Processing Top-k Queries from Samples

Edith Cohen
AT&T Labs–Research
180 Park Avenue
Florham Park, NJ 07932, USA
edith@research.att.com

Nadav Grossaug Haim Kaplan
School of Computer Science
Tel Aviv University
Tel Aviv, Israel
{nadavg,haimk}@cs.tau.ac.il

ABSTRACT

Top-k queries are desired aggregation operations on data sets. Examples of queries on network data include the top 100 source AS's, top 100 ports, or top Domain names over IP packets or over IP flow records. Since the complete dataset is often not available or not feasible to examine, we are interested in processing top-k queries from samples.

If all records can be processed, the top-k items can be obtained by counting the frequency of each item. Even when the full dataset is observed, however, resources are often insufficient for such counting and techniques were developed to overcome this issue. When we can observe only a random sample of the records, an orthogonal complication arises: The top frequencies in the sample are biased estimates of the actual top-k frequencies. This bias depends on the distribution and must be accounted for when seeking the actual value.

We address this by designing and evaluating several schemes that derive rigorous confidence bounds for top-k estimates. Simulations on various data sets that include IP flows data, show that schemes that exploit more of the structure of the sample distribution produce much tighter confidence intervals with an order of magnitude fewer samples than simpler schemes that utilize only the sampled top-k frequencies. The simpler schemes, however, are more efficient in terms of computation.

Our work is basic and is widely applicable to all applications that process top-k and heavy hitters queries over a random sample of the actual records.

1. INTRODUCTION

Top-k computations are an important data processing tool and constitute a basic aggregation query. In many applications, it is not feasible to examine the whole dataset and therefore approximate query processing is performed using a random sample of the records [4, 8, 14, 20, 15, 2]. These applications arise when the dataset is massive or highly distributed [13] such as the case with IP packet traffic that is both distributed and sampled and with Netflow records that are aggregated over sampled packet traces and collected distributively. Other applications arise when the value of the attribute we aggregate over is not readily available and determining it for a given record has associated (computational or other) cost. For example, when we aggregate over the domain name that corresponds to a source or destination IP address, the domain name is obtained via a reverse DNS lookups which we may want to perform on only a sample of the records.

A top-k query over some attribute is to determine the k most common values for this attribute and their frequencies (number of occurrences) over a set of records. Examples of such queries are to determine the top-100 Autonomous Systems destinations, the top-100 applications (web, p2p, other protocols), 10 most popular Web sites, or 20 most common domain names. These queries can be posed in terms of number of IP packets (each packet is considered a record), number of distinct IP flows (each distinct flow is considered a record), or other unit of interest. We are interested in processing top-k queries from a sample of the records. For example, from a sampled packet streams or from a sample of the set of distinct flows. We seek probabilistic or approximate answers that are provided with confidence intervals.

Top-k queries can be contrasted with *proportion* queries. A proportion query is to determine the frequency of a *specified* attribute value over records in a dataset. Examples of proportion queries are to estimate the fraction of IP packets or IP flows that belong to p2p applications, originate from a specific AS, or from a specific Web site.

Processing an approximate proportion query from a random sample is a basic and very well understood statistical problem. The fraction of sampled records with the given attribute value is an unbiased estimator, and confidence intervals are obtained using standard methods.

Processing Top-k queries from samples is more challenging. When the complete data set is observed, we can compute the frequency of each value and take the top-k most frequent values. When we have a random sample of the records, the natural estimator is the result of performing the same action on the sample. That is, obtaining the k most frequent values in the *sample* and proportionally scaling them to estimate the top-k frequency. This estimator, however, is biased upwards: The expectation of the combined frequency of the top-k items in the sample is generally larger than the value of this frequency over the unsampled records. This is a consequence of the basic statistical property that the expectation of the maximum of a set of random variables is generally greater (is at least as large) as the maximum of their expectations. While this bias must be accounted for when deriving confidence intervals and when evaluating the relation between the sampled and the actual top-k sets, it is not easy to capture as it depends on the fine structure of the full distribution of frequencies in the unsampled dataset, which is not available to us.

Overview of contributions

In Sections 3- 7 we devise and evaluate three basic methods to derive confidence intervals for top-k estimates.

- **"Naive" bounds** Let f be the sampled weight of the sample top-k frequencies. We consider the distributions with smallest top-k frequencies that are at least δ likely to have a sample distribution with top-k weight of at least f. We use this frequency to obtain the lower end of our confidence interval. The confidence interval constructed can be viewed as a combination of the maximum possible bias of our top-k estimator on a distribution with the same top-k weight with standard proportion error bounds. The definition of the Naive bound requires us to consider all distributions, which is not computationally feasible. To calculate these bounds, we identify a restricted set of distributions such that it is sufficient to consider these distributions. We are then able to construct a pre-computed table that provides the bound according to the desired confidence level and the value f.

- **CUB bounds** We use the sample distribution to construct a cumulative upper bound (CUB) for the top-i weight for all $i \geq 1$. We then use the CUB to restrict the set of distributions that must be taken into account in the lower bound construction. Therefore, we can potentially obtain much tighter bounds than in the Naive approach. The CUB method, however, is computationally intensive, since we can not use precomputed values.

- **Validation and Cross-validation bounds** We borrow terminology from hypothesis testing. The sample is split into two parts, one is the "learning" part and the other a "testing" part. The sampled top-k set is obtained from the learning part. We then look at the sampled weight of that set in the testing sample to obtain a "lower end" for our confidence interval. We also consider "validation estimators," that are biased to be lower than the top-k weight. These estimators offer an alternative to the positively biased estimator that corresponds to the top-k frequencies in the sample.

We evaluate these methods on a collection of datasets that include IP traffic flow records collected from a large ISP and Web request data. We show (precise characterization is provided in the sequel) that in a sense, the hardest distributions, those with the worst confidence bounds for a given sampled top-k weight, are those where there are many large items that are close in size. Real-life distributions, however, are more Zipf-like and therefore the cross-validation and CUB approaches can significantly outperform the naive bounds. The naive bounds, however, require the least amount of computation.

Our methodology and sampling schemes are applicable to the problem of computing frequent items and heavy hitters from samples. Iceberg queries [12], frequent items, and heavy hitters, are to find items that their frequency is above a certain threshold. When using the sample to find heavy hitters, the likelihood of false positives depends on the underlying distribution. Our approach can be used to derive tight confidence intervals.

Relation to previous work

Most previous work addressed applications where the complete dataset can be observed [19, 7, 5, 18, 16] but resources are not sufficient to compute the exact frequency of each item. The challenge in this case is to find approximate most frequent items using limited storage or limited communication. Examples of such settings are a data stream, data that is distributed on multiple servers, distributed data streams [1], or data that resides on external memory. We address applications where the complete dataset can not be observed or that it is easier to obtain random samples than to observe the complete dataset. The challenge is to estimate actual top frequencies from the available sample frequencies. These two settings are orthogonal. Our techniques and insights can be extended to a combined setting where the application observes a sample of the actual data and the available storage and communication do not allow us to obtain exact sample frequencies. We therefore need to first estimate sample frequencies from the observed sample, and then use these estimates to obtain estimates of the actual frequencies in the original dataset.

A related problem to top-k and heavy hitters computation is to estimate the *size distribution* [17, 18] (estimate the number of items of a certain size, for all sizes). This is a more general problem than top-k and heavy hitters queries and sampling can be very inaccurate for estimating the complete size distribution [8] or the number of distinct items [4]. Clearly, sampling is too lossy for estimating the number of items with frequencies that are well under the sampling rate and techniques that are able to observe the complete dataset are generally much more effective. For estimating top-k or heavy hitters, being able to observe the full data set is helpful [5], but we can obtain good accuracy from samples. The problem of finding top flows from sampled packet traffic was considered in [2], where empirical data was used to evaluate the number of samples required until the top-k set in the sample closely matches the top-k set in the actual distribution. Their work did not include methods to obtain confidence intervals. The performance metrics used in [2] are rank-based rather than weight based. That is, the approximation quality is measured by the difference between the actual rank of a flow (i.e, 3rd largest in size) to its rank in the sampled trace (i.e., 10th largest in side), whereas our metrics are based on the weight (size of each flow). That is, if two flows are of very similar size our metric does not penalize for not ranking them properly with respect to each other as two flows that have different weights. As a results, the conclusion in [5], that a fairly high sampling rate is required may not be applicable under weight-based metrics.

We are not aware of other work that focused on deriving confidence intervals for top-k and heavy hitters estimates that are derived from sampled records. Related work applied maximum likelihood (through the EM Expectation Maximization algorithm) to estimate the size distribution from samples [8, 18]. Unlike our schemes, these approaches do not provide rigorous confidence intervals.

Some work on distributed top-k was motivated by information retrieval applications and assumed sorted accesses to distributed index list: Each remote server maintains its own top-k list and these lists can only be accessed in this order. Algorithms developed in this model included the well known Threshold Algorithm (TA) [10, 11] TPUT [3], and algorithms with probabilistic guarantees [21]. In this model, the cost is measured by the number of sorted accesses. These algorithms are suited for applications where sorted accesses are readily available and more so than random samples such as with search engines results.

2. PRELIMINARIES

Let I be a set of items with weights $w(i) \geq 0$ for $i \in I$. For $J \subset I$, denote $w(J) = \sum_{i \in J} w(i)$. We denote by $T_i(J)$ (top-i set) a set of the i heaviest items in J, and by $B_i(J)$ (bottom-i set) a set of the i lightest items in J. We also denote by $\overline{W}_i(J) = w(T_i(J))$ the weight of the top-i elements in J and by $\underline{W}_i(J) = w(B_i(J))$ the weight of the bottom-i elements in J.

We have access to weighted samples, where in each sample, the probability that an item is drawn is proportional to its weight. In the analysis and evaluation, we normalize the total weight of all items to 1, and use normalized weights for all items. This is done for convenience of presentation and without loss of generality.

The *sample weight* of an item j using a set of samples S is the fraction of times it is sampled in S. We denote the sample weight of item j by $w(S, j)$. We define the sample weight of a subset J of items as the sum of the sample weights of the items in J, and denote it by $w(S, J)$. The sampled top-i and bottom-i sets (the i items with most/fewest samples in S) and their sampled weights are denoted by $T_i(S, J)$, $B_i(S, J)$, $\overline{W}_i(S, J) = w(T_i(S, J))$, and $\underline{W}_i(S, J) = w(B_i(S, J))$, respectively.

2.1 Top-k problem definition

There are several variations of the approximate top-k problem. The most basic one is to estimate $\overline{W}_k(I)$, where I is the distribution from which we get samples. In this problem we are given a set S of random samples with replacements from I and a confidence parameter δ. We are interested in an algorithm that computes an interval $[\ell, u]$ such that $\ell \leq \overline{W}_k(I) \leq u$ with probability $1 - \delta$. We call this problem *approximate top-k weight*.

