Distinct Elements

- We are back to our basic streaming model: $i_1, \ldots, i_n \in [d] = \{1, \cdots, d\}$.
- The frequency vector $x \in \mathbb{Z}^d$: $x_j = |\{t : i_t = j\}|$.

Distinct Elements

- We are back to our basic streaming model: $i_1, \ldots, i_n \in [d] = \{1, \cdots, d\}$.
- The frequency vector $x \in \mathbb{Z}^d$: $x_j = |\{t : i_t = j\}|$.
- *Counting distinct elements*: estimate $||x||_0 := |j : x_j > 0|$ precise up to $(1+\epsilon)$-factor.

Distinct Elements

- We are back to our basic streaming model: $i_1, \ldots, i_n \in [d] = \{1, \cdots, d\}$.
- The frequency vector $x \in \mathbb{Z}^d$: $x_j = |\{t : i_t = j\}|$.
- *Counting distinct elements*: estimate $||x||_0 := |j : x_j > 0|$ precise up to $(1+\epsilon)$-factor.
- Again, we must use space $O(\log d, \frac{1}{\epsilon})$.

An Ideal Algorithm

- If we can make the distribution of $\{i_t\}$ uniform, then it is easier to estimate its size.

An Ideal Algorithm

- If we can make the distribution of $\{i_t\}$ uniform, then it is easier to estimate its size.
- E.g., if we can have an ideal hash function h from $[d]$ to $[0, 1]$, let X be the minimum $h(i_t)$, then $\frac{1}{X}$ seems a reasonable estimate.

An Ideal Algorithm

- If we can make the distribution of $\{i_t\}$ uniform, then it is easier to estimate its size.
- E.g., if we can have an ideal hash function h from $[d]$ to $[0, 1]$, let X be the minimum $h(i_t)$, then $\frac{1}{X}$ seems a reasonable estimate.
- Indeed, suppose we have i.i.d. $X_1, \cdots, X_\ell$ uniformly distributed on $[0, 1]$, let the smallest be $X_{(1)}$.

An Ideal Algorithm

- If we can make the distribution of $\{i_t\}$ uniform, then it is easier to estimate its size.
- E.g., if we can have an ideal hash function h from $[d]$ to $[0, 1]$, let X be the minimum $h(i_t)$, then $\frac{1}{X}$ seems a reasonable estimate.
- Indeed, suppose we have i.i.d. $X_1, \cdots, X_\ell$ uniformly distributed on $[0, 1]$, let the smallest be $X_{(1)}$.
    - $X_{(1)} \leq X_{(2)} \leq \cdots \leq X_{(\ell)}$ are called *order statistics*.

An Ideal Algorithm

- If we can make the distribution of $\{i_t\}$ uniform, then it is easier to estimate its size.
- E.g., if we can have an ideal hash function h from $[d]$ to $[0, 1]$, let X be the minimum $h(i_t)$, then $\frac{1}{X}$ seems a reasonable estimate.
- Indeed, suppose we have i.i.d. $X_1, \cdots, X_\ell$ uniformly distributed on $[0, 1]$, let the smallest be $X_{(1)}$.
  - $X_{(1)} \leq X_{(2)} \leq \cdots \leq X_{(\ell)}$ are called *order statistics*.
  - The distribution of $X_{(1)}$ satisfies Beta distribution $B(1, \ell)$. We have $\mathbf{E}[X_{(1)}] = \frac{1}{\ell+1}$.
- Therefore, $\frac{1}{X} - 1$ is an unbiased estimator of $||x||_0$.

