Corrections to "LD-Sketch: A Distributed Sketching Design for Accurate and Scalable Anomaly Detection in Network Data Streams"

Qun Huang and Patrick P. C. Lee
{qhuang,pclee}@cse.cuhk.edu.hk

October 2014

Abstract

In this article, we describe the corrections to our paper "LD-Sketch: A Distributed Sketching Design for Accurate and Scalable Anomaly Detection in Network Data Streams" published at IEEE INFOCOM 2014. We also clarify the complexity issue raised by some readers.

1 Corrections to Lemmas and Theorems

1.1 Problems

The main problem in the INFOCOM paper is that we incorrectly apply Markov's inequality in Lemma 6 and 7. In the proof of the two lemmas, we bound the error rate by applying the inequality to the random variable $V_{i,j}$. Specifically in Lemma 6, we derive $Pr\{V_{i,j} \geq (k+1)\phi\} \leq (\frac{U}{w(k+1)\phi})$ by Markov's inequality (Lemma 2). However, this is incorrect because key x has been hashed to bucket (i,j) and hence $V_{i,j} = S(x) + \sum_{y \neq x, f_i(y)=j} S(y)$. The Markov's inequality should apply to the latter term $\sum_{y \neq x, f_i(y)=j} S(y)$ instead of the entire $V_{i,j}$.

1.1.1 Applying Markov's inequality correctly

If we apply Markov's inequality correctly, the results and proofs of the lemmas are as follows.

Lemma 6. *For key x with $S(x) < \phi$, it is reported as a heavy hitter with probability at most $(\frac{U}{w(\phi - S(x))})^r$.*

Proof. In heavy hitter detection, we set $T = \phi$. For key x with $S(x) < \phi$, there always exists an integer $k \geq 0$ such that $S(x) < \frac{\phi}{k+2-\frac{1}{k+1}}$. If $V_{i,j} < (k+1)\phi$, by Lemma 4, $S^{up}_{i,j}(x) < (\frac{k+1}{k+2})\phi + (1 - \frac{1}{(k+1)(k+2)})S(x) = (\frac{k+1}{k+2})\phi + (1 - \frac{1}{(k+1)(k+2)})\frac{\phi}{k+2-\frac{1}{k+1}} = \phi$. So x is not reported as a heavy hitter. On the other hand, x is reported as a heavy hitter only if $V_{i,j} \geq (k+1)\phi$ for all row i. ~~By Lemma 2, the probability that $V_{i,j} \geq (k+1)\phi$ is at most $(\frac{U}{w(k+1)\phi})$.~~ Note that $V_{i,j} = S(x) + \sum_{y \neq x, f_i(y)=j} S(y)$. By Markov inequality, $Pr\{V_{i,j} \geq (k+1)\phi\} = Pr\{\sum_{y \neq x, f_i(y)=j} S(y) \geq (k+1)\phi - S(x)\} \leq \frac{U}{w[(k+1)\phi - S(x)]}$. Since the r hash functions are independent, the probability that x is reported as a heavy hitter is $(\frac{U}{w[(k+1)\phi - S(x)]})^r \leq (\frac{U}{w(\phi - S(x))})^r$. □

The following corollary is weaker than our previous claim because it fails to bound the error rate for keys with actual sum between $(1-\epsilon)\phi$ and ϕ.

Corollary. *For key x with $S(x) \leq (1-\epsilon)\phi$, it will be reported as a heavy hitter with probability at most $(\frac{U}{w\epsilon\phi})^r$.*

Theorem 1. *Consider an LD-Sketch with* $w = 2\epsilon H, r = \log \frac{1}{\delta}$ *and* $T = \phi$*. It reports all heavy hitters. A non-heavy hitter with sum less than* $(1-\epsilon)\phi$ *is reported with probability at most* δ*. The expected space is* $O(H \log \frac{1}{\delta})$*. The expected time complexity of updating a data item is* $O(\log \frac{1}{\delta})$*, and that of detection is* $O(H \log \frac{1}{\delta})$*.*

Similarly, Lemma 7 should also be corrected.

Lemma 7. *For key* x *with* $D(x) < (1-\epsilon)\phi$*, it is reported as a heavy changer with probability at most* $(\frac{2U}{w\epsilon\phi})(\frac{2U}{w(\epsilon\phi - S(x))})^r$.

