Sampling Lower Bounds via Information Theory

[Extended Abstract] *

Ziv Bar-Yossef†
IBM Almaden Research Center
650 Harry Road
San Jose, CA 95120, USA
ziv@almaden.ibm.com

ABSTRACT

We present a novel technique, based on the Jensen-Shannon divergence from information theory, to prove lower bounds on the query complexity of sampling algorithms that approximate functions over arbitrary domain and range. Unlike previous methods, our technique does not use a reduction from a decision promise problem. As a result, it gives stronger bounds for functions that possess a large set of inputs, each two of which exhibit a gap in the function value.

We demonstrate the technique with new query complexity lower bounds for three fundamental problems: (1) the "election problem", for which we obtain a quadratic improvement over previous bounds, (2) low rank matrix approximation, for which we prove the first lower bounds, showing that the algorithms given for this problem are almost optimal, and (3) matrix reconstruction.

In addition, we introduce a new method for proving lower bounds on the *expected* query complexity of functions, using the Kullback-Leibler divergence. We demonstrate its use by a simple query complexity lower bound for the mean.

Categories and Subject Descriptors

F.2 [**Analysis of Algorithms and Problem Complexity**]: Miscellaneous

General Terms

Theory

Keywords

query complexity, lower bounds, Jensen-Shannon divergence

*A full version of this paper is available at www.almaden.ibm.com/cs/people/ziv/papers.

†Part of this work was done while the author was a student at U.C. Berkeley and supported by NSF ITR grant CCR-0121555.

1. INTRODUCTION

The increased volume of data generated automatically coupled with the dramatic improvements in storage capability have resulted in the emergence of numerous massive data sets, like the web, Internet traffic, stock market transactions, etc. The typically restricted access to these data sets together with their huge size necessitated the invention of new models of computation. Sampling algorithms—algorithms that query only a few (possibly random) locations of the input data—are the most traditional and well suited for this task.

In this paper we study lower bounds on the *query complexity* of sampling algorithms. Specifically, we develop new techniques for proving lower bounds on the number of queries a sampling algorithm is required to perform in order to approximate a given function accurately with low probability of error. Our lower bounds are applicable to general functions over arbitrary domain and range (even infinite or non-metric spaces) and to all types of approximation. The bounds are stated in terms of simple to calculate properties of functions, which are based on two information-theoretic measures of distance between distributions: the *Jensen-Shannon divergence* [17] and the *Kullback-Leibler divergence* [16]. We demonstrate the strength of our techniques by deriving new bounds for several natural and practical problems, including the "election problem", low rank matrix approximation, matrix reconstruction, and the mean.

Motivation The standard methodology for proving lower bounds for approximation problems is via a reduction from a *decision promise* problem. A decision promise problem for a function f is specified by a pair of disjoint subsets of the domain A and B that exhibit a "gap" in the function's value; that is, for all $a \in A$ and $b \in B$, $f(a)$ is "far" from $f(b)$. The decision problem is then: given an input $\mathbf{x}$ which is promised to be in $A \cup B$, decide whether $\mathbf{x} \in A$ or $\mathbf{x} \in B$. Clearly, any algorithm that accurately approximates f can be used to solve the promise problem, and thus a lower bound for the promise problem implies a hardness of approximation result. This methodology has been extensively used in proving lower bounds for sampling algorithms. Usually, one comes up with two inputs or two distributions over inputs that exhibit a gap, and shows that any sampling algorithm that uses a small number of queries cannot distinguish between the two. The "hardness" of such functions arises from the existence of pairs of inputs that on the one hand exhibit a large gap in the function value, but on the other hand are very "similar" and therefore are not distinguishable in a small number of queries.

There are some functions, however, (see examples below) for which *every* two inputs that exhibit a gap are easily distinguishable. For such functions the paradigm mentioned above fails gloriously. The hardness of this type of functions emanates simply from the fact there are *many* inputs, each two of which exhibit a gap in the function's value. (This can happen, of course, only for functions that have a large range). In this paper we develop an information-theoretic lower bound technique, which is able to capture the hardnesses that emanate from both the similarities among "gapped inputs" and the abundance of such inputs.

