0368-3248-01-Algorithms in Data Mining **Fall 2011**

Lecture 5: Estimating Frequency Moments

Lecturer: Edo Liberty

Warning: *This note may contain typos and other inaccuracies which are usually discussed during class. Please do not cite this note as a reliable source. If you find mistakes, please inform me.*

Assume we have a stream A, of length N which is composed of m different types of items $a_1, \ldots, a_m$ each of which repeats itself $n_1, \ldots, n_m$ times (in arbitrary order) We define the frequency moments f_k as:

$$f_k = \sum_{i=1}^{m} n_i^k$$

Our aim to to process the stream in one element at a time and attain an (ϵ, δ)-approximation. This means, that our estimate is up to multiplicative factor $(1 \pm \epsilon)$ with probability at least $1 - \delta$. Note that f_0 is the number of distinct elements in the stream m and that f_1 is the number of elements N. f_2 is also an important quantity which represents how "skewed" the distribution of the elements in stream is.

Let's first assume that we know N in advance. This is not necessary and we will fix it later. But for now, it makes our analysis simpler.

Let us first define a random variable X. We choose an index $q \in [1, \ldots, N]$ uniformly at random. Let a be the element in place q in the stream, i.e. $a = A_q$. Define by r the number of times a appears in the stream after location q, including. In other words $r = |\{i | A_i = a \ , \ i \geq q\}|$. We define X:

$$X = N(r^k - (r-1)^k)$$

We claim that $E[X] = f_k$. Let us define the variable $e_{i,j}$ which indicates the event that the index q is such that $A_q = a_i$ and a_i appears exactly j times after the location q. Note that the events $e_{i,j}$ are disjoint and that if $e_{i,j}$ happens than r takes the value j. Therefore, $X = \sum_{i,j} e_{i,j} N(j^k - (j-1)^k)$. Moreover, $\Pr[e_{i,j}] = \frac{n_i}{N}\frac{1}{n_i} = \frac{1}{N}$ since the probability of choosing a_i is $\frac{n_i}{N}$ and given that this happens the probability of each index (out of the locations of a_i) is equal to $\frac{1}{n_i}$.

$$\begin{aligned} E[X] &= \sum_{i,j} E[e_{i,j} N(j^k - (j-1)^k)] \\ &= \sum_{i=1}^{m}\sum_{j=1}^{n_i} \Pr[e_{i,j}] N(j^k - (j-1)^k) \\ &= \sum_{i=1}^{m}\sum_{j=1}^{n_i} (j^k - (j-1)^k) \\ &= \sum_{i=1}^{m} n_i^k = f_k \ . \end{aligned}$$

It is somewhat complicated and tedious to compute the variance of X. Citing from the paper [] we use the fact that

$$Var[X] \leq k m^{1-1/k} f_k^2 \ .$$

We define Y as the mean of s different copies of X, $Y = \frac{1}{s}\sum_{i=1}^{s} X_i$. Clearly, $E[Y] = E[X] = f_k$ and $Var[Y] \leq Var[X]/s = k m^{1-1/k} f_k^2/s$. Using Chebyshev's inequality we have that

$$\Pr[|Y - f_k| > \varepsilon f_k] \leq \frac{Var[Y]}{\varepsilon^2 f_k^2} \leq \frac{k m^{1-1/k}}{\varepsilon^2 s} \leq \delta.$$

where the last inequality holds if $s \geq \frac{km^{1-1/k}}{\varepsilon^2\delta}$.

Estimating f_0

This bound is not the most efficient algorithm for approximating the zero'th frequency moment (which is the number of distinct elements, m). Here we will describe a more efficient algorithm which is a merging of ideas from [] and [].

First, assume a hash function $h : a \rightarrow [0, 1]$ uniformly. Let us define a random variable $X = min_i h(a_i)$. Intuitively, X should be roughly $1/m$ and therefore $1/X$ should be a fair estimate of m. This is almost true. In what comes next we make this into an exact statement.

Let us first compute the expectation of X. The distribution function f_X of the X is $f_X(x) = m(1-x)^{m-1}$. This is because, we have m different choices for the minimal element and for every value it takes, x, all the rest $m - 1$ values need to be higher than it (w.p $(1 - x)^{m-1}$). Therefore, u:

$$\begin{aligned}
E[X] &= \int_0^1 xm(1-x)^{m-1}dx \\
&= \int_0^1 (1-y)my^{m-1}dy \\
&= \int_0^1 my^{m-1}dy - \int_0^1 my^m dy \\
&= 1 - \frac{m}{m+1} = \frac{1}{m+1}
\end{aligned}$$

This is after the substitution $y = 1 - x$. We now compute the variance of X. For that we first compute $E[X^2]$.

$$\begin{aligned}
E[X^2] &= \int_0^1 x^2 m(1-x)^{m-1}dx \\
&= \int_0^1 (1-y)^2 my^{m-1}dy \\
&= \int_0^1 my^{m-1}dy - \int_0^1 2my^m dy + \int_0^1 my^{m+1}dy \\
&= 1 - \frac{2m}{m+1} + \frac{m}{m+2} \leq \frac{3}{(m+1)^2}
\end{aligned}$$

Thus, the standard deviation of $\sigma(X)$ is in the same order of magnitude as its expectation $E[X]$. To reduce this ratio we again define $Y = \frac{1}{s}\sum_{i=1}^{s} X_i$ for which $E[Y] = \frac{1}{m+1}$. and $Var[Y] \leq \frac{2}{s(m+1)^2}$.

