CSCE 689: Special Topics in Modern Algorithms for Data Science

Lecture 18

Samson Zhou

Presentation Schedule

- November 27: Chunkai, Jung, Galaxy AI
- November 29: STMI, Anmol, Jason
- December 1: Bokun, Team DAP

Previously: Distinct Elements (F_0 Estimation)

- Algorithm: Set $U_0 = [n]$ and for each i, sample each element of U_{i-1} into U_i with probability $\frac{1}{2}$
- Start index $i = 0$ and track the number $|S \cap U_i|$ of elements of S in U_i
- If $|S \cap U_i| > \frac{2000}{\varepsilon^2} \log n$, then increment $i = i + 1$
- At the end of the stream, output $2^i \cdot |S \cap U_i|$

Previously: Distinct Elements (F_0 Estimation)

- (S' is formed by sampling each item of S with probability p) With probability at least $\frac{9}{10}$,

$$pN - 100\sqrt{pN} \leq |S'| \leq pN + 100\sqrt{pN}$$

- Thus with probability at least $\frac{9}{10}$,

$$N - \frac{100}{\sqrt{p}}\sqrt{N} \leq \frac{1}{p}|S'| \leq N + \frac{100}{\sqrt{p}}\sqrt{N}$$

Previously: Distinct Elements (F_0 Estimation)

- (S' is formed by sampling each item of S with probability p) With probability at least $\frac{9}{10}$,

$$N - \frac{100}{\sqrt{p}}\sqrt{N} \le \frac{1}{p}|S'| \le N + \frac{100}{\sqrt{p}}\sqrt{N}$$

- If $\frac{100}{\sqrt{p}}\sqrt{N} \le \varepsilon N$, then $N - \frac{100}{\sqrt{p}}\sqrt{N} \le \frac{1}{p}|S'| \le N + \frac{100}{\sqrt{p}}\sqrt{N}$ implies

$$(1-\varepsilon)N \le \frac{1}{p}|S'| \le (1+\varepsilon)N$$

Last Time: Sparse Recovery

- Suppose we have an insertion-deletion stream of length $m = \Theta(n)$ and at the end we are promised there are at most k nonzero coordinates
- Goal: Recover the k nonzero coordinates and their frequencies

Last Time: Sparse Recovery

- Suppose at the end we are promised there are at most k nonzero coordinates
- Algorithm: Keep $2k$ running sum of different linear combinations of all the coordinates
- We have $2k$ equations and $2k$ unknown variables
- Correctness can be shown (not quite linear algebra)

Last Time: Sparse Recovery

- Suppose at the end we are promised there are at most k nonzero coordinates

- Algorithm: Keep $2k$ running sum of different linear combinations of all the coordinates

- Space: $O(k)$ words of space

L_0 Sampling

- Given a set S of m elements from $[n]$, let N be the number of distinct elements in S
- Goal: Return a random sample, so that each item from S is chosen with probability $\frac{1}{N} \pm \frac{1}{\mathrm{poly}(N)}$, say $\frac{1}{N} \pm \frac{1}{N^{1000}}$
- Motivation: Data summarization

L_0 Sampling

- Remember reservoir sampling? Does that work?

L_0 Sampling

- Remember reservoir sampling? Does that work? NO!

1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

L_0 Sampling

- Algorithm: What techniques have we learned? What is a good starting point?

Previously: Distinct Elements (F_0 Estimation)

- Algorithm: Set $U_0 = [n]$ and for each i, sample each element of U_{i-1} into U_i with probability $\frac{1}{2}$
- Start index $i = 0$ and track the number $|S \cap U_i|$ of elements of S in U_i
- If $|S \cap U_i| > \frac{2000}{\varepsilon^2} \log n$, then increment $i = i + 1$
- At the end of the stream, output $2^i \cdot |S \cap U_i|$

L_0 Sampling

- Algorithm: Run distinct elements algorithm and at the end of the stream, output a random element of $S \cap U_i$

Insertion-Deletion Streams

- How to perform L_0 estimation?

