Sub-linear Estimation of Entropy and Information Distances

SUDIPTO GUHA
University of Pennsylvania
and
ANDREW MCGREGOR
University of California, San Diego
and
SURESH VENKATASUBRAMANIAN
University of Utah

In many data mining and machine learning problems, the data items that need to be clustered or classified are not arbitrary points in a high-dimensional space, but are distributions, i.e., points on a high-dimensional simplex. For distributions, natural measures are not ℓ_p distances, but information-theoretic measures such as the Kullback-Leibler and Hellinger divergences. Similarly, quantities such as the entropy of a distribution are more natural than frequency moments. Efficient estimation of these quantities is a key component in algorithms for manipulating distributions. Since the data sets involved are typically massive, these algorithms need to have only sub-linear complexity in order to be feasible in practice.

We present a range of sub-linear time algorithms in various oracle models in which the algorithm accesses the data via an oracle that supports various queries. In particular, we answer a question posed by Batu et al. on testing whether two distributions are close in an information-theoretic sense given independent samples. We then present optimal algorithms for estimating various information-divergences and entropy with a more powerful oracle called the combined oracle that was also considered by Batu et al. Finally, we consider sub-linear space algorithms for these quantities in the data-stream model. In the course of doing so, we explore the relationship between the aforementioned oracle models and the data-stream model. This continues work initiated by Feigenbaum et al. An important additional component to the study is considering data streams which are ordered randomly rather than just those which are ordered adversarially.

Categories and Subject Descriptors: F.2.0 [**Analysis of Algorithms and Problem Complexity**]: General; G.3 [**Probability and Statistics**]:

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Data streams, entropy, information divergences, property testing

Authors' addresses: S. Guha, Department of Computer Information Sciences, University of Pennsylvania, 3330 Walnut St, Philadelphia, PA 19104; A. McGregor, Information Theory & Applications Center, University of California, La Jolla, CA 92093; S. Venkatasubramanian, School of Computing, University of Utah, 50 S. Central Campus Drive, Salt Lake City, UT 84112.
S. Guha was supported in part by the Alfred P. Sloan Research Fellowship, NSF Award CCF-0430376. A. McGregor was supported in part by ONR N00014-04-1-0735.

1. INTRODUCTION

There are many settings where the natural unit of data, rather than being a point in a high dimensional vector space, is a distribution defined on a high dimensional simplex. When dealing with distributions, distances arising from information-theoretic considerations are often more natural than distances based on ℓ_p norms. Examples include soft clustering [Tishby et al. 1999], where the membership of a point in a cluster is described by a distribution, and anomaly detection [Krishnamurthy et al. 2005], where the distance between two empirical distributions is used to detect anomalies. Typically, these settings involve large data sets, and so a natural requirement is that we must process these data sets with relatively small space and/or time complexity. In this paper, we examine sub-linear algorithms for estimating properties of distributions.

1.0.0.1 *Entropy and f-Divergences.* We focus on estimating entropy and the Ali-Silvey distances[1], or f-divergences. Entropy was originally introduced by Shannon [Shannon 1948]. It captures the "information content" of a random event. For example, it can be used to lower-bound the compressibility of data and plays a fundamental role in coding and information theory. Recently it has been used in networking applications [Gu et al. 2005; Wagner and Plattner 2005; Xu et al. 2005] where it can be useful when trying to detect anomalous behavior. The f-divergences were discovered independently by Csiszár [Csiszár 1991], and Ali and Silvey [Ali and Silvey 1966]. We start with the necessary definitions.

Definition 1.1 *Entropy and f-Divergences.* Let p and q be two discrete probability distributions defined on base $[n]$. Any convex function f defined on $(0, \infty)$ such that $f(1) = 0$ gives rise to an *f-divergence*,

$$D_f(p, q) = \sum p_i f(q_i/p_i) \ ,$$

where $f(0) = \lim_{t\to 0} f(t)$, $0f(0/0) = 0$, and $0f(a/0) = a \lim_{u\to\infty} f(u)/u$. The *entropy* of a distribution is defined[2] as

$$H(p) = \sum_i -p_i \lg p_i \ .$$

The class of f-divergences includes many commonly used information-theoretic distances, e.g., the (asymmetric) Kullback-Liebler (KL) divergence and its symmetrization, the Jensen-Shannon (JS) divergence; Matsusita's divergence or the squared Hellinger divergence; the (asymmetric) χ^2 divergence and its symmetrization, the Triangle divergence. See Table I. The entropy of a distribution is closely related to the f-divergences. For example,

$$\begin{aligned} \mathrm{JS}(p, q) &= \ln 2 \left(2H\left(\frac{p+q}{2}\right) - H(p) - H(q) \right) \\ \mathrm{KL}(p, u) &= \ln 2 \left(\lg n - H(p) \right) \end{aligned}$$

[1] Many of the measures we consider in this paper are not metrics. Traditionally, the term *divergence* is used to refer to "distances" that are not metric.

[2] Here and throughout we use $\lg x$ to denote $\log_2 x$.

ℓ_1 distance:	$f(u) = \lvert 1 - u \rvert$
Kullback-Liebler (KL) divergence:	$f(u) = u \ln u$
Jensen-Shannon (JS) divergence:	$f(u) = \ln(2/(1+u)) + u \ln(2u/(1+u))$
Hellinger divergence:	$f(u) = (\sqrt{u} - 1)^2$
χ^2 divergence:	$f(u) = (u-1)^2$
Triangle divergence:	$f(u) = (u-1)^2/(u+1)$.

Table I. Commonly used f-divergences where $D_f(p,q) = \sum p_i f(q_i/p_i)$

where u is the uniform distribution and $(p+q)/2$ is the distribution whose i-th component is $(p_i + q_i)/2$.

Results of Csiszár [Csiszár 1991], Liese and Vajda [Liese and Vajda 1987], and Amari [Amari 1985] show that f-divergences are the unique class of distances on distributions that arise from a fairly simple set of axioms, e.g., symmetry, non-decreasing projections, certain direct sum theorems etc., in much the same way that ℓ_2 is a natural measure for points in $\mathbb{R}^n$. Moreover, all of these distances are related to each other (via the Fisher information matrix) [Čencov 1981] in a way that other plausible measures (most notably ℓ_2) are not. In addition, the log-likelihood ratio $\ln(q(x)/p(x))$ is a crucial parameter in Neyman-Pearson style hypothesis testing [Cover and Thomas 1991], and distances based on this, e.g., the KL-divergence and JS-divergence, appear as exponents of error probabilities for optimal classifiers.