Summary of contributions Our main result is a novel technique for proving lower bounds on the query complexity of sampling algorithms using the Jensen-Shannon divergence. The technique is applicable to functions over arbitrary domain and range, and forms a "template" for easily deriving lower bounds for specific functions.

Consider a function $f : \mathcal{X}^n \to \mathcal{Y}$ that possesses ℓ inputs $\mathbf{x}_1, \ldots, \mathbf{x}_\ell \in \mathcal{X}^n$, each two of which exhibit a gap in the value of f (i.e., $f(\mathbf{x}_i)$ is "far" from $f(\mathbf{x}_j)$, for all $i \neq j$). A simple information-theoretic argument can show than any sampling algorithm approximating f would require $\Omega(\log \ell / \log |\mathcal{X}|)$ queries. This bound is useless when $\mathcal{X}$ is large (e.g., $\mathcal{X} = \mathbb{R}$). Moreover, it does not address the hardness arising from similarities among $\mathbf{x}_1, \ldots, \mathbf{x}_\ell$. We prove that any sampling algorithm whose queries on $\mathbf{x}_1, \ldots, \mathbf{x}_\ell$ are i.i.d. (independent and identically distributed) requires $\Omega(\log \ell / JS(\nu_{\mathbf{x}_1}, \ldots, \nu_{\mathbf{x}_\ell}))$ queries to approximate f. Here, $\nu_{\mathbf{x}_j}$ is the joint distribution of queries and their answers when the algorithm runs on input $\mathbf{x}_j$ and $JS(\cdot, \ldots, \cdot)$ is the Jensen-Shannon divergence among these distributions. The $\log \ell$ factor in the lower bound captures the abundance of gapped inputs, and the Jensen-Shannon divergence factor captures the similarities among them.

Lower bounds for sampling algorithms whose queries are i.i.d. are interesting for two reasons. First, in practice many sampling algorithms use i.i.d. (in fact, uniform) queries. Second, and more importantly, for wide classes of functions, sampling algorithm with i.i.d. queries can simulate general, adaptive, sampling with almost no loss of efficiency. For such functions our methods give general query complexity lower bounds. We extend previous known simulations [4, 12] to more general classes of functions, including g-symmetric functions and g-row- and column-symmetric functions (see definitions in Section 4).

Using our technique, we prove new query complexity lower bounds for three fundamental problems. In the election problem the input is a sequence of n votes to m parties. The goal is to estimate the distribution of votes among the parties up to an additive factor of ϵ w.r.t. to statistical (i.e., L_1) distance. This problem, which can be more generally viewed as estimating the distribution of "types" in a given population, is important in database systems for devising good query optimization plans. Batu *et al.* [5] gave an $\Omega(m)$ lower bound for the problem, while [4] showed an $O(m/\epsilon^2)$ upper bound and an $\Omega(1/\epsilon^2)$ lower bound. We prove an optimal lower bound of $\Omega(m/\epsilon^2)$, yielding up to a quadratic improvement over the previous bounds.

In the low rank matrix approximation problem, LRM_k, the input is an $m \times n$ real matrix $\mathbf{A}$, and the goal is to output a rank k matrix $\mathbf{B}$, such that $||\mathbf{A} - \mathbf{B}||_F < ||\mathbf{A} - \mathbf{A}_k||_F + \epsilon ||\mathbf{A}||_F$. Here, $\mathbf{A}_k$ is the best rank k approximation to $\mathbf{A}$ and $||\cdot||_F$ is the Forbenius norm (L_2 norm of the matrix when viewed as a vector). This problem is central in information retrieval applications, such as web search, collaborative filtering, and Latent Semantic Indexing (cf. [11, 2]). $\mathbf{A}_k$ can be computed exactly using Singular Value Decomposition (SVD) [13], however this requires querying the whole input matrix. Frieze, Kannan, and Vempala [11] and Drineas *et al.* [8] showed sampling algorithms for the problem. (Other algorithms were given also in [2, 1, 10]). Both algorithms are given some "advice" about their input in the form of distributions which are close to the row and the column weight distributions of the input matrix. The algorithm of [11] chooses s rows and s columns independently according to the advice distribution and queries the resulting $s \times s$ submatrix ($s = O(k^4/\epsilon^3)$). The algorithm of [8] chooses $O(k/\epsilon^2)$ rows according to the advice distribution and queries all their entries. We prove that (1) if no advice is given, then $\Omega(m+n)$ queries are needed for the problem; and (2) the above mentioned algorithms are optimal up to polynomial factors with respect to the advice they use.

