Count Sketch: Finding Heavy Hitters

COMPCSI 753: Algorithms for Massive Data

Instructor: Ninh Pham

University of Auckland

Auckland, Aug 24, 2020

Basic definitions

- Let $\mathbf{U}$ be a universe of size $\mathbf{n}$, i.e. $\mathbf{U} = \{\mathbf{1, 2, 3, \ldots, n}\}$.
- Cash register model stream:
  - Sequence of $\mathbf{m}$ elements $\mathbf{a_1}, \ldots, \mathbf{a_m}$ where $\mathbf{a_i} \in \mathbf{U}$.
  - Elements of $\mathbf{U}$ may or may not occur once or several times in the stream.
- Finding heavy hitters in data stream (today's lecture):
  - Given a stream, finding frequent items.

Frequent items

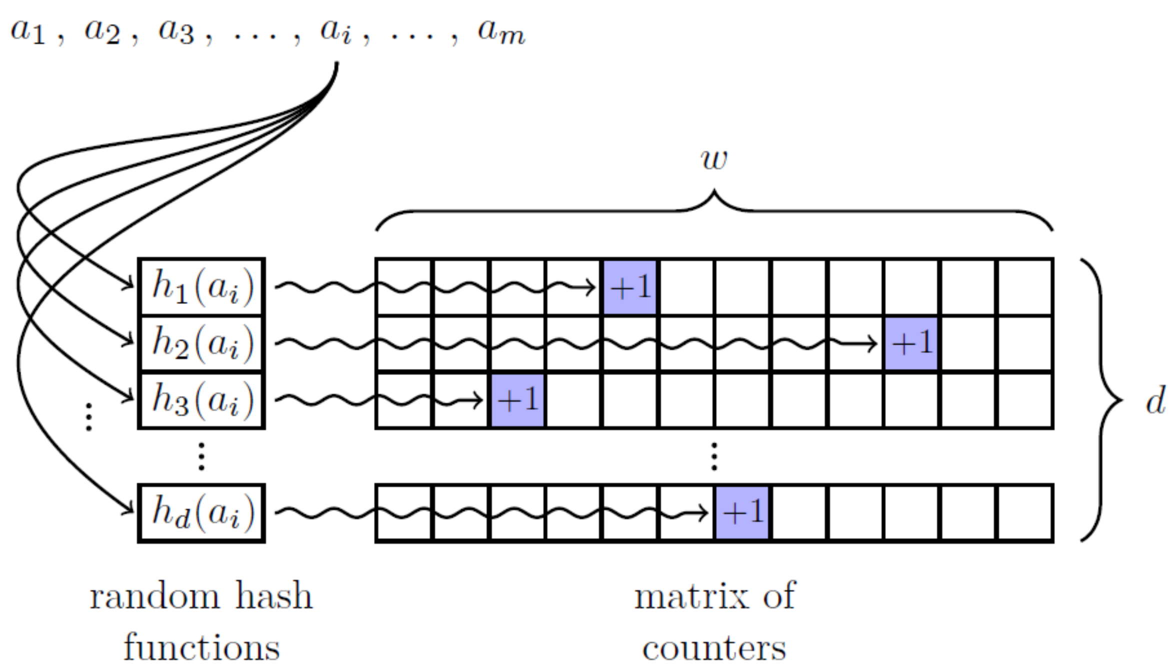
- Each element of data stream is a tuple.
- Given a stream of $\mathbf{m}$ elements $\mathbf{a_1}, \ldots, \mathbf{a_m}$ where $\mathbf{a_i} \in \mathbf{U}$, finding the most/top-$\mathbf{k}$ frequent elements.

- Example:
  - $\{\underline{1}, 2, \underline{1}, 3, 4, 5\} \rightarrow \mathbf{f} = \{\underline{2}, 1, 1, 1, 1\}$
  - $\{\underline{1}, \underline{2}, \underline{1}, 3, \underline{1}, \underline{2}, 4, 5, \underline{2}, 3\} \rightarrow \mathbf{f} = \{\underline{3}, \underline{3}, 2, 1, 1\}$

- We need an approximation solution with much smaller memory with theoretical guarantees.

