Distance Queries from Sampled Data: Accurate and Efficient

Edith Cohen
Microsoft Research
Mountain View, CA, USA
editco@microsoft.com

ABSTRACT

Distance queries are a basic tool in data analysis. They are used for detection and localization of change for the purpose of anomaly detection, monitoring, or planning. Distance queries are particularly useful when data sets such as measurements, snapshots of a system, content, traffic matrices, and activity logs are collected repeatedly.

Random sampling, which can be efficiently performed over streamed or distributed data, is an important tool for scalable data analysis. The sample constitutes an extremely flexible summary, which naturally supports domain queries and scalable estimation of statistics, which can be specified after the sample is generated. The effectiveness of a sample as a summary, however, hinges on the estimators we have.

We derive novel estimators for estimating \( L_p \) distance from sampled data. Our estimators apply with the most common weighted sampling schemes: Poisson Probability Proportional to Size (PPS) and its fixed sample size variants. They also apply when the samples of different data sets are independent or coordinated. Our estimators are admissible (Pareto optimal in terms of variance) and have compelling properties.

We study the performance of our Manhattan and Euclidean distance \( (p = 1, 2) \) estimators on diverse datasets, demonstrating scalability and accuracy even when a small fraction of the data is sampled. Our work, for the first time, facilitates effective distance estimation over sampled data.

1. INTRODUCTION

Data is commonly generated or collected repeatedly, where each instance has the form of a value assignment to a set of keys: Daily summaries of the number of queries containing certain keywords, activity in a social network, transmitted bytes for IP flow keys, performance parameters (delay, throughput, or loss) for IP source destination pairs, environmental measurements for sensor locations, and requests for resources. In these examples, each set of values (instance) corresponds to a particular time or location. The universe of possible key values is fixed across instances but the values of a key are different.

One of the most basic operations in data analysis are Distance queries, which are used to detect, measure, and localize change [7, 26]. Their applications include anomaly detection, monitoring, and planning. Formally, a difference query between instances is specified by a meta-data based selection predicate that is applied to the keys. The result is the distance between the vectors projected on the selected keys. The simple example in Figure 1 shows two instances on 7 keys, and some queries specified on this data using Euclidean and Manhattan distances.

The collection and warehousing of massive data is subject to limitations on storage, throughput, and bandwidth. Even when the data is stored in full, exact processing of queries may be slow and resource consuming. Random sampling of datasets is widely used as a means to obtain a flexible summary over which we can query the data while meeting these limitations [27, 36, 6, 5, 8, 20, 21, 3, 17, 3, 22, 12, 19, 10, 13].

The sampling scheme is applied to our data, and a set of random bits, and returns a small subset of the entries. Our sample includes these entries and their values and has a fraction of the size of the original dataset. The sample only includes nonzero entries, this is particularly important when the data is sparse, meaning that the vast majority of keys have associated value 0. When the values distribution is skewed, we often apply weighted sampling, meaning that the probability a key is sampled depends on its value – Favoring heavier keys allows for more accurate estimates of sums and other statistics.

Perhaps the most basic sampling scheme is Poisson sampling, where keys are sampled independently. Weighting is often done using Probability Proportional to Size (PPS) sampling [23], where the inclusion probability is proportional to the value. Other common sampling schemes are bottom-k (order) samples, which have the advantage over independent sampling of yielding a sample size of exactly k. Bottom-k sampling generalizes reservoir sampling and includes Priority (Sequential Poisson) sampling and weighted sampling without replacement [3, 32, 30, 11, 19, 12, 13]. These sampling schemes are very efficient to apply also when the data is streamed or distributed.

Once we have the sample, we can quickly process approximate queries posed over the original data. This is done by applying an estimator to the sample. We seek estimators that provide good results when a small fraction of the data is sampled. In particular, we would like them to be admissible, that is, optimally use the information we have in the sample, and be efficient to compute. When estimating nonnegative quantities, such as distances, we are interested in nonnegative estimators.

---

1This is a full version of a KDD 2014 paper.
Consider the basic problem of estimating, for a given selection predicate, the sum of values of keys selected by the predicate. This problem is solved well by the classic Horvitz-Thompson (HT) [24] estimator. The estimate is the sum, over sampled keys $i$ satisfying the predicate, of the ratio $v_i/p_i$, where $v_i$ is the value and $p_i$ is the inclusion probability of $i$. This inverse-probability estimate is clearly unbiased (if $v_i > 0 \implies p_i > 0$): if the key is not sampled, the estimate is 0 and otherwise it is $v_i/p_i$, giving expectation $v_i$. The estimate is also nonnegative when values are. Moreover, the inverse-probability estimator is a UMVUE (uniform minimum variance unbiased estimator), meaning that among all unbiased and nonnegative estimators, it minimizes variance point wise. To apply this estimator, we need the inclusion probability $p_i$ to be available to it when $v_i$ is. This is the case with both Poisson and bottom-$k$ sampling, when implemented correctly.

The HT estimate is a sum estimator: conceptually, we apply an estimator to each key in the universe. Keys that are not sampled have an estimate of 0. Keys that are sampled have a positive estimate. We then sum the estimators of different keys. This makes it highly suitable for domain (selection) queries, which aggregate over a subset of keys that can be specified by a predicate.

We now turn to our problem of estimating the distance between instances from their samples. We seek an estimator with similar properties to the HT estimator: a sum estimator, unbiased and non-negative, and with optimal use of the information in the sample.

This problem turns out to be significantly more challenging. One can attempt to apply again the HT estimator: When the outcome reveals the value of the estimated quantity, the estimate is equal to the value divided by the probability of such an outcome. The estimate is 0 otherwise. Inverse probability estimates, however, are inapplicable to distance estimation over weighted samples, since they require that there is a positive probability for an outcome which reveals the exact value of the estimated quantity: the absolute difference between the values of the key in two instances. With multiple instances and weighted sampling, keys that have zero value in one instance and positive value in another have positive contribution to the distance but because zero values are never sampled, there is zero probability for determining the value from the outcome. In the Example in Figure 1, key $c$ has value 5 in the first instance and value 0 in the second, and thus has a contribution of 5 to the $L_1$ distance. The key however will never be included in a weighted sample of instance 2.

When considering multiple instances, we also need to specify how their samples relate to each other. The sampling schemes we mentioned, Poisson and bottom-$k$, specify the distribution for a single instance. Our first requirement, since data of different instances can be distributed or collected in different times, is that the sample of one instance can not depend on values assumed in another [16, 14]. The random bits, however, can be reused (using random hash functions), to make sampling probabilities dependent. The two extremes of the joint distribution of samples of different instances are independent sampling (independent sets of random bits for each instance) and coordinated sampling (identical sets of random bits). With coordinated sampling, which is a form of locality sensitive hashing, similar instances have similar samples whereas independent samples of identical instances can be completely disjoint. Each of these two relations has unique advantages and therefore in our work here we address both: Coordination [4, 34, 31, 33, 6, 5, 8, 20, 21, 3, 12, 22, 13, 16] allows for tighter estimators of many basic queries including distinct counts (set union) [8, 20, 21, 13], quantile sums [16], Jaccard similarity [6, 5], and more recently $L_1$ distance [16]. The drawbacks of coordination are that it results in unbalanced “burden” where same keys tend to be sampled across instances – an issue when, for example, being sampled translates to overhead which we would like to balance across keys. Moreover, while beneficial for some queries, the variance on some queries – notably sum queries that span values from multiple instances (“total number of search queries by Californians on Monday-Wednesday”, from daily summaries) – is larger than with independent sampling – an issue if our samples are primarily used for such queries.

Contributions:
We derive unbiased nonnegative estimators for $L_p^2$ distance queries over weighted samples. These include the two important cases of Manhattan distances $L_1$ and Euclidean distances $L_2$ (which is the square root of $L_2^2$). Our estimators apply with either Poisson or bottom-$k$ sampling. We also address the two important cases where the samples of different instances are independent or coordinated. Our work facilitates, for the first time, the use of weighted samples as summaries that support distance queries.