In the matrix reconstruction problem, the input is an $m \times n$ real matrix $\mathbf{A}$, and the goal is to output any $m \times n$ matrix $\mathbf{B}$ so that $\mathbf{B}$ is close to $\mathbf{A}$. There are two variants of the problem: in the Forbenius variant, MR_F, we need $||\mathbf{A} - \mathbf{B}||_F < \epsilon ||\mathbf{A}||_F$ and in the L_2 variant, MR_2, we need $||\mathbf{A} - \mathbf{B}||_2 < \epsilon ||\mathbf{A}||_F$. (Recall that for a matrix $\mathbf{M}$, $||\mathbf{M}||_2 = \max_{||x||_2 = 1} ||\mathbf{M}x||_2$). The problem is important to collaborative filtering and recommendation systems [10]. Drineas and Kannan [9] show an algorithm for MR_2, which uses advice (the same kind of advice as in [11, 8]). Their algorithm selects $O(1/\epsilon^2)$ rows and columns independently according to the advice distribution and queries all their entries. The authors mention that MR_F requires $\Omega(mn)$ queries, and prove that any algorithm solving MR_2 has to output at least $\Omega((m+n)\log(1/\epsilon))$ bits. It is not clear whether this lower bound implies a sampling lower bound, especially when the sampling algorithm is given advice. We prove that indeed MR_F requires $\Omega(mn)$ queries. We then show that the query complexity of MR_2 is $\Omega(m+n)$, even for sampling algorithms that use the kind of advice used by [9]. This confirms that their algorithm is optimal.

A secondary result of this paper is a new technique for proving tight lower bounds on the *expected* query complexity of symmetric functions. The Jensen-Shannon technique and the techniques discussed in [4] are tight w.r.t. the worst-case query complexity, but give poor bounds (in terms of the error probability) when applied to the expected query complexity. Our new technique uses the KL divergence [16] between distributions. We demonstrate its use by an elementary proof for an optimal lower bound on the expected query complexity of the mean (which was originally proved by Radhakrishnan and Ta-Shma [20]).

Methodology Our lower bounds are based on a reduction from statistical classification to sampling algorithms. We prove that any sampling algorithm approximating a function f derives a classifier for the distributions $\nu_{\mathbf{x}_1}, \ldots, \nu_{\mathbf{x}_\ell}$, where $\mathbf{x}_1, \ldots, \mathbf{x}_\ell$ is the set of "gapped inputs". The classifier uses exactly the same number of samples as the algorithm, and therefore a lower bound on its sample complexity yields a lower bound on the query complexity of the algorithm.

The classification lower bound is proved in two steps. First, we use Fano's inequality from information theory to derive a lower bound of $1 - JS(\nu^q_{\mathbf{x}_1}, \ldots, \nu^q_{\mathbf{x}_\ell})/\log \ell$ on the misclassification probability of error (q is the number of samples used by the classifier). (As a side note, to the best of our knowledge, this lower bound gives an exponential improvement of the dependence on the number of classes ℓ over the best previously known bound due to Lin [17].) We then prove a decomposition property of the Jensen-Shannon divergence: $JS(\nu^q_{\mathbf{x}_1}, \ldots, \nu^q_{\mathbf{x}_\ell}) \le q \cdot JS(\nu_{\mathbf{x}_1}, \ldots, \nu_{\mathbf{x}_\ell})$, which allows us to derive a lower bound on q.