An Ideal Algorithm

- If we can make the distribution of $\{i_t\}$ uniform, then it is easier to estimate its size.
- E.g., if we can have an ideal hash function h from $[d]$ to $[0, 1]$, let X be the minimum $h(i_t)$, then $\frac{1}{X}$ seems a reasonable estimate.
- Indeed, suppose we have i.i.d. $X_1, \cdots, X_\ell$ uniformly distributed on $[0, 1]$, let the smallest be $X_{(1)}$.
  - $X_{(1)} \leq X_{(2)} \leq \cdots \leq X_{(\ell)}$ are called *order statistics*.
  - The distribution of $X_{(1)}$ satisfies Beta distribution $B(1, \ell)$. We have $\mathbf{E}[X_{(1)}] = \frac{1}{\ell+1}$.
- Therefore, $\frac{1}{X} - 1$ is an unbiased estimator of $||x||_0$.
- $\mathrm{Var}[X_{(1)}] = \frac{\ell}{(\ell+1)^2(\ell+2)} \leq \frac{1}{(\ell+1)^2}$.

An Ideal Algorithm

- If we can make the distribution of $\{i_t\}$ uniform, then it is easier to estimate its size.
- E.g., if we can have an ideal hash function h from $[d]$ to $[0,1]$, let X be the minimum $h(i_t)$, then $\frac{1}{X}$ seems a reasonable estimate.
- Indeed, suppose we have i.i.d. $X_1, \cdots, X_\ell$ uniformly distributed on $[0,1]$, let the smallest be $X_{(1)}$.
  - $X_{(1)} \le X_{(2)} \le \cdots \le X_{(\ell)}$ are called *order statistics*.
  - The distribution of $X_{(1)}$ satisfies Beta distribution $B(1,\ell)$. We have $\mathbf{E}[X_{(1)}] = \frac{1}{\ell+1}$.
- Therefore, $\frac{1}{X} - 1$ is an unbiased estimator of $||x||_0$.
- $\mathrm{Var}[X_{(1)}] = \frac{\ell}{(\ell+1)^2(\ell+2)} \le \frac{1}{(\ell+1)^2}$.
- We can apply the Chebyshev bound, although the variance is a bit too large for our purpose.

Flajolet and Martin

- Taking the average of $O(\frac{1}{\epsilon^2})$ independent trials, the estimate of the minimum hash value is within $(1 \pm \epsilon)$ of $\frac{1}{||x||_0+1}$ with constant probability.

Flajolet and Martin

- Taking the average of $O(\frac{1}{\epsilon^2})$ independent trials, the estimate of the minimum hash value is within $(1 \pm \epsilon)$ of $\frac{1}{||x||_0+1}$ with constant probability.
- The median trick again: repeating this $O(\log \frac{1}{\delta})$ times and taking the median, we succeed with probability $1-\delta$.

Flajolet and Martin

- Taking the average of $O(\frac{1}{\epsilon^2})$ independent trials, the estimate of the minimum hash value is within $(1 \pm \epsilon)$ of $\frac{1}{||x||_0+1}$ with constant probability.
- The median trick again: repeating this $O(\log \frac{1}{\delta})$ times and taking the median, we succeed with probability $1 - \delta$.
- This algorithm is due to Flajolet and Martin (1985).

Flajolet and Martin

- Taking the average of $O(\frac{1}{\epsilon^2})$ independent trials, the estimate of the minimum hash value is within $(1 \pm \epsilon)$ of $\frac{1}{||x||_0+1}$ with constant probability.
- The median trick again: repeating this $O(\log \frac{1}{\delta})$ times and taking the median, we succeed with probability $1 - \delta$.
- This algorithm is due to Flajolet and Martin (1985).
- It assumes we have access to ideal hash functions.

Flajolet and Martin

- Taking the average of $O(\frac{1}{\epsilon^2})$ independent trials, the estimate of the minimum hash value is within $(1 \pm \epsilon)$ of $\frac{1}{||x||_0+1}$ with constant probability.
- The median trick again: repeating this $O(\log \frac{1}{\delta})$ times and taking the median, we succeed with probability $1 - \delta$.
- This algorithm is due to Flajolet and Martin (1985).
- It assumes we have access to ideal hash functions.
- Ideas for improvement:
  - Use real hash functions. Discretize the range. Possibly use k-wise independent hash family for appropriate k.

