Frequency moments and Counting Distinct Elements

Lecture 07 06
September 10, 2020
Part I

Estimating Distinct Elements
Distinct Elements

Given a stream $\sigma$ how many distinct elements did we see?

Offline solution via Dictionary data structure
Hashing based idea

- Assume idealized hash function: $h : [n] \rightarrow [0, 1]$ that is fully random over the real interval
- Suppose there are $k$ distinct elements in the stream
- What is the expected value of the minimum of hash values?
Analyzing idealized hash function

**Lemma**

Suppose $X_1, X_2, \ldots, X_k$ are random variables that are independent and uniformly distributed in $[0, 1]$ and let $Y = \min_i X_i$. Then $E[Y] = \frac{1}{(k+1)}$.

**DistinctElements**

Assume ideal hash function $h : [n] \rightarrow [0, 1]$

$y \leftarrow 1$

While (stream is not empty) do

Let $e$ be next item in stream

$y \leftarrow \min(z, h(e))$

EndWhile

Output $\frac{1}{y} - 1$
Analyzing idealized hash function

Lemma
Suppose $X_1, X_2, \ldots, X_k$ are random variables that are independent and uniformly distributed in $[0, 1]$ and let $Y = \min_i X_i$. Then

$$E[Y] = \frac{1}{(k+1)}.$$

Lemma
Suppose $X_1, X_2, \ldots, X_k$ are random variables that are independent and uniformly distributed in $[0, 1]$ and let $Y = \min_i X_i$. Then

$$E[Y^2] = \frac{2}{(k+1)(k+2)} \quad \text{and} \quad \text{Var}(Y) = \frac{k}{(k+1)^2(k+2)} \leq \frac{1}{(k+1)^2}.$$
Analyzing idealized hash function

Apply standard methodology to go from exact statistical estimator to good bounds:

- average $h$ parallel and independent estimates to reduce variance
- apply Chebyshev to show that the average estimator is a $(1 + \epsilon)$-approximation with constant probability
- use preceding and median trick with $O(\log 1/\delta)$ parallel copies to obtain a $(1 + \epsilon)$-approximation with probability $(1 - \delta)$

Total space: $O\left(\frac{1}{\epsilon^2} \log(1/\delta)\right)$ hash values to obtain an estimate that is within $(1 \pm \epsilon)$ approximation with probability at least $(1 - \delta)$.
Algorithm via regular hashing

Do not have idealized hash function.

- Use $h : [n] \rightarrow [N]$ for appropriate choice of $N$
- Use pairwise independent hash family $\mathcal{H}$ so that random $h \in \mathcal{H}$ can be stored in small space and computation can be done in small memory and fast

Several variants of idea with different trade offs between

- memory
- time to process each new element of the stream
- approximation quality and probability of success
Algorithm from BJKST

BJKST-DistinctElements:

\( H \) is a 2-universal hash family from \([n]\) to \([N = n^3]\)

choose \( h \) at random from \( H \)

\( t \leftarrow \frac{c}{\varepsilon^2} \)

While (stream is not empty) do

\( a_i \) is current item

Update the smallest \( t \) hash values seen so far with \( h(a_i) \)

endWhile

Let \( v \) be the \( t' \)th smallest value seen in the hast values.

Output \( tN/v \).
Algorithm from BJKST

**BJKST-DistinctElements:**

- \( \mathcal{H} \) is a 2-universal hash family from \([n]\) to \([N = n^3]\)
- Choose \( h \) at random from \( \mathcal{H} \)
- \( t \leftarrow \frac{c}{\varepsilon^2} \)

While (stream is not empty) do

- \( a_i \) is current item
  - Update the smallest \( t \) hash values seen so far with \( h(a_i) \)

endWhile

Let \( v \) be the \( t \)'th smallest value seen in the hash values.

Output \( tN/v \).

**Memory:** \( t = O(1/\varepsilon^2) \) values so \( O(\log n)/\varepsilon^2 \) bits. Also \( O(\log n) \) bits to store hash function.

**Processing time per element:** \( O(\log(1/\varepsilon)) \) comparisons of \( \log n \) bit numbers by using a binary search tree. And computing hash value.
Intuition for algorithm/analysis

Let $d$ be true number of distinct values in stream. Assume $d > c\epsilon^2$; can keep track of the exact count for small counts. How?
Intuition for algorithm/analysis

Let $d$ be true number of distinct value in stream. Assume $d > c\epsilon^2$; can keep track of the exact count for small counts. How?