The lower bound via the KL divergence is a direct application of a result from statistical sequential analysis, called *the optimality of the sequential probability ratio test* (cf. [21]).

Related work The work most closely related to ours is a previous paper with Kumar and Sivakumar [4], which gave query complexity lower bounds in terms of the *Hellinger distance*. These bounds give tight results for functions that have few "gapped inputs" and weak bounds for functions that have many gapped inputs. We note that our Jensen-Shannon technique is as good as the Hellinger technique even for functions with few gapped inputs, since the Hellinger distance and the Jensen-Shannon divergence are always at most a constant away of each other [T.S. Jayram, private communication, October 2002]. The Hellinger technique, however, has a tighter dependence on the error probability.

Many ad-hoc sampling lower bounds appear in the literature. All of them are tailored to specific problems, and do not present a general technique. Sampling lower bounds in slightly different settings are given in statistics and learning (e.g., via VC dimension and the Cramér-Rao inequality). Previous lower bounds on the misclassification error (such as Stein's Lemma and Chernoff Bound, and also [15, 22, 23, 14, 17]), are either applicable only to two-class classification, are stated in an asymptotic form, or are weaker than the bound we present in this paper.

The rest of the paper is organized as follows. In Section 2 we review the tools from information theory, statistics, and combinatorics we use. In Section 3 we describe the sampling model in detail. In Section 4 we present the simulations of general sampling by i.i.d. sampling for symmetric functions. In Section 5 we describe the reduction from statistical classification to sampling. In Section 6 we prove the classification lower bound via the Jensen-Shannon divergence. In Section 7 we discuss the applications. In Section 8 we present the lower bound technique for the expected query complexity. For lack of space, some of the proofs are omitted. The complete proofs appear in the full version of the paper.

2. PRELIMINARIES

Notations and conventions $[m]$ stands for $\{1, \ldots, m\}$. All logarithms are to the base of 2. We deal only with finite discrete probability spaces. $X \sim \mu$ means that μ is the distribution of the random variable X. $\mu(x)$ is the probability of x under μ.

In order to formulate our lower bound technique in a way that is independent of the particular notion of approximation used, we use the following abstraction of approximation [4]: for a function $f : \mathcal{X}^n \to \mathcal{Y}$ and an error parameter $\epsilon > 0$, an approximation notion is a family of subsets $\{A_{f,\epsilon}(\mathbf{x})\}_{\mathbf{x} \in \mathcal{X}^n}$ of $\mathcal{Y}$. $A_{f,\epsilon}(\mathbf{x})$ is the called the "ϵ approximation set" of $\mathbf{x}$ and includes $f(\mathbf{x})$ and all the values that are considered an ϵ-approximation of $f(\mathbf{x})$. For example, in additive approximation, $A_{f,\epsilon}(\mathbf{x}) = (f(\mathbf{x}) - \epsilon, f(\mathbf{x}) + \epsilon)$, and in relative approximation $A_{f,\epsilon}(\mathbf{x}) = ((1-\epsilon)f(\mathbf{x}), f(\mathbf{x})(1+\epsilon))$. An algorithm Δ is said to (ϵ, δ)-approximate f, if for all inputs $\mathbf{x}$, $\Pr(\Delta(\mathbf{x}) \in A_{f,\epsilon}(\mathbf{x})) \ge 1 - \delta$, where the probability is over the internal coin tosses of Δ.

Statistical distance measures We use the following measures of distance between distributions:

Definition 1. *The* KL divergence *[16] and the* Jensen-Shannon divergence *[17] between two distributions μ_1, μ_2 on $\mathcal{X}$ are:*

$$D_{KL}(\mu_1 \parallel \mu_2) \stackrel{\text{def}}{=} \sum_{x \in \mathcal{X}} \mu_1(x) \log \frac{\mu_1(x)}{\mu_2(x)}.$$
$$JS(\mu_1, \mu_2) \stackrel{\text{def}}{=} \frac{1}{2}\left(D_{KL}(\mu_1 \parallel \tfrac{\mu_1+\mu_2}{2}) + D_{KL}(\mu_2 \parallel \tfrac{\mu_1+\mu_2}{2})\right).$$