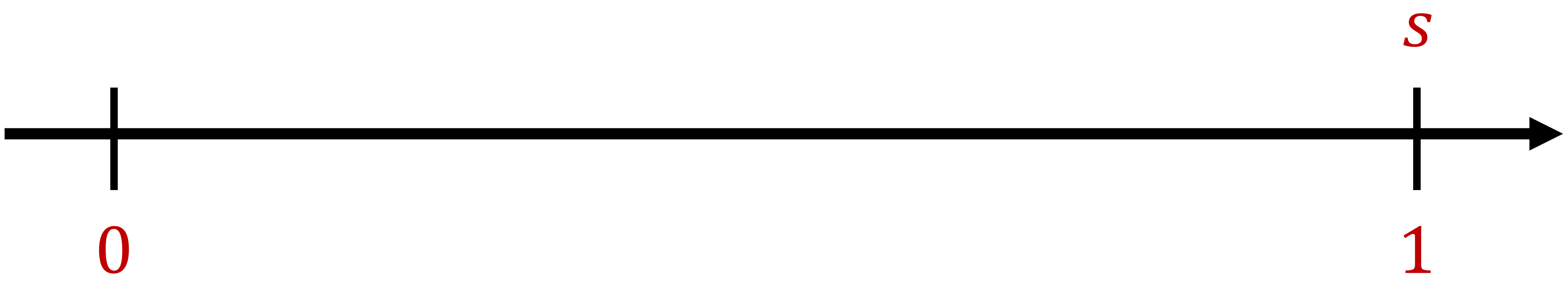
- How to perform L_0 sampling?

Distinct Elements (F_0 Estimation)

- Different, simpler algorithm on insertion-only streams

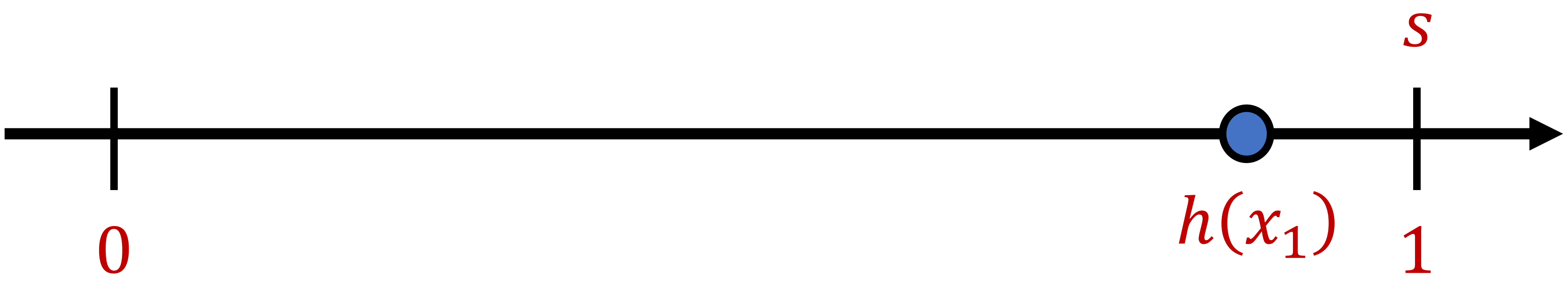
Distinct Elements (F_0 Estimation)

- Let $h: [n] \rightarrow [0,1]$ be a random hash function with a real-valued output
- Initialize $s = 1$
- For $x_1, \ldots, x_m$:
  - $s \leftarrow \min(s, h(x_i))$
- Return $Z = \frac{1}{s} - 1$



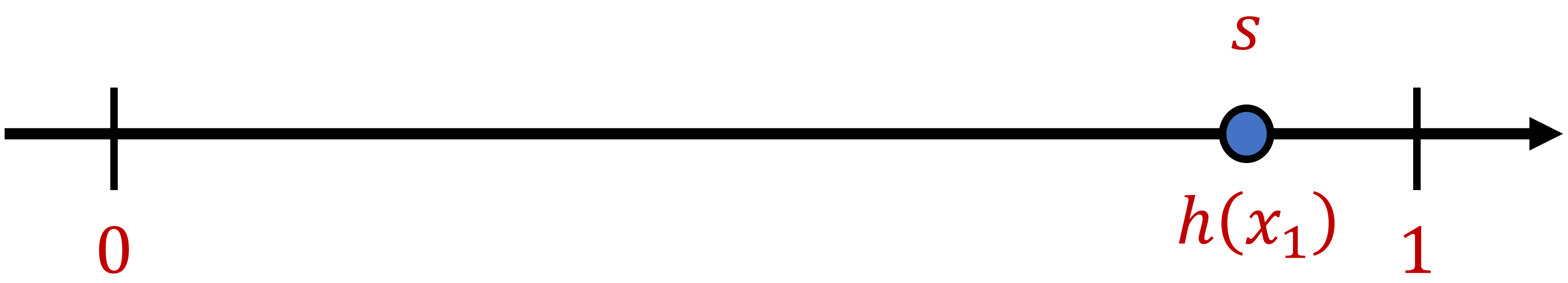
Distinct Elements (F_0 Estimation)