However, the modified Lemma 7 is still problematic. Its result includes term $\epsilon\phi - S(x)$, but our condition is $D(x) < (1-\epsilon)\phi$ so that the results involving $S(x)$ makes no sense.

1.2 Corrections

We change the values of T. We set $T = \epsilon\phi$ ($T = \phi$ in INFOCOM paper) for heavy hitter detection and $T = \epsilon\phi/2$ ($T = \epsilon\phi$ in INFOCOM paper) for heavy changer detection.

We first modify Lemma 3 and Lemma 4 (only Lemma 4 is modified, but we change their order for the ease of presentation). In INFOCOM paper, we derive the estimate errors with respect to $V_{i,j}$. Now, we provide a more accurate error bound. Given any bucket (i,j) and a key x in the bucket (i.e. $f_i(x) = j$), the estimate errors for key x depend on other keys in the same bucket. Thus, we fix the problem by deriving the estimate errors with respect to sum of other keys in bucket (i,j), i.e. $\sum_{y \neq x, f_i(y)=j} S(y) = V_{i,j} - S(x)$. The following two lemmas are the errors of lower and upper estimated sum, respectively.

Lemma 3. *For bucket* (i,j) *and* x *with* $f_i(x) = j$*, if* $\sum_{y \neq x, f_i(y)=j} S(y) \leq \frac{(k+1)^2}{k+2}T$*,*

$$S(x) - \frac{k+1}{k+2}T \leq S^{low}_{i,j}(x) \leq S(x).$$

Proof. From Algorithm 1, $S(x) \geq A_{i,j}[x] = S^{low}_{i,j}(x)$ since $A_{i,j}[x]$ is never incremented due to other items not belonging to x.

Now we prove the lower bound of $S^{low}_{i,j}(x)$. In each expansion number κ ($0 \leq \kappa \leq k$), $A_{i,j}$ contains $l_{i,j} = (\kappa+1)(\kappa+2) - 1$ counters, so key x is decremented by at most $\frac{T}{l_{i,j}+1} = \frac{T}{(\kappa+1)(\kappa+2)}$ (by Lemma 1). Note that once key x is decremented by a value, $(\kappa+1)(\kappa+2)-1$ other keys are also decremented by the same value. Therefore, to achieve the maximum decrement of key x, the total decrement of other keys in the bucket is $\frac{(\kappa+1)(\kappa+2)-1}{(\kappa+1)(\kappa+2)}T$.

Thus, before the expansion number $k+1$, $A_{i,j}[x]$ is decremented by at most $\sum_{\kappa=0}^{k} \frac{T}{(\kappa+1)(\kappa+2)} = \sum_{\kappa=0}^{k}(\frac{1}{\kappa+1} - \frac{1}{\kappa+2})T = (\frac{k+1}{k+2})T$. To achieve this maximum decrement, it requires the total decrement of other keys to be $\sum_{\kappa=0}^{k}(\frac{(\kappa+1)(\kappa+2)-1}{(\kappa+1)(\kappa+2)}T = (k+1)T - \frac{k+1}{k+2}T = \frac{(k+1)^2}{k+2}T$. Since $\sum_{y \neq x, f_i(y)=j} S(y) \leq \frac{(k+1)^2}{k+2}T$, the results follow. □

Lemma 4. *For bucket* (i,j) *and key* x *with* $f_i(x) = j$*, if* $kT < \sum_{y \neq x, f_i(y)=j} S(y) \leq (k+1)T$*,*

$$S(x) \leq S^{up}_{i,j}(x) \leq S(x) + (\frac{k+1}{k+2})T.$$

Proof. From Algorithm 1, $A_{i,j}[x]$ is decremented by at most $e_{i,j}$, so $A_{i,j}[x] \geq S(x) - e_{i,j}$ and hence $S(x) \leq S^{up}_{i,j}(x)$.