Flajolet and Martin

- Taking the average of $O(\frac{1}{\epsilon^2})$ independent trials, the estimate of the minimum hash value is within $(1 \pm \epsilon)$ of $\frac{1}{||x||_0+1}$ with constant probability.
- The median trick again: repeating this $O(\log \frac{1}{\delta})$ times and taking the median, we succeed with probability $1 - \delta$.
- This algorithm is due to Flajolet and Martin (1985).
- It assumes we have access to ideal hash functions.
- Ideas for improvement:
  - Use real hash functions. Discretize the range. Possibly use k-wise independent hash family for appropriate k.
  - The minimum of $h(i_t)$ tends to be voltaile: a single bad event ruins the estimate.

Flajolet and Martin

- Taking the average of $O(\frac{1}{\epsilon^2})$ independent trials, the estimate of the minimum hash value is within $(1 \pm \epsilon)$ of $\frac{1}{||x||_0+1}$ with constant probability.
- The median trick again: repeating this $O(\log \frac{1}{\delta})$ times and taking the median, we succeed with probability $1 - \delta$.
- This algorithm is due to Flajolet and Martin (1985).
- It assumes we have access to ideal hash functions.
- Ideas for improvement:
  - Use real hash functions. Discretize the range. Possibly use k-wise independent hash family for appropriate k.
  - The minimum of $h(i_t)$ tends to be voltaile: a single bad event ruins the estimate.
  - To make the estimate more stable, we may keep track of the $t > 1$ minimum hash values.

KMV

The following KMV (*k* minimum values) algorithm is due to Bar-Yossef, Jayram, Kumar, Sivakumar and Trevisan (2002).

KMV

The following KMV (k minimum values) algorithm is due to Bar-Yossef, Jayram, Kumar, Sivakumar and Trevisan (2002).

- Sample a hash function h from a pairwise independent hash family mapping $[d]$ to $[D]$, for $D \in [d^3, 2d^3]$ that is a power of 2.

KMV

The following KMV (k minimum values) algorithm is due to Bar-Yossef, Jayram, Kumar, Sivakumar and Trevisan (2002).

- Sample a hash function h from a pairwise independent hash family mapping $[d]$ to $[D]$, for $D \in [d^3, 2d^3]$ that is a power of 2.
- Initialize S to $\emptyset$. Set $t = 12/\delta\epsilon^2$.

KMV

The following KMV (k minimum values) algorithm is due to Bar-Yossef, Jayram, Kumar, Sivakumar and Trevisan (2002).

- Sample a hash function h from a pairwise independent hash family mapping $[d]$ to $[D]$, for $D \in [d^3, 2d^3]$ that is a power of 2.
- Initialize S to $\emptyset$. Set $t = 12/\delta\epsilon^2$.
- When i_j arrives, add $h(i_j)$ to S if
  - If $|S| < t$, then add $h(i_j)$ to S;
  - Otherwise, if $h(i_j) < y, \forall y \in S$, then replace the largest element of S by $h(i_j)$.

KMV

The following KMV (k minimum values) algorithm is due to Bar-Yossef, Jayram, Kumar, Sivakumar and Trevisan (2002).

- Sample a hash function h from a pairwise independent hash family mapping $[d]$ to $[D]$, for $D \in [d^3, 2d^3]$ that is a power of 2.
- Initialize S to $\emptyset$. Set $t = 12/\delta\epsilon^2$.
- When i_j arrives, add $h(i_j)$ to S if
    - If $|S| < t$, then add $h(i_j)$ to S;
    - Otherwise, if $h(i_j) < y, \forall y \in S$, then replace the largest element of S by $h(i_j)$.
- For output at the end:
    - If $|S| < t$, return $|S|$.

KMV

The following KMV (k minimum values) algorithm is due to Bar-Yossef, Jayram, Kumar, Sivakumar and Trevisan (2002).