- Let $h: [n] \to [0,1]$ be a random hash function with a real-valued output
- Initialize $s = 1$
- For $x_1, \dots, x_m$:
  - $s \leftarrow \min\big(s, h(x_i)\big)$
- Return $Z = \frac{1}{s} - 1$



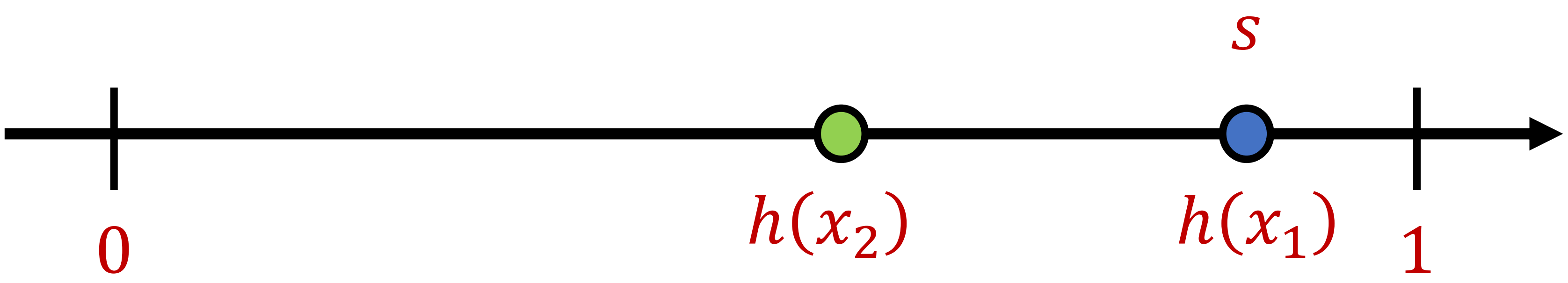
Distinct Elements (F_0 Estimation)

- Let $h: [n] \to [0,1]$ be a random hash function with a real-valued output
- Initialize $s = 1$
- For $x_1, \dots, x_m$:
  - $s \leftarrow \min\big(s, h(x_i)\big)$
- Return $Z = \frac{1}{s} - 1$



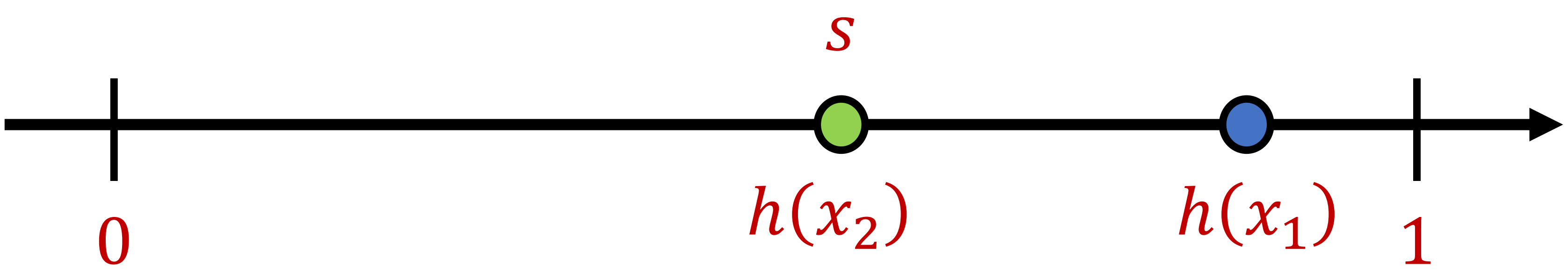
Distinct Elements (F_0 Estimation)

