Lecture 5: Sketch of Big Data

Xiaotie Deng

AIMS Lab
Department of Computer Science
Shanghai Jiaotong University

October 16, 2017

Synopsis Structures

Summary of Data

- Approximately Summarize the properties of data.
  - Random Sampling
  - Sketching
- Synopsis Structure Functionality:
  - Insert, delete, query
  - Merge databases
- Limitation in Sketching Algorithms
  - One or two passes of data in cpu.
  - Limited size of working cpu/memory

Data Stream

- Approximately Summarize the properties of data.
    - Random Sampling
    - Sketching
- Synopsis Structure Functionality:
    - Insert, delete, query
    - Easy to merging databases
- Applications:
    - Network Traffic Management
    - I/O Efficiency
    - Real Time Data

Frequent Elements

Description of Problem

- Data: $\{m_i : i = 1, 2, \cdots, n\}$ where m_i represents frequencies of the type i element.
- Output: the top-k last elements with m_i, $i = 0, 1, 2, \cdots, k$:
- Practicality: Power Law Property of Data.

Misra Gries Algorithm

- Place a counter on the first k distinct elements.
- On the $(k+1)$-st elements, reduce each counter by 1 and remove counters of value zero.
- Report counter value on any query.
- Estimation Error: at most $\frac{m-m'}{k+1}$ less where m total data, m' total data in structure.

Merge of Two Database

- Merge the common element counter, keep distinct counters.
- Remove small counters to keep k largest (by reducing counter then remove counters of value zero.
- Report counter value on any query.
- Estimation Error: at most $\frac{m-m'}{k+1}$ less where m total data, m' total data in structure.

Stream Counting

Morris Counting

- Standard: Use a register and increase by one on reading each item, taking space $O(\log n)$.
- Morris' idea: Tracking $\log n$ using $\log \log n$ bits.
  - Keep a counter x of value "$\log n$".
  - Increase the counter with probability $p = 2^{-x}$.
  - On a query, return $2^x - 1$.

Running Example

Input Data		a	b	c	d	e	f	g	h	i
Counter n	0	1	2	3	4	5	6	7	8	9
Counter x	0	1	1	2	2	2	2	2	3	3
Inc-prob p	1	.5	.5	.25	.25	.25	.25	.25	.125	.125
estimate $\tilde{n}$	0	1	1	3	3	3	3	3	7	7

Expected Returned Value

Theorem: Expected value after reading n input data is n.

- Base case: $n = 0$.
  - Expected returned value at time 0: $n = x = 0$ and $2^x - 1 = 0$
  - True value $n = 0$.
- Assume claim true for $n = k$: $EX[\tilde{n}] = n$.
- Consider $n = k + 1$
  - $EX[\tilde{n} + 1] = EX[2^{X_n}] = \sum_{all\ j \geq 1} P[X_{n-1} = j] EX[2^{X_n} | X_{n-1} = j]$
  - $EX[2^{X_n} | X_{n-1} = j] = P(X_n = j + 1) * 2^{j+1} + P(X_n = j) 2^j$
  - $EX[2^{X_n} | X_{n-1} = j] = 2^{-j} * 2^{j+1} + (1 - 2^{-j}) 2^j = 2^j + 1$
  - $EX[\tilde{n}] = \sum_{all\ j \geq 1} 2^j P[X_{n-1} = j] = EX[\tilde{k} + 1] = k + 1 = n$
- Therefore, $EX[\tilde{n}] = n$ for all $n \geq 0$.

Reference: http://www.cohenwang.com/edith/bigdataclass2013/lectures/lecture1.pdf

Count Distinct Items

The frequency moments of input sequence A

- Input sequence $A = \{a_1, a_2, \cdots, a_m\}$, $a_i \in N = \{1, 2, \cdots, n\}$.
    - $m_i = \{j : a_j = i\}$ represents frequencies of the type i element.
- Output: $F_k = \sum_{i=1}^{n} m_i^k$, $k = 0, 1, 2, \cdots$.
- F_0 number of distinct elements in list, F_1 length of sequence.
- F_2 Ginis index of homogeneity.
- $F_\infty^* = \max_{1 \le i \le n} m_i$.

General Theorem

- Theorem Computing an approximation Y of F_k on the sequence $A = \{a_1, a_2, \cdots, a_m\}$ of members of $N = \{1, 2, \cdots, n\}$ using $O(\frac{k \log 1/\epsilon)}{\lambda^2} n^{1-1/k} (\log n + \log m)$ memory bits, where Y deviates from F_k by more than λF_k is no more than ϵ.

Improved Performance Approximating Distinct Items

Fix a constant $c > 2$. Compute Y of approximation for F_0, the number of distinguished elements in the input sequence A.

- Memory requirement: $\log n$ bits
- Property of output: Probability that the ratio between Y and F_0 is not between $1/c$ and c is at most $2/c$. ($c \geq 2$).

Algorithm

- Choose d: $2^d > n$ and construct the finite field $F = GF(2^d)$.
- N represented as binary vectors of length d in F.
- Algorithm:
  - a, b randomly chosen from F.
  - $\forall a_i \in A$ (in the order of the input sequence), hash a_i to $z_i = a * a_i + b(mod\ F)$ represented by a d-vector in F.
    - z_i uniformly random in F.
  - Define $r_i = r(z_i) = \max\{i : 2^i | z_i\}$.
    - NOTE: there are only upto $\log n$ different values for r_is.
  - Define R to be the largest r_i over all elements of A.
    - $\log n$ different values for R needs $\log \log n$ bits.
- Output $Y = 2^R$.