Recently, these distance measures have been used in more algorithmic contexts, such as natural distances for clustering distributional data [Tishby et al. 1999; Dhillon et al. 2003; Banerjee et al. 2005]. Batu et al. [Batu et al. 2000] gave algorithms for testing closeness of distributions for the ℓ_1 and ℓ_2 distances, and raised the question of testing closeness of distributions under the JS-divergence. They state that they suspect that this is "the most powerful" notion of closeness.

1.0.0.2 *Oracle Models and the Data-Stream Model.* When processing massive amounts of data, it is desirable to use algorithms whose space and/or time complexity is sub-linear in the size of the data set. Two models that have gained significant currency in this context are the oracle model [Kearns et al. 1994; Batu et al. 2005] and the data-stream model [Henzinger et al. 1999; Alon et al. 1999; Feigenbaum et al. 2002a]. In the former, the algorithm accesses the data via an oracle that supports various queries. The query types most commonly considered include returning a sample from the underlying distribution of the data or "probing" a portion of the data. In general, the oracle models are suited for designing sub-linear time algorithms that do not look at the entire data but rather inspect only a few regions of the data in an attempt to test various properties of the data. In contrast, in the data-stream model, all the data is inspected sequentially but the algorithm is only permitted limited space to remember the data that has been seen. In addition, the order in which data is inspected is fixed. We consider estimating entropy and f-divergences in the oracle model and the data-stream model.

A natural question that arises is the relationship of these two models. Feigenbaum

et al. [Feigenbaum et al. 2002b] initiated the study of this problem. They considered testing properties of a length m list of values. They consider processing the list in the data-stream model and in an oracle model in which queries can be made to the value contained in each position of the list. They showed that there exist functions that are easy in their oracle model but hard to test in the data-stream model and vice versa. In particular, they show that testing SORTED-SUPERSET, the property that the first half of the stream (presented in sorted order) contains all the elements of the second half or does not contain at least an ϵ fraction of the elements in the second half; there is an algorithm that uses $O(\log m)$ queries in the oracle model but any single pass streaming algorithm requires $\Omega(m)$ space. Conversely, testing GROUPEDNESS, the property that all identical values in the list appear consecutively, requires $\Omega(\sqrt{m})$ queries in the oracle while it only requires $O(\log m)$ space in the data-stream model.

Given such a result it may appear that the problem of relating oracle models to data-stream models is resolved. However, most of the properties considered by [Feigenbaum et al. 2002b] are dependent on the ordering of the data stream. Properties of the empirical distribution defined by a stream, are invariant under re-ordering of the stream. Furthermore, many of these properties are also invariant under re-labeling the values in the stream. Such properties include the entropy of the data stream or the f-divergence between two empirical distributions. For these properties and others, is it possible to relate various oracle models to the data-stream model?

1.0.0.3 *Our Results and Organization.* In Section 2, we formally define the relevant computational models. In Section 3, we present an algorithm in the generative oracle model for testing if two distributions are close in terms of a range of f-divergences. This answers a question posed by Batu et al. [Batu et al. 2000]. In Section 4, we present an algorithm in the combined oracle model for approximating the f-divergence between two distributions for all bounded f-divergences. We also prove a matching lower bound. In Section 5, we present an algorithm in the combined oracle model for approximating the entropy of a distribution. This improves upon the algorithm in Batu et al. [Batu et al. 2005] and matches a lower bound proved in the same paper.

We then prove the main model-theoretic result of this paper. This relates the computational power of the oracles models to the data-stream models and addresses a question posed by Feigenbaum et al. [Feigenbaum et al. 2002b]. When combined with the algorithmic results in Section 4 and Section 5, this result gives rise to algorithms for approximating f-divergences and entropy in the data-stream model.

Note that this paper is based on work that originally appeared in [Guha et al. 2006] and in that paper there were further results for approximating entropy in the data-stream model including at asymptotic $(e/(e-1)+\epsilon)$-approximation algorithm when the stream is adversarially ordered and a $(1+\epsilon)$-approximation when the stream is randomly ordered. Both algorithms used $\text{poly}(\epsilon^{-1}, \log n, \log m)$ space (where m is the length of the stream). However, these results have been subsumed by a sequence of results [Chakrabarti et al. 2006; Lall et al. 2006; Bhuvanagiri and Ganguly 2006] culminating in an $(1+\epsilon)$-approximation for adversarially ordered streams that uses $O(\epsilon^{-2}\log m)$ space [Chakrabarti et al. 2007].

1.0.0.4 *Notation.* We denote $[n] := \{1, 2, \ldots, n\}$ and write $x \in_R S$ to mean that x is a value chosen uniformly from the set (or multi-set) S. A randomized algorithm is said to,

(1) (ϵ, δ)-approximate a real number Q if it outputs a value $\hat{Q}$ such that $|\hat{Q} - Q| \leq \epsilon Q$ with probability at least $(1 - \delta)$ over its internal coin tosses.
(2) (ϵ, δ)-additively-approximate a real number Q if it outputs a value $\hat{Q}$ such that $|\hat{Q} - Q| \leq \epsilon$ with probability at least $(1 - \delta)$ over its internal coin tosses.
(3) $(\epsilon_1, \epsilon_2, \delta)$-test a real number Q if it outputs FAIL if $Q \geq \epsilon_1$ and outputs PASS if $Q \leq \epsilon_2$ with probability at least $(1 - \delta)$ over its internal coin tosses.

2. MODELS

2.0.0.5 *The Oracle Models.* Two main oracle models have been used in the property testing literature for testing properties of distributions. These are the *generative* and *evaluative* oracle models introduced by Kearns et al. [Kearns et al. 1994]. In the generative oracle model, the oracle only supports the request to sample from the distribution. Specifically, for a distribution $p = \{p_1, \ldots p_n\}$ that is known to the oracle, `sample`(p) returns i with probability p_i. In the evaluative oracle model, a `probe` operation is supported. `probe`(p, i) returns p_i. A natural third model, the *combined* oracle model was introduced by Batu et al. [Batu et al. 2005]. In this model both the `sample` and `probe` operations are supported. In all three models, the complexity of an algorithm is measured by the number of operations made by the algorithm.

2.0.0.6 *The Data-Stream Model.* In the data-stream model, a *stream* of data items may be accessed sequentially. Any algorithm processing this stream has a limited amount of working memory. This is always sub-linear in the length of the stream but typically we require that the amount of working memory is only poly-logarithmic in the length of the data stream. We are primarily in designing algorithms that only make a single pass over the stream but sometimes, when this is not sufficient, we allow an algorithm to take multiple passes over the stream.