On the other hand, let $e'_{i,j} = S(x) - S^{low}_{i,j}(x)$ be the decrements of $A_{i,j}[x]$. Thus, $e_{i,j} - e'_{i,j}$ corresponds to the decrements of $A_{i,j}[y]$ for any $y \neq x$ when $A_{i,j}[x]$ is not decremented, and is contributed by sum of other keys not x in the bucket, whose maximum value is $\sum_{y \neq x, f_i(y)=j} S(y)$. Thus, $e_{i,j} - e'_{i,j}$ ~~achieves maximum if $V_{i,j} - S(x) > kT$, and its value~~ is at most $\sum_{\kappa=0}^{k} \frac{T}{(\kappa+1)(\kappa+2)} = \frac{k+1}{k+2}T$. Since $S^{up}_{i,j}(x) = S^{low}_{i,j}(x) + e_{i,j} = S(x) + (e_{i,j} - e'_{i,j})$, the results follow. □

Based on the new Lemma 4, Lemma 5 is slightly modified.

Lemma 5. *Using LD-Sketch, if key x has $S(x) \geq \phi$, it must be reported as a heavy hitter; if x has $D(x) \geq \phi$, it must be reported as a heavy changer.*

Proof. By Lemma 3, if $S(x) \geq \phi$, then for any bucket (i, j) associated with x, we must have $A_{i,j}[x] = S^{low}_{i,j}(x) > S(x) - T \geq 0$ (note that ~~$T = \phi$~~ $T = \epsilon\phi$ for heavy hitter detection and ~~$T = \epsilon\phi$~~ $T = \epsilon\phi/2$ for heavy changer detection). For heavy changer detection, if $D(x) \geq \phi$, there must be at least one epoch where $S(x) \geq 0$. Therefore, x must be kept in the associative array of its corresponding buckets.

By Lemma 4, we know $S(x) \leq S^{up}_{i,j}$ for every bucket (i, j). If $S(x) \geq \phi$, then $S^{up}_{i,j}(x) \geq \phi$, so x must be reported as a heavy hitter. Also, $D(x) \leq D_{i,j}(x)$ for every bucket (i, j). If $D(x) \geq \phi$, then $D_{i,j}(x) \geq \phi$, so x must be reported as a heavy changer. □

Now we modify Lemma 6 and Lemma 7 as follows.

Lemma 6. *For key x with $S(x) \leq (1-\epsilon)\phi$, it is never be reported as a heavy hitter. For key x with $(1-\epsilon)\phi < S(x) < (1-\epsilon/2)\phi$, it is reported as a heavy hitter with probability at most $(\frac{U}{w\epsilon\phi})^r$.*

Proof. In heavy hitter detection, we set $T = \epsilon\phi$. By Lemma 4, for key x with $S(x) \leq (1-\epsilon)\phi$, $S^{up}_{i,j}(x) \leq S(x) + T \leq \phi$. Thus, key x will never be reported.

For key x with $(1-\epsilon)\phi < S(x) < (1-\epsilon/2)\phi$, there always exists an integer $k \geq 0$ such that $S(x) < (1-\epsilon)\phi + \frac{\epsilon\phi}{k+2}$. If $\sum_{y \neq x, f_i(y)=j} S(y) \leq (k+1)\epsilon\phi$, by Lemma 4, $S^{up}_{i,j}(x) \leq (\frac{k+1}{k+2})\epsilon\phi + S(x) < (\frac{k+1}{k+2})\epsilon\phi + (1-\epsilon)\phi + \frac{\epsilon\phi}{k+2} = \phi$. So x is not reported as a heavy hitter. Thus, x is reported as a heavy hitter only if $\sum_{y \neq x, f_i(y)=j} S(y) \geq (k+1)\epsilon\phi$ for all row i. By Markov's inequality, $Pr\{\sum_{y \neq x, f_i(y)=j} S(y) \geq (k+1)\epsilon\phi\} \leq \frac{U}{w[(k+1)\epsilon\phi]}$. Since the r hash functions are independent, the probability that x is reported as a heavy hitter is $(\frac{U}{w(k+1)\epsilon\phi})^r \leq (\frac{U}{w\epsilon\phi})^r$. □

Lemma 7. *For key x with $D(x) \leq (1-\epsilon)\phi$, it will never be reported as a heavy changer. For key x with $(1-\epsilon)\phi < D(x) < (1-\epsilon/2)\phi$, it is reported as a heavy changer with probability at most $(\frac{6U}{w\epsilon\phi})^r$.*

Proof. For heavy changer detection, we set $T = \epsilon\phi/2$. Denote the sum of key x in last and current epochs by $S^1(x)$ and $S^2(x)$, respectively. Without losing generality, we assume $S^2(x) \geq S^1(x)$ and hence $D(x) = S^2(x) - S^1(x)$.

