$$

We have

$$\mathbf{E}(\mathbf{E}(Y|X)) = \mathbf{E}Y \tag{14}$$

Definition 16. *Conditional variance* $\mathbf{Var}(Y|X)$ *is the variance in the conditional distribution.*

$$\mathbf{Var}(Y|X = x) = \sum_y (y - \psi(x))^2 f_{Y|X}(y|x) \tag{15}$$

This can also be written as

$$\mathbf{Var}(Y|X) = \mathbf{E}\left(Y^2|X\right) - (\mathbf{E}(Y|X))^2$$

and can be manipulated into

$$\mathbf{Var} = \mathbf{E}\mathbf{Var}(Y|X) + \mathbf{Var}\mathbf{E}(Y|X)$$

which partitions the variance of Y*.*

Theorem 17 (Markov's inequality)**.** *Let* $X \geq 0$ *be a random variable. Then for any* $k \geq 1$ *:*

$$\mathbf{Pr}(X \geq k \cdot \mathbf{E}[X]) \leq \frac{1}{k} \tag{16}$$

Theorem 18 (Chebyshev's inequality)**.** *Let* X *be a random variable. For any* $k > 0$*:*

$$\mathbf{Pr}(|X - \mathbf{E}[X]| \geq k \cdot \sqrt{\mathbf{Var}[X]}) \leq \frac{1}{k^2} \tag{17}$$

Theorem 19. *Hoeffding bound] Let* $X_1, X_2, \ldots, X_n \in \{0, 1\}$ *be fully independent ran-dom variables. Let* $X = \sum_i X_i$*. Then:*

$$\mathbf{Pr}(|X - \mathbf{E}[X]| \geq t) \leq 2\exp\left(-\frac{t^2}{n}\right) \tag{18}$$