- Sample a hash function h from a pairwise independent hash family mapping $[d]$ to $[D]$, for $D \in [d^3, 2d^3]$ that is a power of 2.
- Initialize S to $\emptyset$. Set $t = 12/\delta\epsilon^2$.
- When i_j arrives, add $h(i_j)$ to S if
  - If $|S| < t$, then add $h(i_j)$ to S;
  - Otherwise, if $h(i_j) < y, \forall y \in S$, then replace the largest element of S by $h(i_j)$.
- For output at the end:
  - If $|S| < t$, return $|S|$.
  - Otherwise, let X be the largest element in S, return $\frac{tD}{X}$.

Analysis of KMV

Proposition

If Y is a Bernoulli random variable, then $\mathrm{Var}[Y] \leq \mathbf{Pr}[Y = 1]$.

Analysis of KMV

Proposition

If Y is a Bernoulli random variable, then $\mathrm{Var}[Y] \leq \mathbf{Pr}[Y = 1]$.

Proof.

$\mathrm{Var}[Y] = \mathbf{E}[Y^2] - \mathbf{E}[Y]^2 \leq \mathbf{E}[Y^2] = \mathbf{Pr}[Y = 1]$. □

Analysis of KMV

Proposition

If Y is a Bernoulli random variable, then $\mathrm{Var}[Y] \leq \mathbf{Pr}[Y = 1]$.

Proof.

$\mathrm{Var}[Y] = \mathbf{E}[Y^2] - \mathbf{E}[Y]^2 \leq \mathbf{E}[Y^2] = \mathbf{Pr}[Y = 1]$. □

- Let's denote $\ell := ||x||_0$, assume $\epsilon < \frac{1}{2}$, and $d > \frac{2}{\epsilon^2 \delta}$.
- First case of output: if $|S| < t$, what's the chance that $||x||_0 > |S|$?

Analysis of KMV

Proposition

If Y is a Bernoulli random variable, then $\mathrm{Var}[Y] \leq \mathbf{Pr}[Y = 1]$.

Proof.

$\mathrm{Var}[Y] = \mathbf{E}[Y^2] - \mathbf{E}[Y]^2 \leq \mathbf{E}[Y^2] = \mathbf{Pr}[Y = 1]$. □

- Let's denote $\ell := ||x||_0$, assume $\epsilon < \frac{1}{2}$, and $d > \frac{2}{\epsilon^2 \delta}$.
- First case of output: if $|S| < t$, what's the chance that $||x||_0 > |S|$?
  - For any pair of indices, they are mapped to the same address with probability $\frac{1}{D}$.

Analysis of KMV

Proposition

If Y is a Bernoulli random variable, then $\mathrm{Var}[Y] \leq \mathbf{Pr}[Y = 1]$.

Proof.

$\mathrm{Var}[Y] = \mathbf{E}[Y^2] - \mathbf{E}[Y]^2 \leq \mathbf{E}[Y^2] = \mathbf{Pr}[Y = 1]$. □

- Let's denote $\ell := ||x||_0$, assume $\epsilon < \frac{1}{2}$, and $d > \frac{2}{\epsilon^2 \delta}$.
- First case of output: if $|S| < t$, what's the chance that $||x||_0 > |S|$?
  - For any pair of indices, they are mapped to the same address with probability $\frac{1}{D}$.
  - There are $\binom{\ell}{2}$ pairs, so the probability that any clash happens is $\leq \binom{\ell}{2} \cdot \frac{1}{D} \leq \frac{1}{d}$. (Recall $D \geq d^3$.)

Analysis: The interesting case

- The interesting case: $|S| \geq t$.

Analysis: The interesting case

- The interesting case: $|S| \geq t$.
- Recall: X is the largest element in S. We'll bound $\mathbf{Pr}[|\frac{tD}{X} - \ell| > \epsilon\ell]$.

