CS60021: Scalable Data Mining

Streaming Algorithms

Sourangshu Bhattacharya

Frequent count

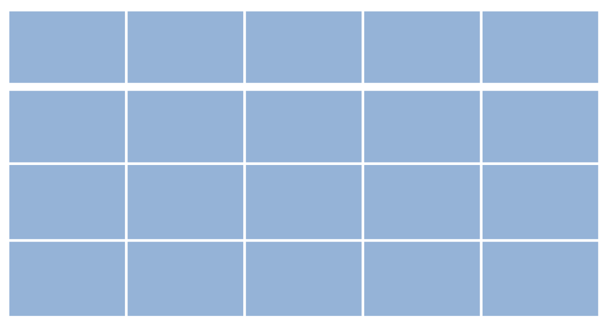
Review: Frequency Estimation

- Given input stream, length m, want a sketch that can answer frequency queries at the end
  - For give item x, return an estimate of the frequency

- Algorithms seen
  - Deterministic counter based algorithms: Misra-Gries, SpaceSaving
  - Count-Min sketch

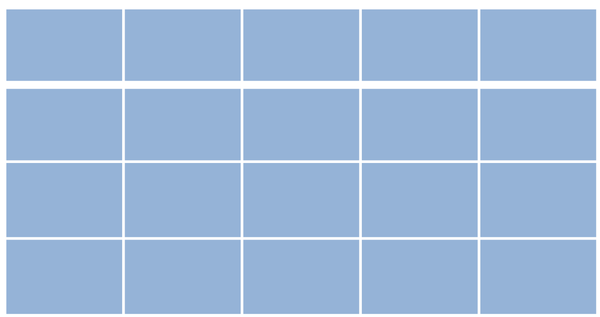
Recall: Count-min sketch

- Model input stream as a vector over U
  - f_x is the entry for dimension x
- Creates a small summary $w \times d$
- Use w hash functions, each maps $U \to [1, d]$



Count-sketch

- Model input stream as a vector over U
  - f_x is the entry for dimension x
- Creates a small summary $w \times d$
- Use w hash functions, $h_i\colon U \rightarrow [1, d]$
- w sign hash function, each maps $\mathrm{g_i}\colon\ U \rightarrow \{-1, +1\}$



Count Min Sketch

Initialize

– Choose $h_1, .., h_w$, $A[w, d] \leftarrow 0$

Process(x, c):

– For each $i \in [w]$, $A[i, h_i(x)] \mathrel{+}= c \times g_i(x)$

Query(q):

– Return median$\{g_i(x) A[i, h_i(x)]\}$

Example

h1			
h2			

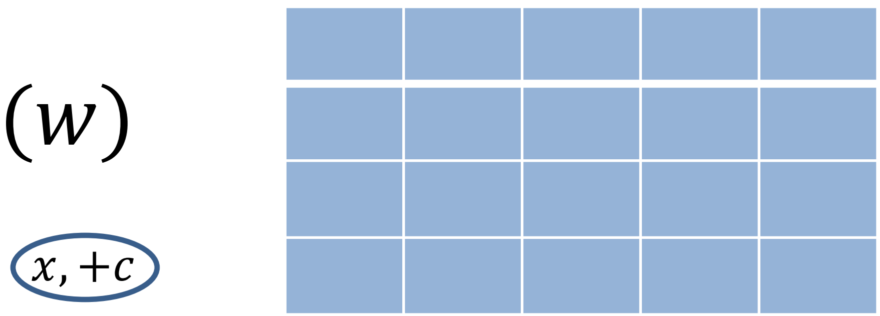
	h1,g1	h2,g2
	2,+	1,+
	3,-	2,+
	1,+	3,-
	2,-	3,+

7

Guarantees

Space = $O(wd)$

Update time = $O(w)$



Each item is mapped to one bucket per row

Guarantees

- $w = \frac{2}{\epsilon^2} \quad \mathrm{d} = \log\left(\frac{1}{\delta}\right)$