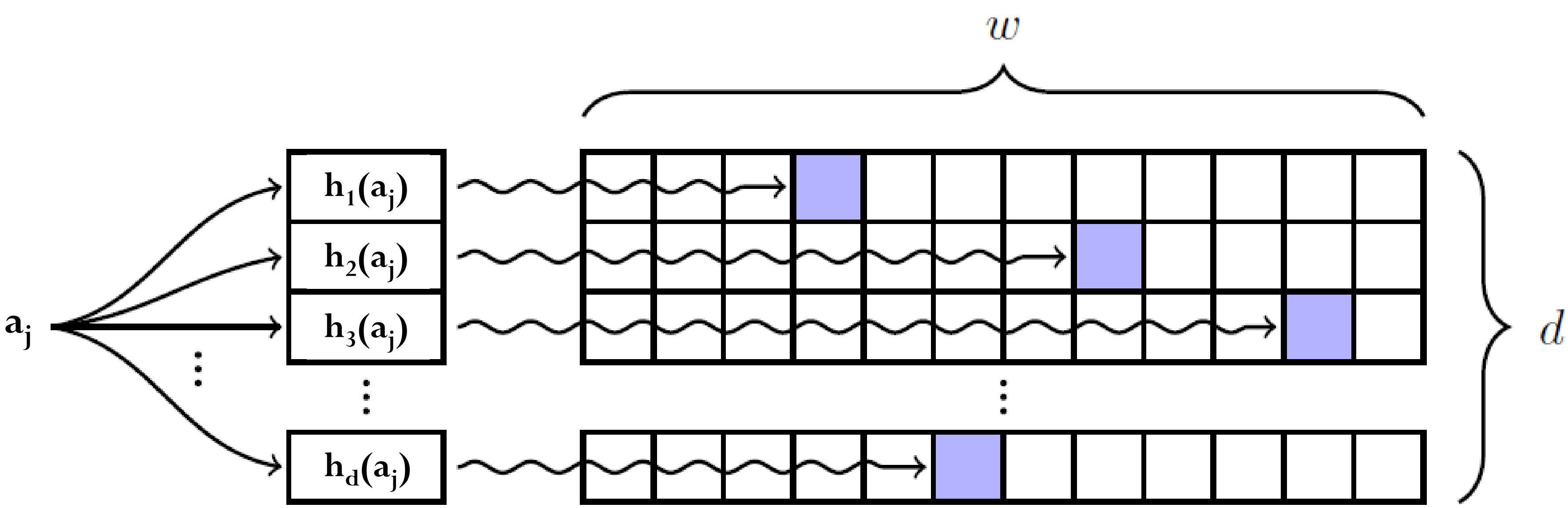
Randomized: CountMin sketch

- Setup:
  - **d** independent **universal** hash functions **h** over range **[0, w)**
  - **d** different counters, $\mathbf{C_1, \ldots, C_d}$. Each of size **w** initialized with **0**s
- Process an element $\mathbf{a_j}$:
  - For each hash function, compute $\mathbf{h_i(a_j)}$ and increment $\mathbf{C_i[h_i(a_j)]}$ by **1**