Analysis: The interesting case

- The interesting case: $|S| \geq t$.
- Recall: X is the largest element in S. We'll bound $\mathbf{Pr}[|\frac{tD}{X} - \ell| > \epsilon\ell]$.
- Consider the event $\frac{tD}{X} > (1+\epsilon)\ell$.

Analysis: The interesting case

- The interesting case: $|S| \geq t$.
- Recall: X is the largest element in S. We'll bound $\mathbf{Pr}[|\frac{tD}{X} - \ell| > \epsilon\ell]$.
- Consider the event $\frac{tD}{X} > (1 + \epsilon)\ell$.
- This happens only if more than t of the ℓ elements are hashed to addresses smaller than $X < \frac{tD}{(1+\epsilon)\ell} \leq \frac{(1-\epsilon/2)tD}{\ell}$.

Analysis: The interesting case

- The interesting case: $|S| \geq t$.
- Recall: X is the largest element in S. We'll bound $\mathbf{Pr}[|\frac{tD}{X} - \ell| > \epsilon\ell]$.
- Consider the event $\frac{tD}{X} > (1+\epsilon)\ell$.
- This happens only if more than t of the ℓ elements are hashed to addresses smaller than $X < \frac{tD}{(1+\epsilon)\ell} \leq \frac{(1-\epsilon/2)tD}{\ell}$.
- W.l.o.g let the ℓ elements be $1, \cdots, \ell$, and let Z_i be the indicator variable for the event $h(i) < \frac{(1-\epsilon/2)tD}{\ell}$.

Analysis: The interesting case

- The interesting case: $|S| \geq t$.
- Recall: X is the largest element in S. We'll bound $\mathbf{Pr}[|\frac{tD}{X} - \ell| > \epsilon\ell]$.
- Consider the event $\frac{tD}{X} > (1+\epsilon)\ell$.
- This happens only if more than t of the ℓ elements are hashed to addresses smaller than $X < \frac{tD}{(1+\epsilon)\ell} \leq \frac{(1-\epsilon/2)tD}{\ell}$.
- W.l.o.g let the ℓ elements be $1, \cdots, \ell$, and let Z_i be the indicator variable for the event $h(i) < \frac{(1-\epsilon/2)tD}{\ell}$.
- Then $\mathbf{E}[Z_i] = (1-\epsilon/2)t/\ell$.

Analysis: The interesting case

- The interesting case: $|S| \geq t$.
- Recall: X is the largest element in S. We'll bound $\mathbf{Pr}[|\frac{tD}{X} - \ell| > \epsilon\ell]$.
- Consider the event $\frac{tD}{X} > (1+\epsilon)\ell$.
- This happens only if more than t of the ℓ elements are hashed to addresses smaller than $X < \frac{tD}{(1+\epsilon)\ell} \leq \frac{(1-\epsilon/2)tD}{\ell}$.
- W.l.o.g let the ℓ elements be $1, \cdots, \ell$, and let Z_i be the indicator variable for the event $h(i) < \frac{(1-\epsilon/2)tD}{\ell}$.
- Then $\mathbf{E}[Z_i] = (1-\epsilon/2)t/\ell$.
- On the other hand, $Z := \sum_{i=1}^{\ell} Z_i \geq t$.

Analysis: The interesting case

- The interesting case: $|S| \geq t$.
- Recall: X is the largest element in S. We'll bound $\mathbf{Pr}[|\frac{tD}{X} - \ell| > \epsilon\ell]$.
- Consider the event $\frac{tD}{X} > (1+\epsilon)\ell$.
- This happens only if more than t of the ℓ elements are hashed to addresses smaller than $X < \frac{tD}{(1+\epsilon)\ell} \leq \frac{(1-\epsilon/2)tD}{\ell}$.
- W.l.o.g let the ℓ elements be $1, \cdots, \ell$, and let Z_i be the indicator variable for the event $h(i) < \frac{(1-\epsilon/2)tD}{\ell}$.
- Then $\mathbf{E}[Z_i] = (1-\epsilon/2)t/\ell$.
- On the other hand, $Z := \sum_{i=1}^{\ell} Z_i \geq t$.
- $\mathrm{Var}[Z_i] \leq \mathbf{Pr}[Z_i] = (1-\frac{\epsilon}{2})t/\ell$.