Our estimators have several compelling properties. Similarly to the HT estimator, our estimators are sum estimators: We estimate $L_p^2$ as a sum, over the selected keys, of nonnegative unbiased estimates of $RG_p = |v_1 - v_2|^p$ of the values assumed by the key (see Figure 1). Our estimators are unbiased, nonnegative, and admissible (Pareto optimal), meaning that another (nonnegative and unbiased) estimator with strictly lower variance on some data must have strictly higher variance on another data. The estimate $RG_p$, obtained for a particular key has high variance, since most likely, the key is not sampled in any instance (in which case the estimate is 0), but unbiasedness allows for diminishing relative error when more keys are selected. The distance $L_p$ can be estimated by the $p$th root of our $L_p^2$ estimate.

Our estimates $RG_p$ can be positive also when the samples do not reveal the respective value $RG_p$ but only partial information on it. This property turns out to be critical for obtaining admissible estimators, what is not possible, as we mentioned above, with the inverse-probability estimate. It also means, however, that the estimators we derive have to be carefully tailored to the power $p$.

For independently-sampled instances, we present an estimator for $RG_p$ of two values ($p > 0$). This derivation uses a technique we presented in [14]. Our estimator, which we call the $L^*$ estimator, is the unique symmetric and monotone admissible estimator, where monotonicity means that the estimate is non-decreasing with the information we can glean from the outcome.

For coordinated samples of instances, we apply our framework of monotone sampling [9]. We derive the $L^*$ and $U^*$ estimators for the $RG_p$ functions, which are admissible, unbiased, and nonnegative. The $L^*$ estimator is monotone and has lower variance for data with small difference (range) whereas the $U^*$ estimator performs better when the range is large. This choice is important, because it allows us to customize estimation to properties of the data set. Network traffic data, for example, is likely to have larger differences and thus the $U^*$ estimator may perform better whereas the $L^*$ estimator would be preferable when differences are smaller. The $L^*$ estimator also exhibits a compelling theoretical property of being “variance competitive” [15], meaning that for all data vectors, its variance is not too far off the minimum possible variance for the vector by a nonnegative unbiased estimator. This property makes the $L^*$ estimator a good default choice.

For $p = 1, 2$, which are the important special cases of the Manhattan and the Euclidean distances, we compute closed form expressions of estimators and their variance and also obtain tighter bounds on the “competitiveness” of the $L^*$ estimator. We evaluate and compare the performance of our $L_1$ and $L_2$ distance estimators on queries over diverse data sets. The queries vary in the support size (number of keys in the data set satisfying the selection
2. PRELIMINARIES

We denote by $v_{ih} \in \mathbb{R}_{\geq 0}$ the value of key $h \in K$ in instance $i \in [r]$ and by the vector $v(h)$, the values of key $h$ in all instances (the column vector $v_i(h)$). The exponentiated range of a vector $v$ is:

$$
RG_p(v) = (\max(v) - \min(v))^p \quad (p > 0)
$$

where $\max(v) \equiv \max_i v_i$ and $\min(v) = \min_i v_i$ are the maximum and minimum entry values of the vector $v$. We omit the subscript when $p = 1$.

We are interested in queries which specify a selected subset $H \subset K$ of keys, through a predicate on $K$, and return

$$
L_p^H(h) = \sum_{h \in H} RG_p(v(h)) .
$$

The $L_p$-distance of two instances $(r = 2)$ is $L_p(H) \equiv (L_p^H(H))^{1/p}$.

When data is sampled, we estimate $L_p^H$ by summing estimates $\hat{RG}_p$ for the respective single-key primitive $\hat{RG}_p(v_i(h))$ over keys $h \in H$. We use nonnegative unbiased estimators for the primitives, which result, from linearity of expectation, in unbiased estimates for the sums. We measure error by the coefficient of variation (CV), which is the ratio of the square root of the variance to the mean. Our estimates for each key have high variance, but when inclusions of different keys are pairwise independent, variance is additive and the CV decreases with $|H|$, allowing for accurate estimates of the sum. Finally, we can estimate $L_p(H)$ by taking the $p$th root of the estimate for $L_p^H(H)$. This estimate is biased, but the error is small when the CV of our $L_p^H$ estimate is small.

A basic component in applying sum estimators is obtaining from basic sampling schemes of instances the respective estimation problems for a single key. We cast the basic sampling schemes discussed in the introduction in the following form, which facilitates separate treatment of each key.

The sampling of the entry $v_{ih}$ is specified by a threshold value $\tau_{ih} \geq 0$, and random seed values $u_{ih} \sim U[0, 1]$ chosen uniformly at random.

$$
h \text{ is sampled in } i \iff v_{ih} \geq \tau_{ih}u_{ih} .
$$

The estimation problem for a single key $h$ is then as follows. We know $\tau = (\tau_1, \ldots, \tau_h)$, the seeds $u = (u_{1h}, \ldots, u_{rh})$, and the results of the sampling for $h$ (the value $v_{ih}$ in all instances $i$ where $h$ was sampled). We apply an estimator $\hat{RG}_h$ to this information, which we refer to as the outcome $S$, to estimate $\hat{RG}_h(v)$. The availability of the seeds $u_{ih}$ to the estimator turns out to be critical for estimation quality [14, 9]. We facilitate it by generating the seeds using random hash functions (pairwise independence between keys suffices for variance bounds). When we treat a single key $h$, we omit the reference to $h$ from the notation.

Sampling scheme of instances. We now briefly return to sampling schemes of instances, and show how we obtain the single-key formulation (3) from them.

With Poisson PPS (Probability Proportional to Size) sampling of instance $i$, each $h$ is sampled with probability proportional to $v_{ih}$. An equivalent formulation is to use a global threshold value $T_i$, such that a key $h$ is sampled if and only if $v_{ih} \geq u_{ih}T_i$. The expected sample size $E[|S|] = \sum_{h \in K} \min\{1, u_{ih}/T_i\}$ is determined by $T_i$. The sampling can be easily implemented with respect to either a desired sample size or a fixed threshold value. The respective estimation problem (3) for key $h$ has $\tau_{ih} = T_i$. As an example, we can obtain a PS Poisson sample of expected size $E[|S|] = 3$ for the instances in Figure 1 using $T_1 = 29/3$ (instance 1) and $T_2 = 33/3 = 11$ (instance 2).

Prior (sequential Poisson) sampling [30, 19, 35] is performed by assigning each key a priority $\tau_{ih} = v_{ih}/u_{ih}$. The sample of instance $i$ includes the $k$ keys with largest priorities, the $(k + 1)$th largest priority $T_i$, and the $k$th largest priority $T_i'$. To obtain the single-key formulation (3) for key $h$, we consider the sampling conditioned on fixing the seeds $u_{ih}$ (and thus the priorities $\tau_{ih}$) for all $j \neq h$ [12, 19]. The effective threshold $\tau_{ih}$ is the $k^{th}$ largest priority in $K \setminus \{h\}$: Key $h$ is then sampled if and only if $v_{ih}/u_{ih} > \tau_{ih}$. When $h \in S$, $\tau_{ih} = T_i$ and when $h \notin S$, $\tau_{ih} = T_i'$.

When sampling several instances, we can make the samples independent when we use independent $u_{ih}$ for all $i$. The samples are coordinated (shared-seed) if the same seed is used for the same key in all instances, that is, $v_{ih} \in V_i \forall i \in [r], u_{ih} = u_{ih} \equiv u_{ih}$.

Figure 2 shows Poisson PPS and priority samples, independent and coordinated, obtained for the two instances in Figure 1 from a random seed assignment $u_{ih}$. The figure also shows the threshold values $\tau_{ih}$, which are available to the estimators $\hat{RG}_h(S)$. As an example, the outcome for key 4, (the outcome is the input to the estimator), is as follows. When instances are independently Poisson sampled the outcome includes: $v_1 = 5$ (key 4 is sampled only in instance 1 so we know $u_{14}$ exactly), $u_1 = 0.15$, $u_2 = 0.36$ (seeds available by applying hash functions to the key), and $T_1 = 29/3$, $T_2 = 11$ (with Poisson PPS sampling, the same threshold applies to all keys in each instance). With coordinated priority sampling of instances the outcome includes $v_1 = 5$, $u = 0.15$, $T_1 = 13.8$ (since key 4 is sampled in instance 1) and $T_2 = 30.4$ (since key 4 is not sampled in instance 2).

From here onward, we focus on estimating $\hat{RG}_p(v)$ for a single key from the respective outcome $S$. We return to sum aggregates only for the experiments in Section 7.