We will consider both streams in which the data items are adversarially ordered and streams in which the data items arrive in a random order. The former is the more usual assumption in the data-stream literature but the later actually dates back to one of the seminal streaming papers by Munro and Paterson [Munro and Paterson 1980]. It was also considered in more recent work by Demaine et al. [Demaine et al. 2002] and Guha and McGregor [Guha and McGregor 2006; 2007a; 2007b].

We consider streams that give rise to a distribution, or distributions, in the following way.

Definition 2.1 *Empirical Distributions.* For a data stream $S = \langle a_1, \ldots, a_m \rangle$ with each data item $a_i \in \{p, q\} \times [n]$ we define two *empirical distributions* p and q as follows. Let $m(p)_i = |\{j : a_j = \langle p, i \rangle\}|$, $m(p) = |\{j : a_j = \langle p, \cdot \rangle\}|$ and $p_i = m(p)_i / m(p)$. Similarly for q. We assume that $m(p) = \Theta(m(q))$.

Note that this model essentially captures the model in which a single data item can encode multiple (p, i) or (q, i) pairs (for some i). This model has been referred

Algorithm *Hellinger-Test*(m, α, ϵ)

1. $m_i^p, m_i^q \leftarrow 0$ for all $i \in [n]$
2. **for** $t = 1$ to m:
3. **do** $i \leftarrow$ `sample`(p) and $m_i^p \leftarrow m_i^p + 1$
4. $i \leftarrow$ `sample`(q) and $m_i^q \leftarrow m_i^q + 1$
5. **return** `FAIL` if

$$\sum_{i \in S} \left(\sqrt{m_i^p/m} - \sqrt{m_i^q/m} \right)^2 > \epsilon/10$$

where $S = \{i : \max\{m_i^p, m_i^q\} \geq mn^{-\alpha}\}$

6. **return** ℓ_2-Tester$(p', q', \epsilon n^{-1/2}/(2\sqrt{2}))$ where p' (and q' analogously) is the distribution formed by the following sampling procedure:

$$i \leftarrow \texttt{sample}(p)$$

$$\texttt{sample}(p') \leftarrow (i \text{ if } i \notin S \text{ and } j \in_R [2n] \setminus [n] \text{ otherwise})$$

Fig. 1. Hellinger-Testing in the Generative Oracle Model

to as the *cash-register* model [Gilbert et al. 2001]. It is a generalization of the *aggregate* model, in which all updates to a given i are grouped together in the ordering of the stream.

3. f-DIVERGENCE TESTING (GENERATIVE ORACLE)

In this section we consider property testing in the generative model for various information theoretic distances. We will present the results for the Hellinger distance. However, the Jensen-Shannon and triangle divergences are constant factor related to the Hellinger distance as follows:

$$\frac{\text{Hellinger}(p,q)}{2} \leq \frac{\Delta(p,q)}{2} \leq \text{JS}(p,q) \leq \ln(2) \cdot \Delta(p,q) \leq 2\ln(2) \cdot \text{Hellinger}(p,q) \ . \quad (1)$$

The second and third inequalities in Eqn. 1 are proved in [Topsøe 2000] and the other two inequalities can be proved using the AM-GM inequality on each term in the sum. Therefore the results presented here naturally imply analogous results for them as well. Our algorithm is presented in Figure 1. In common with the ℓ_1 tester presented in [Batu et al. 2000], our algorithm relies on an ℓ_2 tester as an important sub-routine. Central to the analysis are the following inequalities.

$$\frac{\ell_2^2(p,q)}{2(\ell_\infty(p) + \ell_\infty(q))} \leq \text{Hellinger}(p,q) \leq \sqrt{n}\ell_2(p,q) \ . \quad (2)$$

LEMMA 3.1 ℓ_2 TESTING [BATU ET AL. 2000]. *There exists an* $(\epsilon, \epsilon/2, \delta)$*-tester for* $\ell_2(p,q)$ *using*

$$O(\epsilon^{-4}(b^2 + \epsilon^2\sqrt{b})\log(\delta^{-1}))$$

samples where $b = \max(\ell_\infty(p), \ell_\infty(q))$.

We first prove two preliminary lemmas. Throughout we assume that ϵ is sufficiently small and that n is sufficiently large.

LEMMA 3.2. *Define* $\tilde{p}_i = m_i^p/m$ *and* $\tilde{q}_i = m_i^q/m$. *With* $m = O(\epsilon^{-4} n^\alpha \log(n\delta^{-1}))$ *samples, with probability* $1 - \delta/2$, *the following two conditions hold:*

$$\begin{aligned} &\forall i \notin S, \quad p_i, q_i \leq 2n^{-\alpha} \\ &\forall i \in S, \quad |(\sqrt{p_i} - \sqrt{q_i})^2 - (\sqrt{\tilde{p}_i} - \sqrt{\tilde{q}_i})^2| \leq \epsilon \max\{p_i, q_i\}/100 \ . \end{aligned}$$

PROOF. By applying Chernoff-Hoeffding bounds it is straight-forward to show that with probability at least $1 - \delta/2$,

$$\forall i \in [n], \ |\tilde{p}_i - p_i| \leq \epsilon \max\{p_i, \epsilon^2 n^{-\alpha}\}/1000 \text{ and } |\tilde{q}_i - q_i| \leq \epsilon \max\{q_i, \epsilon^2 n^{-\alpha}\}/1000 \ .$$

Therefore $i \notin S$ implies that $p_i, q_i \leq 2n^{-\alpha}$ as required. Also, if $i \in S$ and $p_i > q_i$ then $p_i \geq n^{-\alpha}/2$ and hence $|\tilde{p}_i - p_i| \leq (\epsilon/1000)p_i$. Let $i \in S$. Without loss of generality assume that $p_i \geq q_i$. Therefore,

$$\begin{aligned} |(\sqrt{\tilde{p}_i} - \sqrt{\tilde{q}_i})^2 - (\sqrt{p_i} - \sqrt{q_i})^2| \ &\leq \ |\tilde{p}_i - p_i| + |\tilde{q}_i - q_i| + 2|\sqrt{\tilde{p}_i \tilde{q}_i} - \sqrt{p_i q_i}| \\ &\leq \ \epsilon p_i/500 + 2|\sqrt{\tilde{p}_i \tilde{q}_i} - \sqrt{p_i q_i}| \ . \end{aligned}$$