Analysis: The interesting case

- The interesting case: $|S| \geq t$.
- Recall: X is the largest element in S. We'll bound $\mathbf{Pr}[|\frac{tD}{X} - \ell| > \epsilon\ell]$.
- Consider the event $\frac{tD}{X} > (1+\epsilon)\ell$.
- This happens only if more than t of the ℓ elements are hashed to addresses smaller than $X < \frac{tD}{(1+\epsilon)\ell} \leq \frac{(1-\epsilon/2)tD}{\ell}$.
- W.l.o.g let the ℓ elements be $1, \cdots, \ell$, and let Z_i be the indicator variable for the event $h(i) < \frac{(1-\epsilon/2)tD}{\ell}$.
- Then $\mathbf{E}[Z_i] = (1-\epsilon/2)t/\ell$.
- On the other hand, $Z := \sum_{i=1}^{\ell} Z_i \geq t$.
- $\mathrm{Var}[Z_i] \leq \mathbf{Pr}[Z_i] = (1-\frac{\epsilon}{2})t/\ell$.
- By pairwise independence we have $\mathrm{Var}[Z] = \sum_i \mathrm{Var}[Z_i] \leq t$.

Analysis of KMV (Cont.)

- We have so far $\{\frac{tD}{X} > (1+\epsilon)\ell\} \Rightarrow \{Z \geq t\}$, $\mathbf{E}[Z] \leq (1-\frac{\epsilon}{2})t$, and $\mathrm{Var}[Z] \leq t$.

Analysis of KMV (Cont.)

- We have so far $\{\frac{tD}{X} > (1+\epsilon)\ell\} \Rightarrow \{Z \geq t\}$, $\mathbf{E}[Z] \leq (1-\frac{\epsilon}{2})t$, and $\mathrm{Var}[Z] \leq t$.
- By Chebysheve inequality, we have

$$\mathbf{Pr}\left[\frac{tD}{X} > (1+\epsilon)\ell\right] \leq \mathbf{Pr}\left[Z \geq t\right] \leq \frac{\mathrm{Var}[Z]}{(\epsilon t/2)^2} \leq \frac{4}{\epsilon^2 t} \leq \frac{\delta}{3}.$$

Analysis of KMV (Cont.)

- We have so far $\{\frac{tD}{X} > (1+\epsilon)\ell\} \Rightarrow \{Z \geq t\}$, $\mathbf{E}[Z] \leq (1-\frac{\epsilon}{2})t$, and $\mathrm{Var}[Z] \leq t$.
- By Chebysheve inequality, we have

$$\mathbf{Pr}\left[\frac{tD}{X} > (1+\epsilon)\ell\right] \leq \mathbf{Pr}\left[Z \geq t\right] \leq \frac{\mathrm{Var}[Z]}{(\epsilon t/2)^2} \leq \frac{4}{\epsilon^2 t} \leq \frac{\delta}{3}.$$

- Almost symmetrically, the event $\{\frac{tD}{X} < (1-\epsilon)\ell\}$ happens only if fewer than t of the ℓ elements are hashed to addresses smaller than $X > \frac{tD}{(1-\epsilon)\ell}$.

Analysis of KMV (Cont.)

