Explain Flajolet Martin Algorithm with example.

written 3.9 years ago by swatisharma (https://www.ques10.c • 1.1k

modified 16 months ago by besharateif739 (https://www.ques10.com • 10

2

49k views

big data analytics (https://www.ques10.com/t/big%20data%20analytics/)

Thanks for excellent explanation. But I have one question. When x=4, Hash function's output is 5 and there are 5 zeros in tail. Therefor our max R is 5 and distinct elements will be 32. But in input stream we have 4 distinct elements. I think, we should not count all-zero element. May you explain more, please?

16 months ago by besharateif739 (https://www.ques10.com/u/355589/ • 10

2 Answers

written 3.9 years ago by

yashbeer (https://www.ques10.com/u/. ★ **10k**

8

5.5k views

Flajolet-Martin algorithm approximates the number of unique objects in a stream or a database in one pass. If the stream contains n elements with m of them unique, this algorithm runs in $O(n)$ time and needs $O(log(m))$ memory.

Algorithm:

1. Create a bit vector (bit array) of sufficient length L, such that $2^L > n$, the number of elements in the stream. Usually a 64-bit vector is sufficient since 2^64 is quite large for most purposes.

2. The i-th bit in this vector/array represents whether we have seen a hash function value whose binary representation ends in 0^i. So initialize each bit to 0.

3. The i-th bit in this vector/array represents whether we have seen a hash function value whose binary representation ends in 0i. So initialize each bit to 0.

4. The i-th bit in this vector/array represents whether we have seen a hash function value whose binary representation ends in 0i. So initialize each bit to 0.

5. Once input is exhausted, get the index of the first 0 in the bit array (call this R). By the way, this is just the number of consecutive 1s (i.e. we have seen $0, 00, \ldots, 0^{R-1}$ as the output of the hash function) plus one.

6. Calculate the number of unique words as $2^R/\phi$, where ϕ is 0.77351. A proof for this can be found in the original paper listed in the reference section.

7. The standard deviation of R is a constant: $\sigma(R) = 1.12$. (In other words, R can be off by about 1 for **1 - 0.68 = 32%** of the observations, off by 2 for about **1 - 0.95 = 5%** of the observations, off by 3 for **1** - **0.997** = **0.3%** of the observations using the Empirical rule of statistics). This implies that our count can be off by a factor of 2 for 32% of the observations, off by a factory of 4 for 5% of the observations, off by a factor of 8 for 0.3% of the observations and so on.

Example:

$S = 1, 3, 2, 1, 2, 3, 4, 3, 1, 2, 3, 1$

$h(x) = (6x + 1)\ mod\ 5$

Assume |b| = 5

x	h(x)	Rem	Binary	r(a)
1	7	2	00010	1

x	h(x)	Rem	Binary	r(a)
3	19	4	00100	2
2	13	3	00011	0
1	7	2	00010	1
2	13	3	00011	0
3	19	4	00100	2
4	25	0	00000	5
3	19	4	00100	2
1	7	2	00010	1
2	13	3	00011	0
3	19	4	00100	2
1	7	2	00010	1

R = max(r(a)) = 5

So no. of distinct elements = $N = 2^R = 2^5 = 32$

- We may want to know how many different elements have appeared in the stream.
- For example, we wish to know how many distinct users visited the website till now or in last 2 hours.
- If no of distinct elements required to process many streams then keeping data in main memory is challenge.
- FM algorithm gives an efficient way to count the distinct elements in a stream.
- It is possible to estimate the no. of distinct elements by hashing the elements of the universal set to a bit string that is sufficiently long.
- The length of the bit string must be sufficient that there are more possible results of the hash function than there are elements in the universal set.

- Whenever we apply a hash function h to a stream element a, the bit string h(a) will end in some number of oS, possibly none.
- Call this as tail length for a hash.
- Let R be the maximum tail length of any a seen so far in the stream.
- Then we shall use estimate 2^R for the number of distinct elements seen in the stream.
- Consider a stream as:

S = {1, 2, 1, 3}

Let hash function be 2x + 2 mod 4

- When we apply the hash function we get reminder represented in binary as follows:

000, 101, 000 considering bit string length as 3.

- Maximum tail length R will be 3.
- No of distinct elements will be $2^R = 2^3 = 8$
- Here the estimates may be too large or too low depending on hash function.
- We may apply multiple hash functions and combine the estimate to get near accurate values.