The Jensen-Shannon divergence can be generalized to measure the mutual distance among more than two distributions, and with non-uniform weights:

Definition 2. *For a distribution λ on $[n]$, and for distributions $\mu_1, \ldots, \mu_n$ on $\mathcal{X}$, let $\mu_\lambda \stackrel{\text{def}}{=} \sum_{i=1}^n \lambda(i)\mu_i$ be the λ-weighted average distribution. The* λ-generalized Jensen-Shannon divergence *among $\mu_1, \ldots, \mu_n$ is:*

$$JS_\lambda(\mu_1, \ldots, \mu_n) \stackrel{\text{def}}{=} \sum_{i=1}^n \lambda(i) \cdot D_{KL}(\mu_i \parallel \mu_\lambda).$$

Information theory In the following $X \sim \mu_X$, $Y \sim \mu_Y$, $Z \sim \mu_Z$ are random variables on domains $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$, respectively. The entropy of X (or, equivalently, of μ_X) is $H(X) \stackrel{\text{def}}{=} \sum_{x \in \mathcal{X}} \mu_X(x) \log \frac{1}{\mu_X(x)}$. The entropy of a Bernoulli random variable with probability of success p is denoted $H_2(p)$. The joint entropy of X and Y is the entropy of the joint distribution (μ_X, μ_Y). The conditional entropy of X given an event A, denoted $H(X \mid A)$, is the entropy of the conditional distribution of μ_X given A. The conditional entropy of X given Y is $H(X \mid Y) \stackrel{\text{def}}{=} \sum_{y \in \mathcal{Y}} \mu_Y(y) H(X \mid Y = y)$. The mutual information between X and Y is $I(X;\, Y) \stackrel{\text{def}}{=} H(X) - H(X \mid Y) = H(Y) - H(Y \mid X)$. The conditional mutual information between X and Y given Z is $I(X;\, Y \mid Z) = H(X \mid Z) - H(X \mid Y, Z) = H(Y \mid Z) - H(Y \mid X, Z)$. Some basic properties of entropy and mutual information we are using in this paper are the following. Proofs can be found in Chapter 2 of [7].

Proposition 1. *Let X, Y, Z be random variables on domains $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$, respectively.*

1. *$H(X) \le \log |\mathcal{X}|$. Equality iff X is uniform on $\mathcal{X}$.*
2. *$H(X, Y) \le H(X) + H(Y)$. Equality iff X, Y are independent. (Subadditivity of entropy).*
3. *For any function f, $I(X;\, f(Y)) \le I(X;\, Y)$. (Data processing inequality).*
4. *$I(X;\, Y, Z) = I(X;\, Y) + I(X;\, Z \mid Y)$. (Chain rule for mutual information).*
5. *$I(X;\, Y) = 0$ iff X, Y are independent.*
6. *If X, Y are jointly independent of Z, then $I(X;\, Y \mid Z) = I(X;\, Y)$.*

The generalized Jensen-Shannon divergence has an equivalent characterization in terms of mutual information (see proof in [3]):

PROPOSITION 2. *$JS_\lambda(\mu_1, \ldots, \mu_n) = I(D; X_D)$, where $D \sim \lambda$, $X_D = \pi(D, X_1, \ldots, X_n)$, $X_j \sim \mu_j$ for all j, and $\pi(d, x_1, \ldots, x_n) = x_d$.*

Set designs A family of subsets $\mathcal{F}_1, \ldots, \mathcal{F}_\ell \subseteq [m]$ is called a (d, k)-design, if (1) for all $j \in [\ell]$, $|\mathcal{F}_j| = d$, and (2) for all $j \neq j' \in [\ell]$, $|\mathcal{F}_j \cap \mathcal{F}_{j'}| \leq k$. We will use $(\frac{m}{2}, \Omega(m))$-designs of exponential size, as given by the following proposition. The proof is a slight modification of the probabilistic method argument of [19], guaranteeing the sets are of size exactly $m/2$ (and not just at most $m/2$).

