Lecture 6

The Tug-of-War Sketch

At this point, we have seen a sublinear-space algorithm — the AMS estimator — for estimating the kth frequency moment, $F_k = f_1^k + \cdots + f_n^k$, of a stream σ. This algorithm works for $k \geq 2$, and its space usage depends on n as $\widetilde{O}(n^{1-1/k})$. This fails to be polylogarithmic even in the important case $k = 2$, which we used as our motivating example when introducing frequency moments in the previous lecture. Also, the algorithm does *not* produce a sketch in the sense of Section 4.2.

But Alon, Matias and Szegedy [AMS99] also gave an *amazing* algorithm that *does* produce a sketch, of logarithmic size, which allows one to estimate F_2. What is amazing about the algorithm is that seems to do almost nothing.

6.1 The Basic Sketch

We describe the algorithm in the turnstile model.

Initialize :
Choose a random hash function $h : [n] \to \{-1, 1\}$ from a 4-universal family ;
$x \leftarrow 0$;

Process (j, c):
$x \leftarrow x + ch(j)$;

Output : x^2

The sketch is simply the random variable x. It is pulled in the positive direction by those tokens j with $h(j) = 1$, and is pulled in the negative direction by the rest of the tokens; hence the name "Tug-of-War Sketch". Clearly, the absolute value of x never exceeds $f_1 + \cdots + f_k = m$, so it takes $O(\log m)$ bits to store this sketch. It also takes $O(\log n)$ bits to store the hash function h, for an appropriate 4-universal family.

6.1.1 The Quality of the Estimate

Let X denote the value of x after the algorithm has processed σ. For convenience, define $Y_j = h(j)$ for each $j \in [n]$. Then $X = \sum_{j=1}^{n} f_j Y_j$. Therefore,

$$\mathbb{E}[X^2] = \mathbb{E}\left[\sum_{j=1}^{n} f_j^2 Y_j^2 + \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} f_i f_j Y_i Y_j\right] = \sum_{j=1}^{n} f_j^2 + \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} f_i f_j \,\mathbb{E}[Y_i]\,\mathbb{E}[Y_j] = F_2\,,$$

where we used the fact that $\{Y_j\}_{j \in [n]}$ are pairwise independent (in fact, they are 4-wise independent, because h was picked from a 4-universal family), and then the fact that $\mathbb{E}[Y_j] = 0$ for all $j \in [n]$. This shows that the algorithm's output, X^2, is indeed an unbiased estimator for F_2.

The variance of the estimator is $\mathrm{Var}[X^2] = \mathbb{E}[X^4] - \mathbb{E}[X^2]^2 = \mathbb{E}[X^4] - F_2^2$. We bound this as follows. By linearity of expectation, we have

$$\mathbb{E}[X^4] = \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \sum_{\ell=1}^n f_i f_j f_k f_\ell \, \mathbb{E}[Y_i Y_j Y_k Y_\ell] .$$

Suppose one of the indices in (i, j, k, ℓ) appears exactly once in that 4-tuple. Without loss of generality, we have $i \notin \{j, k, \ell\}$. By 4-wise independence, we then have $\mathbb{E}[Y_i Y_j Y_k Y_\ell] = \mathbb{E}[Y_i]\,\mathbb{E}[Y_j Y_k Y_\ell] = 0$, because $\mathbb{E}[Y_i] = 0$. It follows that the only potentially nonzero terms in the above sum correspond to those 4-tuples (i, j, k, ℓ) that consist either of one index occurring four times, or else two distinct indices occurring twice each. Therefore we have

$$\mathbb{E}[X^4] = \sum_{j=1}^n f_j^4 \,\mathbb{E}[Y_j^4] + 6 \sum_{i=1}^n \sum_{j=i+1}^n f_i^2 f_j^2 \,\mathbb{E}[Y_i^2 Y_j^2] = F_4 + 6 \sum_{i=1}^n \sum_{j=i+1}^n f_i^2 f_j^2 ,$$

where the coefficient "6" corresponds to the $\binom{4}{2} = 6$ permutations of (i, i, j, j) with $i \neq j$. Thus,

$$\begin{aligned}
\mathrm{Var}[X^2] &= F_4 - F_2^2 + 6 \sum_{i=1}^n \sum_{j=i+1}^n f_i^2 f_j^2 \\
&= F_4 - F_2^2 + 3\left(\Big(\sum_{j=1}^n f_j^2 \Big)^2 - \sum_{j=1}^n f_j^4 \right) \\
&= F_4 - F_2^2 + 3(F_2^2 - F_4) \le 2F_2^2 .
\end{aligned}$$