For key x with $D(x) \leq (1-\epsilon)\phi$, by Lemma 3 and Lemma 4, the estimated change $D_{i,j}(x) < [S^2(x) + T] - [S^1(x) - T] = D(x) + 2T \leq (1-\epsilon)\phi + \epsilon\phi = \phi$, implying that x is not reported as a heavy changer.

For key x with $D(x) < (1-\epsilon/2)\phi$, there always exits an integer $k \geq 0$ such that $D(x) < (1-\epsilon)\phi + \frac{\epsilon\phi}{k+2}$. If $\sum_{y \neq x, f_i(y)=j} S^1(y) < \frac{(k+1)^2}{k+2}T$ in the first sketch and $\sum_{y \neq x, f_i(y)=j} S^2(y) < (k+1)T$ in the second one, by Lemma 3 and Lemma 4, both $S^{up,2}(x) - S^2(x)$ and $S^1(x) - S^{low,1}(x)$ are at most $\frac{k+1}{k+2}T$. The estimated change $D_{i,j}(x) \leq D(x) + 2\frac{k+1}{k+2}T < (1-\epsilon)\phi + \frac{\epsilon\phi}{k+2} + \frac{k+1}{k+2}\epsilon\phi \leq \phi$, implying that x is not reported as a heavy changer. Thus, key x is reported as a heavy changer only if $\sum_{y \neq x, f_i(y)=j} S^1(y) \geq \frac{(k+1)^2}{k+2}T$ or $\sum_{y \neq x, f_i(y)=j} S^2(y) \geq (k+1)T$. The probability of at least one of them occurs is at most $Pr\{\sum_{y \neq x, f_i(y)=j} S^1(y) \geq \frac{(k+1)^2}{k+2}T\} + Pr\{\sum_{y \neq x, f_i(y)=j} S^2(y) \geq (k+1)T\} \leq \frac{2(k+2)U}{w(k+1)^2\epsilon\phi} + \frac{2U}{w(k+1)\epsilon\phi} \leq$

$\frac{6U}{w\epsilon\phi}$. Since the r hash functions are independent, the probability that x is reported as a heavy changer is at most $(\frac{6U}{w\epsilon\phi})^r$. □

Parameter Selection. We express our results in terms of H, where $H = \frac{U}{\phi}$ for heavy hitter detection and $H = \frac{2U}{\phi}$ for heavy changer detection. For heavy hitter detection, LD-Sketch selects ~~$w = 2H$~~, $w = \frac{2H}{\epsilon}, r = \log\frac{1}{\delta}$ For heavy changer detection, LD-Sketch selects ~~$w = \frac{2H}{\epsilon}$~~ $w = \frac{6H}{\epsilon}$ and $r = \log\frac{1}{\delta}$.

Theorem 1. *Consider an LD-Sketch with* ~~$w = 2H$~~ $w = \frac{2H}{\epsilon}$, $r = \log\frac{1}{\delta}$ *and* $T = \epsilon\phi$. *It reports all heavy hitters. A non-heavy hitter with sum less than* $(1-\epsilon)\phi$ *is never be reported. A non-heavy hitter with sum between* $(1-\epsilon)\phi$ *and* $(1-\epsilon/2)\phi$ *is reported with probability at most* δ. *The expected space is* ~~$O(H\log\frac{1}{\delta})$~~$O(\frac{H}{\epsilon}\log\frac{1}{\delta})$. *The expected time complexity of updating a data item is* $O(\log\frac{1}{\delta})$, *and that of detection is* ~~$O(H\log\frac{1}{\delta})$~~$O(\frac{H}{\epsilon}\log\frac{1}{\delta})$

Theorem 2. *Consider two LD-Sketches with* ~~$w = \frac{2H}{\epsilon}$~~ $w = \frac{6H}{\epsilon}$, $r = \log\frac{1}{\delta}$ *and* $T = \epsilon\phi/2$. *They report all heavy changers. A non-heavy changer with difference less than* $(1-\epsilon)\phi$ *never reported. A non-heavy changer with difference between* $(1-\epsilon)\phi$ *and* $(1-\epsilon/2)\phi$ *is reported with probability at most* δ. *The expected space complexity is* $O(\frac{H}{\epsilon}\log\frac{1}{\delta})$. *The expected time complexity of updating a data item is* $\log\frac{1}{\delta}$, *and that of detection is* $O(\frac{H}{\epsilon}\log\frac{1}{\delta})$.

