CSCE 689: Special Topics in Modern Algorithms for Data Science

Lecture 12

Samson Zhou

Presentation Schedule

- September 25: Team DAP, Team Bokun, Team Jason
- September 27: Galaxy AI, Team STMI
- September 29: Jung, Anmol, Chunkai

Last Time: (ε, k)-Frequent Items Problem

- Goal: Given a set S of m elements from $[n]$, an accuracy parameter $\varepsilon \in (0, 1)$, and a parameter k, output a list that includes:
  - The items from $[n]$ that have frequency at least $\frac{m}{k}$
  - No items with frequency less than $(1 - \varepsilon)\frac{m}{k}$

(ε, k)-Frequent Items Problem

- Goal: Given a set S of m elements from $[n]$ that induces a frequency vector $f \in R^n$, an accuracy parameter $\varepsilon \in (0, 1)$, and a parameter k, output a list that includes:
  - The items from $[n]$ that have frequency at least $\frac{\|f\|_1}{k}$
  - No items with frequency less than $(1 - \varepsilon)\frac{\|f\|_1}{k}$

CountMin

- Initalization: Create b buckets of counters and use a random hash function $h: [n] \rightarrow [b]$
- Algorithm: For each update x_i, increment the counter $h(x_i)$

c_1	c_2	c_3	c_4
0	0	0	0

- At the end of the stream, output the counter $h(x_i)$ as the estimate for x_i

CountMin for (ε, k)-Frequent Items Problem

- Claim: For all estimated frequencies $\widehat{f_i}$ by CountMin, we have

$$f_i \leq \widehat{f_i} \leq f_i + \frac{\varepsilon \|f\|_1}{k}$$

- If $f_i \geq \frac{\|f\|_1}{k}$, then $\widehat{f_i} \geq f_i - \frac{\varepsilon \|f\|_1}{k}$ and if $f_i < (1-\varepsilon) \cdot \frac{\|f\|_1}{k}$, then $\widehat{f_i} < f_i - \frac{\varepsilon \|f\|_1}{k}$
- Returning coordinates V_t with $c_t \geq (1-\varepsilon) \cdot \frac{\|f\|_1}{k}$ means:
  - i with $f_i \geq \frac{\|f\|_1}{k}$ will be returned
  - NO i with $f_i < (1-\varepsilon) \cdot \frac{\|f\|_1}{k}$ will be returned

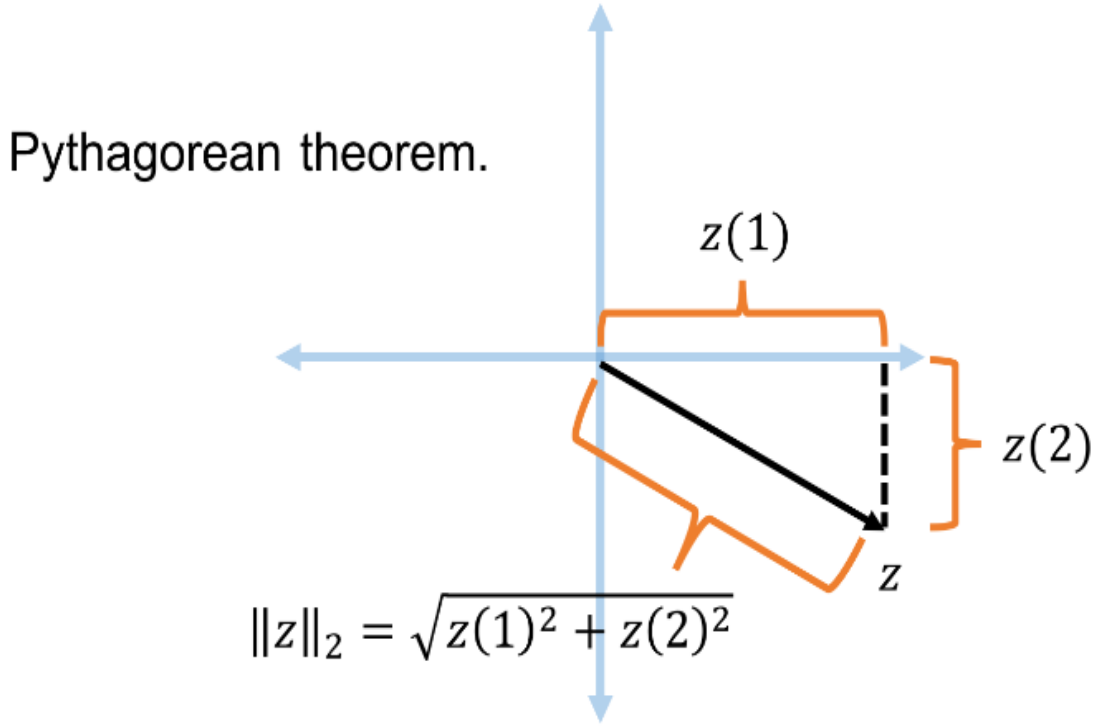
CountMin for (ε, k)-Frequent Items Problem