Estimators: For an outcome $S$, we denote by

$$
S^* = \{v \mid S = S(u,v)\}
$$

the set of all data vectors consistent with $S$. We can equivalently define the outcome as the set $S^*$ since it captures all the information available to the estimator on $v$ and hence on $\hat{RG}_p(v)$. For our example in Figure 2, considering independent Poisson sampling and key 4, the set $S^*$ includes all vectors $(5, x)$ such that $x < u_{2T_2}/u_{1T_1} = 0.36 \cdot 11 = 3.96$. The actual data vector for key 4 is $(5, 0)$, but the outcome only partially reveals it.

We denote by $S$ the set of all possible outcomes, that is, any outcome consistent with any data vector $v$ in our domain. For data $v$, we denote by $S_v$, the probability distribution over outcomes consistent with $v$. As mentioned, we are seeking nonnegative estimators $\hat{RG}_h(S) \geq 0$ for all $S \in S$, since $\hat{RG}_h$ are nonnegative. Since we sum many estimates, we would like each estimate to be unbiased.
### 3. INDEPENDENT PPS SAMPLING

We derive estimators for $\text{RG}_p(v)$, where $v = (v_1, v_2)$ (two instances $r = 2$). The estimation scheme is specified by $\tau = (\tau_1, \tau_2)$. The outcome $S$ is determined by the data vector $v = (v_1, v_2)$ and $u = (u_1, u_2)$, where $u_i \sim U[0, 1]$ are independent. The set $S^*$ of vectors with $S$ is (4).

We derive the L*-estimator, $\text{RG}_P^L(v)$, which is the unique symmetric, monotone, and admissible estimator. The construction adapts a framework from [14], which was used to estimate $\max\{v_2, v_1\}$. We specify an order $\prec$ on the data domain. We then formulate a set of sufficient constraints for an unbiased symmetric and order-optimal estimator $\hat{f}(\prec)$ of $\text{RG}_P(v)$. The constraints, however, do not incorporate nonnegativity, as this results in much more complex dependencies. But if we find a nonnegative solution $\hat{f}(\prec)$, then we find an estimator with all the desired properties. We therefore hope for a good “guess” of $\prec$.

We work with $\prec$ that prioritizes smaller distances, that is, $v \prec \bar{z}$ if and only if $\text{RG}(v) < \text{RG}(\bar{z})$. With each outcome $S \in S$, we associate its determining vector $\phi(S)$, which we define as the $\prec$-minimal vector in (the closure of) $S^*$. The closure is the set obtained when using a non-strict inequality in (4). The determining vector is unique for all outcomes $S$ that are not consistent with $\text{RG}_P(v) = 0$, that is, $\text{RG}_P(v) > 0$ for all $v \in S^*$. As we mentioned earlier, we only need to specify the estimator on these outcomes, since we only consider estimators that are 0 on outcomes consistent with $\text{RG}_P(v) = 0$. The mapping of outcomes $S$ to the determining vector $\phi(S)$ is shown in Table 1 (Bottom).

We now formulate sufficient constraints for $\prec$-optimality. Conditioned on fixing the estimator on outcomes $S$ such that $\phi(S) \prec v$, the “best” we can do, in terms of minimizing variance, is to set it to a fixed value on all outcomes such that $\phi(S) = v$. This fixed value is determined by the unbiasedness requirement. Since $\hat{f}(\prec)$ is the same for all outcomes with same determining vector, we specify it as a function of the determining vector $\hat{f}(\prec)(S)$.

We use the notation $S_\phi(v)$ for the set of outcomes $S$ that are consistent with $\phi$ but also consistent with a vector that precedes $v$:

$$S_\phi(v) = \{S | v \in S^* \land \phi(S) \prec v\}$$

The contribution of the outcomes $S_\phi(v)$ to the expectation of $\hat{f}(\prec)$ when data is $v$ is

$$f_0(v) = \mathbb{E}_{S \sim S_\phi}[I_{S \in S_\phi(v)} \hat{f}(\prec)(S)]$$

where $I$ is the indicator function. We obtain the following sufficient constraints for a $\prec$-optimal unbiased estimator, that may not be nonnegative, but is forced to be 0 on all outcomes consistent with $\text{RG}_P(v) = 0$. For all $v$,

$$\text{pr}_{S \sim S_\phi}[\phi(S) = v] = 0 \implies f_0(v) \equiv f(v)$$

Table 1: Top: Estimator $\text{RG}_P^L$ for $p > 0$ over independent samples, stated as a function of the determining vector $\phi = (\phi_1, \phi_2)$ when $\phi_1 \geq \phi_2$ (case $\phi_2 > \phi_1$ is symmetric). Bottom: mapping of outcomes to determining vectors.
We derive $\hat{R}_{g}^{L}(v)$ by solving the right hand side of (6) for all $v$ such that $Pr_{p,g}(\phi(S) = v) > 0$. The solution $\hat{R}_{g}^{L}(\phi)(p > 0)$ is provided in Table 1 through a mapping of determining vectors to estimate values. The estimator is specified for $\phi_{1} \geq \phi_{2}$, as the other case is symmetric. We can verify that for all $p > 0$, the estimator $\hat{R}_{g}^{L}(v)$ is nonnegative, monotone (for all $y$, $\hat{R}_{g}^{L}(y,x)$ is non-increasing for $x \in (0, y)$) and has finite variances (follow from $\int_{0}^{y} \hat{R}_{g}^{L}(y,x)^{2}dx < \infty$). We can also verify that condition (5) holds. Vectors $v$ with $Pr_{p,g}(\phi(S) = v) = 0$ are exactly those with one positive and one zero entry. We can verify that (5) is satisfied, that is, $E_{p,g} \hat{R}_{g}^{L}(v) = \hat{R}_{g}^{L}(v)$ on these vectors. Table 2 shows explicit expressions of $\hat{R}_{g}^{L}(v)$ and $\hat{R}_{g}^{L}(v)$.

<table>
<thead>
<tr>
<th>$\phi = (\phi_{1}, \phi_{2})$</th>
<th>$\hat{R}_{g}^{L}(\phi)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi = (0,0)$</td>
<td>$0$</td>
</tr>
<tr>
<td>$\phi_{1} \geq \phi_{2} \geq \tau_{2}$</td>
<td>$\tau_{1} \min(1, \frac{\tau_{1}}{\tau_{2}}, \frac{\tau_{1}}{\tau_{1}-\tau_{2}}) + \tau_{1} \max(0, \frac{\tau_{1}}{\tau_{1}-\tau_{2}})$</td>
</tr>
<tr>
<td>$\phi_{1} \geq \phi_{2} \leq \tau_{2}$</td>
<td>$0$</td>
</tr>
</tbody>
</table>

Table 2: Explicit form of estimators $\hat{R}_{g}^{L}(v)$ and $\hat{R}_{g}^{L}(v)$ for $r = 2$ over independent samples. Estimator is stated as a function of the determining vector $(\phi_{1}, \phi_{2})$ when $\phi_{1} \geq \phi_{2}$ (case $\phi_{2} \geq \phi_{1}$ is symmetric).

We now proceed to the derivation. We consider vectors $v$ in increasing order and solve (6) for $f^{(v)}$ on outcomes with determining vector $v = (v, v - \Delta)$, where $v \geq \Delta > 0$.