Using Chebyshev's inequality we get that

$$\Pr[|Y - \frac{1}{m+1}| \geq \frac{\varepsilon/2}{m+1}] \leq \frac{8}{\varepsilon^2 s} \leq \delta$$

if $s \geq \frac{8}{\varepsilon^2\delta}$. Therefore, multiplying this procedure $\frac{8}{\varepsilon^2\delta}$ different hash function and taking their mean minimal value guaranties that with probability at least $1 - \delta$ we have $\frac{1}{m+1}(1 - \varepsilon/2) \leq Y \leq \frac{1}{m+1}(1 + \varepsilon/2)$. In other words: $(m+1)\frac{1}{1+\varepsilon/2} \leq \frac{1}{Y} \leq (m+1)\frac{1}{1-\varepsilon/2}$. But, since $\frac{1}{1-\varepsilon/2} \leq 1 + \varepsilon$ and $1 - \varepsilon \leq \frac{1}{1+\varepsilon/2}$ we get the desired results that $(m+1)(1-\varepsilon) \leq \frac{1}{Y} \leq (m+1)(1+\varepsilon)$

Estimating f_2

We will give here a better estimator of f_2. Assume a hash function $h : a \rightarrow \{-1, 1\}$ with probability 1/2 each. Define $Z = \sum_{i=1}^{N} h(A_i) = \sum_{i=1}^{m} n_i h(a_i)$.

Consider the variable $X = Z^2$. As usual, we will begin with computing the expectation and variance of X.

$$\begin{aligned}
E[X] &= E[Z^2] = E[\sum_{i=1}^{m} n_i h(a_i)^2] \\
&= E[(\sum_{i=1}^{m} n_i h(a_i))(\sum_{i'=1}^{m} n_{i'} h(a_{i'}))] \\
&= \sum_{i=1}^{m}\sum_{i'=1}^{m} n_i n_{i'} E[h(a_i)h(a_{i'})] \\
&= \sum_{i=1}^{m} n_i^2 = f_2
\end{aligned}$$

Similarly,

$$\begin{aligned}
E[X^2] &= E[Z^4] = \sum_{i=1}^{m} n_i^4 + 6 \sum_{1 \le i < i' \le m} n_i^2 n_{i'}^2 \\
Var[X] &= E[X^2] - E^2[X] \le 4 \sum_{1 \le i < i' \le m} n_i^2 n_{i'}^2 \le 2 f_2
\end{aligned}$$

Finally, defining $Y = \frac{1}{s}\sum_{i=1}^{s} X_i$, where each X_i is an independent copy of X we have that:

$$\Pr[|Y - f_2| \ge \varepsilon f_2] \le \delta$$

if $s \ge \frac{2}{\varepsilon^2 \delta}$.

Connection to random projections (next class)

Consider the s hash functions $h_i : a \to \{-1, 1\}$ we used in estimating the second frequency moment. Consider the matrix $H \in \mathbb{R}^{s \times m}$ such that $H(i,j) = h_i(j)$. Also, consider representing each input element a_i by $\vec{a_i}$, the i'th standard basis vector in $\mathbb{R}^m$ (the vector whose i'th entry is equal to 1 and the rest are zero). Analogously, $\vec{A_i}$ is the vector representing the i'th element in the stream. Remember that our estimate Y of f_2 was $\frac{1}{s}\sum_{i=1}^{s} Z_i^2 = ||\frac{1}{\sqrt{s}}\vec{Z}||^2$. Moreover, from the definition of $\vec{Z}$, H, and $\vec{A_i}$ we have that $\vec{Z} = \sum_{i=1}^{N} H\vec{A_i} = H\sum_{i=1}^{N}\vec{A_i} = \frac{1}{\sqrt{s}}H\vec{A}$. Here, $\vec{A} = \sum_{i=1}^{N}\vec{A_i} = [n_1, n_2, \ldots, n_m]$. Note however, that $f_2 = ||\vec{A}||^2$ by definition of the second frequency moment. We get that for any stream and any element frequencies $||\frac{1}{\sqrt{s}}H\vec{A}||^2 \approx_{(\varepsilon,\delta)} ||\vec{A}||^2$. In other words, multiplying any vector $\vec{A}$ by the matrix $\frac{1}{\sqrt{s}}H$ is very likely to preserve its norm. We will see that this phenomenon is in fact more overreaching and has some serious consequences on point ensembles in high dimensional Euclidean spaces.