PROPOSITION 3. *For every m, there exists a $(\frac{m}{2}, \frac{11}{24}m)$-design $\mathcal{F}_1, \ldots, \mathcal{F}_\ell \subseteq [m]$ of size $\ell = 2^{\Omega(m)}$.*

Sequential distributions A sequential distribution on a domain $\mathcal{B}$ is a family of distributions $\mu = \{\mu_{\mathbf{b}}\}_{\mathbf{b}}$ on $\mathcal{B}$, indexed by all the finite length sequence $\mathbf{b} = (b_1, \ldots, b_t)$ of elements from $\mathcal{B}$. The family is used to construct an infinite random sequence of elements of $\mathcal{B}$ as follows: the first element is chosen according to $\mu_\emptyset$; if the first t elements chosen are $\mathbf{b} = (b_1, \ldots, b_t)$, the next element is selected according to $\mu_{\mathbf{b}}$.

For a sequential distribution μ and any integer $t > 0$, μ_t is the following distribution on $\mathcal{B}^t$: $\mu_t(b_1, \ldots, b_t) = \mu_\emptyset(b_1) \cdot \mu_{b_1}(b_2) \cdots \mu_{b_1,\ldots,b_{t-1}}(b_t)$.

A sequential distribution μ on $\mathcal{B}$ is called independent and identically distributed (or "i.i.d." in short), if all the distributions in the family are identical to some distribution ν on $\mathcal{B}$. We call ν the base distribution of μ. Note that for i.i.d. distributions, $\mu_t = \nu^t$ for all t.

3. THE SAMPLING MODEL

A sampling algorithm is a randomized algorithm that has "oracle access" to its input $\mathbf{x} \in \mathcal{X}^n$. In each oracle call, the algorithm queries an index $i \in [n]$, and gets back $\mathbf{x}_i$. The queries are performed sequentially, and can be chosen *adaptively*: that is, based on answers to previous queries the algorithm can choose which index to query next.

Since we are interested in lower bounds for sampling algorithms, we (1) consider only the number of oracle calls, or queries, the algorithm performs, and ignore other resources like time or space; (2) ignore precision issues and assume the algorithm can query even real numbers at unit cost; and (3) restrict to functions over a fixed input length n.

A formal definition of sampling algorithms is given in the full version of the paper and is briefly described next. A sampling algorithm is a communication protocol Δ among three players: Alice, Bob, and a "referee". Bob gets the input $\mathbf{x} \in \mathcal{X}^n$; Alice and the referee get a random string each. In the protocol Alice continuously sends query indices to Bob, and Bob replies with answers to these queries based on his input. The referee monitors the interaction between Alice and Bob until she decides she has enough information to determine the output, at which point she stops the protocol and declares the output.

The protocol proceeds in rounds as follows. Alice starts each round by using her random string and the answers to her previous queries to decide about the next query index $i \in [n]$. She sends this index to Bob, who replies with $\mathbf{x}_i$. The referee, after seeing i and $\mathbf{x}_i$, uses her random string as well as all the previous queries and their answers to decide whether she would like to stop and give an answer.

Query cost The round T at which the referee stops is called the "stopping time" of the protocol. T is a random variable, depending on $\mathbf{x}$ and on the random strings of Alice and the referee. The maximum value of T, over all inputs and random strings, is called the worst-case query cost of the protocol Δ. The maximum expected value of T, where the maximum is over inputs and the expectation is over random strings, is called the expected query cost of Δ.

Public coins vs. private coins We distinguish between two types of sampling protocols. In public-coin protocols, Alice and the referee get the same random string, while in private-coins protocols they get independent random strings. Clearly, any private-coin protocol can be simulated by a public-coin one with the same query cost. Public-coin protocols *represent the most general sampling algorithms possible.* The reason to study private-coin protocols is that our lower bound technique is applicable only to them. An interesting question is then whether any public-coin protocol can be simulated efficiently by a private-coin one. It is possible to adapt a result by Newman [18], which works for general communication complexity protocols, to our restricted model of communication: any public-coin protocol of query cost q can be simulated by a private-coin protocol of query cost $q + O(\log\log|\mathcal{X}|/\log n)$. This implies that for small domains $\mathcal{X}$, private-coin protocols are almost as good as public-coin protocols. We currently do not know whether there could be a large gap between the two for large domains.