- **Case**: $v - \Delta \geq \tau_{2}$. The outcomes always reveals the second entry. The determining vector is $v$ when $u_{1} \tau_{1} \leq v$, which happens with probability $\min(1, \frac{\tau_{1}}{\tau_{1}})$. Otherwise, the outcome is consistent with $(v - \Delta, v - \Delta)$ and the estimate is 0. We solve the equality $\Delta^{p} = \min(1, v/\tau_{1}) \hat{R}_{g}^{L}(v)$, obtaining

$$
\hat{R}_{g}^{L}(v, v - \Delta) = \frac{\tau_{1}}{\min(\tau_{1}, v/\tau_{1})} \Delta^{p}.
$$

(7)

- **Case**: $v - \Delta < \tau_{2}$. The determining vector is $(v, v - \Delta)$ with probability $\min(1, \frac{\tau_{1}}{\tau_{1}}) \frac{v - \Delta}{\tau_{2}}$. Otherwise, it is $(v, v - y)$ for some $y < \Delta$. We use (6) to obtain an integral equation:

$$
\Delta^{p} = \frac{\min(\tau_{1}, v/\tau_{1})}{\tau_{1}} \frac{v - \Delta}{\tau_{2}} \hat{R}_{g}^{L}(v, v - \Delta) + \frac{\min(\tau_{1}, v/\tau_{1})}{\tau_{1}} \frac{v - \Delta}{\tau_{2}} \int_{max(0,v-v_{2})}^{\Delta} \hat{R}_{g}^{L}(v, y)dy
$$

Taking a partial derivative with respect to $\Delta$, we obtain

$$
\frac{\partial \hat{R}_{g}^{L}(v, v - \Delta)}{\partial \Delta} = \frac{\tau_{1} \tau_{2}}{\min(\tau_{1}, v/\tau_{1})} \frac{\Delta^{p-1}}{v - \Delta}
$$

We use the boundary value for $\Delta = \max(0, v - \tau_{2})$:

$$
\hat{R}_{g}^{L}(v, \min(v, \tau_{2})) = \frac{\tau_{1}}{\min(\tau_{1}, v/\tau_{1})} \max(0, v - \tau_{2})^{p},
$$

and obtain the solution

$$
\hat{R}_{g}^{L}(v, v - \Delta) = \frac{p \tau_{1} \tau_{2}}{\min(\tau_{1}, v/\tau_{1})} \int_{max(0,v-v_{2})}^{\Delta} \frac{\Delta^{p-1}}{v - y} dy + \frac{\tau_{1}}{\min(\tau_{1}, v/\tau_{1})} \max(0, v - \tau_{2})^{p},
$$

(8)

The special case $\tau_{1} = \tau_{2} = \tau$: The estimators $\hat{R}_{g}^{L}(v)$ and $\hat{R}_{g}^{L}(v)$ as a function of the determining vector and their variance are provided in Tables 3 and 4. For data vectors where $v_{1} \geq v_{2} \geq \tau$, $\hat{R}_{g}^{L}(v_{1} - v_{2})$ and $\var[\hat{R}_{g}^{L}(v_{1} - v_{2})]$ is 0. If $v_{1} \geq \tau \geq v_{2}$, $\hat{R}_{g}^{L}(v_{1} - v_{2}) = \ln \frac{v_{2}}{v_{1}} + \ln \frac{v_{1}}{v_{2}}$ and $\var[\hat{R}_{g}^{L}(v_{1} - v_{2})]$ is $\frac{2\tau v_{1} v_{2}}{v_{1}^{2} + v_{2}^{2}}$. Finally, if $v_{2} \leq v_{1} \leq \tau$, $\hat{R}_{g}^{L}(v_{1} - v_{2}) = \frac{v_{2}^{2}}{v_{1}} \ln \frac{v_{2}}{v_{1}}$ and

$\var[\hat{R}_{g}^{L}(v_{1} - v_{2})] = \frac{v_{1} v_{2}}{\tau^{2}} (\frac{\tau^{2} v_{1} v_{2}}{v_{1}^{2}} - (v_{1}^{2} - v_{2}^{2}))^{2} +$$\frac{(1 - \frac{v_{1}^{2}}{\tau^{2}})(v_{1} - v_{2})^{2}}{\tau^{2}}$ $+$$\frac{v_{1}^{2}}{\tau^{2}} \int_{v_{1}}^{v_{2}} \frac{(\frac{\tau^{2} v_{1} v_{2}}{v_{1}^{2}} - (v_{1} - v_{2})^{2})dy}{v_{1}^{2}}$ $= 2(\tau)^{2}((1 - \frac{v_{1}^{2}}{\tau^{2}}) \ln \frac{v_{1}}{v_{2}} - \frac{v_{2}^{2}}{v_{1}^{2}} - (v_{1} - v_{2})^{2})$.

Table 3: $\hat{R}_{g}^{L}(v)$ and its variance for independent samples.

Similarly, for $v_{2} \leq v_{1} \leq \tau$, $\hat{R}_{g}^{L}(v_{1} - v_{2}) = 2(\tau)^{2}(\ln \frac{v_{2}}{v_{1}} - \frac{v_{2}^{2}}{v_{1}^{2}} - v_{1} - v_{2})^{2}$.

4. SHARED-SEED SAMPLING

We derive estimators for $R_{g}(v)(p > 0)$, where $v = (v_{1}, \ldots, v_{r})$ for $r > 2$. The sampling uses the same random seed $u$ for all entries. The outcome $(S, (u, v))$ is determined by the data $v$ and a scalar seed value $u \in (0, 1)$, drawn uniformly at random: Entry $i$ is included in $S$ if and only if $v_{i} \geq \tau_{1} u$.

We apply our work on estimators for monotone sampling [9] to derive two unbiased nonnegative admissible estimators: The $L^{*}$ estimator $\hat{R}_{g}^{L}(\phi)$ and the $U^{*}$ estimator $\hat{R}_{g}^{L}(\phi)$. We present closed form expressions of estimators and variances when $\tau$ has all entries equal (to the scalar $\tau$). The derivations easily extend to non-uniform $\tau$.

<table>
<thead>
<tr>
<th>Condition</th>
<th>$R_{g}(\phi)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi_{1} \geq \phi_{2} \geq \tau$</td>
<td>$\phi_{1} - \phi_{2} \geq \tau$</td>
</tr>
<tr>
<td>$\phi_{1} \geq \tau \geq \phi_{2}$</td>
<td>$\phi_{1} - \phi_{2} \geq \tau$</td>
</tr>
<tr>
<td>$\phi_{2} &lt; \phi_{1} \leq \tau$</td>
<td>$\phi_{1} - \phi_{2} \geq \tau$</td>
</tr>
</tbody>
</table>

Table 4: $\hat{R}_{g}^{L}(v)$ and its variance for independent samples.
The set of data vectors consistent with outcome \( S(u, v) \) is 
\[ S^* = \{ z | \forall i \in [r], i \notin S \Rightarrow z_i = v_i, \ i \in S \Rightarrow z_i < \tau_i u \}. \]
Observe that the sets \( S^*(u, z) \) are the same for all consistent data vectors \( z \in S^*(u, v) \). Fixing the data \( v \), the set \( S^*(u, v) \) is non-decreasing with \( u \), which means that the information on the data that we can glean from the outcome can only increase when \( u \) decreases. This makes the sampling scheme monotone in the randomization, which allows us to apply the estimator derivations in [9].

### The lower bound function

The derivations use the lower bound function \( \text{RG}_p \), which maps an outcome \( S \) to the infimum of \( \text{RG}_p \) values on vectors that are consistent with the outcome:
\[ \text{RG}_p(S) = \inf_{\forall i \in S} \text{RG}_p(v). \]
For \( R \), the lower bound is the difference between a lower bound on the maximum entry and an upper bound on the minimum entry.
\[ \text{RG}(S) = \max_i v_i - \min_{i \in S} \min_{i \notin S} \tau_i u. \]
The lower bound on \( \text{RG}_p \) is the \( p \)th power of the respective bound on \( \text{RG} \), that is, \( \text{RG}_p(S) = \text{RG}(S)^p \). For \( S(u, v) \), we use the notation \( \text{RG}_p(S(u, v)) \equiv \text{RG}_p(u, v) \). For all-entries-equal \( \tau \):

| condition | \(|S|\) | RG(S) |
|-----------|--------|--------|
| \( u > \frac{\max(u)}{\tau} \) | 0 | 0 |
| \( \frac{\max(u)}{\tau} \geq u \geq \frac{\min(u)}{\tau} \) | 1 \( \Rightarrow \) | \( \max(v) - u \tau \) |
| \( u < \frac{\min(u)}{\tau} \) | \( r \) | \( \text{RG}(v) \) |

#### Figure 3: Left: The lower bound function and corresponding lower hull for example vectors and \( p \in \{0.5, 1, 2\} \). Right: the corresponding optimal, \( L^* \), and \( U^* \) estimates on outcomes consistent with the vector.