A possible variation is to compute a set T of k items, and a fraction ϵ, as small as possible, such that $w(T) \geq (1 - \epsilon)\overline{W}_k(I)$ with probability $1 - \delta$. If we are interested in absolute error rather than relative error then we require that $w(T) \geq \overline{W}_k(I) - \epsilon$ with probability $1 - \delta$. We call this problem *approximate top-k set*.

Note that in the *approximate top-k set* problem we do not explicitly require to obtain an estimate of $w(T)$. In case we can obtain such an estimate then we also obtain good bounds on $\overline{W}_k(I)$.

The relation between these two variants is interesting. It seems that approximating the top-k weight rather than finding an actual approximate subset is an easier problem (requires fewer samples). As we shall see, however, there are families of distributions on which it is much easier to obtain an approximate subset.

There are stronger versions of the approximate top-k weight problem and the approximate top-k set problem. Two natural ones are the following. We define here the "set" version of these problem. The definition of the "weight" version is analogous.

- *All-prefix approximate top-k set:* Compute an ordered set of k items such that with probability $1-\delta$ for any $i = 1, \ldots, k$, the first i items have weight that is approximately $\overline{W}_i(I)$. We can require either a small relative error or a small absolute error.

- *Per-item approximate top-k set:* Compute an ordered set of k items such that with probability $1-\delta$ for any $i = 1, \ldots, k$, the ith item in the set has weight that approximately equals $(\overline{W}_i(I) - \overline{W}_{i-1}(I))$ (the weight of the ith heaviest item in I). Here too we can require either a small relative error or a small absolute error.

Satisfying the stronger definitions can require substantially more samples while the weaker definitions suffice for many applications. It is therefore important to distinguish the different versions of the problem. We provide algorithms and results for obtaining an approximate top-k weight, some of our techniques also extend to other variants.

2.2 Confidence bounds

We recall that for the approximate top-k weight problem we require that the interval $[\ell, u]$ produced by the algorithm would contain the weight of $T_k(I)$ with probability $1 - \delta$. That is if we run our algorithm many times then it would be "correct" in at least $1-\delta$ fraction of its runs. We also separately consider the two one-sided bounds on $\overline{W}_k(I)$. This holds for other versions of the problem as well when we estimate other parameters. In general we use the following standard statistical definitions.

We say that u is a $(1-\delta)$-*confidence upper bound* for a parameter ξ of a distribution I, if the value of ξ in I is not larger than u with probability $(1 - \delta)$. (This probability is over the draw of the random samples.) We define $(1 - \delta)$-*confidence lower bound* for ξ analogously. We say that $[\ell, u]$ is a $(1 - \delta)$-*confidence interval* for ξ, if the value of ξ is not larger than u and not smaller than ℓ with probability $(1 - \delta)$.

If $U(\delta_1)$ is a $(1 - \delta_1)$-confidence upper bound for a value and $L(\delta_2)$ is a $(1 - \delta_2)$-confidence lower bound for the same value, then $(U(\delta_1) + L(\delta_2))/2 \pm (U(\delta_1) - L(\delta_2))/2$ is a $(1 - \delta_1 - \delta_2)$-confidence interval for the value. We refer to $\pm(U(\delta_1) - L(\delta_2))/2$ as the *error bars* and to $(U(\delta_1) + L(\delta_2))/2$ as the *estimate*.

Bounds for proportions. Consider a sample of size s obtained for a proportion query, with $\hat{p}s$ positive samples. Let $U(h, s, \delta)$ be the largest value q such that a proportion q is at least δ likely to have at most h positive samples in a sample of size s. Then it is easy to see that $U(s\hat{p}, s, \delta)$ a $(1 - \delta)$-*confidence upper bound* on the proportion p.

Similarly, let $L(h, s, \delta)$ be the smallest value q such that a proportion q is at least δ likely to have at least h positive samples in a sample of size s. Then $L(s\hat{p}, s, \delta)$ is a $(1 - \delta)$-*confidence lower bound* on the proportion p.

Exact values of these bounds are defined by the Binomial distribution. Approximations can be obtained using Chernoff bounds, tables produced by simulations, or via the Poisson or Normal approximation. The Normal approximation applies when $ps \geq 5$ and $s(1-p) \geq 5$. The standard error is approximated by $\sqrt{\hat{p}(1 - \hat{p})/s}$.

Difference of two proportions. We use $(1 - \delta)$-confidence upper bounds for the *difference of two proportions*. Suppose we have n_1 samples from a Binomial distribution with mean p_1 and n_2 samples from a Binomial distribution with mean p_2. Denote the respective sample means by $\hat{p}_1$ and $\hat{p}_2$. Observe that the expectation of $\hat{p_1} - \hat{p_2}$ is $p_1 - p_2$.

We use the notation $C(\hat{p_1}, n_1, \hat{p_2}, n_2, \delta)$ for the $(1-\delta)$-confidence upper bound on $p_1 - p_2$.

We can apply bounds for proportions to bound the difference: It is easy to see that $U(n_1\hat{p}_1, n_1, \delta/2) - L(n_2\hat{p}_2, n_2, \delta/2)$ is a $(1 - \delta)$-confidence upper bound on the difference $p_1 - p_2$. This bound, however, is not tight. The prevailing statistical method is to use the Normal Approximation (that is based on the fact that if the two random variables are approximate Gaussians, so is their difference). The Normal approximation is applicable if $p_1 n_1, (1-p_1)n_1$, $p_2 n_2$ and $(1 - p_2)n_2 > 5$. The approximate standard error on the difference estimate $\hat{p_1} - \hat{p_2}$ is $\sqrt{\hat{p}_1(1 - \hat{p}_1)/n_1 + \hat{p}_2(1 - \hat{p}_2)/n_2}$.

2.3 Cumulative confidence bounds

Consider an (arbitrary) distribution on $[0, 1]$ with cumulative distribution function $F()$. That is, for all $0 \leq x \leq 1$, the probability of drawing a value that is at most x is $F(x)$.

For a random sample S of points from the distribution. Let $\hat{F}(x)$ be the fraction of the points in S which are smaller than x. We would like to obtain a (simultaneous) $(1 - \delta)$-confidence upper bounds for $F(b)$ for all $b \geq 0$. Observe that this is a generalization of proportion estimation: Proportion estimation is equivalent to estimating or obtaining an upper bound on a single point $p = F(a)$

without estimating $F(b)$ for all $b > 0$.

The cumulative bounds we consider are derived with respect to a certain $0 \leq a \leq 1$. We obtain a $(1-\delta)$-confidence multiplicative error bound for $F(b)$ for all $b \geq a$.

We define the random variable $\epsilon(a, S)$ to be $\max_{x>a} \frac{F(x)-\hat{F}(x)}{F(x)}$. Let $R(p, s, \delta)$ be the smallest fraction such that for every distribution $F()$ and a such that $F(a) \geq p$, and a random sample S of size s drawn from F, we have that $\epsilon(a, S) \leq R(p, s, \delta)$ with probability $1-\delta$ (over the choice of S).

We define the cumulative $(1-\delta)$-confidence upper bound on $F(b)$ for all $b \geq a$ as follows. Let $\hat{p} = \hat{F}(a)$. We look for the largest q such that $q(1 - R(q, s, \delta)) \leq \hat{p}$. The cumulative upper bound is $\frac{\hat{F}(x)}{1-R(q,s,\delta)}$ for every $x \geq a$.

We also consider cumulative bounds that are multiplicative on $b \geq a$ *and* additive on $b < a$. We refer to these bounds as *cumulative+ bounds.* Define

$$\epsilon^+(a, S) = \max\left\{\epsilon(a, S), \max_{x<a} \frac{F(x) - \hat{F}(x)}{F(a)}\right\}.$$

Let $R^+(p, s, \delta)$ be the smallest fraction such that for every distribution $F()$ and a such that $F(a) \geq p$, and a random sample S of size s drawn from F, we have that $\epsilon^+(a, S) \leq R(p, s, \delta)$ with probability $1-\delta$ (over the choice of S). The cumulative+ $(1-\delta)$-confidence upper bound on $F(b)$ for all $b \geq 0$ is defined as follows. Let $\hat{p} = \hat{F}(a)$. We look for the largest q such that $q(1-R(q, s, \delta)) \leq \hat{p}$. The cumulative+ upper bound is $\frac{\hat{F}(x)}{1-R(q,s,\delta)}$ for every $x \geq a$ and $\hat{F}(x) + qR(q, s, \delta)$ for every $x \leq a$.

It is known that $R(p, s, \delta)$ and $R^+(p, s, \delta)$ are not much larger than the relative error in estimating a proportion p using s draws with confidence $1-\delta$. Furthermore they have the same asymptotic behavior as proportion estimates as s grows [6]. Simulations show that we need about 25% more samples for the cumulative upper bound to be as tight as an upper bound on a proportion $F(a)$.

2.4 Data Sets

We use 4 data sets of IP flows collected on a large ISP network in a 10 minute interval during October, 2005. We looked at aggregations according to IP source address (366K distinct values), IP destination address (517K distinct values), source port (55K distinct values), and destination port (57k distinct values). We also use three additional Web traffic datasets. *WorldCup* World Cup 98 May 1 Web server logs with 4021 distinct items. *Dec-64*: Web proxy traces that were taken at Digital Equipment Corporation on September 16, 1996, 497597 items. *Lbl-100*: 30 days of all wide-area TCP connections between the Lawrence Berkeley Laboratory (LBL) and the rest of the world, 13783 distinct items. Figure 1 shows the top-k weights for these distributions that show an obvious Zipf-like form.

3. BASIC BOUNDS FOR TOP-K SAMPLING

When estimating a proportion, we use the fraction of positive examples in the sample as our estimator. We then determine a confidence interval for this estimate. Using the notation we introduced earlier, we can use the interval from $L(\hat{p}s, s, \delta)$ to $U(\hat{p}s, s, \delta)$ as a 2δ confidence interval. It is also well understood how to obtain the number of samples needed for proportion estimation within some confidence and error bounds when the proportion is at least p.

When estimating the top-k weight from samples, we would like to derive confidence intervals and also to determine the size of a fixed sample needed to answer a top-k query when the size of the top-k set is at least p.

The natural top-k candidate is the set of k most sampled items. The natural estimator for the weight of the top-k set is the sampled weight of the sampled top-k items. This estimator, however, is inherently biased. The expectation of the sampled weight of the sample top-k is always at least as large and generally is larger than the size of the top-k set. The bias depends on the number of samples and vanishes as the number of samples grows. It also depends on the distribution. To design estimation procedures or to obtain confidence intervals for a top-k estimate we have to account for both the standard error, as in proportion estimation, and for the bias.

3.1 Top-k versus proportion estimation

We show that top-1 estimation is at least as hard as estimating a proportion. Intuitively, we expect this to be the case since we do not need to only estimate the size of a particular set but also to bound away the size of all items.