- We have so far $\{\frac{tD}{X} > (1+\epsilon)\ell\} \Rightarrow \{Z \geq t\}$, $\mathbf{E}[Z] \leq (1-\frac{\epsilon}{2})t$, and $\text{Var}[Z] \leq t$.
- By Chebysheve inequality, we have

$$\mathbf{Pr}\left[\frac{tD}{X} > (1+\epsilon)\ell\right] \leq \mathbf{Pr}\left[Z \geq t\right] \leq \frac{\text{Var}[Z]}{(\epsilon t/2)^2} \leq \frac{4}{\epsilon^2 t} \leq \frac{\delta}{3}.$$

- Almost symmetrically, the event $\{\frac{tD}{X} < (1-\epsilon)\ell\}$ happens only if fewer than t of the ℓ elements are hashed to addresses smaller than $X > \frac{tD}{(1-\epsilon)\ell}$.
- Let Z_i be the indicator variable for the event $h(i) < \frac{tD}{(1-\epsilon)\ell}$.

Analysis of KMV (Cont.)

- We have so far $\{\frac{tD}{X} > (1+\epsilon)\ell\} \Rightarrow \{Z \geq t\}$, $\mathbf{E}[Z] \leq (1-\frac{\epsilon}{2})t$, and $\mathrm{Var}[Z] \leq t$.
- By Chebysheve inequality, we have

$$\mathbf{Pr}\left[\frac{tD}{X} > (1+\epsilon)\ell\right] \leq \mathbf{Pr}\left[Z \geq t\right] \leq \frac{\mathrm{Var}[Z]}{(\epsilon t/2)^2} \leq \frac{4}{\epsilon^2 t} \leq \frac{\delta}{3}.$$

- Almost symmetrically, the event $\{\frac{tD}{X} < (1-\epsilon)\ell\}$ happens only if fewer than t of the ℓ elements are hashed to addresses smaller than $X > \frac{tD}{(1-\epsilon)\ell}$.
- Let Z_i be the indicator variable for the event $h(i) < \frac{tD}{(1-\epsilon)\ell}$.

$$\frac{t}{(1-\epsilon)\ell} \geq \mathbf{E}\left[Z_i\right] \geq \frac{t}{(1-\epsilon)\ell} - \frac{1}{D} \geq \frac{(1+\epsilon)t}{\ell} - \frac{1}{D} \geq \frac{(1+\epsilon/2)t}{\ell}.$$

Analysis of KMV

$$\mathrm{Var}\left[Z_i\right] \leq \mathbf{E}\left[Z_i\right] \leq \frac{t}{(1-\epsilon)\ell} \leq \frac{2t}{\ell}.$$

Analysis of KMV

$$\mathrm{Var}\left[Z_i\right] \le \mathbf{E}\left[Z_i\right] \le \frac{t}{(1-\epsilon)\ell} \le \frac{2t}{\ell}.$$

Let Z be $\sum_{i=1}^{\ell} Z_i$, then $\mathbf{E}[Z] \ge (1+\frac{\epsilon}{2})t$, $\mathrm{Var}[Z] \le 2t$.

Analysis of KMV

$$\mathrm{Var}\left[Z_i\right] \leq \mathbf{E}\left[Z_i\right] \leq \frac{t}{(1-\epsilon)\ell} \leq \frac{2t}{\ell}.$$

Let Z be $\sum_{i=1}^{\ell} Z_i$, then $\mathbf{E}[Z] \geq (1+\frac{\epsilon}{2})t$, $\mathrm{Var}[Z] \leq 2t$.
By Chebyshev inequality,

$$\mathbf{Pr}\left[\frac{tD}{X} < (1-\epsilon)\ell\right] \leq \mathbf{Pr}\left[Z < t\right] \leq \frac{\mathrm{Var}[Z]}{(\epsilon t/2)^2} \leq \frac{8}{\epsilon^2 t} \leq \frac{2\delta}{3}.$$

Analysis of KMV

$$\mathrm{Var}\left[Z_i\right] \le \mathbf{E}\left[Z_i\right] \le \frac{t}{(1-\epsilon)\ell} \le \frac{2t}{\ell}.$$

Let Z be $\sum_{i=1}^{\ell} Z_i$, then $\mathbf{E}[Z] \ge (1+\frac{\epsilon}{2})t$, $\mathrm{Var}[Z] \le 2t$.
By Chebyshev inequality,

$$\mathbf{Pr}\left[\frac{tD}{X} < (1-\epsilon)\ell\right] \le \mathbf{Pr}\left[Z < t\right] \le \frac{\mathrm{Var}[Z]}{(\epsilon t/2)^2} \le \frac{8}{\epsilon^2 t} \le \frac{2\delta}{3}.$$

Combining everything, we have that with probability at least $1-\delta$, $|\frac{tD}{X} - \ell| \le \epsilon\ell$.