Ideal hash function maps to real interval $[0, 1]$. Instead we map to integers in big range: $1$ to $N = n^3$. 

$$0 \to 0 \quad 1 \to N-1$$ 

$N = n^3$ 

$0 \sim 1 \quad 1 \sim \cdots \sim N$
Intuition for algorithm/analysis

Let $d$ be true number of distinct value in stream. Assume $d > c\varepsilon^2$; can keep track of the exact count for small counts. How?

Ideal hash function maps to real interval $[0, 1]$. Instead we map to integers in big range: 1 to $N = n^3$.

If $h$ were truly random min hash value is around $N/(d + 1)$.
Intuition for algorithm/analysis

Let $d$ be true number of distinct value in stream. Assume $d > c\varepsilon^2$; can keep track of the exact count for small counts. How?

Ideal hash function maps to real interval $[0, 1]$. Instead we map to integers in big range: 1 to $N = n^3$.

If $h$ were truly random min hash value is around $N/(d + 1)$

t’th minimum hash value $v$ to be around $tN/(d + 1)$. 
Intuition for algorithm/analysis

Let $d$ be true number of distinct value in stream. Assume $d > c\epsilon^2$; can keep track of the exact count for small counts. How?

Ideal hash function maps to real interval $[0, 1]$. Instead we map to integers in big range: 1 to $N = n^3$.

If $h$ were truly random min hash value is around $N/(d + 1)$

$t$'th minimum hash value $v$ to be around $tN/(d + 1)$. $\Rightarrow v$

Hence $tN/v$ should be around $d + 1$

$t$'th min hash value more robust estimator than minimum hash value and incorporates the averaging trick to reduce variance
Let $d$ be actual number of distinct values in a given stream (assume $d > c/\epsilon^2$). Let $D$ be the output of the algorithm which is a random variable.

$$D = \frac{\eta N}{\sqrt{c}}$$
Let $d$ be actual number of distinct values in a given stream (assume $d > c/\varepsilon^2$). Let $D$ be the output of the algorithm which is a random variable.

**Lemma**

$\Pr[D < (1 - \varepsilon)d] \leq 1/6.$

**Lemma**

$\Pr[D > (1 + \varepsilon)d] \leq 1/6.$

Hence $\Pr[|D - d| \geq \varepsilon d] < 1/3$. Can do median trick to reduce error probability to $\delta$ with $O(\log 1/\delta)$ parallel repetitions.
Analysis

For simplicity assume no collisions. Prove following as exercise.

**Lemma**

Since $N = n^3$ the probability that there are no collisions in $h$ is at least $1 - 1/n$.

Recall

**Lemma**

$X = X_1 + X_2 + \ldots + X_k$ where $X_1, X_2, \ldots, X_k$ are pairwise independent. Then $\text{Var}(X) = \sum_i \text{Var}(X_i)$.

$$
\frac{1}{1-\epsilon} = 1 + \epsilon + \epsilon^2 \ldots \Rightarrow 1 + \epsilon \leq \frac{1}{1-\epsilon} \leq 1 + \frac{3\epsilon}{2} \text{ for } \epsilon < 1/2.
$$

$$
\frac{1}{1+\epsilon} = 1 - \epsilon + \epsilon^2 \ldots \Rightarrow 1 - \epsilon \leq \frac{1}{1+\epsilon} \leq 1 - \frac{\epsilon}{2}.
$$
Let \( b_1, b_2, \ldots, b_d \) be the distinct values in the stream. Recall \( D = tN/v \) where \( v \) is the \( t \)'th smallest hash value seen.

- Each \( b_i \) hashed to a uniformly random bucket from 1 to \( N \)
- Consider buckets in interval \( I = [1..\frac{tN}{d}] \)
- Expected number of distinct items hashed into \( I \) is \( t \)
- Estimate \( D < (1 - \epsilon)d \) implies less than \( t \) hashed in interval \( I_1 = [1..\frac{tN}{(1-\epsilon)d}] \) when expected is \( \frac{t}{1-\epsilon} \)
- Estimate \( D > (1 + \epsilon)d \) implies more than \( t \) hashed in interval \( I_2 = [1..\frac{tN}{(1+\epsilon)d}] \) when expected is \( \frac{t}{(1+\epsilon)} \).
- Use Chebyshev to analyse “bad” event probabilities via pairwise independence of hash function.
Let $b_1, b_2, \ldots, b_d$ be the distinct values in the stream.
Recall $D = tN/v$ where $v$ is the $t$'th smallest hash value seen.