First assume that $q_i \geq \epsilon^2 n^{-\alpha}$. Therefore,

$$2|\sqrt{\tilde{p}_i \tilde{q}_i} - \sqrt{p_i q_i}| \leq 2\sqrt{p_i q_i}|\sqrt{\tilde{p}_i \tilde{q}_i/(p_i q_i)} - 1| \leq 2\epsilon p_i/500 \ .$$

Alternatively, if $q_i \leq \epsilon^2 n^{-\alpha}$ then,

$$\begin{aligned} 2|\sqrt{\tilde{p}_i \tilde{q}_i} - \sqrt{p_i q_i}| \ &\leq \ 2\sqrt{p_i}|\sqrt{(1 - \epsilon/1000)(q_i - \epsilon^3 n^{-\alpha}/1000)} - \sqrt{q_i}| \\ &\leq \ 2\sqrt{p_i}\sqrt{\epsilon^3 n^{-\alpha}/500)} \\ &\leq \ 2\epsilon p_i/\sqrt{250\epsilon^{-1}} \ , \end{aligned}$$

where the second inequality follows because $x^{1/2}$ is a concave function. In either case, for sufficiently small ϵ, $|(\sqrt{\tilde{p}_i} - \sqrt{\tilde{q}_i})^2 - (\sqrt{p_i} - \sqrt{q_i})^2| \leq \epsilon p_i/100$ as required. □

LEMMA 3.3. *Let* p *and* q *be two distributions on* $[n]$ *and let* $S \subset [n]$. *Define a distribution* p',

$$p'_i \ = \ \begin{cases} p_i & \text{if } i \in [n] \setminus S \\ 0 & \text{if } i \in S \\ (\sum_{j \in S} p_j)/n & \text{if } i \in [2n] \setminus [n] \end{cases} \ .$$

Let q' *be defined analogously. Then,*

$$\sum_{i \notin S} (\sqrt{p_i} - \sqrt{q_i})^2 \leq \textit{Hellinger}(p', q') \leq \textit{Hellinger}(p, q) \ .$$

PROOF. The first inequality is immediate because of term-by-term dominance. To bound the second term we need to show that,

$$\sum_{i \in S} (\sqrt{p_i} - \sqrt{q_i})^2 \geq n \left(\sqrt{\frac{\sum_{i \in S} p_i}{n}} - \sqrt{\frac{\sum_{i \in S} q_i}{n}} \right)^2 \ = \left(\sqrt{\sum_{i \in S} p_i} - \sqrt{\sum_{i \in S} q_i} \right)^2 \ .$$

We will first show that $(\sqrt{p_i} - \sqrt{q_i})^2 + (\sqrt{p_j} - \sqrt{q_j})^2 \geq (\sqrt{p_i + p_j} - \sqrt{q_i + q_j})^2$.

This is because,

$$\begin{aligned}
& (\sqrt{p_j q_i} - \sqrt{p_i q_j})^2 && \geq\ 0 \\
\Rightarrow\quad & p_j q_i + p_i q_j && \geq\ 2\sqrt{p_i p_j q_i q_j} \\
\Rightarrow\quad & (p_i + p_j)(q_i + q_j) && \geq\ p_i q_i + p_j q_j + 2\sqrt{p_i p_j q_i q_j} \\
\Rightarrow\quad & 2\sqrt{(p_i + p_j)(q_i + q_j)} && \geq\ 2\sqrt{p_i q_i} + 2\sqrt{p_j q_j} \\
\Rightarrow\quad & (\sqrt{p_i} - \sqrt{q_i})^2 + (\sqrt{p_j} - \sqrt{q_j})^2 && \geq\ (\sqrt{p_i + p_j} - \sqrt{q_i + q_j})^2 \ .
\end{aligned}$$

Therefore, by "merging" the probability mass on all indices in S we decrease the Hellinger distance as required. □

We are now ready to prove the main theorem of this section.

THEOREM 3.4 HELLINGER TESTING. *There exists an* $(\epsilon, \epsilon^2/(256n^{1-\alpha}), \delta)$*-tester for Hellinger*$(p, q)$ *with sample complexity*

$$O(\log\left(\delta^{-1}\right) \max\{\epsilon^{-2} n^{\alpha} \log n, \epsilon^{-4}(n^{-2\alpha+2} + \epsilon^2 n^{1-\alpha/2})\}) \ .$$

Observe that setting $\alpha = 2/3$ *yields an algorithm with sample complexity*

$$O(\epsilon^{-4} n^{2/3} \log n \log \delta^{-1}) \ .$$

PROOF. Let $A = \sum_{i \in S} (\sqrt{p_i} - \sqrt{q_i})^2$ and $B = \sum_{i \notin S} (\sqrt{p_i} - \sqrt{q_i})^2$. By Lemma 3.2, we estimate A with an additive error of at most $(\epsilon/100) \sum_i (p_i + q_i) = \epsilon/50$.

(1) If Hellinger$(p, q) > \epsilon$ then either A is bigger than $\epsilon/2$ or B is bigger than $\epsilon/2$. If A is bigger than $\epsilon/2$ then our estimate of A is bigger than $\epsilon(1/2 - 1/50)$ and thus $\sum_{i \in S} (\sqrt{\tilde{p}_i} - \sqrt{\tilde{q}_i})^2 > \epsilon/10$ and we fail. Otherwise if B is bigger than $\epsilon/2$. Therefore, appealing to Eq. 2 and Lemma 3.3 (note that p' and q' are on base $[2n]$) we deduce that,

$$\epsilon/2 \leq \text{Hellinger}(p', q') \leq \sqrt{2n}\ell_2(p', q') \ .$$

Hence $\ell_2(p', q') \geq \epsilon n^{-1/2}/(2\sqrt{2})$. Consequently the ℓ_2 test fails.

(2) If Hellinger$(p, q) < \epsilon^2/(256n^{1-\alpha})$ then $A < \epsilon^2/(256n^{1-\alpha})$ and we pass the first test because our estimate of A is at most $\epsilon^2/(256n^{1-\alpha}) + \epsilon/50 < \epsilon/10$ (for sufficiently large n.) By Lemma 3.2, $\max(\ell_\infty(p'), \ell_\infty(q')) \leq 2n^{-\alpha}$. Therefore, appealing to Eq. 2 and Lemma 3.3,

$$n^{\alpha} \ell_2^2(p', q')/8 \leq \text{Hellinger}(p', q') \leq A + B < \epsilon^2/(256n^{1-\alpha})$$

implies that the second test passes since $n^{\alpha}\ell_2^2(p', q') \leq \epsilon^2/(32n^{1-\alpha})$ and thus $\ell_2(p', q') \leq \frac{\epsilon}{4\sqrt{2n}}$.