- Let $h: [n] \to [0,1]$ be a random hash function with a real-valued output
- Initialize $s = 1$
- For $x_1, \dots, x_m$:
  - $s \leftarrow \min\left(s, h(x_i)\right)$
- Return $Z = \frac{1}{s} - 1$



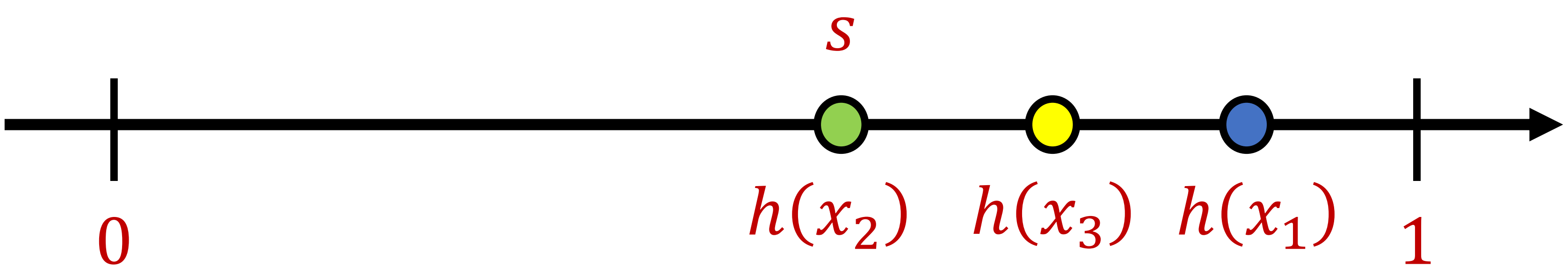
Distinct Elements (F_0 Estimation)

- Let $h: [n] \to [0,1]$ be a random hash function with a real-valued output
- Initialize $s = 1$
- For $x_1, \ldots, x_m$:
  - $s \leftarrow \min\left(s, h(x_i)\right)$
- Return $Z = \frac{1}{s} - 1$



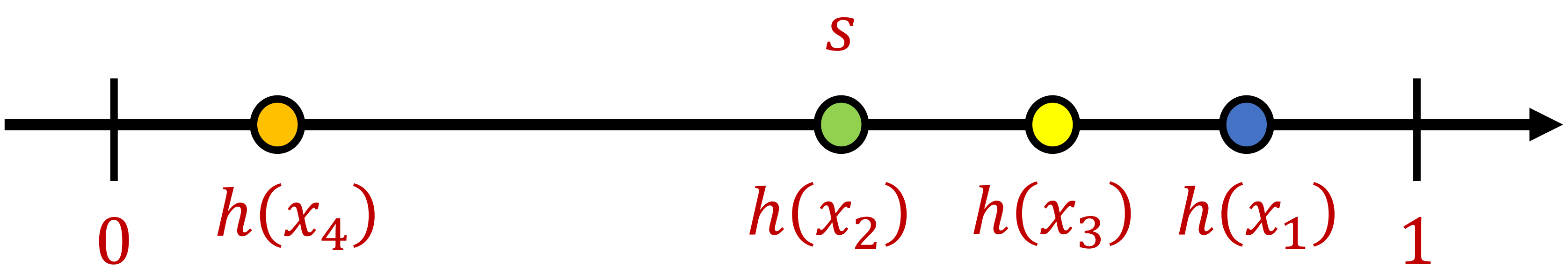
Distinct Elements (F_0 Estimation)

- Let $h: [n] \rightarrow [0,1]$ be a random hash function with a real-valued output
- Initialize $s = 1$
- For $x_1, \dots, x_m$:
  - $s \leftarrow \min\left(s, h(x_i)\right)$
- Return $Z = \frac{1}{s} - 1$



