Streaming

Inge Li Gørtz

Counting (Distinct) Elements in a Stream

- Applications
  - IP traffic logs
    - How many distinct IP addresses used a given link to send their traffic from the beginning of the day, or how many distinct IP addresses are currently using a given link on ongoing flow?
    - How many flows comprised one packet only (i.e., rare flows)?
    - What are the top k heaviest flows during the day, or currently in progress?
  - Search engine query logs.
    - How many distinct queries in a list of queries?

Probabilistic Counting

Probabilistic Counting

- Counting. Count number of elements in the stream.

- Exact. Need $\lceil \lg m \rceil$ bits.

Probabilistic Counting

- Exercise. Alice is thinking of a number between 0 and m. She wants to tell Bob which number she is thinking of, but can only use a limited number of bits.
  - Exact. Need $\lceil \lg m \rceil$ bits.
  - Approximate. What is the best estimate Bob can get if Alice can only use:
    - $\lceil \lg m \rceil - 1$ bits?

Probabilistic Counting

- Exercise. Alice is thinking of a number between 0 and m. She wants to tell Bob which number she is thinking of, but can only use a limited number of bits.
    - Exact. Need $\lceil \lg m \rceil$ bits.
    - Approximate. What is the best estimate Bob can get if Alice can only use:
        - $\lceil \lg m \rceil - 1$ bits?

Probabilistic Counting

- Exercise. Alice is thinking of a number between 0 and m. She wants to tell Bob which number she is thinking of, but can only use a limited number of bits.
  - Exact. Need $\lceil \lg m \rceil$ bits.
  - Approximate. What is the best estimate Bob can get if Alice can only use:
    - $\lceil \lg m \rceil - 1$ bits?
    - $\lceil \lg \lg m \rceil$ bits?

Probabilistic Counting

- Algorithm.

$X \leftarrow 0$
while (stream is not empty) do
    toss a biased coin that is heads with probability $1/2^X$
    if heads then
        $X \leftarrow X + 1$
Output $2^X - 1$

- X_i = value of X after i elements seen. Let $Y_i = 2^{X_i}$.
- Claim. $E[Y_m] = m + 1$.
- Expected number of bits needed: $E[\log X_m] = E[\log \log Y_m] = O(\log \log m)$.

Probabilistic Counting

- Claim. $E[Y_m] = m + 1$.
- Proof. By induction on m.

```
X ← 0
while (stream is not empty) do
    toss a biased coin that is heads with probability 1/2^X
    if heads then
        X ← X + 1
Output 2^X − 1
```

$$E[Y_m] = E[2^{X_m}] = \sum_{j=0}^{\infty} 2^j \cdot P[X_m = j]$$

$$= \sum_{j=0}^{\infty} 2^j \cdot \left(P[X_{m-1} = j] \cdot (1 - \frac{1}{2^j}) + P[X_{m-1} = j - 1] \cdot \frac{1}{2^{j-1}} \right)$$

$$= \sum_{j=0}^{\infty} 2^j \cdot P[X_{m-1} = j] + \sum_{j=0}^{\infty} 2^j \cdot \left((2 \cdot P[X_{m-1} = j-1] - P[X_{m-1} = j]) \cdot \frac{1}{2^j} \right)$$

$$= \sum_{j=0}^{\infty} 2^j \cdot P[X_{m-1} = j] + \sum_{j=0}^{\infty} \left(2 \cdot P[X_{m-1} = j-1] - P[X_{m-1} = j] \right)$$

$$= E[Y_{m-1}] + 1$$

$$= (m - 1 + 1) + 1 \quad = m + 1$$

Counting Distinct Elements

Counting Distinct Elements

- Goal. Output an (ε, δ)-estimate of the number d of distinct elements in the stream.

- (ε, δ)-estimate.

$$P\left[\left|\frac{A(s)}{d} - 1\right| > \varepsilon\right] < \delta$$

  where $A(s)$ is the output of algorithm A on stream s.

- AMS Algorithm.
  - Simple
  - Median trick
  - Tail bounds

Pairwise Independent Hash Functions

- Pairwise Independent Hash Functions. A family of functions $\mathscr{H} = \{h \mid h : U \to [m]\}$ is pairwise independent if the following two conditions hold:
    1. $\forall x \in U$, the random variable $h(x)$ is uniformly distributed in $[m]$,
    2. $\forall x \neq y \in U$, the random variables $h(x)$ and $h(y)$ are independent.

- Pairwise Independent Hash Functions. A hash function $h : U \to [m]$ is pairwise independent if for all $x \neq y \in U$ and $q, r \in [m]$:

$$P[h(x) = q \wedge h(y) = r] = \frac{1}{m^2}$$

AMS algorithm

- zeros(p). Number of zeros that the binary representation of p ends with.
- Intuition. Assume we have a large stream s of uniformly distributed numbers from [m].
  - 1/2 of the numbers ends with 0.
  - 1/4 of the numbers ends with 00.
  - 1/8 of the numbers ends with 000.
  - ……

Therefore: let z = $\max_{x \in s} \text{zeros}(x)$

- If z = 1, then it is likely that the number of distinct integers is $2^1 = 2$.
- If z = 2, then it is likely that the number of distinct integers is $2^2 = 4$.
- If z = 3, then it is likely that the number of distinct integers is $2^3 = 8$.
- …….