The sample complexity follows from Lemma 3.1 and Lemma 3.2. □

Batu et al. [Batu et al. 2000] discuss lower bounds for ℓ_1 property testing. Their arguments consider distributions such that either $p_i = q_i$ or one of p_i, q_i is 0. For these distributions $\Delta(p, q) = \ell_1(p, q) = \text{Hellinger}(p, q) = \text{JS}(p, q)$. There arguments suggest that $O(n^{2/3})$ samples are necessary to test if such distributions are close.

4. f-DIVERGENCE TESTING (COMBINED ORACLE)

In this section we consider property testing in the combined oracle model for all bounded f-divergences. Define a *conjugate* $f^*(u) = uf(\frac{1}{u})$. We can write any f-Divergence as,

$$D_f(p,q) = \sum_{i:p_i>q_i} p_i f(q_i/p_i) + \sum_{i:q_i>p_i} q_i f^*(p_i/q_i) \ .$$

We start with the following preliminary lemma that demonstrates that we may assume that $f(u) \in [0, f(0)]$ and $f^*(u) \in [0, f^*(0)]$ for $u \in [0,1]$ where both $f(0) = \lim_{u\to 0} f(u)$ and $f^*(0) = \lim_{u\to 0} f^*(u)$ exist and are positive if f is bounded.

LEMMA 4.1. *For any bounded D_f, there exists another f-divergence D_g such that,*

(1) $D_f(p,q) = D_g(p,q)$ for all distributions p and q.
(2) $g(0) = \lim_{u\to 0} g(u)$ exists and $0 \le g(u) \le g(0)$ for all $u \in (0,1]$.
(3) $g^*(0) = \lim_{u\to 0} g^*(u)$ exists and $0 \le g^*(u) \le g^*(0)$ for all $u \in (0,1]$.

PROOF. Note that D_f bounded implies that $f(0) = \lim_{u\to 0} f(u)$ exists. Otherwise

$$D_f((1/2,1/2),(0,1)) = 1/2(f(0)+f(2))$$

would not be finite. Similarly $f^*(0) = \lim_{u\to 0} f^*(u) = \lim_{u\to 0} uf(1/u)$ exists because otherwise

$$D_f((0,1),(1/2,1/2)) = 0.5 \lim_{u\to 0} uf(1/u) + f(1/2)$$

would not be finite. Let $c = -\lim_{u\to 1^-} f(u)/(1-u)$. This limit exists because f is convex and defined on $(0,\infty)$. Note that $c = f'(1)$ if the derivative exists at 1. Then $g(u) = f(u) - c(u-1)$ satisfies the necessary conditions because $g(1) = f(1) = 0$, g is convex, and

$$\lim_{u\to 1^-} \frac{g(1)-g(u)}{1-u} = \lim_{u\to 1^-} \left(c - \frac{f(u)}{1-u} \right) = 0 \ .$$

□

Although the above may appear simple, it actually allows us to break the divergence into small, positive components. This allows us to use sharp concentration bounds.

THEOREM 4.2. *There exists an (ϵ,δ)-approximation algorithm for any τ-bounded D_f in the combined oracle model making $O(\tau\epsilon^{-2}\log(\delta^{-1})/D_f(p,q))$ queries.*

PROOF. Consider the value $(a+b)/(2\tau)$ added to E in each iteration. This is a random variable with range $[0,1]$ and mean $D_f(p,q)/(2\tau)$. By applying the Chernoff-Hoeffding bounds,

$$\Pr\left[|E - m\frac{D_f(p,q)}{2\tau}| < \epsilon m \frac{D_f(p,q)}{2\tau} \right] \le 2e^{-\epsilon^2 D_f(p,q)m/6\tau} \le 1-\delta \ .$$

Therefore E is an (ϵ,δ)-approximation for $mD_f(p,q)/2\tau$. Hence, $2\tau E/m$ is an (ϵ,δ)-approximation for $D_f(p,q)$ as required. □

Algorithm *Combined Oracle Distance Testing*(m, τ)
1. $E \leftarrow 0$
2. **for** $t = 1$ to m:
3. **do** $i \leftarrow$ `sample`(p) and $x =$ `probe`(q, i)/`probe`(p, i)
4. If $x > 1$ then $a \leftarrow f(x)$ else $a \leftarrow 0$
5. $j \leftarrow$ `sample`(q) and $x =$ `probe`(q, j)/`probe`(p, j)
6. If $x < 1$ then $b \leftarrow f^*(1/x)$ else $b \leftarrow 0$
7. $E \leftarrow (a + b)/2\tau + E$
8. **return** $2\tau E/m$

Fig. 2. Divergence-Testing in the Combined Oracle Model

We now prove a corresponding lower bound that shows that our algorithm is tight. Note that while it is relatively simple to see that there exists two distributions that are indistinguishable with less than $o(1/\ell_1)$ oracle queries, it requires some work to also show a lower bound with a dependence on ϵ. Further note that the proof below also gives analogous results for JS, Hellinger and Δ.

THEOREM 4.3 ℓ_1 LOWER BOUND. *Any $(\epsilon, 1/4)$-approximation algorithm of ℓ_1 in the combined oracle model requires $\Omega(\epsilon^{-2}/\ell_1)$ queries.*

PROOF. Let p and q^r be the distributions on $[n]$ described by the following two probability vectors:

$$p = (1 - 3a/2, \overbrace{3a\epsilon/2k, \ldots, 3a\epsilon/2k}^{k/\epsilon}, 0, \ldots, 0)$$

$$q^r = (1 - 3a/2, \overbrace{0, \ldots 0}^{r}, \overbrace{3a\epsilon/2k, \ldots, 3a\epsilon/2k}^{k/\epsilon}, 0, \ldots, 0)$$