LEMMA 3.1. *Let A be an algorithm that approximates the top-1 weight in a distribution with confidence $1-\delta$. We can use A to derive an algorithm A' for a proportion estimation query. The accuracy of A' in estimating a proportion p is no worse than the accuracy of A on a distribution with top-1 weight equal to p.*

PROOF. An input to A' is a set S' of s coin flips of a coin with bias p. Algorithm A' translates S' to a sample S from a distribution D in which we have one item b of weight p and every other item has negligible small weight. We generate S by replacing each positive sample in S' by a draw of b and every negative example by a draw of a different element (a unique element per each negative example). Algorithm A' applies A to S and returns the result. □

It is also not hard to see that the top-k problem is at least as hard as the top-1 problem (or as the top-i problem for $i < k$.). This is obvious for the stronger (per item) versions of the top-k problem but also holds for the top-k weight and the top-k set problems. To see this, consider a stream of samples for a top-1 problem. Label the jth sample of item i by the label $(i, U[0, \ldots, k-1])$ (where $U[\ldots]$ is a Uniform random selection). This is equivalent to drawing from a distribution where each item is partitioned to k same-size parts. The top-k weight in this distribution is the same as the top-1 weight in the original distribution.

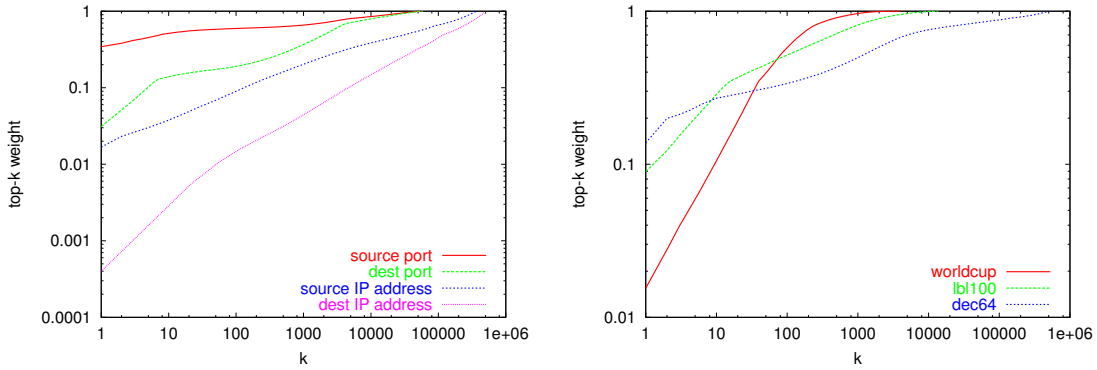
Note that the reduction from top-1 to proportion is not applicable to the version of the top-1 problem where we only want the set, without an approximation of the weight itself.

4. THE NAIVE CONFIDENCE INTERVAL

Suppose that we sampled s times and observed that the sampled weight of the sampled top-k set is $\hat{f}$. For a given s, $\hat{f}$, k, and δ, we define $L_k(\hat{f}s, s, \delta)$ to be the smallest f' such that there exists a distribution with top-k weight that is at most f' such that using s samples, the sampled weight of the sampled top-k set is at least δ likely to be at least f. We similarly define $U_k(\hat{f}s, s, \delta)$ to be the largest f' such that there is a distribution with top-k weight that is at least f' such that using s samples, the sampled weight of the sampled top-k set is at least δ likely to be at most f. It follows from the definitions that $U_k(\hat{f}s, s, \delta)$ (respectively, $L_k(\hat{f}s, s, \delta)$) is a $(1-\delta)$-confidence upper (respectively, lower) bound on the top-k weight.

These definitions do not provide a way to computationally obtain these bounds, since they require us to consider all possible distributions of items weights.

We first consider the upper bound and show that the proportion $(1-\delta)$-confidence upper bound can be used as an upper bound on the top-k weight:



IP flows Web datasets

Figure 1: top-k weights for test distributions

LEMMA 4.1. $U_k(\hat{f}s, s, \delta) \leq U(\hat{f}s, s, \delta)$.

Lemma 4.1 is an immediate corollary of the following lemma and monotonicity of $U(\hat{f}s, s, \delta)$ (with respect to $\hat{f}$).

LEMMA 4.2. *The distribution function of the sampled weight of the sampled top-k dominates that of the sampled weight of the top-k set. That is, for all $\alpha > 0$,*

$$Prob\{\overline{W}_k(S, I) \geq \alpha\} \geq Prob\{w(S, T_k(I)) \geq \alpha\}\,.$$

In particular, $E(\overline{W}_k(S, I)) \geq \overline{W}_k(I)$ (the expectation of the sampled weight of the sampled top-k set is an upper bound on the actual top-k weight.)

PROOF. Observe that the sample weight of the sample top-k is at least the sample weight of the actual top-k set (assume top-k set is unique using arbitrary tie breaking). □

We next consider obtaining a lower bound on the top-k weight. The definition of $L_k(\hat{f}s, s, \delta)$ was with respect to all distributions. The following Lemma restricts the set of distributions that we have to consider. We can then compute $L_k(\hat{f}s, s, \delta)$ using simulations on the more restricted set of distributions.

Let I_1 and I_2 be two distributions. We say that I_1 *dominates* I_2 if for all $i \geq 1$, $\overline{W}_i(I_1) \geq \overline{W}_i(I_2)$.

The next Lemma shows that if I_1 dominates I_2 then the probability distribution function of the sampled weight of the sampled top-i for I_1 dominates that of I_2.

LEMMA 4.3. *If the weighted set I_1 dominates I_2 then for any $k \geq 1$, and number of samples $s \geq 1$, the distribution function of the sampled weight of the sampled top-k with I_1 dominates the distribution function for I_2: that is, for any t, the probability that the sampled top-k would have at least t samples with I_1 is at least as large as with I_2.*

PROOF. We prove the claim for two distributions I_1 and I_2 that are identical except for two items b_1 and b_2. In I_2 the items b_1 and b_2 have weights w_1 and w_2, respectively. In I_1 the items b_1 and b_2 have weights $w_1 + \Delta$ and $w_2 - \Delta$, respectively for some $\Delta \geq 0$. Clearly if the claim holds for I_1 and I_2 as above then it holds in general. This is true since given any two distributions I_1 and I_2 such that I_1 dominates I_2 we can find a sequence of distributions $I_2 = I^0, I^1, \ldots, I^\ell = I_1$ where for every $0 \leq j < \ell$, I_2^{j+1} is obtained from I_2^j by shifting Δ weight from a smaller item to a larger one.

Consider a third distribution I_3 that is identical to I_1 and I_2 with respect to all items other than b_1 and b_2. The distribution I_3, similar to I_1, has an item b_1 with weight w_1, and it also has two items b_2 of weight $w_2 - \Delta$ and b_3 of weight Δ.

We sample s items from I_2 by sampling s items from I_3 and considering any sample of b_2 or b_3 as a sample of b_2. Similarly we sample s items from I_1 by sampling s items from I_3 and considering a sample from b_2 as a sample of b_2 and a sample of either b_1 or b_2 as a sample of b_1.

Suppose we sample a set S of s items from I_3 and map them as above to a sample S_1 of s items from I_1 and to a sample S_2 of s items from I_2. We show that for every k and t, $Prob\{\overline{W}_k(S_1, I_1) \geq t\}$ is not smaller than $Prob\{\overline{W}_k(S_2, I_2) \geq t\}$.

Fix the number of samples of each item different of b_1, b_2, and b_3, fix the number of samples of b_3 to be r, and fix the number of samples of b_1 and b_2 together to be m. Consider only samples S of I_3 that satisfy these conditions. We look at the probability space conditioned on these choices where the only freedom that we have left is to split the combined m draws of b_1 and b_2, between b_1 and b_2. We show that in this conditioned space for every k and t, $Prob\{\overline{W}_k(S_1, I_1) \geq t\}$ is not smaller than $Prob\{\overline{W}_k(S_2, I_2) \geq t\}$.

Over this conditioned probability space, for a fix $j \geq m/2$, consider the event A_j where the number of samples of b_1 in S is j and the number of samples of b_2 in S is $m - j$. Consider also the event A_{m-j} where the number of samples of b_1 is $m - j$ and the number of samples of b_2 is j. In A_j the the maximum among the weights of b_1 and b_2 in S_1 is $\max\{j + r, m - j\} = j + r$, and the maximum among the weights of b_1 and b_2 in S_2 is $\max\{j, m - j + r\}$ which is smaller than $j + r$. On the other hand, in A_{m-j} the maximum among the weights of b_1 and b_2 in S_1 is $\max\{m - j + r, j\}$, and the maximum among the weights of b_1 and b_2 in S_2 is $\max\{m - j, j + r\} = j + r$.

Consider the weight of the top-k set of S_2 in A_{m-j}, and the weight of the top-k set of S_1 in A_{m-j}. If both are at least t then they both are at least t in A_j, and both $Prob\{\overline{W}_k(S_1, I_1) \geq t\}$ and $Prob\{\overline{W}_k(S_2, I_2) \geq t\}$ equal 1. However it could be that in A_{m-j} the weight of the top-k set of S_2 is larger than t but the weight of the top-k set in S_1 is smaller than t. However if this is indeed the case in A_{m-j}, then in A_j the weight of the top-k set of S_1 is larger than t but the weight of the top-k set in S_2 is smaller than t.

Let $a = b_1/(b_1 + b_2 - \Delta)$. Since

$$Prob\{A_j\} = \left(\begin{array}{c} m \\ j \end{array} \right) a^j (1-a)^{m-j} \geq$$
$$\left(\begin{array}{c} m \\ m-j \end{array} \right) (1-a)^j (a)^{m-j} = Prob\{A_{m-j}\},$$

it follows that $Prob\{\overline{W}_k(S_1, I_1) \geq t\}$ is not smaller than $Prob\{\overline{W}_k(S_2, I_2) \geq t\}$. □

Lemma 4.3 identifies the family of "worst-case" distributions among all distributions that have top-k weight equal to f. That is, for any threshold t and for any i, one of the distributions in this family maximizes the probability that the sampled weight of the sampled top-i exceeds t. Therefore, to find $L_k(\hat{f}s, s, \delta)$, instead of all distributions, we can consider the more restricted set of most-dominant distributions.

The most-dominant distribution is determined once we fix both the weight f of the top-k, and the weight $0 < \ell \leq f/k$ of the kth largest item. The top-1 item in this distribution has weight $f-(k-1)\ell/k$, the next $k-1$ heaviest items have weight ℓ, next there are $\lfloor (1-f)/\ell \rfloor$ items of weight ℓ and then possibly another item of weight $1 - \ell\lfloor (1-f)/\ell \rfloor$. Example is provided in Figure 3. Fix the weight f of the top-k. Let G_ℓ be the most dominant distribution with value ℓ for the kth largest item. We can use simulations to determine the threshold value t_ℓ so that with confidence at most δ, the sampled weight of the sampled top-k in s samples from G_ℓ is at least t_ℓ. We associate f with the value $f_m = \max_\ell t_\ell$. Clearly f_m decreases with f. The value $L_k(\hat{f}s, s, \delta)$ is the largest f such that $f_m \leq \hat{f}$. This mapping from the observed value $\hat{f}$ to the lower bound f_m can be computed once and stored in a table, or can be produced on the fly as needed.

