CS 229r: Algorithms for Big Data Fall 2017-10-9 Monday

Lecture 5-Sketch of Big Data(1)

Prof. Xiaotie Deng *Scribe: Ranzhen Li*

1 Synopsis of Structure

1.1 Summary of Data

Statistical properties: Random sampling, sketchig.

Synopsis Structure Functionlity: Insert, delete, query and Merge databases.

Limitation(Sketching): One or two passes of data in cpu, limited size of working cpu or memory.

Applications: Network traffic management, I/O effciency and real time data.

2 Frequent Elements

2.1 Description of Problem

Brief: Read the data in one pass and output the top-k frequent element.

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

Simple Algorithm: Creat a counter for each distinct element and count plus 1 when its next occurrence.

Problem: The simple algorithm requires about $nlogm$ bits space, where n is the number of distinct elements and m is the total number of each element. When n counters are too large then only k counters are affordable. Hence we can consider the Misra Gries Algorithm.

2.2 Misra Gries Algorithm

Steps:

- 1. Place a counter on the first k distinct elements.
- 2. On the $(k+1)_{\text{st}}$ different element, reduce each counter by 1 and remove element which value of counters is zero.
- 3. Report counter value on any query.

Estimation Error: At most $\frac{m-m'}{k+1}$ less where m is the real total data's counts, m' total data's counts in structure. When meet different element, every count in current array should sub 1, so the total number should sub $k+1$. Then the times of sub 1 of an element at most $\frac{m-m'}{k+1}$.

Table 1: Example of Misra Gries Algorithm

Step No	current			not read								
1	a	b	c	d	a	b	c	c	e	f	d	a
	1	2	1									
2		b			a	b	c	c	e	f	d	a
	0	1	0									
3	a	b	c						e	f	d	a
	1	2	2									
4		b	c							f	d	a
	0	1	1									
5	f	b	c								d	a
	1	1	1									
6												a
	0	0	0									
7	a											
	1	0	0									
max Estimation Error: $\frac{m-m'}{k+1} = \frac{13-1}{4} = 3$												

Example: Show in Table.1

- Input data stream: a b c b d a b c c e f d a
- The size of counter(k): 3
- Total number: 13
- Top-k real frequents: a-3, b-3, c-3

2.3 Merge of Two Database

Steps:

- 1.Merge the common element counter, keep distinct counters.
- 2.Remove small counters to keep k largest (by reducing counter then remove counters of value zero.
- 3.Report counter value on any query.

Estimation Error: At most $\frac{m-m'}{k+1}$ less where m is the real total data's counts, m' total data's counts in structure.

Example: Show in Table.2

Table 2: Example of Merge of Two Database

Step No	group 1			group 2			total number in structrue
0	a	b	c	c	d	e	
	15	10	5	5	4	3	30
1	a	b	c		d	e	
	15	10	10		4	3	35
2	a	b	c			e	
	11	6	6			3	23
3	a	b	c				
	8	3	3				14
max Estimation Error							$\frac{m-m'}{k+1} = \frac{42-14}{4} = 7$
real Estimation Error							$a:7,\ b:7,\ c:7$

3 Stream Counting

3.1 Power Law Distribution

- Uniform $f(x) = 1,\ x \in [0, 1]$
- Normal $f(x) = e^{-x^2}$
- (negative) Exponential $f(x) = e^{-x},\ x \geq 0$
- Power Law $f(x) = c \cdot x^{-\alpha},\ x \geq 0$, usually $\alpha \in [2, 3]$

What happense when $\alpha = 1$?

$$f(x) = \frac{c}{x},\ x \geq 0$$

3.2 Morris Counting

Standard: Use a register and increase by one on reading each item, taking space $O(\log n)$.

Morris' idea: Instead of tracking n using $\log n$ bits, tracking $\log n$ using $\log\log n$ bits. When a counter is BIG, less bits are significant any more. For example, when $n = (100000000)_B$, it doesn't matter whether $n = (100000000)_B$ or $n = (100000010)_B$, the left most 1 (in the 9_{th} bit) is important.

Steps:

- 1. Keep a counter x of value "$\log n$"
- 2. Increase the counter with probability $p = 2^{-x}$.
- 3. On a query, return $2^x - 1$.

Example: Show in Table.3

3.3 Expected Returned Value

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

Table 3: Example of Morris Counting

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	0	0.5	0.5	0.25	0.25	0.25	0.25	0.25	0.125	0.125
Estimate $\hat{n}$	0	1	1	3	3	3	3	3	7	7

Proof:

- Base case: n = 0.
  - Exprected returned value at time 0: $n = x = 0$ and $2^x - 1 = 0$
  - True value $n = 0$.
- Assume claim true for $n = k : \mathbf{EX}[\hat{n}] = n$.
- Consider $n = k + 1$

$$\begin{aligned}
\mathbf{EX}[2^{X_n}|X_{n-1} = j] &= P[X_n = j+1]2^{j+1} + P[X_n = j]2^j \\
&= 2^{-j}2^{j+1} + (1 - 2^{-j})2^j \\
&= 2^j + 1
\end{aligned}$$

$$\begin{aligned}
\mathbf{EX}[\hat{n}] = \mathbf{EX}[2^{X_n} - 1] &= \sum_{\text{all},j\leq 1} P[X_{n-1} = j]\mathbf{EX}[2^{X_n}|X_{n-1} = j] - 1 \\
&= \sum_{\text{all},j\geq 1} P[X_{n-1} = j]\mathbf{EX}[2^{X_n}|X_{n-1} = j] - 1 \\
&= \sum_{\text{all},j\geq 1} P[X_{n-1} = j](2^j + 1) - 1 \\
&= \sum_{\text{all},j\geq 1} P[X_{n-1} = j]2^j + \sum_{\text{all},j\geq 1} P[X_{n-1} = j] - 1 \\
&= n
\end{aligned}$$

- Therefore, $\mathbf{EX}[\hat{n}] = n$ for all $n \geq 0$.

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

4 Count Distinct Items

Next course (2017-10-11 Wednesday)