- Summary: CountMin can be used to solve the (ε, k)-frequent items problem on an insertion-deletion stream
- CountMin uses $O\left(\frac{k}{\varepsilon}\log n\right)$ bits of space
- CountMin is a deterministic algorithm
- CountMin *never* underestimates the true frequency

Recall: Euclidean Space and L_2 Norm

- For $z \in R^n$, the L_2 norm of z is denoted by $\|z\|_2$ and defined as:

$$\|z\|_2 = \sqrt{z_1^2 + z_2^2 + \cdots + z_n^2}$$



- For $x, y \in R^n$, the distance function D is denoted by $\|\cdot\|_2$ and defined as $\|x - y\|_2$

Trivia Question #7 (Norms)

- For $x \in R^n$, which of the following is (the most) true?

- $\|x\|_2 > \|x\|_1$
- $\|x\|_2 \geq \|x\|_1$
- $\|x\|_2 = \|x\|_1$
- $\|x\|_2 \leq \|x\|_1$
- $\|x\|_2 < \|x\|_1$
- None of these are true characterizations of the relationship between $\|x\|_2$ and $\|x\|_1$

Trivia Question #8 (Norms)

- For $x \in R^n$, how much large can $\|x\|_1/\|x\|_2$ be?

- $O(n)$
- $O(\sqrt{n})$
- $O(\log n)$
- $O(1)$

(ε, k)-Frequent Items Problem

- Goal: Given a set S of m elements from $[n]$ that induces a frequency vector $f \in R^n$, an accuracy parameter $\varepsilon \in (0, 1)$, and a parameter k, output a list that includes:
  - The items from $[n]$ that have frequency at least $\frac{\|f\|_1}{k}$
  - No items with frequency less than $(1 - \varepsilon)\frac{\|f\|_1}{k}$

L_2 Heavy-Hitters

- Goal: Given a set S of m elements from $[n]$ that induces a frequency vector $f \in R^n$, an accuracy parameter $\varepsilon \in (0,1)$, and a parameter k, output a list that includes:
  - The items from $[n]$ that have frequency at least $\frac{\|f\|_2}{k}$
  - No items with frequency less than $(1-\varepsilon)\frac{\|f\|_2}{k}$

L_2 Heavy-Hitters

- Goal: Given a set S of m elements from $[n]$ that induces a frequency vector $f \in R^n$ and a threshold parameter $\varepsilon \in (0,1)$, output a list that includes:
  - The items from $[n]$ that have frequency at least $\varepsilon \cdot \|f\|_2$
  - No items with frequency less than $\frac{\varepsilon}{2} \cdot \|f\|_2$

L_2 Estimation

- Goal: Given a set S of m elements from $[n]$ that induces a frequency vector $f \in R^n$ and an accuracy parameter $\varepsilon \in (0,1)$, output a $(1+\varepsilon)$-approximation to $\|f\|_2$

- Find Z such that $(1-\varepsilon) \cdot \|f\|_2 \leq Z \leq (1+\varepsilon) \cdot \|f\|_2$

- Find Z' such that $(1-\varepsilon) \cdot \|f\|_2^2 \leq Z' \leq (1+\varepsilon) \cdot \|f\|_2^2$

L_2 Estimation

- How to do?

L_2 Estimation

- How to do?

- Sorry, won't reveal until next lecture (wait, don't we already have a tool for this)?

- Assume for now we are given $\|f\|_2$

Revisiting CountMin

- Initalization: Create b buckets of counters and use a random hash function $h: [n] \to [b]$
- Algorithm: For each insertion (or deletion) to x_i, increment (or decrement) the counter $h(x_i)$

c_1	c_2	c_3	c_4
0	0	0	0

- At the end of the stream, output the counter $h(x_i)$ as the estimate for x_i

CountMin and the Power of Random Signs

- Initalization: Create b buckets of counters and use a random hash function $h: [n] \to [b]$ and a uniformly random sign function $s: [n] \to \{-1, +1\}$, i.e., $\Pr[s(i) = +1] = \Pr[s(i) = -1] = \frac{1}{2}$
- Algorithm: For each insertion (or deletion) to x_i, change the counter $h(x_i)$ by $s(x_i)$ (or $-s(x_i)$)

c_1	c_2	c_3	c_4
0	0	0	0

- At the end of the stream, output the quantity $s(x_i) \cdot h(x_i)$ as the estimate for x_i

CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
0	0	0	0	0	0	0

1

c_1	c_2	c_3	c_4
0	0	0	0

CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	0	0	0	0	0	0

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$

1

c_1	c_2	c_3	c_4
0	0	0	0

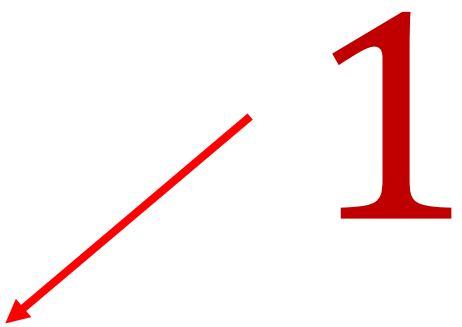
CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	0	0	0	0	0	0

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$



c_1	c_2	c_3	c_4
0	0	0	0

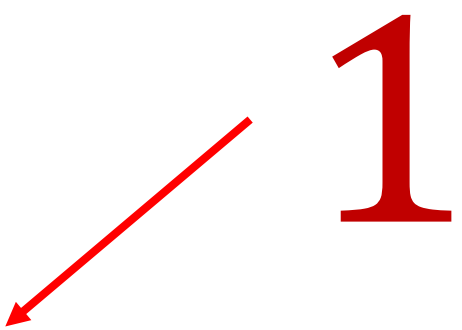
CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	0	0	0	0	0	0

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$



c_1	c_2	c_3	c_4
1	0	0	0

CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	0	0	0	0	0	0

3

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$

c_1	c_2	c_3	c_4
1	0	0	0

CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	0	1	0	0	0	0

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$

3

c_1	c_2	c_3	c_4
1	0	1	0

CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	0	1	0	0	0	0

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$

2

c_1	c_2	c_3	c_4
1	0	1	0

CountSketch

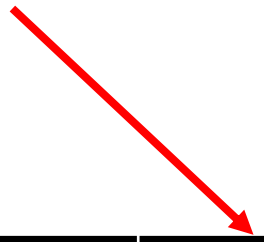
f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	1	1	0	0	0	0

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$

2



c_1	c_2	c_3	c_4
1	0	1	1

CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	1	1	0	0	0	0

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$

1

c_1	c_2	c_3	c_4
1	0	1	1

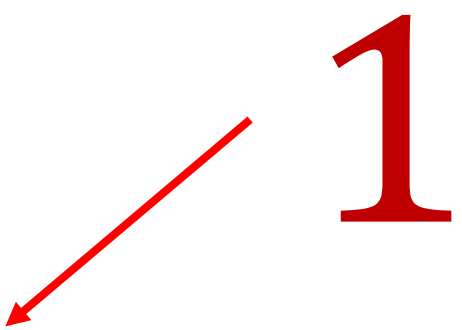
CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
2	1	1	0	0	0	0

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$



c_1	c_2	c_3	c_4
2	0	1	1

CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
2	1	1	0	0	0	0

5

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$

c_1	c_2	c_3	c_4
2	0	1	1

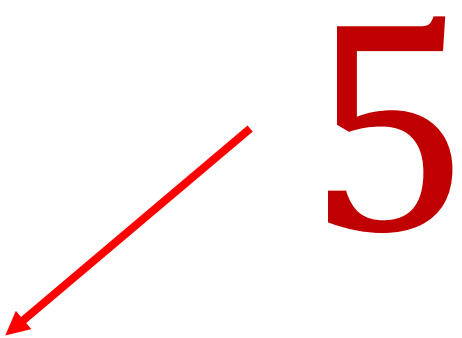
CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
2	1	1	0	1	0	0

$h(x) = 3x + 2 \pmod 4$

$s(x) = +1$ for $x \in \{1,2,3\}$

$s(x) = -1$ for $x \in \{4,5\}$



c_1	c_2	c_3	c_4
1	0	1	1

CountSketch

f_1	f_2	f_3	f_4	f_5	f_6	f_7
2	1	1	0	1	0	0

- What is the estimation for f_4?
- What about f_3?
- What about f_5? What about f_1?

$h(x) = 3x + 2 \pmod 4$
$s(x) = +1$ for $x \in \{1,2,3\}$
$s(x) = -1$ for $x \in \{4,5\}$

c_1	c_2	c_3	c_4
1	0	1	1