Randomized: CountMin sketch

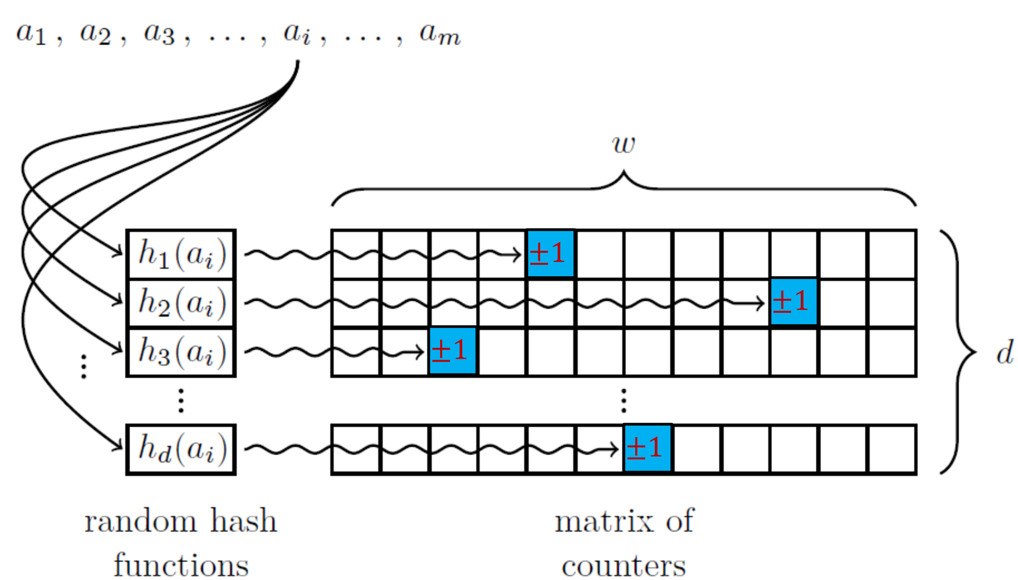
- Query: How many times $\mathbf{a_j}$ occurred?
  - For each hash function, compute $\mathbf{h_i(a_j)}$ and get $\mathbf{C_i[h_i(a_j)]}$
  - Return $\mathbf{min(C_1[h_1(a_j)], \ldots, C_d[h_d(a_j)])}$



