Fair-Count-Min Sketch

By Sai Male

Preliminaries

- Let D be a data stream.
- Consists of elements from a finite universe.
- Let $f(x)$ denote the true frequency of an element x in the data stream.
- $0 \leq f(x) \leq |D|$
- Additive error: $\hat{f}(x) - f(x)$
- Multiplicative error (approximation factor): $\alpha(x) = \frac{f(x)}{\hat{f}(x)}$

Traditional Count-Min (CM) Sketch

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

4 x 5 array with 4 different hash functions

- Used for frequency estimation.
- Maintains a d x w array of counters initialized to 0.
- d hash functions.
- $\hat{f}(x) =$ min count of all the rows.
- Note: $\hat{f}(x) \geq f(x)$ always.

CM Sketch's Guarantee

With $w = \left\lceil \frac{e}{\varepsilon} \right\rceil$ and $d = \left\lceil \ln\left(\frac{1}{\delta}\right) \right\rceil$.

The CM sketch satisfies:

$\hat{f}(x) \leq f(x) + \varepsilon N$, where $N = |D|$ for every $x \in U$ with probability at least $1 - \delta$.

Problem Statement

- CM sketch introduces additive error.
- Case: low-frequency element collides with a high-frequency element.
- Overestimation of low-frequency element.
- While the estimate of the high-frequency element is barely affected.
- Creates an unfairness gap.

Fair-Count-Min (FCM) Sketch

- Variant of the CM sketch.
- Solution: divides low-frequency elements and high-frequency elements into two groups.
- Groups: ℓ, h.
- Notation: Group fairness $= \alpha(G)$
- $\alpha(G) = \frac{1}{|G|}\sum_{x \in G} \alpha(x)$

FCM Sketch (continuation)

- Difference: FCM uses semi-uniform hashing.
- Hashing function:

$$h_i(x): \begin{cases} \ell \rightarrow [w_\ell] \\ h \rightarrow w_\ell + [w_h] \end{cases}$$

- $w = w_\ell + w_h$

Low region High region

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

$w_\ell = 2, w_h = 3$

FCM Sketch: Width Allocation & Guarantee

- Choose w_ℓ such that $\alpha(\ell)$ and $\alpha(h)$ are close.
- $\frac{\mathrm{w}_\ell}{\mathrm{w}_h} = \frac{\mathrm{n}_\ell}{\mathrm{n}_h}$
- $\mathrm{w}_\ell \approx \frac{\mathrm{n}_\ell}{\mathrm{n}_\ell + \mathrm{n}_h}\mathrm{w}$
- More elements in a group means more columns allocated.
- When d = 1, total additive error is lower.

Similarities & Differences

Similarities of FCM and CM:

- Maintain a 2D array.
- Costs $O(d)$ time per update and per query.
- Takes $O(dw)$ space.

Difference: FCM restricts which columns an element is mapped to.

Experiment

- $\mathrm{U} = \{0,1,2,3,4,5,6,7,8,9\}$
- $\ell = \{0,1,2,3,4\}, h = \{5,6,7,8,9\}$
- Probabilities set for each element.
- A stream of 5000 samples generated.
- $\mathrm{w} = 6, \mathrm{d} = 1$
- $\mathrm{w}_{\ell} = 3, \mathrm{w}_{h} = 3$
- 1 row, so 1 hash function.
- Hash function: x mod w

Results: Additive Errors

Item x	CM additive error	FCM additive error
0	902	182
1	927	232
2	763	0
3	658	60
4	0	98
5	0	763
6	60	658
7	98	0
8	350	828
9	182	902

Table 1: Per item additive error for CM and FCM.

- CM assigns a large additive error to elements 0 – 3, while high elements are mostly unaffected.
- Under FCM, the large errors are within the high group instead.

Results: Multiplicative Errors

Item x	$\alpha_{\mathrm{CM}}(x)$	$\alpha_{\mathrm{FCM}}(x)$
0	0.062	0.248
1	0.096	0.297
2	0.314	1.000
3	0.217	0.752
4	1.000	0.703
5	1.000	0.520
6	0.938	0.578
7	0.904	1.000
8	0.686	0.480
9	0.783	0.422

Table 2: Per item multiplicative error for CM and FCM (rounded to 3 d.p.).

- The multiplicative errors for elements 0 and 1 are close to 0 under CM.
- Whereas, under FCM, they increase.

Results: Price of Fairness

- $PoF = A_{\mathrm{FCM}} - A_{\mathrm{CM}}$
- $PoF > 0$ indicates a cost from enforcing group fairness.
- Values near zero indicate negligible cost.
- Negative values indicate that FCM reduces total additive error.

Method	Total Additive Error
CM	3940
FCM	3723

Table 4: Total additive error for CM and FCM.

Price of fairness (PoF) $= A_{\mathrm{FCM}} - A_{\mathrm{CM}} = -217$.

Conclusion & Future Work

- Use FCM when the universe naturally splits into groups and one group is much heavier than the other.
- Use CM when all elements have similar frequencies.
- Run experiments with CM and FCM that use more rows.
- Develop methods to fairly handle elements that shift from high frequency to low frequency over time.

References

- G. Cormode and S. Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. Journal of Algorithms, 55(1):58–75, 2005.
- N. Shahbazi, S. Sintos, and A. Asudeh. Fair-count-min: Frequency estimation under equal group-wise approximation factor. arXiv preprint arXiv:2505.18919, 2025.

- Code: https://github.com/nitar31/cm_vs_fcm

Questions?