AMS Algorithm

- AMS Algorithm

```
Choose a random function h: [n] → [n] from a family of pairwise independent hash functions
z←0
while (an item x arrives) do
      if zeros(h(x)) > z then
         z ← zeros(h(x))

Output 2^{z+1/2}
```

- Let z' be the value of z when algorithm ends and $d' = 2^{z+1/2}$ be the estimate returned by the algorithm.
- Want to bound probability that d' is far from d:

$$P[d' \geq 3d] \text{ and } P[d' \leq d/3]$$

AMS Algorithm

- Let z' be the value of z when algorithm ends and $d' = 2^{z+1/2}$ be the estimate returned by the algorithm.
- Want to bound probability that d' is far from d'.
    - Bound

$$P[d' \geq 3d] \text{ and } P[d' \leq d/3]$$

```
Choose a random function h: [n] → [n] from a family of
pairwise independent hash functions
z←0
while (an item x arrives) do
        if zeros(h(x)) > z then
            z ← zeros(h(x))

Output 2^{z+1/2}
```

AMS Algorithm

- Let z' be the value of z when algorithm ends and $d' = 2^{z+1/2}$ be the estimate returned by the algorithm.
- Want to bound probability that d' is far from d'.
  - Bound

$$P[d' \geq 3d] \text{ and } P[d' \leq d/3]$$

- Let a be the smallest integer such that $2^{a+1/2} \geq 3d$.
- Let b be the smallest integer such that $2^{b+1/2} \leq d/3$.
- Let Y_r = #distinct items in the stream such that zeros$(h(j)) \geq r$.
- Then

$$P[d' \geq 3d] = P[z' \geq a] = P[Y_a > 0] = ??$$

and

$$P[d' \leq d/3] = P[z' \leq b] = P[Y_{b+1} = 0] = ??$$

Choose a random function h: [n] → [n] from a family of pairwise independent hash functions
z←0
while (an item x arrives) do
if zeros(h(x)) > z then
z ← zeros(h(x))

Output $2^{z+1/2}$

AMS Algorithm Analysis

- Goal. Bound $P[Y_a > 0]$ and $P[Y_{b+1} = 0]$.

- Define

$$X_{r,j} = \begin{cases} 1 & \text{if zeros}(h(j)) \geq r \\ 0 & \text{otherwise} \end{cases} \implies Y_r = \sum_{j:f_j>0} X_{r,j}$$

- Expected value of $X_{j,r}$

$$E[X_{r,j}] = P[\text{zeros}(h(j)) \geq r] = \frac{1}{2^r}$$

- Expected value of Y_r

$$E[Y_r] = E[\sum_{j:f_j>0} X_{r,j}] = \sum_{j:f_j>0} \frac{1}{2^r} = \frac{d}{2^r}$$

- Variance of Y_r

$$\text{Var}[Y_r] = \sum_{j:f_j>0} \text{Var}[X_{r,j}] \leq \sum_{j:f_j>0} E[X_{r,j}^2] = \sum_{j:f_j>0} E[X_{r,j}] = \frac{d}{2^r}$$

AMS Algorithm Analysis

- Goal. Bound $P[Y_a > 0]$ and $P[Y_{b+1} = 0]$.
- Have $E[Y_r] = \text{Var}[Y_r] = \dfrac{d}{2^r}$
- By Markov's inequality

$$P[Y_a > 0] = P[Y_a \geq 1] \leq \frac{E[Y_a]}{1} = \frac{d}{2^a} \leq \frac{\sqrt{2}}{3}$$

- By Chebychev's inequality

$$P[Y_{b+1} = 0] = P[\,|\,Y_{b+1} - E[Y_{b+1}]\,|\geq d/2^{b+1}]$$

$$= P\left[\,|\,Y_{b+1} - E[Y_{b+1}]\,| \geq \frac{d/2^{b+1}}{\sqrt{\text{Var}[Y_{b+1}]}} \cdot \sqrt{\text{Var}[Y_{b+1}]}\right]$$

$$\leq \frac{\text{Var}[Y_{b+1}]}{(d/2^{b+1})^2} = \frac{2^{b+1}}{d} \leq \frac{\sqrt{2}}{3}$$

Markov's inequality

$$P[X \geq t] \leq \frac{E[X]}{t}$$

Chebychev's inequality

$$P[\,|\,X - E[X]\,| \geq t\sqrt{\text{Var}[X]}] \leq \frac{1}{t^2}$$

Median trick

- Run $O(\log(1/\delta))$ parallel and independent copies of the algorithm and output the median.
- The probability of success is at least $1 - \delta$.
- Gives a $(1/3, \delta)$- estimate.
- Time and space. $O(\log(1/\delta)\log n)$.