$D < (1 - \epsilon)d$ iff $v > \frac{tN}{(1-\epsilon)d}$. Implies less than $t$ hash values fell in the interval $I = [1..\frac{tN}{(1-\epsilon)d}]$. 

\[
\Pr[D < (1 - \epsilon)d] \leq 1/6.
\]
Analysis

Lemma

\[ \Pr[D < (1 - \epsilon)d] \leq \frac{1}{6}. \]

Let \( b_1, b_2, \ldots, b_d \) be the distinct values in the stream.
Recall \( D = \frac{tN}{v} \) where \( v \) is the \( t \)'th smallest hash value seen.

\[ D < (1 - \epsilon)d \text{ iff } v > \frac{tN}{(1-\epsilon)d}. \]
Implies less than \( t \) hash values fell in the interval \( I = [1..\frac{tN}{(1-\epsilon)d}] \). What is the probability of this event?

Let \( X_i \) be indicator for \( h(b_i) \leq \frac{tN}{(1-\epsilon)d} \).
And \( X = \sum_{i=1}^{d} X_i \) is number that hashed to \( I \)

\[ \Pr[D < (1 - \epsilon)d] = \Pr[X < t]. \]
Let $X_i$ be indicator for $h(b_i) \leq \frac{tN}{(1-\epsilon)d}$. And $X = \sum_{i=1}^{d} X_i$

Since $h(b_i)$ is uniformly distributed in $\{1, \ldots, N\}$, $E[X_i] = \Pr[X_i = 1] = \frac{t}{(1-\epsilon)d} \geq (1 + \epsilon)\frac{t}{d}$. 
Let $X_i$ be indicator for $h(b_i) \leq \frac{tN}{(1-\epsilon)d}$. And $X = \sum_{i=1}^{d} X_i$

- Since $h(b_i)$ is uniformly distributed in $\{1, \ldots, N\}$, $E[X_i] = \Pr[X_i = 1] = \frac{t}{(1-\epsilon)d} \geq (1 + \epsilon)t/d$.
- $E[X] \geq (1 + \epsilon)t$. 

Recall $\Pr[D < (1 - \epsilon)d] = \Pr[X < t]$.

Thus $D < (1 - \epsilon)d$ only if $X < t$. Use Chebyshev to upper bound this probability.
Let $X_i$ be indicator for $h(b_i) \leq \frac{tN}{(1-\epsilon)d}$. And $X = \sum_{i=1}^{d} X_i$

- Since $h(b_i)$ is uniformly distributed in $\{1, \ldots, N\}$, 
  $E[X_i] = \Pr[X_i = 1] = \frac{t}{(1-\epsilon)d} \geq (1 + \epsilon) t/d$.
- $E[X] \geq (1 + \epsilon) t$.

Recall $\Pr[D < (1 - \epsilon)d] = \Pr[X < t]$ 

Thus $D < (1 - \epsilon)d$ only if $X - E[X] < \epsilon t$. Use Chebyshev to upper bound this probability.
Let $X_i$ be indicator for $h(b_i) \leq \frac{tN}{(1-\epsilon)d}$. And $X = \sum_{i=1}^{d} X_i$

- Since $h(b_i)$ is uniformly distributed in $\{1, \ldots, N\}$, $E[X_i] = \Pr[X_i = 1] = \frac{t}{(1-\epsilon)d} \geq (1 + \epsilon/2)t/d$.

- $E[X] \geq (1 + \epsilon)t$.

- $X_i$ is a binary rv hence $Var(X_i) \leq E[X_i] \leq (1 + 3\epsilon/2)t/d$. 


Analysis

Let $X_i$ be indicator for $h(b_i) \leq \frac{tN}{(1-\epsilon)d}$. And $X = \sum_{i=1}^{d} X_i$

- Since $h(b_i)$ is uniformly distributed in $\{1, \ldots, N\}$,
  $E[X_i] = \Pr[X_i = 1] = \frac{t}{(1-\epsilon)d} \geq (1 + \epsilon/2)t/d$

- $E[X] \geq (1 + \epsilon)t$.