### The \( L^* \) estimator

The \( L^* \) estimator [9] is defined for any monotone estimation problem for which an unbiased and nonnegative estimator exists. This estimator is specified as a function of the corresponding lower bound function. For \( \text{RG}_p \) we have
\[ \forall S(\zeta, v), \text{RG}_p(S) = \frac{\text{RG}_p(\zeta, v)}{\zeta} = \int_\zeta^1 \text{RG}_p(u, v) \frac{du}{u^2}. \]

From [9], we know that \( \text{RG}_p(L) \) has the following properties:

- It is nonnegative and unbiased.
- It is the unique (up to equivalence) admissible unbiased nonnegative monotone estimator, meaning that the estimate is non-decreasing with \( u \).
- It is \( <^* \)-optimal with respect to the partial order \( <^* \)
\[ v <^* z \iff \text{RG}_p(v) < \text{RG}_p(z). \]

\( <^* \)-optimality with respect to this particular order means that any estimator with a strictly lower variance for a data vector must have strictly higher variance on some vector with a smaller range – this means that the \( L^* \) estimator “prioritizes” data where the range (or difference when aggregated) is small.

The \( L^* \) estimator has finite variances when the monotone estimation problem admits a nonnegative estimator with finite variances. It is also 4-competitive in terms of variance [15], meaning that for any data vector, the ratio of the expectation of the square to the minimum one possible for the data via an unbiased nonnegative estimator is at most 4
\[ \forall \nu, \mathbb{E}_{S \sim \nu} [\text{RG}_p(S)^2] \leq 4 \mathbb{E}_{S \sim \nu} [\text{RG}_p(\nu)^2], \]
where \( \text{RG}_p(\nu) \) is a nonnegative unbiased estimator which minimizes the variance for \( \nu \) (We will present a construction of this estimator). Competitiveness, is a strong property that means that for all data vectors, the variance under the \( L^* \) estimator is not too far off the minimum possible variance for that vector by a nonnegative unbiased estimator.

For our sampling scheme with all entries equal \( \tau \), we define \( \max(v) \equiv \max_i v_i \) (which is available from \( S \) whenever \( |S| > 0 \)) and \( v_{\min} = \min(v) \) if \( |S| = r \) and \( v_{\min} = u \tau \) otherwise, which is also always available from \( S \), the estimator is
\[ \text{RG}_p(L)(S) = \begin{cases} 0 & |S| = 0 \\ |S| \geq 1: \left( \max(v) - v_{\min} \right)^p \max\left\{ 1, \frac{\tau}{\min_{i \in S} \tau_i} \right\} - \int_{\min_{i \in S} \tau_i}^{\max(v)} (\max(v) - x)^p dx \end{cases} \]

Estimators and variance for \( R \) and \( \text{RG}_2 \) are provided in Tables 6 and 7. We also compute a tight ratio on variance competitiveness for \( p = 1, 2 \):

**Lemma 4.1.**
\[ \forall \nu, \mathbb{E}_{S \sim \nu} [\text{RG}_p(S)^2] \leq 2 \mathbb{E}_{S \sim \nu} [\text{RG}_p(\nu)^2] \]
\[ \forall \nu, \mathbb{E}_{S \sim \nu} [\text{RG}_2(S)^2] \leq 2.5 \mathbb{E}_{S \sim \nu} [\text{RG}_2(\nu)^2] \]

### The \( U^* \) estimator

The \( U^* \) estimator [9] is the solution of the integral equation
\[ \forall S(\zeta, v), \text{RG}_p(\zeta, v) = \sup_{x \in S^+} \inf_{0 \leq \eta < \zeta} \max_{u \leq v} \text{RG}_p(u, v) du \]

From [9], we know that \( \text{RG}_p(U) \) has the following properties:

- It is nonnegative and unbiased.
- It is \( <^* \)-optimal with respect to the partial order \( <^* \)
\[ v <^* z \iff \text{RG}_p(v) > \text{RG}_p(z). \]

This means that the \( U^* \) estimator “prioritizes” data where the range (or difference when aggregated) is large. In particular, it is the nonnegative unbiased estimator with minimum variance on data with \( \min(v) = 0 \).
The solution for all-entries equal \( \tau \) is provided as Algorithm 1 (see Appendix A for calculation). The estimator is admissible and has finite variances for all data vectors.

The estimator \( \text{RG}_p^{(w)} \) and its variance for \( p = 1, 2 \) are provided in Tables 8 and 9 (See Appendix B for details).

\( \nu \)-optimality. We say that an estimator is \( \nu \)-optimal (for data \( v \)) if amongst all estimators that are nonnegative and unbiased for all data, it has the minimum possible variance when the data is \( v \). In order to measure the competitiveness of our estimators, as in Lemma 4.1, we derive an expression for the \( \nu \)-optimal estimate values \( \text{RG}_p^{(w)} \). It turns out that the values assumed by a \( \nu \)-optimal estimator on outcomes consistent with \( v \) are unique (almost everywhere) on \( S_v \) [15]. Note that there is no single estimator that is \( \nu \)-optimal for all \( v \), that is, there is no uniform minimum variance unbiased nonnegative estimator for \( \text{RG}_p \). Therefore, the \( \nu \)-optimal estimates for all possible values of \( v \) can not be combined into a single estimator.

We now obtain an explicit representation of \( \text{RG}_p^{(w)} \). We use the notation \( H_{\text{RG}_p}^{(w)}(u) \) for the lower boundary of the convex hull (lower hull) of \( \text{RG}_p^{(w)}(u, v) \) and the point \((1, 0)\). This function is monotone non-increasing in \( u \) and therefore differentiable almost everywhere. We apply the following.

**Theorem 4.1.** [15] A nonnegative unbiased estimator \( \text{RG}_p \), minimizes \( \text{VAR}_{S_v} \text{RG}_p \) \( \iff \) almost everywhere on \( S_v \)

\[
\text{RG}_p^{(w)}(u) = -\frac{d H_{\text{RG}_p}^{(w)}(u)}{du}. \tag{13}
\]

The estimates (13) are monotone non-increasing in \( u \).

We can now specify \( \text{RG}_p^{(w)} \) for PPS sampling with all-entries-equal \( \tau \). The function \( \text{RG}_p^{(w)}(u, v) = \max\{0, \max(v) - \tau\} \) for \( u \geq \max(v) \), and equal to \( \text{RG}_p^{(w)}(u) \) for \( u < \max(v) \). Therefore for \( u \geq \max(v) \), the lower hull is \( H_{\text{RG}_p}^{(w)}(u) = 0 \) and \( \text{RG}_p^{(w)}(u) = 0 \).

For \( p \leq 1 \), the function is concave for \( u \in \left[ \min(v), \max(v) \right] \). The lower hull is therefore a linear function for \( u \leq \max(v) \), when \( \max(v) \leq \tau \), \( H_{\text{RG}_p}^{(w)}(u) = \text{RG}_p^{(w)}(v)(1 - u \frac{\tau}{\max(v)}) \) and when \( \max(v) \geq \tau \), \( H_{\text{RG}_p}^{(w)}(u) = \text{RG}_p^{(w)}(v) - u(\text{RG}_p^{(w)}(v) - \max(v) - \tau)^p \).

The \( \nu \)-optimal estimates are therefore constant for \( u \leq \min\{1, \max(v), \frac{\max(v)}{\tau}\} \): \( \text{RG}_p^{(w)}(u) = \text{RG}_p^{(w)}(\max(v)) \) when \( \max(v) \leq \tau \), and \( \text{RG}_p^{(w)}(u) = \text{RG}_p^{(w)}(v) - (\max(v) - \tau)^p \) when \( \max(v) \geq \tau \).

For \( p > 1 \), \( \text{RG}_p^{(w)}(u, v) \) is convex for \( u \in \left[ \min(v), \max(v) \right] \). Geometrically, the lower hull follows the lower bound function for \( u > \alpha \), where \( \alpha \) is the point of the slope where the lower bound function is equal to the slope of a line segment connecting the current point to the point \((0, \text{RG}_p(v))\). For \( u \leq \alpha \), the lower hull follows this line segment and is linear. Formally, the point \( \alpha \) is the solution of

\[
\text{RG}_p(v) = (\max(v) - x\tau)^{p-1}(p\tau + \max(v) - x\tau). \tag{14}
\]

If there is no solution \( \alpha \in \left[ \min(v), \min\{1, \max(v)\} \right] \), we use \( \alpha = \min\{1, \max(v)\} \). The estimates for \( u \in [\alpha, \min\{1, \max(v)\}] \) are \( \text{RG}_p^{(w)}(u, v) = -\frac{d \text{RG}_p(v)}{du}(u - \tau)^{p-1} \) and for \( u \leq \alpha \), \( \text{RG}_p^{(w)}(u) = \text{RG}_p(v)(\max(v) - \alpha)^p \).