return the minimum of values in blue cells

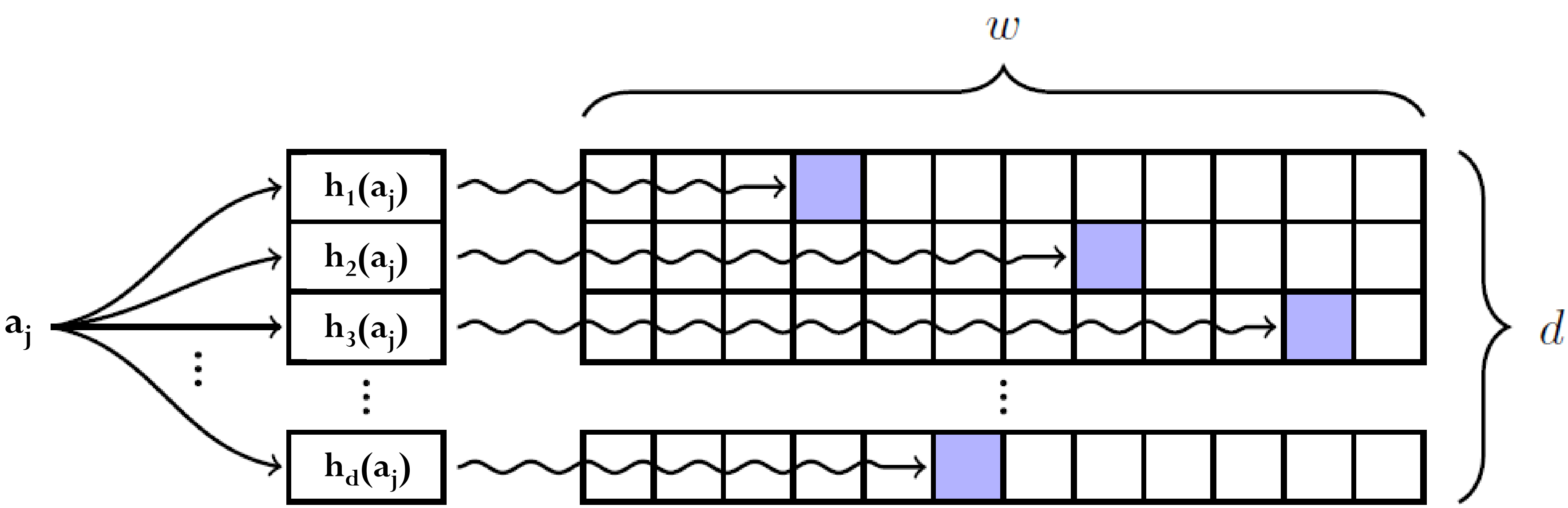
Randomized: Count sketch

- Setup:
  - **d** independent 2-wise hash functions **h** over range **[0, w)**.
  - **d** independent 2-wise hash functions **s** over range {**+1, -1**}.
  - **d** different counters, $\mathbf{C_1, \ldots, C_d}$. Each of size **w** initialized with **0**s.
- Process an element $\mathbf{a_j}$:
  - For each hash function, compute $\mathbf{h_i(a_j)}$ and increment $\mathbf{C_i[h_i(a_j)]}$ by $\mathbf{s(a_j)}$.



Randomized: Count sketch

- Query: How many times $\mathbf{a_j}$ occurred?
  - For each hash function, compute $\mathbf{h_i(a_j)}$ and get $\mathbf{C_i[h_i(a_j)]}$.
  - Return $\mathbf{median(C_1[h_1(a_j)], \ldots, C_d[h_d(a_j)])}$.



return the median of values in blue cells

2-wise hash function family

- 2-wise hash function definition:
  - A family of hash function $\mathbf{H} = \{\mathbf{h} : \mathbf{U} \rightarrow \{\mathbf{0}, \mathbf{1}, \ldots, \mathbf{w-1}\}\}$ is 2-wise independent if for any **2** distinct keys $\mathbf{x_i} \neq \mathbf{x_j} \in \mathbf{U}$ and **2** hash values (not necessary distinct) $\mathbf{y_i}, \mathbf{y_j} \in \{\mathbf{0}, \mathbf{1}, \ldots, \mathbf{w-1}\}$, we have

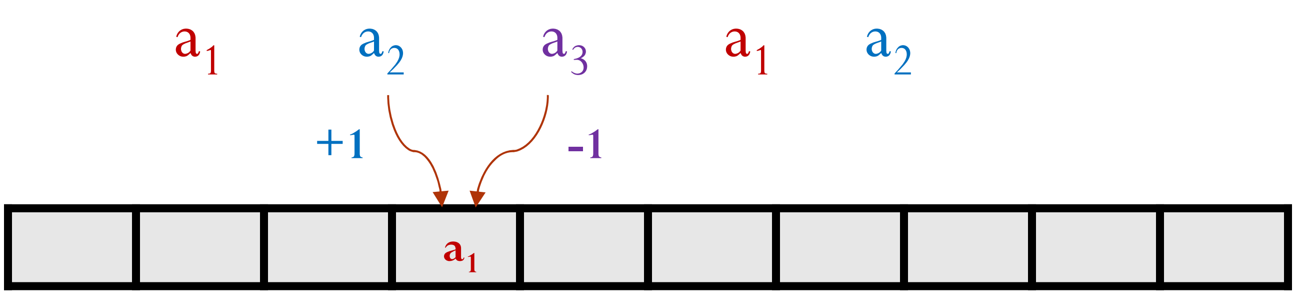
$$\mathbf{Pr_h[h(x_i) = y_i \wedge h(x_j) = y_j] = 1/w^2}$$

- On our CountSketch:
  - Given two different items $\mathbf{a_i} \neq \mathbf{a_j}$, what is the prob. $\mathbf{a_i}$ and $\mathbf{a_j}$ collide?