- $X_i$ is a binary rv hence $\text{Var}(X_i) \leq E[X_i] \leq (1 + 3\epsilon/2)t/d$.

- $X_1, X_2, \ldots, X_d$ are pair-wise independent random variables hence $\text{Var}(X) = \sum_i \text{Var}(X_i) \leq (1 + 3\epsilon/2)t$. 

---

Chandra (UIUC) CS498ABD Fall 2020 16 / 20
Analysis

Let \( X_i \) be indicator for \( h(b_i) \leq \frac{tN}{(1-\epsilon)d} \). And \( X = \sum_{i=1}^{d} X_i \)

- Since \( h(b_i) \) is uniformly distributed in \( \{1, \ldots, N\} \), \( E[X_i] = Pr[X_i = 1] = \frac{t}{(1-\epsilon)d} \geq (1 + \epsilon/2) t/d \)
- \( E[X] \geq (1 + \epsilon) t. \)
- \( X_i \) is a binary rv hence \( \text{Var}(X_i) \leq E[X_i] \leq (1 + 3\epsilon/2) t/d. \)
- \( X_1, X_2, \ldots, X_d \) are pair-wise independent random variables hence \( \text{Var}(X) = \sum_{i} \text{Var}(X_i) \leq (1 + 3\epsilon/2) t. \)

By Chebyshev:

\[
\Pr[X < t] \leq \Pr[|X - E[X]| > \epsilon t] \leq \frac{\text{Var}(X)}{\epsilon^2 t^2} \leq \frac{(1 + 3\epsilon/2)}{c}
\]
Let $X_i$ be indicator for $h(b_i) \leq \frac{tN}{(1-\epsilon)d}$. And $X = \sum_{i=1}^{d} X_i$

- Since $h(b_i)$ is uniformly distributed in $\{1, \ldots, N\}$, 
  $E[X_i] = \Pr[X_i = 1] = \frac{t}{(1-\epsilon)d} \geq (1 + \epsilon/2)t/d$

- $E[X] \geq (1 + \epsilon)t$.

- $X_i$ is a binary rv hence $Var(X_i) \leq E[X_i] \leq (1 + 3\epsilon/2)t/d$.

- $X_1, X_2, \ldots, X_d$ are pair-wise independent random variables hence $Var(X) = \sum_i Var(X_i) \leq (1 + 3\epsilon/2)t$.

By Chebyshev:

$$Pr[X < t] \leq Pr[|X - E[X]| > \epsilon t] \leq \frac{Var(X)}{\epsilon^2 t^2} \leq \frac{(1 + 3\epsilon/2)/c}{1/6}$$

Choose $c$ sufficiently large to ensure ratio is at most $1/6$. 
Lemma

\[ \Pr[D > (1 + \epsilon)d] \leq 1/6]. \]

Let \( b_1, b_2, \ldots, b_d \) be the distinct values in the stream.
Recall \( D = tN/v \) where \( v \) is the \( t \)'th smallest hash value seen.

\[ D > (1 + \epsilon)d \iff v < \frac{tN}{(1+\epsilon)d}. \]

Implies more than \( t \) hash values fell in the interval \([1..\frac{tN}{(1+\epsilon)d}]\).
Analysis

Lemma

\[ \Pr[D > (1 + \epsilon)d] \leq 1/6]. \]

Let \( b_1, b_2, \ldots, b_d \) be the distinct values in the stream. Recall \( D = tN/v \) where \( v \) is the \( t \)'th smallest hash value seen.

\[ D > (1 + \epsilon)d \text{ iff } v < \frac{tN}{(1+\epsilon)d}. \] Implies more than \( t \) hash values fell in the interval \([1..\frac{tN}{(1+\epsilon)d}]\). What is the probability of this event?

Let \( X_i \) be indicator for \( h(b_i) \leq \frac{tN}{(1+\epsilon)d} \).