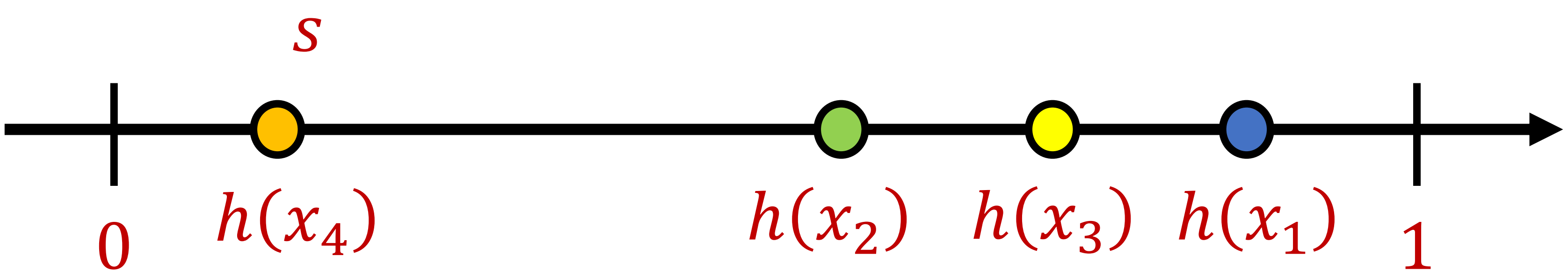
Distinct Elements (F_0 Estimation)

- Let $h: [n] \to [0,1]$ be a random hash function with a real-valued output
- Initialize $s = 1$
- For $x_1, \dots, x_m$:
  - $s \leftarrow \min\left(s, h(x_i)\right)$
- Return $Z = \frac{1}{s} - 1$



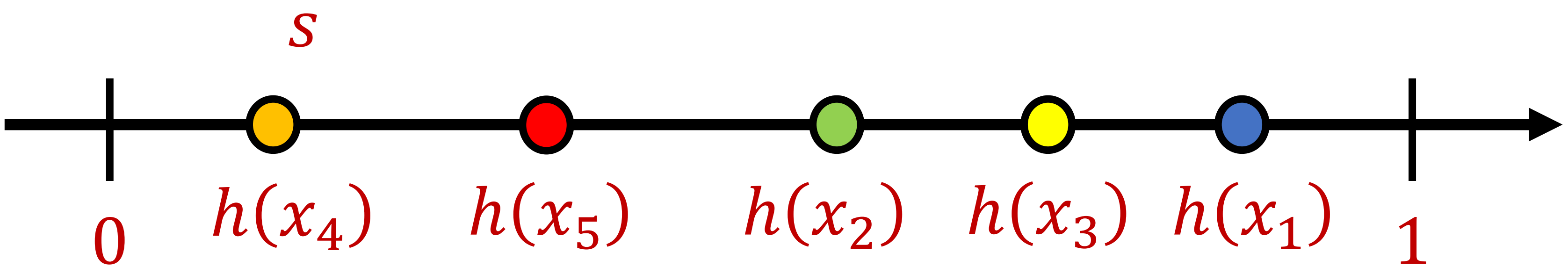
Distinct Elements (F_0 Estimation)

- Let $h: [n] \to [0,1]$ be a random hash function with a real-valued output
- Initialize $s = 1$
- For $x_1, \ldots, x_m$:
  - $s \leftarrow \min(s, h(x_i))$
- Return $Z = \frac{1}{s} - 1$



Distinct Elements (F_0 Estimation)

- Let $h: [n] \to [0,1]$ be a random hash function with a real-valued output
- Initialize $s = 1$
- For $x_1, \ldots, x_m$:
  - $s \leftarrow \min\big(s, h(x_i)\big)$
- Return $Z = \frac{1}{s} - 1$



Distinct Elements (F_0 Estimation)

- After all stream updates are processed, s is the minimum of N points chosen uniformly at random from $[0,1]$, where N is the number of distinct elements

- Intuition: The larger the value of N, the smaller we expect s to be

Distinct Elements (F_0 Estimation)

- Can show: $E[s] = \frac{1}{N+1}$
- Also can show that $|s - E[s]| \leq \varepsilon \cdot E[s]$ implies $(1 - 2\varepsilon)N \leq Z \leq (1 + 4\varepsilon)N$
- Can show: $\mathrm{Var}[s] \leq \frac{1}{(N+1)^2}$ so by taking the mean of $O\left(\frac{1}{\varepsilon^2}\right)$ independent instances, we get that $|s - E[s]| \leq \varepsilon \cdot E[s]$ with probability $\frac{2}{3}$

Distinct Elements (F_0 Estimation)

- Space guarantee: $O\left(\frac{1}{\varepsilon^2}\right)$ independent instance, each independent instance keeps a single word of space