Note that for the top-1 problem, Lemma 4.3 provides us with a *single "worst-case" most-dominant distribution*: Since we only need to consider distributions where the "kth" (in this case, the top) item is f: the distribution has $\lfloor 1/f \rfloor$ items of weight f and possibly an additional item of weight $1 - f\lfloor 1/f \rfloor$.

The Naive confidence interval. We obtained our first method to derive a confidence interval for a top-k weight estimate. Suppose after s samples we observe that the sampled weight of the sampled top-k set is $\hat{f}$.

We use the estimate $(L_k(\hat{f}s, s, \delta/2) + U(\hat{f}s, s, \delta/2))/2$ with error bars of $\pm(U(\hat{f}s, s, \delta/2) - L_k(\hat{f}s, s, \delta/2))/2$. Since the two one-sided confidence intervals are not symmetric, we can reduce the combined width of the error bars by using a different confidence level for the upper and lower bounds: For $0 < \delta' < \delta$ we can use the estimate $(L_k(\hat{f}s, s, \delta') + U(\hat{f}s, s, \delta - \delta'))/2$ with error bars $\pm(U(\hat{f}s, s, \delta') - L_k(\hat{f}s, s, \delta - \delta'))/2$.

This estimate applies to the weight of the top-k set. We next consider the problem of bounding the (real) weight of the sampled top-k set:

LEMMA 4.4. *$L_1(\hat{f}s, s, \delta)$ is a $(1-\delta)$-confidence lower bound on the weight of the sampled top-1 item.*

PROOF. We first define $L'_k(\hat{f}s, s, \delta)$, the $(1-\delta)$-confidence lower bound on the actual weight of the sampled top-k set. It is defined as the minimum, over distributions I, of the minimum value ℓ, such that the probability is at least δ that the following combined property holds for the sampled top-k set:

- the sampled weight is at least $\hat{f}$, and
- the actual weight is at most ℓ.

It is easy to see that $L'_k(\hat{f}s, s, \delta) \leq L_k(\hat{f}s, s, \delta)$, since if we restrict the set of distributions considered when calculating L'_k to those with top-k weight that is at most ℓ, we obtain $L_k(\hat{f}s, s, \delta)$.

For $k = 1$, it is easy to see that equality holds, that is, $L'_1(\hat{f}s, s, \delta) = L_1(\hat{f}s, s, \delta)$. Consider a distribution with items of weight larger than ℓ. It is easy to see that removal of these items or replacing them with items of weight smaller than ℓ only increases the probability that the sampled top-k set has the combined property. □

For $k > 1$ we conjecture the following:

CONJECTURE 4.5. *$L_k(\hat{f}s, s, \delta)$ is a $(1-\delta)$-confidence lower bound on the weight of the sampled top-k set.*

To prove the conjecture we need to show that $L'_k(\hat{f}s, s, \delta) = L_k(\hat{f}s, s, \delta)$, that is, there is distribution that minimize ℓ that has top-k weight that is at most ℓ.

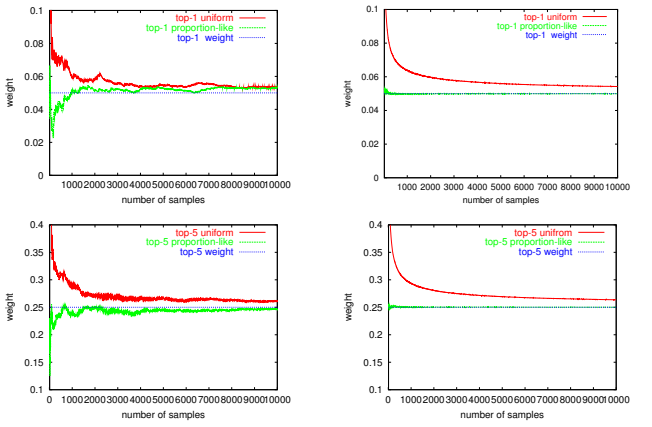
Our experimental observations support the conjecture in that the actual weight of the top-k weight lies inside the confidence interval.

4.1 Asymptotics of the Naive estimator

For a given distribution I, and given ϵ and δ, one can consider the smallest number of samples such that the sampled weight of the sampled top-1 item is in the interval $(1 \pm \epsilon)\overline{W}_1(I)$ with confidence $1 - \delta$. When we take the maximum of this number of samples over all distributions of top-1 weight f, we obtain the smallest number of samples that suffices to answer a top-1 query for a specified δ and ϵ, when the base distribution has top-1 weight at least f. The most dominant distribution with top-1 item of weight f has $1/f$ items of weight f. For this distribution, we need each of the $1/f$ items to be estimated to within $(1+\epsilon)$ with confidence $1 - f\delta$. Using multiplicative Chernoff bounds we obtain that the number of samples needed is $O(f^{-1}\epsilon^{-2}(\ln \delta^{-1} + \ln f^{-1}))$. This dependence is *super linear* in f^{-1}. This can be contrasted with the number of samples needed to estimate a proportion of value at least p, for a given ϵ, δ, and p. From Chernoff bounds we have $O(p^{-1}\epsilon^{-2} \ln \delta^{-1})$, which is *linear* in p^{-1}.

The naive bounds are derived under "worst-case" assumptions on the distribution, and therefore subjected to the $O(f^{-1}\epsilon^{-2}(\ln \delta^{-1} + \ln f^{-1}))$ dependence. A distribution where all items other than the top-1 are tiny behaves like a proportion and we obtain a good estimate of the top-1 weight after $O(f^{-1}\epsilon^{-2} \ln \delta^{-1})$ samples. Zipf-like distributions, that arise in natural settings, have asymptotic that is closer to proportion estimation when the distribution is more skewed.

This point is demonstrated in Figure 2. The figure shows sampling from a distribution with top-1 item that is of weight 0.05. It shows the sampled weight of the sampled top-1 item on a uniform distribution where there are 20 items of weight 0.05 each. It also shows the sampled weight of a sampled top-1 item in a distribution where there is a single item of weight 0.05 and other items have infinitesimally small weight. The averaging of the expected sampled weight of the sampled top-1 over 1000 runs illustrates the bias of the estimator on the two distributions. Evidently, the bias quickly vanishes on the second distribution but is significant for the first distribution. The naive confidence bound accounts for this maximum possible bias, so even on this simple distribution, after $10,000$ samples would only be able to guarantee a 5% error bars. The figure shows a similar situation when we measure the sampled weight of the top-5 items in a distribution with 5 items of weight 0.05 each and all other items infinitesimally small The convergence is similar to that of estimating a proportion of 0.25; When there are 20 items of weight 0.05, convergence is much slower and there is a significant bias.



A single set of samples average over 1000 sets of samples

Figure 2: Convergence of top-k estimator. The top figures are for top-1 item of weight 0.05. The bottom figures for top-5 items of weight 0.25. The curve "top-1 uniform" shows the sample weight of the sampled top-1 item in a uniform distribution. The curve "top-1 proportion-like" shows the sample weight of the sampled top-1 item from a distribution with a single item of weight 0.05 and all the rest infinitesimally small. The plots for top-5 are annotated similarly.

These arguments indicate that the Naive estimator provides us with a pessimistic lower bounds that also exhibit worse asymptotics than what we can hope to obtain for some natural distributions. We therefore devise and evaluate procedures to derive tighter lower bounds by exploiting more information on the distribution.

5. CUB BOUNDS

The derivation of CUB bounds resembles that of the Naive bound. As with the Naive bound, we look for the distribution with the smallest top-k weight that is at least δ likely to have sampled top-k weight that "matches our sample." The difference is that we not only look at the sampled top-k weight but use further statistics on the sample to further restrict the set of distributions we have to consider. By doing this, we are able to tighten the bound.

The bound is derived in two steps: (for $\delta' \leq \delta$)

1. *Cumulative upper bound (CUB) derivation:* We obtain $(1-\delta')$-confidence cumulative upper bounds on the weight of $\overline{W}_i(I)$ for all $i \geq 0$ (see Subsection 2.3). We obtain $R_1 \leq \cdots \leq R_k \leq R_{k+1} \leq \cdots$ such that for all $i \geq 1$, R_i is an upper bound on the top-i weight.

2. *Lower bound derivation:* We derive a $(1-(\delta-\delta'))$ confidence lower bound $L'_k(\{R_i\}, \hat{f}s, s, \delta-\delta')$ as follows. We consider all distributions that are consistent with the obtained CUB, that is, J such that $\overline{W}_i(J) \leq R_i$ for all $(i \geq 0)$. We look for the distribution J with smallest top-k weight $\overline{W}_k(J)$ that is at least $(\delta-\delta')$ likely to have a sampled top-k weight of at least $\overline{W}_k(S, I)$. The lower bound is then set to $\overline{W}_k(J)$.

Correctness is immediate. Consider a distribution. The probability that the cumulative upper bound obtained for it fails (even for one value) is at most δ'. If the distribution obeys the cumulative upper bound derived for it then the probability that the lower bound derived in the second step is incorrect is at most $(\delta-\delta')$. Therefore, for any distribution, the probability that it does not lie in its confidence interval is at most δ.

We derive a $(1-\delta)$ confidence lower bound $L_k(\{R_i\}, \hat{f}s, s, \delta)$ on the top-k weight as follows. (The Naive bound is $L_k(\hat{f}s, s, \delta) \equiv L_k(\{1, 1, 1, \ldots\}, \hat{f}s, s, \delta)$.) Similarly to the Naive bound, we restrict the set of distributions considered for the lower bound derivation by only considering the representative set of most dominant distributions. Applying Lemma 4.3 (similarly to its usage for the Naive bounds), we obtain that the most dominant distributions that conform to $\{R_i\}$ upper bounds is determined once we fix the top-k weight f and the weight $\ell \leq f/k$ of the kth heaviest item. For $i > k$, the weight of the ith item is as large as possible given that it is no larger than the $(i-1)$th item and that the sum of the top-i items is at most R_i. If R_i for $i < k$ are not restricted $(R_1 = R_2 = \ldots = R_{k=1} = 1)$, then the k-heaviest items are as in the naive bounds: the top-1 weight is $f-(k-1)\ell$ and the next $k-1$ heaviest items have weight ℓ. Otherwise, each of the first k items has weight at least ℓ, with as much weight as possible placed on earlier items. Formally, let $1 \leq j \leq k$ be the minimum such that $\sum_{h=1}^{j} R_h + (k-j)\ell \geq f$. The most dominant distribution is such that the top $j-1$ items have weights $R_1, \ldots, R_{j-1}$; the items $j+1, \ldots k$ have weight ℓ; and the jth item has weight $f - \sum_{h=1}^{j-1} R_j - (k-j)\ell$.

Figure 3 shows most dominant distributions for $k = 100$ with top-k weight equal to 0.4 that are constructed subject to CUB constraints R_i for $i \geq 100$ and $R_1 = \ldots = R_{99} = 1$. The dotted lines show the most dominant distributions without the CUB constraints. The figure helps visualize the benefit of CUB: The CUB constraints reduce the size and the number of larger non top-k items and by doing so reduce the bias of the top-k estimator (the sampled weight of the sample top-k).