Let $r_1 = k\epsilon^{-1}/3$ and $r_2 = k\epsilon^{-1}/3 + k$. Then $\ell_1(p, q^{r_1}) = a$ and $\ell_1(p, q^{r_2}) = a(1 + 3\epsilon)$. Hence to $1 + \epsilon$ approximate the distance between p and q^r we need to distinguish between the cases when $r = r_1$ and $r = r_2$. Consider the distributions p' and $q^{r\prime}$ formed by arbitrarily permuting the base sets of the p and q^r. Trivially, $\ell_1(p', q^{r\prime}) = \ell_1(p, q^r)$. We will show that, without knowledge of the permutation, it is impossible to estimate this distance with $o(1/(\epsilon^2 a))$ oracle queries. We reason this by first disregarding the value of any "blind probes", i.e., a probe `probe`(p', i) or `probe`(q', i) for any i that has not been returned as a sample. This is the case because, by choosing $n \gg k/(a\epsilon^2)$ we ensure that, with arbitrarily high probability, for any $o(1/(\epsilon^2 a))$ set of i's chosen from any $n - o(1/(a\epsilon^2))$ sized subset of $[n]$, ${p_i}' = {q_i^r}' = 0$. This is the case for both r_1 and r_2. Let $I = \{i : p_i \text{ or } q_i^r = 3a\epsilon/(2k)\}$ and $I_1 = \{i \in I : p_i \neq q_i^r\}$. Therefore determining whether $r = r_1$ or r_2 is equivalent to determining whether $|I_1|/|I| = 1/2$ or $1/2 + \frac{9\epsilon}{8+6\epsilon}$. We may assume that every time an algorithm sees i returned by `sample`(p) or `sample`(q), it learns the exact values of p_i and q_i for free. Furthermore, by making k large ($k = \omega(1/\epsilon^3)$ suffices) we can ensure that no two `sample` oracle queries will ever return the same $i \in I$ (with high probability.) Hence distinguishing between $|I_1|/|I| = 1/2$ and $1/2 + \frac{9\epsilon}{8+6\epsilon}$ is analogous to distinguishing between a fair coin and a $\frac{9\epsilon}{8+6\epsilon} = \Theta(\epsilon)$ biased coin. It is well known that the latter requires $\Omega(1/\epsilon^2)$ samples. Unfortunately only $O(1/a)$ samples return an $i \in I$ since with probability $1 - 3a/2$ we output an $i \notin I$ when

Algorithm *Combined Oracle Entropy Testing*(m)
1. $E \leftarrow 0$
2. **for** $t = 1$ to m:
3. **do** $i \leftarrow \texttt{sample}(p)$
4. $p_i \leftarrow \texttt{probe}(p, i)$
5. if $p_i \geq n^{-3}$ then $a \leftarrow \lg(1/p_i)/(3 \lg n)$ else $a \leftarrow 0$
6. $E \leftarrow a + E$
7. **return** $3E \lg n/m$

Fig. 3. Entropy-Testing in the Combined Oracle Model

sampling either p or q. The bound follows. □

5. ENTROPY TESTING (COMBINED ORACLE)

In this subsection, we present a simple algorithm that achieves the optimal bounds for estimating the entropy in the combined oracle model. Note that this algorithm improves upon the previous upper bound of Batu et al. [Batu et al. 2005] by a factor of $\Omega(\log n/H)$ where H is the entropy of the distribution. The authors of [Batu et al. 2005] showed that their algorithms were tight for $H = \Omega(\log n)$; we show that the upper and lower bounds match for arbitrary H. The algorithm is presented in Figure 3. It is structurally similar to the algorithm given in [Batu et al. 2005] but uses a cut-off that will allow for a tighter analysis via Chernoff bounds.

The next lemma estimates the contribution of the unseen elements and that leads to the main theorem about estimating entropy in the combined oracle model.

LEMMA 5.1. *For any* $S \subset [n]$, $\sum_{i \in S} p_i \lg 1/p_i \leq \lg(n/\sum_{i \in S} p_i) \sum_{i \in S} p_i$.

THEOREM 5.2. *There exists an* (ϵ, δ)*-approximation algorithm for* $H(p)$ *in the combined oracle model making* $O(\epsilon^{-2} H^{-1} \log(n) \log(\delta^{-1}))$ *queries.*

PROOF. We restrict our attention to the case when $H(p) > 1/n$ and $\epsilon > 1/\sqrt{n}$ since otherwise we can trivially find the entropy exactly in $O(\epsilon^{-2} H^{-1})$ time by simply probing each of the n p_i's. Consider the value a added to E in each iteration. This is a random variable with range $[0, 1]$ since $p_i \geq 1/n^3$ guarantees that $-\lg(1/p_i)/(3 \lg n) \leq 1$. Now, the combined mass of all p_i such that $p_i < 1/n^3$ is at most $1/n^2$. Therefore, by Lemma 5.1, the maximum contribution to the entropy from such i is less than $n^{-2} \lg n^3 \leq n^{-1/2} n^{-1}/3 \leq \epsilon H(p)/3$ for sufficiently large n. Hence the expected value of a is between $(1 - \epsilon/3) H(p)/(3 \lg n)$ and $H(p)/(3 \lg n)$ and therefore, if we can $1 + \epsilon/2$ approximate $E[a]$ then we are done. By applying the Chernoff-Hoeffding bounds,

$$\Pr[|E - mE[a]| < (\epsilon/2) mE[a]] \leq 2e^{-(\epsilon/2)^2 mE[a]/3} .$$

Therefore with $O(1/(\epsilon^2 E[a]) \log(\delta^{-1})) = O(\epsilon^{-2} H^{-1} \log(n) \log(\delta^{-1}))$ samples/probes the probability that we do not $1 + \epsilon/2$ approximate $E[a]$ is at most δ. □

6. ALGORITHMS IN THE DATA-STREAM MODEL

In this section we relate the computational power of the oracle models to the data-stream model. In the process of doing so, we present data stream algorithms for approximating entropy and f-divergences.

6.1 Relating Oracle Models to the Data-Stream Model

We direct the reader to [Bar-Yossef 2002] for a detailed treatment of the relative computational power of the data stream and generative sampling models. Here we restrict ourselves to comparing the combined oracle model to the data-stream model. Specifically we consider a combined oracle that "knows" the empirical distribution defined by the stream. We will show how to emulate any combined-oracle algorithm for a *symmetric* function in the data-stream model.

Definition 6.1. We say that a function $f : \mathbb{R}^n \to \mathbb{R}$ is *symmetric*, if for all $p_1, \ldots, p_n \in \mathbb{R}, i, j \in [n]$,

$$\begin{aligned} & f(p_1, \ldots, p_{i-1}, p_i, p_{i+1}, \ldots, p_{j-1}, p_j, p_{j+1}, \ldots, p_n) \\ = \; & f(p_1, \ldots, p_{i-1}, p_j, p_{i+1}, \ldots, p_{j-1}, p_i, p_{j+1}, \ldots, p_n) \ . \end{aligned}$$

Symmetry is a desirable and often-assumed property of functions on distributions. It is a special case of general invariance under coordinate re-parametrizations [Čencov 1981].

First we will show that we can always express an algorithm for a symmetric function in the combined oracle model in a canonical form where the algorithm first samples and then probes the samples along with a few other elements. The idea would be to view the original algorithm, after the sampling stages and probing of the samples, as a randomized decision tree that we rewrite as an oblivious decision tree along the lines of Bar-Yossef et al. [Bar-Yossef et al. 2001; Bar-Yossef 2002]. We start with the necessary definitions.

