Massive Data Algorithmics

The Streaming Model

Lecture 16: Estimating Frequency Moments

The Problem

- Frequecy
  - Input: the stream $\sigma = < a_1, \cdots, a_m >$ where $a_i \in [n]$
  - Frequent number of item j: $f_j = |\{i : a_i = j\}|$,
  - Frequency vector: $F = (f_1, f_2, \cdots, f_n)$
  - Frequency moments: $F_k = ||F||_k^k = \sum_{j=1}^n f_j^k$
  - F_0: the number of distint items
  - F_1: the number of items (i.e. m)
- Problem: Estimating the frequency moments
  - Input: the stream $\sigma = < a_1, \cdots, a_m >$ where $a_i \in [n]$
  - Output: An estimation for $F_k = f_1^k + \ldots + f_n^k = ||F||_k^k$.

Note: we have to report an output upon arrival of a_i for any i

AMS Estimator

Initialize : $(m, r, a) \leftarrow (0, 0, 0)$;

Process j:

$m \leftarrow m + 1$;
$\beta \leftarrow$ random bit with $\Pr[\beta = 1] = 1/m$;
if $\beta = 1$ **then**
  $a \leftarrow j$;
  $r \leftarrow 0$;
if $j = a$ **then**
  $r \leftarrow r + 1$;

Output : $m(r^k - (r-1)^k)$;

General Idea

- Pick a token a from the stream σ u.a.r.
- Assume we pick a from the position i (i.e. $a_i = a$)
- Count the number of occurrences of a after the position i (i.e. $r = |\{k : k \geq i, a_k = a\}|$)
- The basic estimator of F_k is then defined to be $m(r^k - (r-1)^k)$

Analysis

- Lines 3-5 select a token a u.a.r.
  - The probability a_i to be selected is $\frac{1}{i} \cdot (1 - \frac{1}{i+1}) \cdot (1 - \frac{1}{i+2}) \cdots (1 - \frac{1}{m}) = \frac{1}{m}$
  - This is equivalent to (i) pick a random token $a \in [n]$ with $\Pr(a = j) = f_j/m$ for each $j \in [n]$, and then (ii) pick one of the f_a occurrences of a in σ u.a.r.
- Let A and R be the (random) values of a and r after the algorithm has proccessed σ, and let X be the output of the algorithm.
- Consider the event $A = j$ for some particular $j \in [n]$. R is equally likely to be any of the values $\{1, 2, \ldots, f_j\}$.
- Therefore, $E(X|A = j) = E(m(R^k - (R-1)^k)|A = j) = \sum_{i=1}^{f_j} \frac{1}{f_j} \cdot m(i^k - (i-1)^k) = \frac{m}{f_j}(f_j^k - 0^k)$
- $E(X) = \sum_{j=1}^{n} \Pr(A = j) E(X|A = j) = \sum_{j=1}^{n} \frac{f_j}{m} \cdot \frac{m}{f_j} \cdot f_j^k = F_k$

Analysis

- We must bound $Var(X)$ from above.
- $Var(X) \leq E(X^2) = \sum_{j=1}^{n} \frac{f_j}{m} \sum_{i=1}^{f_j} \frac{1}{f_j} \cdot m^2 (i^k - (i-1)^k)^2 = m \sum_{j=1}^{n} \sum_{i=1}^{f_j} (i^k - (i-1)^k)^2$
- We know $x^k - (x-1)^k \leq kx^{k-1}$
- Then, $Var(X) \leq m \sum_{j=1}^{n} \sum_{i=1}^{f_j} k i^{k-1} (i^k - (i-1)^k) \leq m \sum_{j=1}^{n} k f_j^{k-1} \sum_{i=1}^{(} i^k - (i-1)^k) = m \sum_{j=1}^{n} k f_j^{k-1} f_j^k = kF_1 F_{2k-1}$
- It is possible to show $Var(X) \leq kF_1 F_{2k-1} \leq kn^{1-1/k} F_k^2$ (For proof see the lecture note)

The Median-of-Means Improvement

- Unfortunately we can not apply the median trick. This is because the variance is so large that we are unable to bound below $1/2$ the probability if an ε relative deviation in th estimator.
- So, we must first bring the variance down by averaging a number of independent copies of the basic estimator and then apply the median trick.
- The next theorem quantifies this precisely.

The Median-of-Means Improvement

Theorem: Let $X_{i,j}$ and X be the independent random variable s.t. $E(X_{i,j}) = E(X) = Q$ where $i = 1, \cdots, t = O(\log(1/\delta))$, and $j = 1, \cdots, \ell = \frac{3Var(X)}{\varepsilon^2 E(X)^2}$. Let $Z = \text{median}_{1 \le i \le t}(\frac{1}{\ell}\sum_{j=1}^{\ell} X_{i,j})$. Then, we have $\Pr(|Z - Q| \ge \varepsilon Q) \le \delta$.

Proof:

- Let $Y_i = \frac{1}{\ell}\sum_{j=1}^{\ell} X_{i,j}$
- $E(Y_i) = Q$, $Var(Y_i) = \frac{1}{\ell^2}\sum_{j=1}^{\ell} Var(X_{i,j}) = \frac{Var(X)}{\ell}$
- $\Pr(|Y_i - Q|) \ge \varepsilon Q) \le \frac{Var(Y_i)}{(\varepsilon Q)^2} = \frac{Var(X)}{k\varepsilon^2 E(X)^2)} = \frac{1}{3}$

The Median-of-Means Improvement

- $\ell = \frac{3Var(X)}{\varepsilon^2 E(X)^2} \leq \frac{3kn^{1-1/k}F_k^2}{\varepsilon^2 F_k^2} = \frac{3k}{\varepsilon^2} n^{1-1/k}$
- space: $O(\frac{1}{\varepsilon^2} \cdot \log \frac{1}{\delta} \cdot kn^{1-1/k}(\log n + \log m))$

The General Idea Behind all Randomization Algorithms

- Assume we want to design a (ε, δ)-randomization approximation algorithm for estimating $f(\sigma)$.
- Design an algorithm whose ouput X is a random variable s.t. $E(X) = f(\sigma)$.
- To bound X is not far from $E(X)$, we need to compute $Var(X)$, and then apply the Chebyshev inequality to bound $\Pr(|X - E(X)| \geq \varepsilon\sqrt{Var(X)})$.
- To decrease the variance we can run k copies and then take the average
- To decrease the probability error we can run k copies and take the median

References

- **Data Stream Algorithms** (Chapter 5)
  Lecture notes by A. Chakrabbarti and D. College