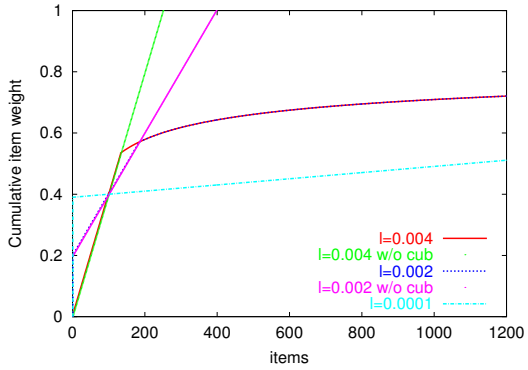


Figure 3: Most dominant distributions for $k = 100$ and top-k weight 0.4. These distributions are for $\ell = 0.004$ (Uniform), $\ell = 0.002$, and $\ell = 0.0001$. We also show the distributions subject to upper bounds R_i for $i \geq 100$. The distribution with $\ell = 0.0001$ is not affected by the CUB.

We use simulations on these most-dominant distributions to determine the probability that the sampled weight of the sampled top-k matches or exceeds the observed one.

Since there are many parameters in the upper bound ($\{R_i\}$ for $i \geq 1$), we can not use a precomputed table for the lower bound $L_k(\{R_i\}, \hat{f}s, s, \delta))$ like we could do for the naive bound $L_k(\hat{f}s, s, \delta)$. Therefore the CUB bounds are much more computationally intensive than the Naive bounds.

The confidence interval obtained applies to the weight $\overline{W}_k(I)$ of the top-k set. Using similar arguments to the naive derivation,

for $k = 1$, the confidence interval applies to the actual weight of the sampled top-1 set (see Lemma 4.4). We conjecture that it also applies to the actual weight of the set I_k when $k > 1$ (see Conjecture 4.5).

The CUB method is based on performing simulations based on statistics derived from the sample and in that it is related to statistical bootstrap method [9].

6. CROSS VALIDATION METHODS

We apply *validation* and *cross validation* frameworks and borrow terminology from hypothesis testing literature. In that context, the sample is split into *learning* and *testing* parts. A hypothesis is constructed using the learning subsample and its error rate is computed on the testing data. This is used to estimate the generalization error of a model learning procedure. In our analogous setting, we compute the sampled top-k set from the learning data, and estimate its weight using the testing data. Since the learning and testing parts are independent, the expectation of the sample weight of that set in the testing data is equal to its actual weight, which is at most the top-k weight. We then apply proportion bounds to obtain a ($(1-\delta)$-confidence) lower bound on the top-k weight.

LEMMA 6.1. *The distribution of the sampled weight of any* particular *size-k subset is dominated by the sample weight distribution of the top-k set.*

PROOF. Any fixed k-subset has weight at most that of the top-k. □

Note that just like the case of applying proportion upper bounds to the estimate that is biased upwards, the application of proportion lower bounds to validation estimators is pessimistic in that it is applied to a quantity that its expectation is below that of the top-k weight. Proportion bounds are calculated to be correct for unbiased quantities. Therefore, we expect the fraction of runs on which the estimate is incorrect to be lower than the corresponding δ value. In particular for smaller number of samples when the bias is larger.

We start with the plain *split-sample validation* for which we obtain error bars using proportion bounds. This method allows us to obtain a top-k candidate and obtain a lower bound on its actual weight. The statistics literature shows that extensions on this validation method, that are referred to as *cross validation* exhibit better performance in the hypothesis testing context. We obtain top-k estimators from analogous cross-validation methods: the *f-fold cross validation* and the *leave-m_ℓ-out cross validation*. These estimates allow us to derive lower bounds on the top-k weight. The expectation of these estimators is equal to the expectation of the actual weight of a sampled top-k set obtained in a sample of size equal to that of the learning sample. For a larger learning set, the expectation is higher and closer to the top-k weight, and therefore allows for tighter bound. On the other hand, the variance of the estimate depends on the size of the testing set and the cross validation method. We study these tradeoffs and the derivation of confidence intervals.

We also study the derivation of upper bounds on the difference between the weight of our set to that of the actual top-k set. That is, upper bound the potential increase in weight by exchanging items in our candidate set with items outside it. We apply a variant of the split-sample validation method to directly bound this difference.

6.1 Split-sample (hold out) Validation

We denote the learning sample by S_u and the testing sample S_ℓ and their respective sizes by m_u and m_ℓ. We have $m_u = m_\ell = s/2$. The sampled top-k set in the learning sample, $I_{k,u} = T_k(S_u, I)$, is our top-k candidate and its sampled weight $w(S_u, I_{k,u})$ in the learning sample is a sample from a quantity that upper bounds $\overline{W}_k(I)$ and hence used to derive an upper bound on the top-k weight. The sampled weight of $I_{k,u}$ in the testing sample is used to derive a lower bound. Since S_ℓ is independent of S_u, the expectation of the sampled weight of $I_{k,u}$ in S_ℓ is the actual weight of $I_{k,u}$. In fact, the distribution of $w(S_\ell, I_{k,u})$ is a Binomial random variable of sampling a proportion $w(I_{k,u})$ m_ℓ times. Since $w(I_{k,u}) \leq \overline{W}_k(I)$, the expectation of the estimator is a lower bound on $\overline{W}_k(I)$. When computing error bars, both $w(S_u, I_{k,u})$ and $w(S_\ell, I_{k,u})$ can be treated as proportion samples from proportions that are at least and at most the top-k weight, respectively.

For the upper bound we can use $U(\delta) = U(m_u w(S_u, I_{k,u}), m_u, \delta)$ or the generally tighter upper bound derived from the complete sample $U(\delta) = U(mw(S, I_k), m, \delta)$. For the lower bound we use $L(\delta) = L(m_\ell w(S_\ell, I_{k,u}), m_\ell, \delta))$.

We therefore have $(U(\delta/2) + L(\delta/2))/2$ as our top-k weight estimate. The error bars are $\pm(U(\delta/2) - L(\delta/2))/2$. Note that the estimate is valid not only for the top-k weight but also for the actual weight of the set $I_{k,u}$.

6.2 r-fold Cross Validation

In 2-fold ("double") cross validation the sample is again split into two equal parts S_u and S_ℓ. We compute the sampled top-k sets in both S_u and S_ℓ. Denote the two sets by $I_{k,u}$ and $I_{k,\ell}$. Denote by I_k the sampled top-k set in the full sample. For the lower bound we use $L(\delta) = L((m_\ell w(S_\ell, I_{k,u}) + m_u w(S_u, I_{k,\ell}))/2, s/2, \delta)$. We argue that this is a $(1-\delta)$-confidence lower bound on the top-k weight: The $s = m_u + m_\ell$ samples are taken from two different proportions, but both these proportions are at most the top-k weight. The expectation of this 2-fold estimate is the same as for the split sample estimator, but the motivation for introducing this refinement is that we reduce the variance by averaging over two sets.

We conjecture that this bound is also applicable to the weight of the set I_k:

CONJECTURE 6.2. *$L(\delta)$ is a $(1-\delta)$-confidence lower bound on the weight of the set I_k.*

This approach can be extended to r-fold cross validation where the sample is split into r equal parts. For each part, we compute the sampled top-k set on the learning set that contains the other $r-1$ parts and then compute its weight on the held-out part. We denote the r-fold cross validation estimate by X_r.

LEMMA 6.3. *For any r, $E(X_r) \leq \overline{W}_k(I)$.*

PROOF. For each part, we have s/r independent samples from a proportion that is the actual weight of some k-subset (therefore is at most the top-k weight). The proof follows from linearity of expectation. (Note that there is dependence between different parts.) □

As noted above, the expectation of X_r is equal to the expectation of the actual weight of a sampled top-k set obtained using $(1-1/r)s$ samples.

6.3 Leave-out Cross Validation.

Leave-m_ℓ-out cross validation is a "smoothed" version of r-fold cross validation.

Consider some fixed $k \leq m_u \leq m-1$. The estimator J_{m_u} is the average, over all subsets $S_u \subset S$ of size $|S_u| = m_u$, of the sampled weight in $S_\ell = S \setminus S_u$ of the sampled top-k subset in S_u. (When there are multiple items with kth largest number of samples we emulate uniform at random selection among them to determine

which ones are included in the sampled top-k set. This selection is also factored into the estimators by averaging over all selections.) We expect the leave-out estimators to perform better than the r-fold estimators since we expect the variance of J_{m_u} to be at most that of X_r with $r = s/m_\ell$.

LEMMA 6.4. *For all m_u, $E(J_{m_u}) \leq \overline{W}_k(I)$.*

PROOF. Consider a particular size-m_u subset of the sample specified by its positions in the sample. The sampled weight in $S_\ell = S \setminus S_u$ of the sampled top-k subset in S_u is equivalent to taking $|S_\ell|$ independent samples from a proportion equal to the weight of the sampled top-k set in S_u, which by definition is at most $\overline{W}_k(I)$. The proof follows by linearity of expectation. □

The expectation of the estimator J_{m_u} is equal to the expectation of the actual weight of the sampled top-k set in a sample of size m_u.

Computing leave-out estimators. As the leave-out estimators are defined over all possible subsets, direct computation can be prohibitive. The following Lemma provides us with a computationally easy way to obtain approximate values for the leave-out estimators. For a multiset S, integer k, and item id i, let $P(i, k, m, S)$ be the probability that i is in the top-k items in a random m-size subset of S. We account for "partial fit" in the definition of $P(i, k, m, S)$. Consider some subset of S of size m. If the count of i exceeds that of the kth most frequent item, the contribution is 1. If it is strictly lower than the frequency of the kth most frequent item in the subset then the contribution is 0. Otherwise, let b be the total number of items with frequency equal to that of the kth most frequent item, and let c be the number of such items in the top-k set. The contribution is then c/b. $P(i, k, m, S)$ is the average of these contributions over all possible m-subsets of S.

LEMMA 6.5. *Let i be the id of the ith most common item in S and let a_i be its number of occurrences. For any m_u,*

$$J_{m_u} = \sum_i a_i P(i, k, m_u, S \setminus \{i\}) .$$

To estimate J_{m_u} we use subsets of size $m_u + 1$ from S. From each sample, we can compute a contribution to $P(i, k, m_u, S \setminus \{i\})$ for all i by carefully accounting for the occurrences of item with index id i.

Leave-1-out. The leave-1-out and the s-fold estimators are the same. This estimator can be efficiently computed from the sample counts of items. Consider a sample and let $a_1 \geq a_2 \geq a_3 \cdots$ be the sampled counts of items. Let $t_{k+1} \geq 1$ be the number of items with frequency equal to a_{k+1}. Let n ($t_{k+1} - 1 \geq n \geq 0$) be the number of such items in the sampled top-k set. The estimate is

$$X_s \equiv J_{s-1} = \left(\frac{1}{s}\right)\left(\sum_{i|a_i-1>a_{k+1}} a_i + \left(\frac{n+1}{t_{k+1}+1}\right)\sum_{i|a_i-1=a_{k+1}} a_i\right) .$$

The first terms account for the contribution of items that definitely remain in the modified top-k set after "loosing" the leave-out sample. This includes all items that their count in the sample is larger than $a_{k+1} + 1$. The second term accounts for items that are "partially" in the top-k set after loosing the leave-out sample. By partially we mean that there are more items with that frequency than spots for them in the new top-k set. The hypothesis testing literature indicates that leave-1-out cross validation performs well but has the disadvantage of being computationally intensive. In our setting, the computation of the estimator is immediate from the sampled frequencies. This estimator has a maximal size learning set, of size $s - 1$, and therefore its expectation is closest to the top-k weight among all the cross validation estimators.