Figure 3 (top) illustrates \( \text{RG}_p^{(w)}(u, v) \) and the corresponding lower hull \( H_{\text{RG}_p}^{(w)}(u) \) as a function of \( u \) for example vectors with \( p \in \{0.5, 1, 2\} \).

Now that we expressed \( \text{RG}_p^{(w)}(S) \) on all outcomes consistent with \( v \), we can compute for (any vector \( v \)), the minimum possible

\[
\text{VAR}_{S_v}[\text{RG}_p^{(w)}(u)] = \int_0^1 \text{RG}_p^{(w)}(u)^2 du - \text{RG}_p^{(w)}(v)^2. \tag{14}
\]

We use the expectation of the square to measure the “variance competitiveness” of estimators (Since the second summand in (14) is the same for all estimators). The ratio for a particular \( v \) is the expectation of the square for \( v \) divided by the optimal one which is \( E_{S \sim S_v}[\text{RG}_p^{(w)}(S)^2] = \int_0^1 \text{RG}_p^{(w)}(u)^2 du \). The competitive ratio is the maximum ratio over all \( v \).

**The optimally competitive (OC) estimator.** We used a program to compute the estimator with minimum competitive ratio. The estimator was computed for an outcome that revealed \( \max(v) \) and provided an upper bound \( x < \max(v) \) on \( \min(v) \). The domain was discretized and the estimates were computed iteratively for decreasing \( x \) so that the estimates satisfy a certain ratio \( c \). We then performed a search to find the minimum \( c \) for which the computation is successful.

**Choosing between the \( L^* \), \( U^* \), and \( OC \) estimators.** Figure 3 shows the \( \nu \)-optimal estimates and the \( L^* \) and \( U^* \) estimators for example vectors, illustrating the monotonicity of \( L^* \) and how the estimators relate to each other. The estimators and their variances depend only on \( \tau \) and the maximum and minimum entry values \( \max(v) \) and \( \min(v) \). We study the variance for all-entries-equal \( \tau \) and \( \max(v) \leq \tau \).

Figure 5 shows the expectation of the square of the \( L^* \) estimator over independent samples and of the \( L^* \), \( U^* \), and \( OC \) estimators over shared-seed samples. This is as a function of the ratio \( c \) of the minimum to the maximum value in the data vector. The expectation of the square plotted is the ratio to the minimum possible expectation of the square for the data point (over shared-seed samples), as a function of the ratio \( \min(v)/\max(v) \). Estimator \( \text{RG}_p^{(L)} \) over independent samples, and \( \text{RG}_p^{(L)} \), \( \text{RG}_p^{(U)} \), and the optimally competitive estimator over shared-seed samples. Sampling with all-entries equal \( \tau \).
satisfies

\[ \text{VAR}_o[\hat{R}_G^{(U)}] \leq \text{VAR}_o[\hat{R}_G^{(L)}] \iff \min(v) \leq \phi_p \cdot \max(v) \]

For \( p = 1 \), \( \phi_1 \approx 0.285 \) (is the solution of the equality \((1 - x)/(2x) = \ln(1/x))\). For \( p = 2 \), \( \phi_2 \approx 0.258 \).

This suggests selecting an estimator according to expected characteristics of the data. If typically \( R_G(v) > (1 - \phi_p) \max(v) \), we choose \( \hat{R}_G^{(U)} \). If typically \( R_G(v) < (1 - \phi_p) \max(v) \) we choose \( \hat{R}_G^{(L)} \), and otherwise, we choose the OC estimator.

The variance of the \( L^* \) estimator over independent samples and of the \( L^* \) and \( U^* \) estimators over shared-seed samples is illustrated in Figure 4. The figure also illustrates the relation between the variance of the shared-seed \( L^* \) and \( U^* \) estimators. When \( \max(v) / \tau \ll 1 \) (which we expect to be a prevailing scenario), \( \text{VAR}[\hat{R}_G^{(L)}] \) is nearly at most 2 times \( \text{VAR}[\hat{R}_G^{(U)}] \) but as \( \min(v) \rightarrow \max(v) \), the ratio \( \text{VAR}[\hat{R}_G^{(U)}]/\text{VAR}[\hat{R}_G^{(L)}] \) is not bounded. When \( \max(v) / \tau \ll 1 \), the variance of the \( U^* \) estimator is close to 0, and \( \text{VAR}[\hat{R}_G^{(L)}]/\text{VAR}[\hat{R}_G^{(U)}] \) is not bounded. Interestingly, for \( p = 2 \), the variance of the \( U^* \) estimator is always at least \( \frac{2}{\tau} \text{RG}_4(v) \), and thus, using (12), the variance of the \( L^* \) estimator is at most 4.4 times the variance of the \( v \)-optimal (and thus of the \( U^* \) estimator).

5. EXTENSIONS

The one-sided distance, which isolates the growth or decline components, is defined as \( L^{p+}_p(H) = \sum_{h \in H} R_{G^{p+}}(v(h)) \), where \( R_{G^{p+}}(v) = \max\{0, v_1 - v_2\}^p \). We can use any of our \( R_G \) estimators \( R_{G^{p+}}(S) \) to estimate \( R_{G^{p+}} \) as follows: If \( S^* \) includes \( v \) such that \( v_1 \leq v_2 \) then \( R_{G^{p+}}(S) = 0 \). Otherwise, \( R_{G^{p+}}(S) = R_{G^{p+}}(S) \).

We can symmetrically define \( L^{p-}_p \) and \( R_{G^{p-}} \) and \( R_{G^{p-}} \).

Our derivations can be extended to other sampling schemes. One such extension is to weighted sampling without replacement (PP-SWR), which is bottom-\( k \) sampling with priorities \( \tau_{ih} = \frac{\ln(1 - u_{ih})}{v_{ih}} \) [33, 32, 11, 12].

6. RELATED WORK

Prior to our work, the only distance estimator we are aware of which obtained good estimate with small fraction of data sampled is for \( L_1 \) over coodinated samples. This estimator uses the relation \([v_1 - v_2] = \max\{v_1, v_2\} - \min\{v_1, v_2\} \approx \min(v) \cdot \tau \) to obtain an indirect estimate as the difference of two inverse probability estimates for the maximum and minimum [16]. Our \( U^* \) estimator for \( p = 1 \) is a strengthening of this \( L_1 \) estimator.

Distance estimation over unweighted coordinated [28] or independent [14] sampling is a much simpler problem. With unweighted sampling, the inclusion probability of positive entries is independent of their weight. When carefully implemented, we can use inverse-probability estimates, which as discussed in the introduction, do not work with weighted sampling. An unweighted sample, however, is much less informative of its size when the data is skewed, as “heavy hitters” can be easily missed out. The estimators we develop here are applicable with, and take advantage of, weighted sampling.

Distance estimation of vectors (each instance in our terminology is presented as a vector of key values) was extensively studied using linear sketches, which are random linear projections, e.g. [25, 2, 1, 18]. Random projections have the property that the difference vector of sketches is the sketch of the difference of the two vectors. This means they are tailored for distance-like queries which aggregate over functions of the coordinate-wise differences. In particular, with linear sketches we can accurately estimate distances that are very small relative to the input vectors norms, whereas even with weighted sampling, accuracy depends on the relation of the distance to the vectors norms. A significant disadvantage of linear sketches, however, is lack of flexibility: Unlike samples, they do not support domain (selection) queries that are specified after the summary structure is computed and can only estimate distance between the full vectors. Moreover, each sketch is tailored for a specific metric, such as \( L_2^2 \). In particular, linear sketches are not suitable at all for estimating one-sided distances. Lastly, linear
sketches have size which often depends (poly) logarithmically on the number of keys. To summarize, the two techniques, sampling and linear sketches, have different advantages. Sampling provides much greater flexibility in terms of supported queries and often admits a smaller summary structure.