$$\mathbf{Pr_h[h(a_i) = h(a_j)] = 1/w^2 * w = 1/w}$$

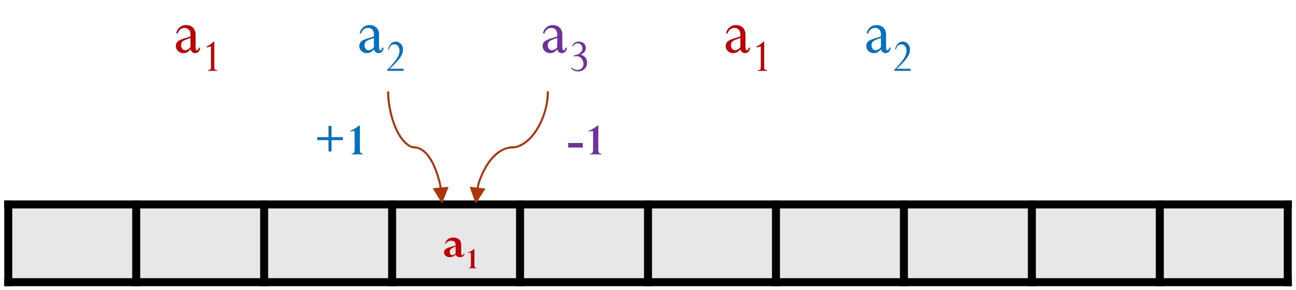
Analysis on 1 array of counters

- Notation:
  - Stream of $\mathbf{m}$ items $\{\mathbf{a_1, \ldots, a_m}\}$ from the universe $\mathbf{U}$ of size $\mathbf{n}$.
  - Frequency vector $\mathbf{f} = \{\mathbf{f_1, \ldots, f_n}\}$, $\|\boldsymbol{f}\|_2^2 = \sum_{i=1}^{n} \boldsymbol{f}_i^2$
- Question:
  - Given a particular item $\mathbf{a_1}$, how many times $\mathbf{a_i} \neq \mathbf{a_1}$ collide by $\mathbf{h}$?



Analysis on 1 array of counters

- Question:
  - Given a particular item $\mathbf{a_1}$, how many times $\mathbf{a_i} \neq \mathbf{a_1}$ collide by $\mathbf{h}$.



- Answer:
  - Let $\mathbf{X_i}$ be contribution of $\mathbf{a_i}$ in the bucket $\mathbf{h(a_1)}$.
  - $X_i = \begin{cases} f_i, & \text{which happens with prob. } 1/2w \\ 0, & \text{which happens with prob. } 1 - 1/w \\ -f_i, & \text{which happens with prob. } 1/2w \end{cases}$
  - Error source caused by the item $\mathbf{a_i}$: $\mathbf{E[X_i] = 0}$

Analysis on 1 array of counters

- Observation: Our estimate is unbiased.
- Question: How large error do we have?

- Analysis of variance of contribution of a particular item $\mathbf{a_i}$:
  - Let $\mathbf{X_i}$ be contribution of $\mathbf{a_i}$ in the bucket $\mathbf{h(a_1)}$.
  - $X_i = \begin{cases} f_i, & \text{which happens with prob. } 1/2w \\ 0, & \text{which happens with prob. } 1-1/w \\ -f_i, & \text{which happens with prob. } 1/2w \end{cases}$
  - Expectation: $\mathbf{E[X_i] = 0}$.
  - Variance: $\mathbf{Var\ [X_i] = E[X_i^2] - (E[X_i])^2 = f_i^2/w.}$

