COMP 480/580 — Probabilistic Algorithms and Data Structure Feb 18, 2020

Lecture 10: Stream Estimation 2: Count Sketches

Lecturer: Anshumali Shrivastava *Scribe By: Amber Lu*

1 Count-Min Sketch

Let's say that we are interested in counting the frequency of visitation for different URLs. For example, if a single person visits Facebook, then we would increment the count for Facebook by 1. Then, rather than storing all hits to all URLs, which is too large to feasibly be stored, we consider turnstil streaming models.

The problem can be described like such. We have a vector L of elements which is so large that we can never store it. As we can never store it, we cannot access its actual elements at time t. Instead, we make a sketch, an approximation of L, of the form (i, Δ_i), for $i > 0$ and time $= t$, where i represents the index that is being accessed and Δ_i the increment of change at the index. The only updates to the model are done in this manner, hence the name "turnstile". The size of this sketch is much smaller than L, and thus can be stored.

1.1 Negative counts

In this subsection, we explore the case when $\Delta_i < 0$. When $\Delta_i < 0$, as long as $c_i > 0$, the count-min sketch proof holds true because the count is still being overestimated. The minimum of $\hat{c}_i$ will always be itself.

That count-min sketch does not allow for negative c_i and always over-estimates could be seen as weaknesses of the algorithm. We introduce a new algorithm which, while very similar, aims to address these issues.

2 Count Sketch Algorithm

We now look at the count sketch algorithm. It introduces a second hash function, which we will refer to as $g(i)$, which randomly maps between $\{-1, 1\}$. In all cases $g(i) = g(i)$, which means that inputting the same value will always end in the same result.

An advantage over count-min sketch possessed by count sketch is that count sketch allows for $c_i < 0$. Furthermore, by introducing this second hash function $g(i)$, the overestimation of $\hat{c}_i$ present in count-min hash due to taking the minimum no longer applies.

Furthermore, it is not dependent on the Markov approximation, count sketch allows for counts of negative values. Since count sketch cannot guarantee that $c_i > 0$, we will use Chebyshev rather than Markov approximation to demonstrate our estimator is reasonable.

2.1 Definitions

Count sketch involves two functions, $update(i, \Delta_i)$ and $query(i)$. **Update** is used to update the sketch at position i with increment Δ_i. **Query** is used to check the value of i at time t.
We have two hash functions,

$$h : [1...N] \xrightarrow{\text{2-universal}} [1 \dots R]$$
$$g : [1...N] \xrightarrow{\text{2-universal}} \{-1, 1\}$$

Algorithm 1: Update

```
Input: position $i$, increment $\Delta_i$
for $j = i\ to\ d$ do
    $S[j, h_j(i)] += \Delta_i * g_j(i)$
end
```

Algorithm 2: Query algorithm

```
Input: position $i$
Output: The value of position $i$ at time $t$
return $\frac{median}{j}[S(j, h_j(i)) * g_j(i)]$;
```

2.2 Proof

The count, $\hat{c}_i$ can be calculated using $g_i * S(h(i))$. First, looking at just a singular row of the count sketch matrix for simplicity, $g_i * S(h(i)) =$

$$c_i * g_1^2(i) + \sum_{j=1,j\neq i}^{N} \mathbb{1}_{j\in h(i)} * c_j * g_1(i) * g_1(j)$$

We can use this to next calculate the expected value. The expected value, can be calculated in a similar manner to count-min sketch, merely updating it to reflect the new $g(i)$:

$$E[\hat{c}_i] = c_i + \sum_{j=1,j\neq i}^{N}$$
$$= c_i + E[\sum_{i=1,j\neq i}^{N} \mathbb{1}_{j\in h(i)} * c_j * g_1(i) * g_1(j)]$$

Knowing that $E[g_1(i)] = 0$ allows us to reduce to

$$E[\hat{c}_i] = c_i$$

We can do something similar to figure out the variance, which we will need for the Chebyshev inequality. The variance is

$$var(\hat{c}_i) = E[(\hat{c}_i - c_j)^2]$$
$$= E[\sum_{j=1,j\neq i}^{N} \mathbb{1}_{j\in h_j(i)} * c_j^2 * g_1^2(j) * g_1^2(i)]$$
$$= E[\sum_{j=1,j\neq i}^{N} c_j^2 \mathbb{1}_{j\in h_1(i)} + c_i^2 * \frac{1}{R}$$

Just like the expectation, we can eliminate the entire right term of the equation due to $E[g(i)] = 0$, giving us:

$$var(\hat{c}_i) = E[\sum_j \mathbb{1}_j * c_j^2]$$
$$\leq \frac{1}{R} * \sum^2 = \epsilon \sum^2$$

Finally, we use the Chebyshev approximation to demonstrate that the update function is a reasonable estimator for i.

$$Pr(|X - E[X]| > a) < \frac{var(X)}{a^2}$$
$$Pr(|X - E[X]| > k\sigma) < \frac{1}{k^2}$$
$$Pr(Err > \sqrt{\frac{3}{R}\epsilon^2}) < \frac{1}{3} \text{ where } k = \sqrt{3}$$

The results of Chebyshev show that we have a reasonable estimator for c_i.