7. EXPERIMENTAL EVALUATION

We selected several datasets that have the form of values assigned to a set of keys, on two instances, and natural selection predicates. We consider the $L_1$ and $L_2^2$ distances on keys satisfying these predicates. Properties of the data sets with selections are summarized below and in Table 5.

- **destIP** (IP packet traces): keys: (anonymized) IP destination addresses. value: the number of IP flows to this destination IP. Instances: two consecutive time periods. Selection: all IP destination addresses in a certain subnet.


- **Surnames and OSPD8**: keys: all words (terms). value: the number of occurrences of the term in English books digitized by Google and published within the time period [29]. Instances: the years 2007 and 2008. Surnames selection: the 18.5 $\times$ 10^{3} most common surnames in the US. OSPD8 selection: the 7.5 $\times$ 10^{4} 8 letter words that appear in the Official Scrabble Players Dictionary (OSPD).

Each of the two instances were Poisson PPS [23] sampled (see Section 2) with different sampling threshold $T$, to obtain a range of sample sizes. We used both coordinated (shared-seed) and independent sampling of the two instances.

We study the quality of the $L_p^p$ estimates obtained from our $RG_p$ estimators. We estimate $L_p^p = \sum_{h \in H} RG_p(v(h))$ as the sum over selected keys $H$ of $RG_p$ estimates: $\hat{L}_p^p = \sum_{h \in H} \hat{RG}_p(v(h))$. We consider the estimator $\hat{RG}_p^L(L)$ for independent samples (Section 3) and the estimators $\hat{RG}_p^L(L)$ and $\hat{RG}_p^L(U)$ for coordinated samples (Section 4). To apply the estimators, we apply the selection predicate to sampled keys to identify all the ones satisfying the predicate. The estimators are then computed for keys that are sampled in at least one instance (the estimate is 0 for keys that are not sampled in any instance and do not need to be explicitly computed).

Since all our $RG_p$ estimators are unbiased and nonnegative, so is the corresponding sum estimate $\hat{L}_p^p$. The variance is additive and is $\sum_{h \in H} \text{VAR}_{G_k(h)} [\hat{RG}_p]$. We measure the performance of the estimators using the variance normalized by the square of the expectation, which is the squared coefficient of variation $CV^2(\hat{L}_p^p) = \frac{\sum_{h \in H} \text{VAR}_{G_k(h)} [\hat{RG}_p]}{\sum_{h \in H} \hat{RG}_p(v(h))}$. This is the same as the normalized mean squared error (MSE) for our unbiased estimators.

Figure 6 shows the $CV^2$ of our $L_p^p$ estimators ($p = 1, 2$) as a function of the sampled fraction of the dataset. We can see qualitatively, that all estimators, even over independent samples, are satisfactory, in that the CV is small for a sample that is a small fraction of the full data set. The estimator $\hat{RG}_p^L(L)$ over coordinated shared-seed samples outperforms, by orders of magnitude, the estimator $\hat{RG}_p^L(U)$ over independent samples. The gap widens for more aggressive sampling (higher $T$).

On the IP flows and WWW logs data, there are significant differences on the values of keys between instances: the $L_1$ distance is a large fraction of the total sum of values $\sum_{h \in H} \sum_{i \in [2]} t_h(i)$. Therefore, for the destIP and Server selections, $\hat{RG}_p^L(U)$ outperforms $\hat{RG}_p^L(L)$ on shared-seed samples. On the term count data there is typically a small difference between instances. We can see that for the Surnames and OSPD8 selections, $\hat{RG}_p^L(L)$ outperforms $\hat{RG}_p^L(U)$ on shared seed samples. These trends are more pronounced for the Euclidean distance ($p = 2$). In this case, on Surnames and OSPD8, $\hat{RG}_p^L(L)$ over independent samples outperforms $\hat{RG}_p^L(U)$ over shared-seed samples. We can see that we can significantly improve accuracy by tailoring the selection of the estimator to properties of the data. The performance of the $U^*$ estimator, however, can significantly diverge for very similar instances whereas the competitive $L^*$ estimator is guaranteed not to be too far off. Therefore, when there is no prior knowledge on the difference, we suggest using the $L^*$ estimator.

The datasets also differ in the symmetry of change. The change is more symmetric in the IP flows and WWW logs data $L_{p+} \approx L_{p-}$ whereas there is a general growth trend $L_{p+} > L_{p-}$ in the term count data. Estimator performance on one-sided distances (not shown) is similar to the corresponding distance estimators.

8. CONCLUSION

Distance queries are essential for monitoring, planning, and anomaly and change detection. Random sampling is an important tool for retaining the ability to query data under resource limitations. We provide the first satisfactory solution for estimating $L_p$ distance from sampled data sets. Our solution is comprehensive, covering common sampling schemes. It is supported by rigorous analysis and novel techniques. Our estimators scale well with data size and we demonstrated that accurate estimates are obtained for queries with small support size.

Acknowledgement. The author is grateful to Haim Kaplan for many comments, helpful feedback, and suggesting the use of the ngrams data.

<table>
<thead>
<tr>
<th>Condition</th>
<th>$\text{VAR}_{RG_p} [\hat{RG}_p^L(L)]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>S</td>
</tr>
<tr>
<td>$</td>
<td>S</td>
</tr>
</tbody>
</table>

Table 6: $\hat{RG}_p^L(L)$ and variance for shared-seed sampling.

9. REFERENCES


Table 5: Datasets with subset selections. Table shows total number of distinct keys $H$ satisfying selection predicate that had a positive value in at least one of the two instances, corresponding percentage in each instance, total sum of values $\sum_i v_{ih}$, and fraction (shown as percentage), sum in each instance $i = 1, 2$: $\sum_i v_{ih}$ and normalized $L_1, L_{1+}$ and $L_{1-}$ distances.

Table 7: $R_2^{(L)}$ and variance for shared-seed sampling

Figure 6: Queries (left to right): destIP, Server, Surnames, OSPD8. Plot shows CV$^2$ of $L_2^p$ estimate for fraction of sampled items from the query support. Top shows $p = 1 (L_1)$ bottom shows $p = 2 (L_2^2)$.  


Algorithm 1 $\hat{R}_p^{(U)}(S)$

if $|S| = 0$ then return $0 \quad \triangleright \text{from hereafter } |S| > 0$

$m \leftarrow \max_{i \in S} v_i \quad \triangleright \text{if } |S| < r \text{ then } n \leftarrow 0$

else $n \leftarrow \min_{i \in S} v_i \quad \triangleright n = \min(S)$

if $n \geq \tau$ then return $(m - n)^p \quad \triangleright \text{case: } \min(v) \geq \tau$

if $p \leq 1$ then
  if $n=0$ then return $m^p - \frac{x}{\min(m,\tau)^p}$
  else return $\left( \frac{(m-n)^p - \min(m,n)^p}{\min(m,\tau)^p} \right)$

else return $\left( \frac{(m-n)^p - \min(m,n)^p}{\min(m,\tau)^p} \right)$

$m \leftarrow \max_{i \in S} v_i$ if $|S| \geq r$ then $n \leftarrow 0$

else $n \leftarrow \min_{i \in S} v_i \quad \triangleright n = \min(S)$

if $n \geq \tau$ then return $(m - n)^p \quad \triangleright \text{case: } \min(v) \geq \tau$

else return $\left( \frac{(m-n)^p - \min(m,n)^p}{\min(m,\tau)^p} \right)$

\[
\hat{R}_p^{(U)}(\zeta, \psi) = \inf_{0 \leq \eta < \zeta} \int_{\zeta}^{\psi} \left( \frac{\max(v) - \psi}{\zeta - \psi} \right)^p R^{(U)}(u, v) du
\]

\[
R^{(U)}(u, v) = \frac{\max(v) - \psi}{\zeta - \psi} R^{(U)}(u, v) du
\]

\[
\frac{\partial h(\eta)}{\partial \eta} = \frac{(\max(v) - \eta)^{p-1}}{1 - \eta} \left( -p + \frac{\max(v) - \psi}{1 - \psi} \right)
\]

The derivative is $0$ at $\eta = \frac{p \tau - \max(v)}{\tau(p - 1)}$.