For distributed detection, we replace U and ϕ with U/q and $(1-\gamma)\phi$, respectively. Thus, Theorem 3 and 4 are as follows.

Theorem 3. *Consider an LD-Sketch with* ~~$w = \frac{2dH}{(1-\gamma)q}$~~ $w = \frac{2dH}{(1-\gamma)q\epsilon}$, $r = \log\frac{1}{d\delta}$, *and* ~~$T = \frac{(1-\gamma)\epsilon\phi}{d}$~~ $T = \frac{(1-\gamma)\epsilon\phi}{2d}$. *A heavy hitter is missed with probability at most* $1-(1-e^{-\frac{\phi}{2d}\gamma^2})^d$~~, while a~~. *A non-heavy hitter with the true sum less than* ~~$(1-\gamma)\phi$~~$(1-\gamma)(1-\epsilon)\phi$ *is never reported. A non-heavy hitter with sum between* $(1-\gamma)(1-\epsilon)\phi$ *and* $(1-\gamma)(1-\epsilon/2)\phi$ *is reported with probability at most* δ. *The expected space complexity is* ~~$O(\frac{dH}{q(1-\gamma)}\log\frac{1}{d\delta})$~~ $O(\frac{dH}{q\epsilon(1-\gamma)}\log\frac{1}{d\delta})$. *The expected time complexity of updating a data item is* $O(\log\frac{1}{d\delta})$, *and that of detection is* ~~$O(\frac{dH}{q(1-\gamma)}\log\frac{1}{d\delta})$~~ $O(\frac{dH}{q\epsilon(1-\gamma)}\log\frac{1}{d\delta})$.

Theorem 4. *Consider two LD-Sketches with* ~~$w = \frac{2dH}{(1-\gamma)q\epsilon}$~~ $w = \frac{6dH}{(1-\gamma)q\epsilon}$, $r = \log\frac{1}{d\delta}$, *and* ~~$T = \frac{(1-\gamma)\epsilon\phi}{d}$~~ $T = \frac{(1-\gamma)\epsilon\phi}{2d}$. *A heavy changer is missed with probability at most* $1-(1-e^{-\frac{\phi}{2d}\gamma^2})^d$~~, while a~~. *A non-heavy changer with the true difference less than* ~~$(1-\gamma)\phi$~~$(1-\gamma)(1-\epsilon)\phi$ *is never reported. A non-heavy changer with difference between* $(1-\gamma)(1-\epsilon)\phi$ *and* $(1-\gamma)(1-\epsilon/2)\phi$ *is reported with probability at most* δ. *The expected space complexity is* $O(\frac{dH}{q(1-\gamma)\epsilon}\log\frac{1}{d\delta})$. *The expected time complexity of updating a data item is* $\log\frac{1}{d\delta}$, *and that of detection is* $O(\frac{dH}{q(1-\gamma)\epsilon}\log\frac{1}{d\delta})$.

2 Time Complexity

Readers may think that we need a priority queue to get the minimum counter in associative array in Line 10 Algorithm 1. It requires extra $\log\frac{H}{\epsilon}$ operations for each data item to maintain the priority queue. However, we don't need to maintain the priority array. Instead, we go through the entire associative array to find out the minimum counter. The resulting amortized overhead is insignificant for two reasons. First, we do not always go through the associative array. Note that for each data item, we may update the counter directly (Lines 3-6), or expand the array (Lines 22-24), or decrement keys (Lines 10-20). Directly updating and expanding requires $O(1)$ operations. We go through the entire associative array only when we decrement keys.

Second, the length of the associative arrays is small so that the complexity of going through the entire array can be regarded as a constant if we select parameters appropriately. We have analyzed that the average

length of associative array is small in Lemma 8. Lemma 9 discusses the overall complexity of LD-Sketch. We also verify that the length of associative array is constant in our experiments.