Index-oblivious sampling A sampling protocol Δ is called index-oblivious, if the referee's decisions (i.e., when to stop and what value to output) depend only on the answers to queries and not explicitly on the indices queried.

Query distributions Any sampling algorithm Δ and any input $\mathbf{x} \in \mathcal{X}^n$ induce a sequential distribution $\mu^{\mathrm{I}}_{\mathbf{x}} = \{\mu^{\mathrm{I}}_{\mathbf{x},\mathbf{i}}\}_{\mathbf{i}}$ on $[n]$, which we call the index distribution of Δ on $\mathbf{x}$. $\mu^{\mathrm{I}}_{\mathbf{x},\mathbf{i}}$ is the distribution of the random index i Alice chooses at the $t+1$-st round, given that the previous indices queried were $\mathbf{i} = (i_1, \ldots, i_t)$ and their answers were $\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_t}$.

Related families of distributions are the answer distribution and the query distribution. The answer distribution is the sequential distribution $\mu^{\mathrm{A}}_{\mathbf{x}} = \{\mu^{\mathrm{A}}_{\mathbf{x},\mathbf{a}}\}_{\mathbf{a}}$ on $\mathcal{X}$, where $\mu^{\mathrm{A}}_{\mathbf{x},\mathbf{a}}$ is the distribution of the answer a_{t+1}, given that the first t answers were $\mathbf{a} = (a_1, \ldots, a_t)$. The query distribution is the sequential distribution $\mu_{\mathbf{x}} = \{\mu_{\mathbf{x},\mathbf{i}}\}_{\mathbf{i}}$ on $[n] \times \mathcal{X}$, where $\mu_{\mathbf{x},\mathbf{i}}$ is the distribution of the index-answer pair $(i, \mathbf{x}_i)$ given the history $\mathbf{i}$. Note that if the index distribution is i.i.d., then so are the answer distribution and the query distribution.

Two index distributions we are going to focus on in this paper are the following: (1) Uniform sampling without replacement: For $t = 0, \ldots, n$, and for any t distinct indices $i_1, \ldots, i_t \in [n]$, $\mu_{i_1,\ldots,i_t}$ is the uniform distribution on $[n] \setminus \{i_1, \ldots, i_t\}$. (2) Uniform sampling with replacement: For any $\mathbf{i}$, $\mu_{\mathbf{i}}$ is uniform on $[n]$. Note that uniform sampling with replacement is an i.i.d. sequential distribution, while uniform sampling without replacement is not.

Sampling with advice Standard sampling algorithms do not have any prior information about the input on which they run. A non-standard type of sampling algorithms are ones that are given some external "advice" about their input. Formally, it means that the query distribution of Alice on input $\mathbf{x}$ depend not only on her random string and the answers from Bob, but also on the "advice".

4. SIMULATIONS FOR SYMMETRIC FUNCTIONS

For any function with certain "symmetry" properties, we can prove that any sampling algorithm approximating the function can be simulated with almost the same query cost by a sampling algorithm whose query distribution on all inputs is uniform with replacement. Thus, for such functions query complexity lower bounds with respect to algorithms that have i.i.d. query distributions imply general query complexity lower bounds. In the following we extend previously known such simulations [4, 12] to a larger class of functions.

Symmetric functions Given an input $\mathbf{x} \in \mathcal{X}^n$ and a permutation $\pi \in S_n$, we define the permutation of $\mathbf{x}$ according to π to be the input $\pi(\mathbf{x})$ which satisfies for all $i \in [n]$, $\pi(\mathbf{x})_i = \mathbf{x}_{\pi^{-1}(i)}$.

