Problem 47

Ryan Burmeister

March 8, 2016

The Flajolet_Martin algorithm is used to approximate the number of unique elements in a set. Using the algorithm also ensure that only a single pass over the set is required and that the amount of space required is logarithmic compared to the maximum possible number of distinct elements.

The algorithm works by initializing a vector of length L to all 0s. We then assume that we have a hash function which maps an input x to a value in the range $\left[0, 2^L - 1\right]$. We then determine the first index for which the bit is 1 in the hash. This index is then set to 1 in our vector of length L. R is then defined to be the smalles index in our vector which is 0. By performing the computation $2^R/\phi$, where $\phi \approx 0.77351$, we can estimate the cardinality of our vector.

From an intuitive standpoint, this algorithm seems to make a lot of sense. The probability that each hash is even or odd is $\frac{1}{2}$. Therefore, the first bit in our vector will be set to 1 approximately 50% of the time. The chances that more significant bits are set to 1 occur with probability $\frac{1}{2^n}$ given that our hash function uniformly distributes the output.

Example:

$$
\begin{gathered}
13 = 1101_2 \rightarrow 0,\ 0^{th} \text{ bit in vector is set to 1} \\
8 = 1000_2 \rightarrow 3,\ 3^{rd} \text{ bit in vector is set to 1} \\
12 = 1100_2 \rightarrow 2,\ 2^{nd} \text{ bit in vector is set to 1} \\
9 = 1001_2 \rightarrow 1,\ 0^{th} \text{ bit in vector is set to 1} \\
10 = 1010_2 \rightarrow 1,\ 1^{st} \text{ bit in vector is set to 1}
\end{gathered}
$$

vector = 01111
The esitmate of the cardinality of the set from which the hashes were drawn is $2^4/0.77351 \approx 20.57$.

Sources: Wikipedia - Flajolet-Martin Algorithm
Flajolet, Philippe; Martin, G. Nigel (1985). "Probabilistic counting algorithms for data base applications"