6.4 Bounding the variance.

The choice of the particular cross validation estimator, selecting r for the r-fold estimators or m_u for the leave-out estimators reflects the following tradeoffs. The expectation of these estimators is the expectation of the actual weight of the sampled top-k set in a sample of the size of the learning set. This expectation is non-decreasing with the number of samples and gets closer to $\overline{W}_k(I)$ with more samples in the learning set. (Moreover, the distribution of the sampled top-k weight with fewer samples dominates that taken with more samples). Therefore, it is beneficial to use larger learning sets. (larger r or smaller m_u). In the extreme, the leave-1-out estimator is the one that maximizes the expectation of the estimator. However, smaller size test sets and dependencies between learning sets can increase the variance of the estimator. The effect of that on the derived lower bound depends on both the actual variance and on how tightly we can bound this variance. In our evaluation, we consider both the empirical performance of these estimators and the rigorous confidence intervals we can derive for them.

As we did with the 2-fold estimator, we can apply proportion lower bounds to the cross validation estimators as follows: We can treat the estimate as a Binomial random variable with m_u (or s/r) independent samples. This computation is pessimistic from two reasons. The first is the application of a proportion bound to a biased quantity. The second reason is that the calculation assumes a binomial distribution with s/r independent trials, and therefore does not account for the benefit of the cross validation averaging over multiple test sets. These effect worsens for larger values of r. In the experimental evaluation, we consider both the empirical performance of the estimators (in terms of expectation and the average squared and absolute error), and the quality of the confidence intervals. For confidence intervals, we use two approaches to derive lower bounds: The first is the pessimistic rigorous approach. The second is a heuristic that "treats" the estimate as a binomial with s independent trials and applies a proportion $L(sX_r, s, \delta)$ lower bound. We refer to this heuristic as *r-fold with s* and carefully evaluate its empirical correctness.

6.5 Weight difference to the top-k weight

We next consider the goal of obtaining a $(1-\delta)$-confidence upper bound on the difference $\overline{W}_k(I) - w(I_{k,u})$ between the weight of our output set $I_{k,u}$ to that of the true top-k set.

A more refined question is "by how much can we possibly increase the weight of our set by exchanging items from $I_{k,u}$ with items that are in $I \setminus I_{k,u}$?" It is a different question than bounding the weight of the set. For example, in some cases we can say that "we are 95% certain that our set is the (exact) top-k set." which is something we can not conclude from confidence bounds on the weight.

We use the basic split-sample validation approach, where the top-k candidate set, $I_{k,u}$, is derived from the learning sample S_u. The testing sample S_ℓ is then used to bound the amount by which we can increase the weight of the set $I_{k,u}$ by exchanging a set of items from $I_{k,u}$ with a set of items of the same cardinality from $I \setminus I_{k,u}$.

Denote by $J_i = T_i(S_\ell, I \setminus I_{k,u})$ ($1 \leq i \leq k$) the sampled top-i items in $I \setminus I_{k,u}$ using samples S_ℓ. Denote by $H_j = B_j(S_\ell, I_{k,u})$

the sampled bottom-j items in $I_{k,u}$ using samples S_ℓ. Let $C_j \equiv C(w(S_\ell, J_j), m_\ell, w(S_\ell, H_j), m_\ell, \delta)$ (C_j is a $(1-\delta)$-confidence upper bound on the difference of two proportions (see Section 2) applied to $w(S_\ell, J_j)$ and $w(S_\ell, H_j)$ with sample size m_ℓ.)

LEMMA 6.6. *$\max_{1 \le j \le k} C_j$ is a $(1-\delta)$-confidence upper bound on the amount by which we can increase the weight of the set $I_{k,u}$ by exchanging items. (Hence, it is also a $(1-\delta)$-confidence upper bound on the difference $\overline{W}_k(I) - w(I_{k,u})$.)*

PROOF. The maximal amount by which we can increase the weight of $I_{k,u}$ by exchanging items is equal to

$$\max_{1 \le j \le k} \overline{W}_j(I \setminus I_{k,u}) - \underline{W}_j(I_{k,u}) \ .$$

It follows that if C_j is a $(1-\delta)$-confidence upper bound on the difference $\overline{W}_j(I \setminus I_{k,u}) - \underline{W}_j(I_{k,u})$, then $\max_{1 \le j \le k} C_j$ is a $(1-\delta)$-confidence upper bound on the maximum increase (and therefore on the difference $\overline{W}_k(I) - w(I_{k,u})$.)

It remains to show that C_j is a $(1-\delta)$-confidence upper bound on $\overline{W}_j(I \setminus I_{k,u}) - \underline{W}_j(I_{k,u})$. We use the samples S_ℓ to obtain upper bound on the weight of the top-i elements in $I \setminus I_{k,u}$ and lower bound on the weight of the bottom-i elements in $I_{k,u}$. By definition, $w(H_j) \ge \underline{W}_j(I_{k,u})$, and therefore $w(S_\ell, H_j) = \underline{W}_j(S_\ell, I_{k,u})$ is a sample from a proportion that is at least $\underline{W}_j(I_{k,u})$. Similarly, $w(J_i) \le \overline{W}_j(I \setminus I_{k,u})$, and therefore $w(S_\ell, J_i)$ is a sample from a proportion that is at most $\overline{W}_j(I \setminus I_{k,u})$. Therefore, C_j is also a $(1-\delta)$-confidence upper bound on the difference $\overline{W}_j(I \setminus I_{k,u}) - \underline{W}_j(I_{k,u})$. □

7. EVALUATION RESULTS

The algorithms were evaluated on all data sets, for top-100 and top-1, and $\delta = 0.1$ and $\delta = 0.01$. In the evaluation we consider the tightness of the estimates and confidence intervals. For the heuristic r-fold with s lower bounds we also consider correctness.

7.1 Quality of different estimators

We empirically evaluate the expectation, square error, and average absolute error of the (positively biased) sampled weight of the sample top-k items ("upper"), and the negatively-biased split-sample, 2-fold, 10-fold, and s-fold estimators. We also consider two combined estimators: the average of the upper and the s-fold estimators (s-fold+upper) and the average of the upper and the 2-fold estimators (2-fold+upper). The expectation of these estimators shows their bias, the square and absolute error reflect both the bias and the variance of these estimators. The results for four datasets are shown in Figure 4. The average square error and the average absolute error had close correspondence. We show the average absolute value of the relative error. The figures show that the bias decreases with r for the r-fold estimators. The split sample and the 2-fold estimators have the same expectation and therefore split sample averages are not shown. The absolute error and variance measures vary: 2-fold is always at least as good as split-sample and on some datasets have considerably smaller variance. In most cases, the s-fold and 10-fold estimators have smaller variance than the 2-fold estimator. The upper estimator is more often worse or comparable to the s-fold estimator. The combined estimators perform very well. In most cases they had the smallest error and bias.

7.2 Confidence intervals

We evaluate the tightness of confidence bounds obtained via rigorous methods by considering the average value of the bound over many runs. The upper and lower bounds provided are $(1-\delta)$-confidence bound. The five lower bound methods that are compared are the Naive bound, the CUB bound, the split-sample and 2-fold bounds (with $s/2$ proportion correction), and the 10-fold bound (with $s/10$ proportion correction). The split-sample bound has the same expectation as the 2-fold bound, and therefore it is not shown in the plots.

We precomputed, using multiple simulation runs, tables for the $(1-\delta)$-confidence bounds $U(s\hat{p}, s, \delta)$, $L(s\hat{p}, s, \delta)$ (for proportions), and $L_k(\hat{f}s, s, \delta)$ (for the Naive lower bound). The bound for the Naive lower bound was generated using a simulations on families of most dominant distributions. The proportion bounds were used to derive the upper bound, and the lower bound for the split-sample and for the 2-fold methods. The $L_k(\hat{f}s, s, \delta)$ tables were used for the Naive lower bound. The precomputation of these tables made the implementation of the Naive method very efficient. The implementation of the CUB method involved constructing and running simulations on families of most-dominant distributions on each run of the algorithms. For the CUB method, these families depend on the cumulative upper bounds obtained, and we could not use precomputed tables. As a result, the CUB method is considerably more computation intensive.

We evaluate two varieties of the CUB bounds. The first one (CUB) derives R_i only for $(i \ge k)$ by bounding only multiplicative error for $i \ge k$; with this method we use $R_1 = \ldots = R_{k-1} = 1$. The second one (CUB+) also bounds the additive error and obtains R_i for $1 \le i \le k-1$. For a given confidence level, the bounds R_i obtained by CUB+ are tighter for $(i < k)$ but weaker for $i \ge k$ than the bounds obtained by CUB. There is a difference between CUB and CUB+ only for $k > 1$.

The results for selected datasets and parameters (k and δ) are provided in Figure 5. The figures also show the top-k weight $\overline{W}_k(I)$, the sampled weight of the sampled top-k weight (that has expectation at least $\overline{W}_k(I)$ and gets closer to $\overline{W}_k(I)$ as the number of samples grows) the actual weight of the sampled top-k set (that has expectation at most $\overline{W}_k(I)$ and also gets closer to $\overline{W}_k(I)$ as the number of samples grows).

The Naive lower bound is almost always the lowest (least tight) bound and is outperformed by the CUB and 2-fold bounds. The 10-fold bound is sometimes below Naive, because of the pessimistic $s/10$ trials proportion adjustment. In some cases, the Naive bound was tighter than the 2-fold bound. This can happen on distributions that are closer to the "most dominant distributions" on which the Naive bound is tight and the 2-fold method, that utilizes half the samples, is not. On our datasets, we observed that Naive is tighter in distributions where the top-k weight is most of the total weight. The CUB bound was tighter than the 2-fold bound on more distributions, but there were also many distributions where the 2-fold bound was tighter. The CUB+ bounds were slightly tighter than the CUB bounds.