6.2 The Final Sketch

As before, having bounded the variance, we can design a final sketch from the above basic sketch by a median-of-means improvement. By Lemma 5.4.1, this will blow up the space usage by a factor of

$$\frac{O(1) \cdot \mathrm{Var}[X^2]}{\varepsilon^2\, \mathbb{E}[X^2]^2} \cdot \log \frac{1}{\delta} \le \frac{O(1) \cdot 2F_2^2}{\varepsilon^2 F_2^2} \cdot \log \frac{1}{\delta} = O\left(\frac{1}{\varepsilon^2} \log \frac{1}{\delta} \right)$$

in order to give an (ε, δ)-approximation. Thus, we have estimated F_2 using space $O(\varepsilon^{-2} \log(\delta^{-1})(\log m + \log n))$, with a sketching algorithm that in fact computes a *linear* sketch.

6.2.1 A Geometric Interpretation

The AMS Tug-of-War Sketch has a nice geometric interpretation. Consider a final sketch that consists of t independent copies of the basic sketch. Let $M \in \mathbb{R}^{t \times n}$ be the matrix that "transforms" the frequency vector $\mathbf{f}$ into the t-dimensional sketch vector $\mathbf{x}$. Note that M is not a fixed matrix but a random matrix with ± 1 entries: it is drawn from a certain distribution described implicitly by the hash family. Specifically, if M_{ij} denotes the (i, j)-entry of M, then $M_{ij} = h_i(j)$, where h_i is the hash function used by the ith basic sketch.

Let $t = 6/\varepsilon^2$. By stopping the analysis in Lemma 5.4.1 after the Chebyshev step (and before the "median trick" Chernoff step), we obtain that

$$\Pr_M \left[\left| \frac{1}{t} \sum_{i=1}^t x_i^2 - F_2 \right| \ge \varepsilon F_2 \right] \le \frac{1}{3} .$$

Thus, with probability at least $2/3$, we have

$$\left\|\frac{1}{\sqrt{t}} M\mathbf{f}\right\|_2 \;=\; \frac{1}{\sqrt{t}}\|\mathbf{x}\|_2 \;\in\; \left[\sqrt{1-\varepsilon}\cdot\|\mathbf{f}\|_2, \sqrt{1+\varepsilon}\|\mathbf{f}\|_2\right] \;\subseteq\; [(1-\varepsilon)\|\mathbf{f}\|_2, (1+\varepsilon)\|\mathbf{f}\|_2] \,.$$

This can be interpreted as follows. The (random) matrix $M/\sqrt{t}$ performs a "dimension reduction", reducing an n-dimensional vector $\mathbf{f}$ to a t-dimensional sketch $\mathbf{x}$ (with $t = O(1/\varepsilon^2)$), while preserving ℓ_2-norm within a $(1 \pm \varepsilon)$ factor. Of course, this is only guaranteed to happen with probability at least $2/3$. But clearly this correctness probability can be boosted to an arbitrary constant less than 1, while keeping $t = O(1/\varepsilon^2)$.

The "amazing" AMS sketch now feels quite natural, under this geometric interpretation. We are simply using dimension reduction to maintain a low-dimensional image of the frequency vector. This image, by design, has the property that its ℓ_2-length approximates that of the frequency vector very well. Which of course is what we're after, because the second frequency moment, F_2, is just the square of the ℓ_2-length.

Since the sketch is linear, we now also have an algorithm to estimate the ℓ_2-difference $\|\mathbf{f}(\sigma) - \mathbf{f}(\sigma')\|_2$ between two streams σ and σ'.