\[
\frac{\partial h(\eta)}{\partial \eta} = \left( \frac{\max(v) - \eta)^{p-1}}{1 - \eta} \right)
\]

APPENDIX

A. DERIVATION OF $\hat{R}_p^{(U)}$

If $\zeta > \max(v)$ then $\hat{R}_p^{(U)}(\zeta, \psi) = 0$ and $\hat{R}_p^{(U)}(\zeta, \psi) = 0$. Otherwise, when $\min(v) < \zeta \leq \max(v)$, noting that the supremum is obtained by a vector $v'$ with maximum entry $\max(v)$ and minimum entry $0$,

\[
\hat{R}_p^{(U)}(\zeta, \psi) = \inf_{0 \leq \eta < \zeta} \int_{\zeta}^{\psi} \left( \frac{\max(v) - \psi}{\zeta - \psi} \right)^p R^{(U)}(u, v) du
\]

If $\zeta \leq \min(v)$,

\[
\hat{R}_p^{(U)}(\zeta, \psi) = \frac{\max(v) - \psi}{\zeta - \psi} R^{(U)}(u, v) du
\]

\[
\int_{\zeta}^{\psi} R^{(U)}(u, v) du = R^{(U)}(\zeta, \psi) \text{ and the infimum is the derivative of the lower bound function, and thus, } \hat{R}_p^{(U)}(\zeta, \psi) = p \tau (\max(v) - \zeta)^{p-1}.
\]

Table 8: $\hat{R}_p^{(U)}$ and variance for share-seed sampling.


\[
\nu \left( \min(v) \geq \tau \right) \quad \nu(\max(v) - \tau) \quad \nu(\max(v) - \tau - \min(v))
\]

\[
\nu \left( \min(v) \geq \tau \right) \quad \nu(\max(v) - \tau) \quad \nu(\max(v) - \tau - \min(v))
\]

\[
\nu \left( \min(v) \geq \tau \right) \quad \nu(\max(v) - \tau) \quad \nu(\max(v) - \tau - \min(v))
\]

\[
\nu \left( \min(v) \geq \tau \right) \quad \nu(\max(v) - \tau) \quad \nu(\max(v) - \tau - \min(v))
\]
If \( \eta_0 \) is outside \((0, 1)\), the infimum is obtained at \( \eta = 0 \) and the estimate is 
\[
\hat{\text{RG}}_{(U)}(\zeta, \nu) = \max(\nu)\tau \quad \text{for} \quad \zeta \leq \min(\nu)/\tau.
\]
Otherwise, if \( \eta_0 \in (0, 1) \), the infimum is achieved at \( \eta_0 \). Using (15), the estimate is 
\[
\hat{\text{RG}}_p(\zeta, \nu) = \frac{\tau - \max(\nu)}{\min(\nu)} \left( \frac{\tau}{\min(\nu)} - 1 \right)
\]
for \( \zeta < \min(\nu)/\tau \).

B. VARIANCE OF \( \hat{\text{RG}}_{(U)} \) AND \( \hat{\text{RG}}_p(\zeta, \nu) \)

The estimators \( \hat{\text{RG}}_{(U)} \) and \( \hat{\text{RG}}_p(\zeta, \nu) \), provided in Tables 8 and 9, are obtained by substituting \( p = 1 \) and \( p = 2 \) respectively in Algorithm 1. We calculate the variance of these estimators.

Variance of \( \text{RG}^{(U)} \): When \( \max(\nu) \leq \tau \), we have \( \text{RG}^{(U)} = \tau \) for \( \zeta \in \left( \frac{\min(\nu)}{\tau}, \frac{\max(\nu)}{\tau} \right) \) and \( \text{RG}^{(U)} = 0 \) otherwise. Hence,
\[
\text{VAR}[\hat{\text{RG}}^{(U)}] = (\text{RG}(\nu))^2 \left( \frac{1}{\tau} - \frac{\text{RG}(\nu)}{\tau} \right) + (\tau - \text{RG}(\nu))^2 \frac{\text{RG}(\nu)}{\tau} = \text{RG}(\nu)\tau - \text{RG}(\nu)^2
\]

C. VARIANCE OF \( \hat{\text{RG}}_p(\zeta, \nu) \) AND \( \hat{\text{RG}}_p(\zeta, \nu) \)

The estimators \( \hat{\text{RG}}_p(\zeta, \nu) \) and \( \hat{\text{RG}}_p(\zeta, \nu) \), provided in Tables 6 and 7 are obtained using (10). We calculate their variance.

Variance of \( \text{RG}^{(L)} \): When \( \max(\nu) \leq \tau \), we have \( \text{RG}^{(L)} = \tau \) for \( \zeta \leq \min(\nu)/\tau \) and \( \text{RG}^{(L)} = 0 \) otherwise. Hence,
\[
\text{VAR}[\hat{\text{RG}}^{(L)}] = (\text{RG}(\nu))^2 \left( \frac{1}{\tau} - \frac{\text{RG}(\nu)}{\tau} \right) + (\tau - \text{RG}(\nu))^2 \frac{\text{RG}(\nu)}{\tau} = \text{RG}(\nu)\tau - \text{RG}(\nu)^2
\]
The variance is
\[
\text{VAR}[\hat{R}_G(L)|v] = (1 - \frac{\max(v)}{\tau})\bar{R}_G(v)^2 + \int_{\min(v)}^{\max(v)} (\bar{R}_G(v) - \tau \ln(\frac{\max(v)}{\tau}))^2 dy + \frac{\min(v)}{\tau}(\bar{R}_G(v) - \tau \ln(\frac{\max(v)}{\min(v)}))^2
\]
\[
= -2\tau \min(v) \ln(\frac{\max(v)}{\min(v)}) + 2\bar{R}_G(v)\tau - (\bar{R}_G(v))^2
\]

When \(\min(v) \leq \tau \leq \max(v)\), \(\hat{R}_G(L) = \max(v) - \tau + \tau \ln(\frac{1}{\tau})\), when \(1 \leq |S| \leq r - 1\) and \(\hat{R}_G(L) = 2\bar{R}_G(v)\tau - (\bar{R}_G(v))^2\), when \(|S| = r\). The variance is \(\text{VAR}[\hat{R}_G(L)|v] = (\tau)^2 - \min(v)^2 - 2\tau \min(v) \ln(\frac{\tau}{\min(v)})\).

**Variance of \(\hat{R}_G(L)^2\):**

If \(\min(v) < \tau \leq \max(v)\), \(\hat{R}_G(L)^2 = \max(v)^2 - (\tau)^2 + 2\tau(\bar{R}_G(v) - \max(v) + \max(v) \ln \frac{\max(v)}{\tau})\), when \(|S| \in [r - 1]\) and \(\hat{R}_G(L)^2 = \max(v)^2 - (\tau)^2 + 2\tau(\min(v) - \max(v) + \max(v) \ln \frac{\tau}{\max(v)})\), when \(|S| = r\). The variance is
\[
\text{VAR}[\hat{R}_G(L)^2|v] = 4\max(v) \min(v)\tau(\min(v) - 2\max(v)) \ln(\frac{\tau}{\min(v)})
\]
\[
+ 4\max(v) \min(v)(\tau)^2 + (\tau)^3 - 6\max(v) \min(v)^2\tau - \min(v)^4
\]
\[
- 4\max(v)^2\min(v)^2 + 4\max(v)\min(v)^3 + 4\max(v)^3\tau^2
\]
\[
- 2\max(v)(\tau)^3 + 8\min(v)^3\tau
\]

If \(\max(v) < \tau\), \(\hat{R}_G(L)^2 = 2\tau(\bar{R}_G(v) - \max(v) + \max(v) \ln \frac{\max(v)}{\tau})\), when \(|S| \in [r - 1]\) and \(\hat{R}_G(L)^2 = 2\tau(\min(v) - \max(v) + \max(v) \ln \frac{\max(v)}{\min(v)})\), when \(|S| = r\). The variance is
\[
\text{VAR}[\hat{R}_G(L)^2|v] = -4\tau \max(v) \min(v) \ln(\frac{\max(v)}{\min(v)})(2\max(v) - \min(v)) +
\]
\[
+ 2\tau \frac{3}{5}(5\max(v)^3 - 9\max(v)\min(v)^2 + 4\min(v)^3) - \bar{R}_G^4(v)
\]