Observed error-rates for top-k weight. We considered the observed error rates of the $(1-\delta)$-confidence upper bounds and the $(1-\delta)$-confidence lower bounds obtained via rigorous methods (Naive, CUB, 2-fold with $s/2$ correction and 10-fold with $s/10$ correction). The observed error rate is the fraction of runs on which the lower bound was higher (or the upper bound was lower) than the top-k weight. Tables 1, 2, and 3 show the error rates for the upper bound and Naive and CUB lower bounds. The results are aggregated across different numbers of samples, for each dataset and k. The expectation of this error rate is lower than δ and on most instances (an instance is specified by the dataset, k, δ, method, and number of samples), the error rate was well below δ. This was the case since these bounds are pessimistic.

dataset,k	$\delta = 0.1$	$\delta = 0.01$
dec64 1	0.101	0.005
dec64 100	0.005	0
destport 1	0.084	0.002
destport 100	0	0
destIP 1	0	0
destIP 100	0	0
lbl100 1	0.11	0.012
lbl100 100	0.008	0
srcport 1	0.101	0.006
srcport 100	0.016	0
srcIP 1	0.077	0.002
srcIP 100	0	0
worldcup 1	0.05	0.001
worldcup 100	0.008	0

Table 1: Observed error rate of $(1 - \delta)$-confidence upper bound.

dataset,k	$\delta = 0.1$		$\delta = 0.01$	
	weight	set	weight	set
dec64 1	0.003	0.003	0	0
dec64 100	0	0	0	0
destport 1	0.001	0.002	0	0
destport 100	0	0	0	0
destIP 1	0	0	0	0
destIP 100	0	0.001	0	0
lbl100 1	0.003	0.003	0	0
lbl100 100	0	0	0	0
srcport 1	0.024	0.024	0	0
srcport 100	0	0	0	0
srcIP 1	0.001	0.001	0	0
srcIP 100	0	0	0	0
worldcup 1	0	0.004	0	0
worldcup 100	0	0	0	0

Table 2: Observed error rate of the $(1 - \delta)$-confidence Naive lower bound on top-k weight and top-k set.

dataset,k	$\delta = 0.1$		$\delta = 0.01$	
	weight	set	weight	set
dec64 1	0.018	0.018	0	0
dec64 100	0	0	0	0
destport 1	0.022	0.022	0.001	0.002
destport 100	0	0	0	0
destIP 1	0.005	0.089	0	0.033
destIP 100	0	0	0	0
lbl100 1	0.025	0.025	0.002	0.002
lbl100 100	0	0	0	0
srcport 1	0.041	0.041	0.005	0.005
srcport 100	0.001	0.017	0	0.001
srcIP 1	0.036	0.038	0.002	0.002
srcIP 100	0	0	0	0
worldcup 1	0.007	0.011	0.002	0.004
worldcup 100	0	0	0	0

Table 3: Observed error rate of the $(1 - \delta)$-confidence CUB lower bound on top-k weight and top-k set.

dataset,k	split-sample		2-fold	
	$\delta = 0.1$	$\delta = 0.01$	$\delta = 0.1$	$\delta = 0.01$
dec64 1	0.108	0.004	0.034	0
dec64 100	0	0	0.002	0
destport 1	0.079	0.003	0.029	0
destport 100	0	0	0.004	0
destIP 1	0.017	0.001	0.031	0
destIP 100	0	0	0.006	0
lbl100 1	0.107	0.003	0.034	0
lbl100 100	0.006	0	0.003	0
srcport 1	0.121	0.008	0.035	0
srcport 100	0.004	0	0.002	0
srcIP 1	0.091	0	0.037	0
srcIP 100	0	0	0.001	0
worldcup 1	0.064	0.001	0.041	0
worldcup 100	0.007	0	0.006	0

Table 4: Observed error rates of the $(1 - \delta)$-confidence split-sample and 2-fold lower bounds on top-k weight.

Observed error-rates for top-k set. We also considered the error rates of the $(1 - \delta)$-confidence lower bounds with respect to the "top-k set" metric, that is the fraction of runs on which the actual weight of the top-k set in the sample is below the respective lower bound. The actual weight of the top-k set in the sample is always below the top-k set. Therefore, the observed error rate should be higher than for the "top-k weight" metric. Tables 2 and 3 list the observed error rates for the Naive and CUB lower bounds. The results are aggregated across different numbers of samples, for each dataset and $k = 1, 100$. We observed that across all instances, the error rates were consistent with the respective lower bounds, that is, the error rate was below δ or otherwise close to δ within the applicable standard error. These observations support Conjecture 6.2 and its CUB variant and further suggest that a counterpart of this conjecture may also hold to the r-fold lower bounds. The conjecture, which we proved only for the Naive and CUB bounds for $k = 1$, states than the lower bounds on the top-k weight also apply for the top-k set.

Observed error-rates for split-sample and 2-fold. We compared the observed error rates for the top-k weight of the $(1 - \delta)$-confidence lower bounds obtained via the split-sample and the 2-fold methods. Recall that both estimators have the expectation (and therefore the same bias). We expected the 2-fold method to have lower variance and the observed error rates highly support this expectation: For $\delta = 0.1$, the average error rate over split-sample instances was 0.044 and was only 0.015 over 2-fold instances. For $\delta = 0.01$, the respective error rates were 0.0016 and $2.3e-05$. A more detailed summary is provided in Table 4 (error rates are aggregated across different numbers of samples for each dataset and k).

Heuristic cross validation bounds. We evaluated the observed error rates of the heuristic cross validation lower bounds r-fold with s. The observed error rates for s-fold with s are listed in Table 5 (aggregated across all numbers of samples, for each dataset and $k = 1, 100$). On the majority of instances, the error rate did not exceed the corresponding δ value. For the top-k set version, the bounds were often too loose. Since the heuristic lower bounds are tighter than with the rigorous methods, the results suggest that this might be a reasonable heuristic for top-k weight, but not for top-k set. The empirically good performance of the 10-fold and s-fold

dataset,k	$\delta = 0.1$		$\delta = 0.01$	
	weight	set	weight	set
dec64 1	0.097	0.097	0.002	0.002
dec64 100	0.006	0.139	0	0.012
destport 1	0.082	0.087	0.002	0.003
destport 100	0.001	0.115	0	0.009
destIP 1	0.069	0.147	0.004	0.037
destIP 100	0	0.156	0	0.028
lbl100 1	0.102	0.102	0.001	0.001
lbl100 100	0.02	0.135	0	0.006
src4600 1	0.117	0.117	0.008	0.008
src4600 100	0.009	0.099	0	0.002
srcIP 1	0.102	0.104	0.003	0.003
srcIP 100	0.004	0.149	0	0.009
worldcup 1	0.089	0.146	0.004	0.014
worldcup 100	0.028	0.157	0	0.013

Table 5: Observed error rates of the $(1-\delta)$-confidence s-fold with s heuristic lower bound on top-k weight and top-k set.

estimators suggests that there might be a a way to derive tighter rigorous bounds on their variance.

7.3 Bounding the difference to the top-k weight

We evaluated the method (Section 6.5) that directly bounds the difference between the weight of the empirical top-k set to the weight of the best alternative set of size k. We used the Normal approximation to bound the differences of proportions (see Section 2.2).

If we attempt to derive such bounds using methods that provide a confidence interval, we can use the width of the confidence interval, that is, the difference between the upper and the lower bounds. If we use $(1-\delta)$-confidence bounds for the upper and the lower bounds, the confidence level we can provide on the difference is $\delta + (1-\delta)\delta \approx 2\delta$.[1] Figure 6 shows the average width of this interval for the Naive bound, the CUB bound, and the 2-fold bound with $\delta = 0.2$ and $\delta = 0.02$. It also shows the bound that is derived using the direct method for confidence levels $\delta = 0.2$ and $\delta = 0.02$.

The direct bounds are not always tighter than the derived 2-fold, CUB, and Naive bounds, but on many instances are significantly tighter. The bounds obtained as the width of the confidence intervals are always positive whereas the direct method can sometimes provide a negative bound on the difference. The interpretation of a negative bound is that we are $(1-\delta)$-confident that replacing items from our set with the heaviest items that are not in our set will decrease the weight of the set by at least this amount. In particular, the direct method enables us to derive confidence interval for our set being the exact unique top-k set.

8. CONCLUSION AND FUTURE DIRECTIONS

We developed several rigorous methods to derive confidence intervals for approximate top-k weight and top-k set queries over a sample of the dataset. We also propose and evaluate different estimators. Our work provides basic statistical tools for applications that provide only sampled data. The methods we developed vary in the amount of computation required and in the tightness of the bounds. Generally, methods that are able to uncover and exploit more of the structure of the sample distribution provide tighter bounds, but can also be more computationally intensive.

We plan to extend our methodology to applications where the available storage is too limited to obtain the full sample distribution. An example is applications where the sampled records are distributed in many locations or occur as a data stream. For these applications, we need to asses what information to gather on the sample distribution and to derive estimators and confidence intervals that are based on partial information of the sample distribution. In addition, we would like to consider a sequential settings where the algorithm can adaptively increase the number of samples until it can answer a query with specified precision and confidence bounds.

[1]The validity of this depends on Conjecture 6.2 and its extension to CUB and 2-fold lower bounds. These conjectures, that empirically were correct on our datasets, state that the respective derived lower bound applies not only to the top-k weight but also to the actual weight of the sampled top-k set.

9. REFERENCES

[1] B. Babcock and C. Olsten. Distributed top-k monitoring. In *SIGMOD*. ACM, 2003.

[2] C. Barakat, G. Iannaccone, and C. Diot. Ranking flows from sampled traffic. In *Proceedings of the 2005 ACM conference on Emerging network experiment and technology (CoNext)*. ACM, 2005.

[3] P. Cao and Z. Wang. Efficient top-k query calculation in distributed networks. In *Proc. 23rd Annual ACM Symposium on Principles of Distributed Computing*. ACM-SIGMOD, 2004.

[4] M. Charikar, S. Chaudhuri, R. Motwani, and V. Narasayya. Towards estimation error guarantees for distinct values. In *Proceedings of ACM Principles of Database Systems*, 2000.

[5] M. Charikar, K. Chen, and M. Farach-Colton. Finding frequent items in data streams. In *Proceedings of the International Colloquium on Automata, Languages and Programming (ICALP)*, pages 693–703, 2002.

[6] E. Cohen, H. Kaplan, Y. Mansour, and M. Sharir. Tail estimates for dependent increasing sums. Manuscript, 2006.

[7] G. Cormode and S. Muthukrishnan. What's hot and what's not: tracking most frequent items dynamically. In *Proceedings of ACM Principles of Database Systems*, 2003.

[8] N. Duffield, C. Lund, and M. Thorup. Estimating flow distributions from sampled flow statistics. In *Proceedings of the ACM SIGCOMM'03 Conference*, pages 325–336, 2003.

[9] B. Efron and R. Tibshrani. *An Introduction to the Bootstrap*. Chapman and Hall, New York, 1993.

[10] R. Fagin. Combining fuzzy information from multiple systems. *J. Comput. System Sci.*, 58, 1999.

[11] R. Fagin, A. Lotem, and M. Naor. Optimal aggregation algorithms for middleware. In *Proceedings of the 24th ACM Symposium on Principles of Database Systems*. ACM-SIGMOD, 2001.

[12] M. Fang, N. Shivakumar, H. Garcia-Molina, R. Motwani, and J. D. Ullman. Computing iceberg queries efficiently. In *Proceedings of the Twenty-fourth International Conference on Very Large Databases*, pages 299–310, 1998.