Definition 6.2 *Decision Trees.* A *randomized decision tree* for a function f is a decision tree having three types of nodes: a *query* node that asks for the value of an input parameter and maps the resulting value to a choice of child node to visit; a *random choice* node, where the child node is chosen at random; and an *output* node, where an answer is expressed as a function of all queries thus far is returned.

An *oblivious decision tree* is one where the queries are made independently of the input, or the random choices in the algorithm. Formally, suppose we have a tree T with worst-case query complexity u. Then an *I-relabeling* of T by $I = \{i_1, \ldots i_u\}$ relabels all query nodes of depth j by the query to i_j, yielding the tree T^I. An oblivious decision tree is then a pair T, Δ_u, where T is a decision tree with worst-case complexity u and Δ_u is a distribution on $[n]^u$. A computation on an oblivious decision tree consists of two steps: (1) sample u elements I from Δ_q, (2) Relabel T to T^I and run it on input x.

An important step in our argument will be the transition between using a randomized decision tree and an oblivious decision tree. We will do this using the following result due to Bar-Yossef [Bar-Yossef 2002].

Lemma 6.3 [Bar-Yossef 2002, Lemma 4.17]. *Let T be a randomized decision tree that computes an (γ, δ)-approximation to a symmetric function f with u queries in the worst case and u_E queries in the expected case with the expectation taken over the random choices used by T. Then, there is an oblivious decision tree (T, W_u) of worst-case query complexity u and expected query complexity u_E that computes an (γ, δ)-approximation to f where W_u is the uniform-without-replacement distribution.*

The next lemma shows how any combined oracle algorithm can be transformed into one of a canonical form.

LEMMA 6.4 CANONICAL FORM ALGORITHM. *Let $\mathcal{A}$ be a (γ, δ)-approximation algorithm for a symmetric function f using (worst-case) t oracle queries to a combined oracle. Then, there exists a (γ, δ)-approximation canonical algorithm $\mathcal{A}'$ that uses (worst case) $3t$ oracle queries.*

PROOF. Note that `sample` does not take a parameter and therefore only the number of samples we make can depend on the outcome of probes we may do. However, we know that there can be at most t samples taken. Hence if we request t samples initially we can assume that we do not need to do any further sampling. Note that we have at most doubled the number of oracle queries. Let S be the set of i's seen as samples. Then, for each value $i \in S$ we perform `probe`(p, i). This only adds t queries to the complexity.

We now have a randomized algorithm that takes as input the outcome from our t samples and the value p_i for all $i \in S$ and performs a series of further probes. Note that since all samples have already been made, this phase of the computation can be viewed as a randomized decision tree. But now we use the fact that the function is symmetric and appeal to Lemma 6.3 to argue that this randomized decision tree can be rewritten as an oblivious decision tree. In such a tree, all queries can be decided in advance and we now have an algorithm of the desired canonical form. □

We are now ready to prove the main structural result of this section. The central idea for simulating a combined oracle algorithm in two passes of an adversarially ordered stream is to simulate the `sample` queries in the first pass and simulate the `probe` queries in the second pass. If the stream is randomly ordered, we will be able to do both in the same pass by using, roughly speaking, the prefix of the random order stream as a source for `sample` oracle queries.

THEOREM 6.5. *Let $\mathcal{A}$ be a (γ, δ)-approximation algorithm for a symmetric function f with t-query complexity in the combined oracle model. Then, there exist algorithms returning a (γ, δ)-approximation for f that use $O(t)$ space[3] and make either a) a single pass over a randomly-ordered stream or b) two passes over an adversarially ordered stream.*

PROOF. Assume $\mathcal{A}$ is in the canonical form. We first consider a stream in random order. Consider the following streaming algorithm that uses $O(t)$ space. We store the first t items in the data stream, $P = (\langle p, i_1\rangle, \langle p, i_2\rangle, \ldots \langle p, i_t\rangle)$. Now for each $i \in S = \{i_1, i_2, \ldots i_t\}$ we set up a counter that will be used to maintain as exact count of the frequency of i. We now chose t values, S' from $k \in [n] \setminus S$ uniformly at random (without replacement) and set up a counter for each of these t values. We also maintain a counter to estimate the length of the stream m. At the end of the data stream we claim that we can simulate the oracles queries made by $\mathcal{A}$. The only difficulty in establishing this claim is showing that we can use the elements of P to simulate the sample oracle queries. Ideally we would like to claim that we can just return i_j on the j-th sample query. However, to genuinely simulate the sample

[3]We measure space in word use and assume that any value in $[n]$ or $[m]$ can be expressed in a single word of space.

queries we must rather sample with replacement from P. This can be achieved in the obvious way: on the j-th sample queries we output i_j with probability $(m-j+1)/m$ and otherwise output $i_{j'}$ where j' is chosen uniformly at random from $[j-1]$. We thus can emulate the sample queries made by $\mathcal{A}$. The probes performed by $\mathcal{A}$, can also be emulated because for each i_j we have maintained counters that give us p_{i_j} and for each $k \in S'$ we know p_k.

For a stream in adversarial order things are simpler. In the first pass we generate our random sample (with replacement) using standard techniques. In the second pass we count the exact frequencies of the relevant i. □

The proof can be generalized to the case of computing a symmetric function of two distributions. We say that such a function $f : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is *symmetric*, if for all $p_1, \ldots, p_n, q_1, \ldots, q_n \in \mathbb{R}, i, j \in [n]$,

$$\begin{aligned} & f(p_1, \ldots, p_i, \ldots, p_j, \ldots, p_n, q_1, \ldots, q_i, \ldots, q_j, \ldots, q_n) \\ = \; & f(p_1, \ldots, p_j, \ldots, p_i, \ldots, p_n, q_1, \ldots, q_j, \ldots, q_i, \ldots, q_n) \; . \end{aligned}$$

The only important caveat is that, in the random-order result, we need $m(p) = \Theta(m(q))$ such that, with high probability, there are t elements of the form $\langle p, i\rangle$ (for some i) and t elements of the form $\langle q, i\rangle$ (for some i) in the first $O(t)$ data items.

6.2 Data-Stream Algorithms for Approximating Entropy and f-Divergences

The algorithmic results in Section 4 and Section 5, when combined with the results in the previous section naturally gives rise to the following theorem.