$Y_1 \ldots . Y_w$ be the w estimates,
i.e. $Y_i = g_i(x) A[i, h_i(x)], \quad \widehat{f_x} = \underset{i}{\mathrm{median}}\, Y_i$

$$E[Y_i] = E\big[g_i(x)\, A[i, h_i(x)]\big] = E\left[g_i(x) \sum_{h_i(y)=h_i(x)} f_y\, g_i(y)\right]$$

Guarantees

$$E[Y_i] = E\big[g_i(x)\, A[i, h_i(x)]\big] = E\left[g_i(x) \sum_{h_i(y)=h_i(x)} f_y\, g_i(y)\right]$$

Notice that for $x \neq y$, $E[g_i(x)\; g_i(y)] = 0$!

$$E[Y_i] = g_i(x)^2 f_x = f_x$$

We analyse the variance in order to bound the error

For simplicity assume hash functions all independent

Variance analysis

$w = \log \frac{1}{\delta}$

$d = \frac{3}{\varepsilon^2}$

Using simple algebra, as well as independence of hash functions,

$$var(Y_i) = \frac{\left(\sum_y f_y^2 - f_x^2\right)}{d} \le \frac{|f|_2{}^2}{d} \qquad |f|_2^2 = \sum_x f_x^2$$

Using Chebyshev's inequality

δ

$$\Pr[\,|Y_i - f_x| > \epsilon |f|_2] \le \frac{1}{d\epsilon^2} \le \frac{1}{3} \qquad \boxed{d = \frac{3}{\epsilon^2}}$$

$\Pr(|X-\mu| \geq k\sigma) \leq \frac{1}{k^2}$

Finally, use analysis of median-trick with $w = \log\left(\frac{1}{\delta}\right)$

$\sigma = \frac{|f|_2}{\sqrt{d}}$ $\quad k = \varepsilon\sqrt{d}$

Final Guarantees

- Using space $O\left(\overbrace{\frac{1}{\epsilon^2}\log\left(\frac{1}{\delta}\right)}\log(n)\right)$, for any query x, we get an estimate, with prob $1\ -\delta$

$$f_x\ \ -\epsilon|f|_2 \leq f_x \leq f_x + \epsilon|f|_2$$

Comparisons

Algorithm	$\hat{f}_x - f_x$	Space $\times$ $log(n)$	Error prob	Model
Misra-Gries	$[-\epsilon\|f\|_1, 0]$	$1/\epsilon$	0	Insert Only
SpaceSaving	$[0, \epsilon\|f\|_1]$	$1/\epsilon$	0	Insert Only
CountMin	$[0, \epsilon\|f\|_1]$	$\log\left(\frac{1}{\delta}\right)/\epsilon$	δ	Insert+Delete, strict turnstile
CountSketch	$[-\epsilon\|f\|_2, \epsilon\|f\|_2]$	$\log\left(\frac{1}{\delta}\right)/\epsilon^2$	δ	Insert+Delete

Summary

$A_1, \ldots, A_k$

$A_1 + \ldots + A_k$

- CM and Count Sketch to answer point queries about frequencies
  - two user-defined parameters, ϵ and δ
  - Linear sketch, hence can be combined across distributed streams
- Count Sketch handle departures naturally
  - Even if –ve frequencies are present
  - For CM, need strict turnstile
- Extensions to handle range queries and others…
- Actual performance much better than theoretical bound

References:

- Count-sketch:
  - Lecture slides by Graham Cormode
    http://dmac.rutgers.edu/Workshops/WGUnifyingTheory/Slides/cormode.pdf
  - Lecture notes by Amit Chakrabarti:
    http://www.cs.dartmouth.edu/~ac/Teach/data-streams-lecnotes.pdf
  - Sketch techniques for approximate query processing, Graham Cormode.
    http://dimacs.rutgers.edu/~graham/pubs/papers/sk.pdf

- Moment estimation:
  - Mining massive Datasets by Leskovec, Rajaraman, Ullman