Key Ideas

- $z_i = a * a_i + b (mod\ F)$ is a random variable in $GF(2^d)$.
- If $a_i = a'_i$, $z_i = z'_i$.
- As $0 < z_i < 2^{\log n}$, $0 \leq r(z_i) \leq \log n$, $0 \leq R \leq \log n$ R requires $\log \log n$ bits.
- Hash function on $GF(2^d)$ requires $\log n$ bits.
- The more distinct members are in A, the bigger the value R.

Construct Filed GF(2^d)

- Z_p for primes p, e.g., Z_2.
- Irreducible polynomials, and its representation by vector in F_2.
- Mathematical operations in $+, -, *, /$.
- An example $x^3 + x + 1 (mod\ 2)$.

Probabilistic Inequalities

- Markov Inequality: $Pr[X \geq d] \leq \frac{EX[X]}{d}$ for random variable $X \geq 0$.
  - $EX[X] = \int xf(x)dx \geq d * \int_{x \geq d} f(x)dx = d * Pr[X \geq d]$.
- Chebyshev Inequality: $Pr[|X - \mu| \geq k\sigma] \leq \frac{1}{k^2}$.
  - $LHS = Pr((X-\mu)^2 \geq k^2\sigma^2) \leq EX[(X-\mu)^2]/(k\sigma)^2 = \frac{1}{k^2}$
- Chernoff Bound: https://crypto.stanford.edu/ blynn/pr/chernoff.html

Correctness

- Let the correct answer is F_0, the set of distinct elements in A.
- Consider the probability Y deviate significantly
  - $r(z_i) \geq r$ holds with probability 2^{-r} (number of ending 0s).
  - $Pro[r(z_i) \geq r, r(z_j) \geq r] = 2^{-2r}$, as z_i's are pairwise independent.
- Define $W_x(r) = 1$ if $r(ax + b) \geq r$ and $Z_r \equiv \sum_{x \in F_0} W_x(r)$ is the number of variables which has at least r rightmost bits of all zeros in its binary representation.

Correctness II

- By linearity of expectation, $E[Z_r] = F_0/2^r$.
- By pairwise independence of r_i, r_j for $a_i \neq a_j$, the variance of Z_r, $\sigma^2(Z_r) = F_0 \frac{1}{2^r}(1 - \frac{1}{2^r}) < F_0/2^r$

Correctness III

- Choose the smallest constant r_c such that $2^{r_c} > cF_0$, $Pr(Y > cF_0) \leq Pr(Z_{r_c} \geq 1) \leq E[Z_r] = \frac{F_0}{2^r} < 1/c$ by
  - Markov Inequality: $Pro[X \geq a] \leq \frac{E[X]}{a}$.
- NOTE: r_c is chosen for the purpose of proof only. c is determined later.
- Next, consider the case: $c * 2^r < F_0$.

Correctness IV

- Here choose r_d the largest integer $r : 2^r < F_0/d$
- $Pr(Y \leq F_0/d) \leq Pr(Z_{r_d+1} = 0) \leq Var(Z_{r_d+1})/(E[Z_{r_d+1}])^2 < 1/E[Z_{r_d+1}] = 2^{r_d+1}/F_0 < \frac{2}{d}$ by Chebyshev's Inequality and conditions $Var(Z_r) < F_0/2^r = E(Z_r)$ and $F_0/2^{r_d} = E(Z_{r_d}) > d$.
- Use Chebyshev Inequality: $Pr(Z_{r_d+1} = 0) \leq Pr(|Z_{r_d+1} - E[Z_{r_d+1}]| \geq F_0/2^{r_d+1}) \leq \frac{VAR(Z_{r_d+1})}{(F_0/2^{r_d+1})^2}$.

Correctness V

- We now estimate the probability the output $Y = 2^R$
- $Pr[F_0/d \leq Y \leq cF_0] \leq 1 - (\frac{1}{c} + \frac{2}{d})$.
- The two inequalities bounds the probability Y is bounded between $1/d$ and $c \geq 0$ of the true value with probability $1 - (1/c + 2/d)$.

Algorithm II (second part)

1. Given input $a_1, a_2, \cdots, a_n \in [0, 1]$. Choose two random variables $X, Y \in U[0, 1]$ uniformly in $[0, 1]$. Compute $A_i = a_i X + Y - \lfloor a_i X + Y \rfloor$.
   - What is probability distribution of A_i?
   - Are A_i, A_j independent? What is their joint distribution?
   - What happens if $\forall i : a_i \in GF(p)$ and $X, Y \in GF(p)$ uniformly chosen, where $GF(p)$ is the prime field, consisting of $\{0, 1, 2 \cdots, p - 1\}$ under the (mod p) arithmetic operations.
2. Implement the algorithm for counting distinct element.
   - Show a step by step running example.
   - And find a appropriate choices of parameters c and d to achieve the best approximation with respect to the exact number.
   - Compare how close is the theoretical approximation with practical output.