[13] P. Gibbons and S. Tirthapura. Estimating simple functions on the union of data streams. In *Proceedings of the 13th Annual ACM Symposium on Parallel Algorithms and Architectures*. ACM-SIGMOD, 2001.

[14] N. Hohn and D. Veitch. Inverting sampled traffic. In *Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*, pages 222–233, 2003.

[15] T. Johnson, S. Muthukrishnan, and I. Rozenbaum. Sampling algorithms in a stream operator. In *SIGMOD*. ACM, 2005.

[16] K. Keys, D. Moore, and C. Estan. A robust system for accurate real-time summaries of internet traffic. *ACM SIGMETRICS Performance Evaluation Review*, 33, 2005.

[17] A. Kumar, M. Sung, J. Xu, and J. Wang. Data streaming algorithms for efficient and accurate estimation of flow size distribution. *ACM SIGMETRICS Performance Evaluation Review*, 32, 2004.

[18] A. Kumar, M. Sung, J. Xu, and E. W. Zegura. A data streaming algorithm for estimating subpopulation flow size distribution. *ACM SIGMETRICS Performance Evaluation Review*, 33, 2005.

[19] G. Manku and R. Motwani. Approximate frequency counts over data streams. In *International Conference on Very Large Databases (VLDB)*, pages 346–357, 2002.

[20] T. Mori, M. Uchida, R. Kawahara, J. Pan, and S. Goto. Identifying elephant flows through periodically sampled packets. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, 2004.

[21] M. Theobald, G. Weikum, and R. Schenkel. Top-k query evaluation with probabilistic guarantees. In *Proceedings of the 30th VLDB Conference*, 2004.

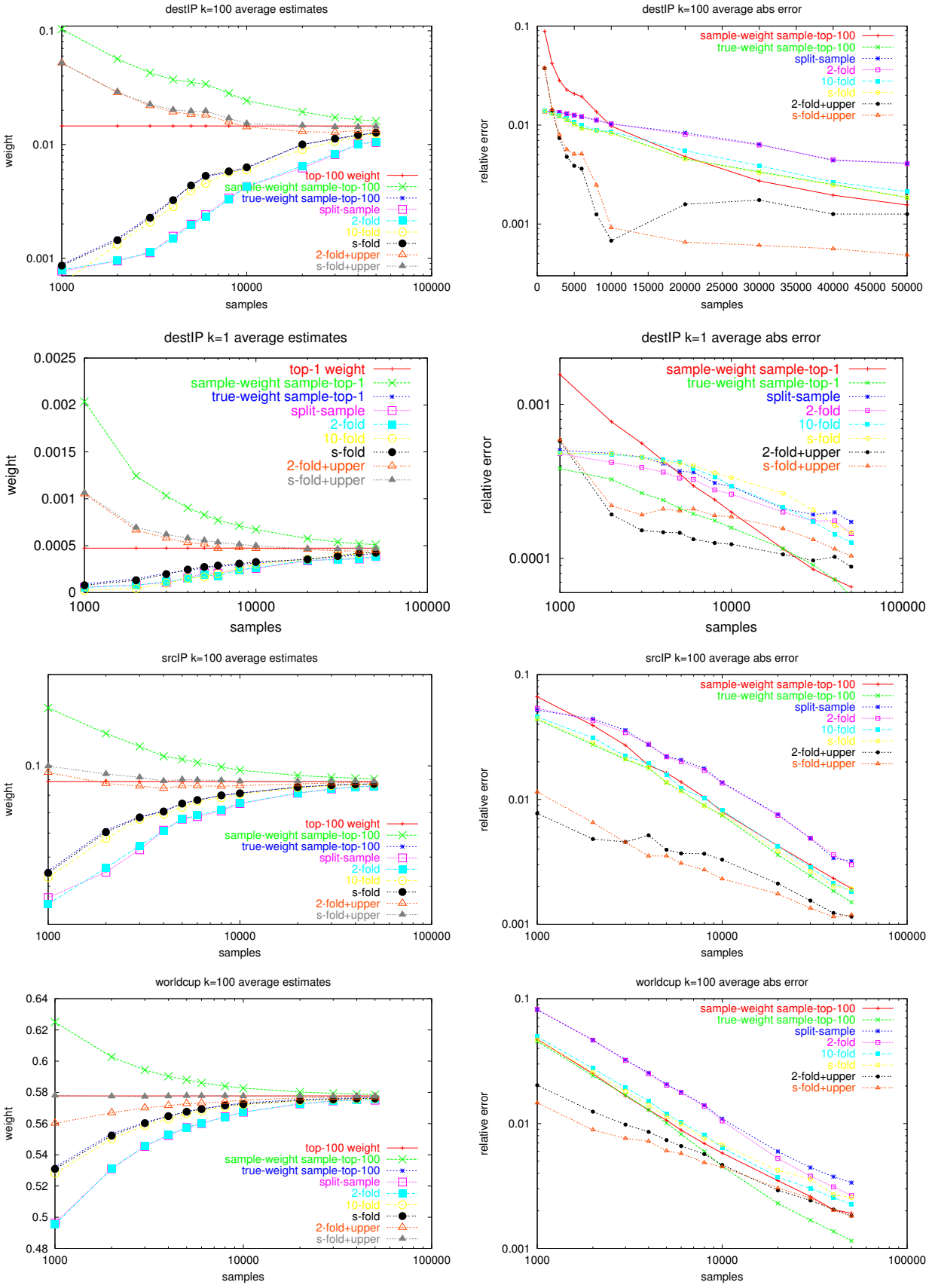


Figure 4: Average value (left) and corresponding average absolute error (right) of top-k estimators (averaged over 500 runs)

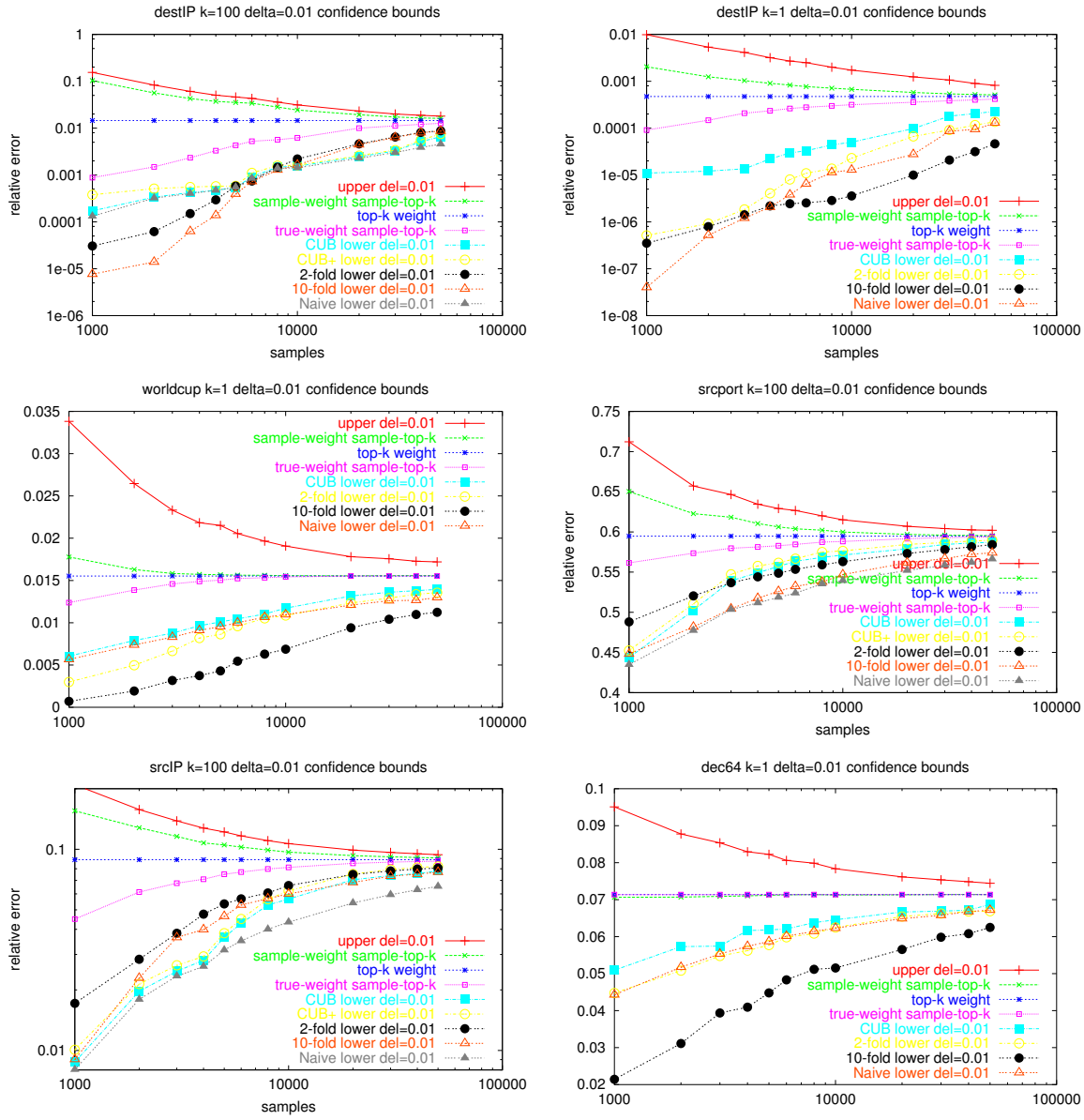


Figure 5: $(1-\delta)$**-confidence upper and lower bounds, by different methods, averaged over 500 runs**

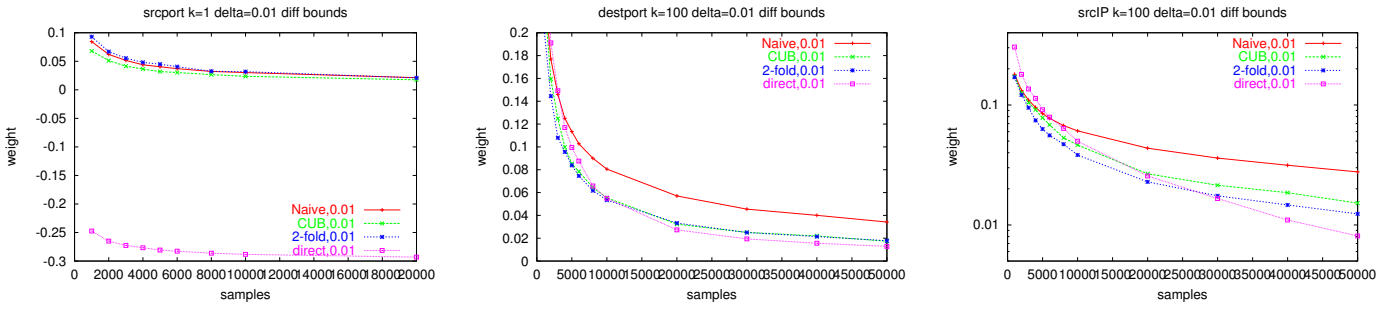


Figure 6: Upper bound on the difference between the weight of our sampled top-k set to the weight of the best alternative set (averaged over 500 runs)