THEOREM 6.6. *In two passes of an adversarially-ordered stream, there exist algorithms that return,*

(1) *An (ϵ, δ)-approximation of entropy H using $O(\epsilon^{-2} H^{-1} \log n \log \delta^{-1})$ space.*
(2) *An (ϵ, δ)-approximation of a bounded f-Divergence D_f using $O(\epsilon^{-2} D_f^{-1} \log \delta^{-1})$ space.*

If the stream is randomly ordered then one pass suffices in each case.

REFERENCES

ALI, S. M. AND SILVEY, S. D. 1966. A general class of coefficients of divergence of one distribution from another. *J. of Royal Statistical Society, Series B 28*, 131–142.

ALON, N., MATIAS, Y., AND SZEGEDY, M. 1999. The space complexity of approximating the frequency moments. *Journal of Computer and System Sciences 58,* 1, 137–147.

AMARI, S. 1985. *Differential-geometrical methods in statistics.* Springer-Verlag, New York.

BANERJEE, A., MERUGU, S., DHILLON, I. S., AND GHOSH, J. 2005. Clustering with bregman divergences. *Journal of Machine Learning Research 6*, 1705–1749.

BAR-YOSSEF, Z. 2002. The complexity of massive data set computations. Ph.D. thesis, University of California at Berkeley.

BAR-YOSSEF, Z., KUMAR, R., AND SIVAKUMAR, D. 2001. Sampling algorithms: lower bounds and applications. *ACM Symposium on Theory of Computing*, 266–275.

BATU, T., DASGUPTA, S., KUMAR, R., AND RUBINFELD, R. 2005. The complexity of approximating the entropy. *SIAM J. Comput. 35,* 1, 132–150.

BATU, T., FORTNOW, L., RUBINFELD, R., SMITH, W. D., AND WHITE, P. 2000. Testing that distributions are close. In *IEEE Symposium on Foundations of Computer Science*. 259–269.

Bhuvanagiri, L. and Ganguly, S. 2006. Estimating entropy over data streams. In *ESA*. 148–159.

Chakrabarti, A., Cormode, G., and McGregor, A. 2007. A near-optimal algorithm for computing the entropy of a stream. In *ACM-SIAM Symposium on Discrete Algorithms*. 328–335.

Chakrabarti, A., Do Ba, K., and Muthukrishnan, S. 2006. Estimating entropy and entropy norm on data streams. In *Symposium on Theoretical Aspects of Computer Science*. 196–205.

Cover, T. M. and Thomas, J. A. 1991. *Elements of Information Theory*. Wiley Series in Telecommunications. John Wiley & Sons, New York, NY, USA.

Csiszár, I. 1991. Why least squares and maximum entropy? an axiomatic approach to inference for linear inverse problems. *Ann. Statist.*, 2032–2056.

Demaine, E. D., López-Ortiz, A., and Munro, J. I. 2002. Frequency estimation of internet packet streams with limited space. In *European Symposium on Algorithms*. 348–360.

Dhillon, I. S., Mallela, S., and Kumar, R. 2003. A divisive information-theoretic feature clustering algorithm for text classification. *Journal of Machine Learning Research 3*, 1265–1287.

Feigenbaum, J., Kannan, S., Strauss, M., and Viswanathan, M. 2002a. An approximate L^1 difference algorithm for massive data streams. *SIAM Journal on Computing 32,* 1, 131–151.

Feigenbaum, J., Kannan, S., Strauss, M., and Viswanathan, M. 2002b. Testing and spot-checking of data streams. *Algorithmica 34,* 1, 67–80.

Gilbert, A. C., Kotidis, Y., Muthukrishnan, S., and Strauss, M. 2001. Surfing wavelets on streams: One-pass summaries for approximate aggregate queries. In *International Conference on Very Large Data Bases*. 79–88.

Gu, Y., McCallum, A., and Towsley, D. 2005. Detecting anomalies in network traffic using maximum entropy estimation. In *Internet Measurement Conference*. 345–350.

Guha, S. and McGregor, A. 2006. Approximate quantiles and the order of the stream. In *ACM Symposium on Principles of Database Systems*. 273–279.

Guha, S. and McGregor, A. 2007a. Lower bounds for quantile estimation in random-order and multi-pass streaming. In *International Colloquium on Automata, Languages and Programming*. 704–715.

Guha, S. and McGregor, A. 2007b. Space-efficient sampling. In *AISTATS*. 169–176.

Guha, S., McGregor, A., and Venkatasubramanian, S. 2006. Streaming and sublinear approximation of entropy and information distances. In *ACM-SIAM Symposium on Discrete Algorithms*. 733–742.

Henzinger, M. R., Raghavan, P., and Rajagopalan, S. 1999. Computing on data streams. *External memory algorithms*, 107–118.

Kearns, M. J., Mansour, Y., Ron, D., Rubinfeld, R., Schapire, R. E., and Sellie, L. 1994. On the learnability of discrete distributions. In *ACM Symposium on Theory of Computing*. 273–282.

Krishnamurthy, B., Venkatasubramanian, S., and Madhyastha, H. V. 2005. On stationarity in internet measurements through an information-theoretic lens. In *ICDE Workshops*. 1185.

Lall, A., Sekar, V., Ogihara, M., Xu, J., and Zhang, H. 2006. Data streaming algorithms for estimating entropy of network traffic. In *SIGMETRICS/Performance*, R. A. Marie, P. B. Key, and E. Smirni, Eds. ACM, 145–156.

Liese, F. and Vajda, F. 1987. Convex statistical distances. *Teubner-Texte zur Mathematik, Band 95, Leipzig*.

Munro, J. I. and Paterson, M. 1980. Selection and sorting with limited storage. *Theor. Comput. Sci. 12*, 315–323.

Shannon, C. E. 1948. A mathematical theory of communication. *Bell System Technical Journal 27*, 379–423 and 623–656.

Tishby, N., Pereira, F., and Bialek, W. 1999. The information bottleneck method. In *Proceedings of the 37-th Annual Allerton Conference on Communication, Control and Computing*. 368–377.

Topsøe, F. 2000. Some inequalities for information divergence and related measures of discrimination. *IEEE Transactions on Information Theory 46,* 4, 1602–1609.

Čencov, N. N. 1981. Statistical decision rules and optimal inference. *Transl. Math. Monographs, Am. Math. Soc. (Providence)*.

Wagner, A. and Plattner, B. 2005. Entropy based worm and anomaly detection in fast IP networks. In *IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*. 172–177.

Xu, K., Zhang, Z.-L., and Bhattacharyya, S. 2005. Profiling internet backbone traffic: behavior models and applications. In *SIGCOMM*. 169–180.

Received December 2006; accepted ?????