And \( X = \sum_{i=1}^{d} X_i \)

\[ \Pr[D > (1 + \epsilon)d] = \Pr[Y > t]. \]
Let $X_i$ be indicator for $h(b_i) \leq \frac{tN}{(1+\epsilon)d}$. And $X = \sum_{i=1}^{d} X_i$

- Since $h(b_i)$ is uniformly distributed in $\{1, \ldots, N\}$, $E[X_i] = \Pr[X_i = 1] = \frac{t}{(1+\epsilon)d} \leq (1 - \epsilon/2)t/d$.
- $E[X] \leq (1 - \epsilon/2)t$.
- $X_i$ is a binary rv hence $Var(X_i) \leq E[X_i] \leq (1 - \epsilon/2)t/d$.
- $X_1, X_2, \ldots, X_d$ are pair-wise independent random variables hence $Var(X) = \sum_i Var(X_i) \leq (1 - \epsilon/2)t$. 

By Chebyshev:
\[ \Pr[X > t] \leq \frac{1}{4} \frac{Var(X)}{t^2} \leq \frac{1}{4} \frac{(1 - \epsilon/2)}{c} \]

Choose $c$ sufficiently large to ensure ratio is at most $1/6$. 

Chandra (UIUC)  
CS498ABD  
Fall 2020
Let $X_i$ be indicator for $h(b_i) \leq \frac{tN}{(1+\epsilon)d}$. And $X = \sum_{i=1}^{d} X_i$

- Since $h(b_i)$ is uniformly distributed in $\{1, \ldots, N\}$, 
  $E[X_i] = \Pr[X_i = 1] = \frac{t}{(1+\epsilon)d} \leq (1 - \epsilon/2)t/d$.
- $E[X] \leq (1 - \epsilon/2)t$.
- $X_i$ is a binary rv hence $Var(X_i) \leq E[X_i] \leq (1 - \epsilon/2)t/d$.
- $X_1, X_2, \ldots, X_d$ are pair-wise independent random variables hence $Var(X) = \sum_i Var(X_i) \leq (1 - \epsilon/2)t$.

By Chebyshev:

$$\Pr[X > t] \leq \Pr[|X - E[X]| > \epsilon t/2] \leq 4 Var(X)/\epsilon^2 t^2 \leq 4(1 - \epsilon/2)/c$$
Let $X_i$ be indicator for $h(b_i) \leq \frac{tN}{(1+\epsilon)d}$. And $X = \sum_{i=1}^{d} X_i$

- Since $h(b_i)$ is uniformly distributed in $\{1, \ldots, N\}$, $E[X_i] = \Pr[X_i = 1] = \frac{t}{(1+\epsilon)d} \leq (1 - \epsilon/2)t/d$.
- $E[X] \leq (1 - \epsilon/2)t$.
- $X_i$ is a binary rv hence $\text{Var}(X_i) \leq E[X_i] \leq (1 - \epsilon/2)t/d$.
- $X_1, X_2, \ldots, X_d$ are pair-wise independent random variables hence $\text{Var}(X) = \sum_i \text{Var}(X_i) \leq (1 - \epsilon/2)t$.

By Chebyshev:

$$\Pr[X > t] \leq \Pr[|X - E[X]| > \epsilon t/2] \leq 4 \text{Var}(X)/\epsilon^2 t^2 \leq 4(1 - \epsilon/2)/c$$

Choose $c$ sufficiently large to ensure ratio is at most $1/6$. 
Where did we use the fact that $d \geq c/\epsilon^2$?
Where did we use the fact that \( d \geq c/\epsilon^2 \)?

Analysis need to be more careful in using \( \frac{N}{(1-\epsilon)d} \) and \( \frac{N}{(1+\epsilon)d} \) since we need to round them to nearest integer; technically have to use floor and ceilings. If \( d > c/\epsilon^2 \) then rounding error of 1 does not matter — adds only \( \epsilon d \) error.

We avoid floor and ceiling etc in lecture for clarity.
Summary on Distinct Elements

- with \( O\left(\frac{1}{\varepsilon^2} \log(1/\delta) \log n \right) \) bits algorithm output estimate \( D \) such that \(|D - d| \leq \varepsilon d\) with probability at least \((1 - \delta)\)

- Best known memory bound: \( O\left(\frac{\log(1/\delta)}{\varepsilon^2} + \log n \right) \) bits and for any fixed \( \delta \) this meets lower bound within constant factors. Both lower bound and upper bound quite technical — potential reading for projects.

- Continuous monitoring: want estimate to be correct not only at end of stream but also at all intermediate steps. Can be done with \( O\left(\frac{\log \log n + \log(1/\delta)}{\varepsilon^2} + \log n \right) \) bits.

- Deletions allowed! Can also be done. More on this later.