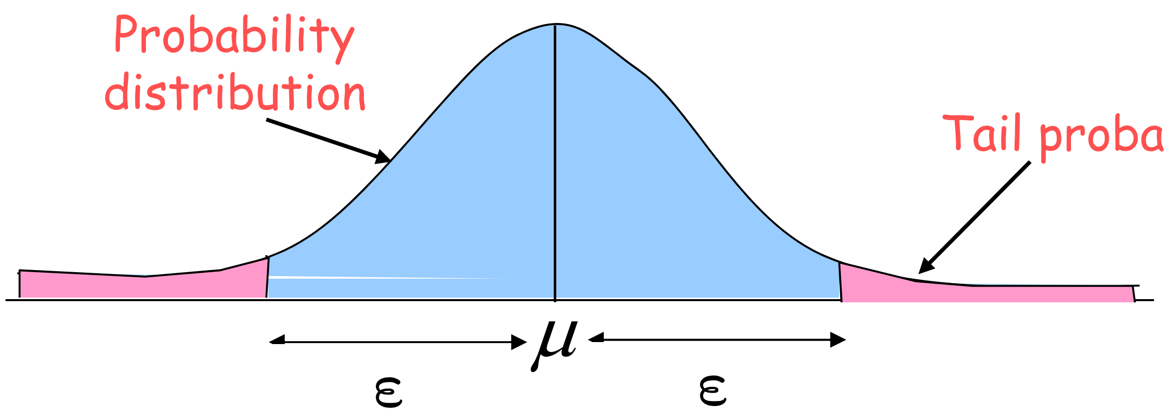
Analysis on 1 array of counters

- Observation: Our estimate is unbiased.
- Question: How large error do we have?

- Analysis of variance of error:
  - Let $\mathbf{X_2, \ldots, X_n}$ be contributions of $\mathbf{a_2, \ldots, a_n}$ in the bucket $\mathbf{h(a_1)}$.
  - Let $\mathbf{Y = X_2 + \ldots + X_n}$ be the total contributions by other item $\mathbf{a_i \neq a_1}$.
  - Variance of error:

$$\mathbf{Var[Y] = Var[X_2] + \ldots + Var[X_n] = (f_2^2 + \ldots + f_n^2)/w \leq \|f\|_2^2/w}.$$

Basic tools: Tail inequality



- Chebyshev's inequality for $\mathbf{E[Y] = \mu}$ and $\mathbf{Var[Y] = \sigma^2}$:

$$\Pr[\,|Y - \mu| \geq \varepsilon] \leq \frac{\sigma^2}{\varepsilon^2} \text{ for any } \varepsilon > 0.$$

Analysis on 1 array of counters

- Question: How large error do we have?

- Analysis of variance of error:
  - Let $\mathbf{X_2, \ldots, X_n}$ be contributions of $\mathbf{a_2, \ldots, a_n}$ in the bucket $\mathbf{h(a_1)}$.
  - Let $\mathbf{Y = X_2 + \ldots + X_n}$ be the total contributions by other item $\mathbf{a_i \neq a_1}$.
  - Expectation: $\mathbf{E[Y] = 0}$.
  - Variance: $\mathbf{Var[Y] = Var[X_2] + \ldots + Var[X_n] = (f_2^2 + \ldots + f_n^2)/w} \leq \frac{\|f\|_2^2}{w}$

- Chebyshev's inequality:
  - $\mathbf{Pr}\,[\ |\mathbf{Y}| \geq \varepsilon\|f\|_2\ ] \leq \frac{1}{w\varepsilon^2} = \mathbf{1/2}$ if we choose $\mathbf{w = 2/\varepsilon^2}$.
  - The error is at most $\varepsilon\|f\|_2$ with the probability $\mathbf{1/2}$.

Analysis on d arrays of counters

- Boosting the accuracy:
  - Using **d** independent hash functions corresponding to **d** independent arrays of counters.
  - $\mathbf{F_1 = median(C_1[h_1(a_1)], \ldots , C_d[h_d(a_1)]) = median(f'_1, f'_2, \ldots, f'_d)}$.
- Analysis:
  - $\mathbf{E [f'_1] = E [f'_2] = \ldots = E [f'_d] = f_1}$.
  - Choose $\mathbf{d = \log(1/\delta)}$ and apply Chernoff bound, we have

$$\mathbf{Pr [\mid F_1 - f_1 \mid \leq \varepsilon \|f\|_2] \geq 1 - \delta}$$

- With probability at least **1 - δ**, we have

$$\mathbf{f_1 - \varepsilon \|f\|_2 \leq F_1 \leq f_1 + \varepsilon \|f\|_2}$$

Homework

- Implement the CountSketch algorithm on the dataset from assignment 1:
  - Description: Each line (doc ID, word ID, freq.) as a stream tuple.
  - Query: What are the most and top-**10** frequent word ID have been used?
- What are the average errors of CountMin Sketch and CountSketch?