Analysis of KMV

$$\mathrm{Var}\left[Z_i\right] \le \mathbf{E}\left[Z_i\right] \le \frac{t}{(1-\epsilon)\ell} \le \frac{2t}{\ell}.$$

Let Z be $\sum_{i=1}^{\ell} Z_i$, then $\mathbf{E}[Z] \ge (1+\frac{\epsilon}{2})t$, $\mathrm{Var}[Z] \le 2t$.
By Chebyshev inequality,

$$\mathbf{Pr}\left[\frac{tD}{X} < (1-\epsilon)\ell\right] \le \mathbf{Pr}\left[Z < t\right] \le \frac{\mathrm{Var}[Z]}{(\epsilon t/2)^2} \le \frac{8}{\epsilon^2 t} \le \frac{2\delta}{3}.$$

Combining everything, we have that with probability at least $1-\delta$, $|\frac{tD}{X} - \ell| \le \epsilon\ell$.
Space usage:

- Storing the hash takes space $O(\log D) = O(\log d)$.

Analysis of KMV

$$\mathrm{Var}\left[Z_i\right] \leq \mathbf{E}\left[Z_i\right] \leq \frac{t}{(1-\epsilon)\ell} \leq \frac{2t}{\ell}.$$

Let Z be $\sum_{i=1}^{\ell} Z_i$, then $\mathbf{E}[Z] \geq (1+\frac{\epsilon}{2})t$, $\mathrm{Var}[Z] \leq 2t$.
By Chebyshev inequality,

$$\mathbf{Pr}\left[\frac{tD}{X} < (1-\epsilon)\ell\right] \leq \mathbf{Pr}\left[Z < t\right] \leq \frac{\mathrm{Var}[Z]}{(\epsilon t/2)^2} \leq \frac{8}{\epsilon^2 t} \leq \frac{2\delta}{3}.$$

Combining everything, we have that with probability at least $1-\delta$, $|\frac{tD}{X} - \ell| \leq \epsilon\ell$.
Space usage:

- Storing the hash takes space $O(\log D) = O(\log d)$.
- Storing S takes space $tO(\log D) = O(\frac{\log d}{\epsilon^2\delta})$.

Analysis of KMV

$$\mathrm{Var}\left[Z_i\right] \le \mathbf{E}\left[Z_i\right] \le \frac{t}{(1-\epsilon)\ell} \le \frac{2t}{\ell}.$$

Let Z be $\sum_{i=1}^{\ell} Z_i$, then $\mathbf{E}[Z] \ge (1+\frac{\epsilon}{2})t$, $\mathrm{Var}[Z] \le 2t$.
By Chebyshev inequality,

$$\mathbf{Pr}\left[\frac{tD}{X} < (1-\epsilon)\ell\right] \le \mathbf{Pr}\left[Z < t\right] \le \frac{\mathrm{Var}[Z]}{(\epsilon t/2)^2} \le \frac{8}{\epsilon^2 t} \le \frac{2\delta}{3}.$$

Combining everything, we have that with probability at least $1-\delta$, $|\frac{tD}{X} - \ell| \le \epsilon\ell$.
Space usage:

- Storing the hash takes space $O(\log D) = O(\log d)$.
- Storing S takes space $tO(\log D) = O(\frac{\log d}{\epsilon^2\delta})$.
- The optimal algorithm uses space $O(\log d + \epsilon^{-2})$!