A symmetric function is one which is invariant under permutations of its inputs: $f(\pi(\mathbf{x})) = f(\mathbf{x})$ for all $\mathbf{x}$ and π. Many basic functions like average, minimum, and median are symmetric. A generalization of this notion is the following: for some function $g : \mathcal{Y} \times S_n \to \mathcal{Y}$, f is called g-symmetric, if for all $\mathbf{x}$ and π, $g(f(\pi(\mathbf{x})), \pi) = f(\mathbf{x})$. That is, $f(\pi(\mathbf{x}))$ is not necessarily the same as $f(\mathbf{x})$, but with a simple transformation, one can use $f(\pi(\mathbf{x}))$ to get $f(\mathbf{x})$. Symmetric functions are g-symmetric with $g(y, \pi) = y$. Another interesting sub-class of g-symmetric functions are ones in which $f(\pi(\mathbf{x})) = \pi(f(\mathbf{x}))$. In this case $g(y, \pi) = \pi^{-1}(y)$. We call such functions permutation-commutative. The identity function, for example, is permutation-commutative. Next, we define the notion corresponding to g-symmetric functions, when dealing with approximations of functions:

DEFINITION 3. *Let $f : \mathcal{X}^n \to \mathcal{Y}$ be a function with approximation notion $A_{f,\epsilon}$. For a function $g : \mathcal{Y} \times S_n \to \mathcal{Y}$, we call f (g, ϵ)-symmetric, if for all inputs $\mathbf{x} \in \mathcal{X}^n$, for all permutations $\pi \in S_n$, and for all $y \in A_{f,\epsilon}(\pi(\mathbf{x}))$, $g(y, \pi) \in A_{f,\epsilon}(\mathbf{x})$.*

[4] showed how to efficiently simulate general sampling algorithms approximating symmetric functions by sampling algorithms whose query distributions on all inputs are uniform with replacement. We observe that almost the same simulation works for (g, ϵ)-symmetric functions. The simulation, described in the following theorem, has the additional properties that it uses only private coins and that it is index-oblivious.

THEOREM 1. *Let Δ be any sampling protocol that (ϵ, δ)-approximates a (g, ϵ)-symmetric function $f : \mathcal{X}^n \to \mathcal{Y}$. Let q be the worst-case query cost of Δ and let q_E be its expected query cost. Then, there is a private-coin sampling protocol $\hat{\Delta}$ that $(\epsilon, 2\delta)$-approximates f and whose index distributions on all inputs are uniform with replacement. If g depends only on its first argument and if $q \le \sqrt{2\delta n}$, then $\hat{\Delta}$ is also index-oblivious. The worst-case query cost of $\hat{\Delta}$ is at most $2q$ and its expected query cost is at most $2q_E$.*

The theorem follows from two simulation lemmas. The first lemma shows how to simulate any sampling algorithm approximating a (g, ϵ)-symmetric function by a sampling algorithm whose queries are uniform without replacement. The second lemma shows how to simulate uniform queries without replacement by uniform queries with replacement. Proofs of these lemma can be found in [4, 3] and in the full version of this paper.

Matrix-symmetric functions A function f on $m \times n$ real matrices is called row-symmetric, if it is invariant under permutations of its input matrix's rows. That is, $f(\mathbf{A}) = f(\mathbf{\Sigma A})$ for all matrices $\mathbf{A}$ and all row permutation matrices $\mathbf{\Sigma}$. Similarly, f is column-symmetric, if $f(\mathbf{A}) = f(\mathbf{A\Pi})$ for all $\mathbf{A}$ and all column permutation matrices $\mathbf{\Pi}$. The rank of a matrix is an example of a function which is both row-symmetric and column-symmetric.

Similar to the definitions of g-symmetric functions and permutation-commutative functions, we define g-row symmetric functions and row permutation commutative functions. The low-rank matrix approximation problem and the matrix reconstruction problem are row and column permutation commutative (see Section 7). Again, we define the corresponding notion for approximation of functions:

DEFINITION 4. *Let $f : M_{m\times n}(\mathbb{R}) \to \mathcal{Y}$ be a function with approximation notion $A_{f,\epsilon}$. For a function $g : \mathcal{Y} \times S_m \to \mathcal{Y}$, we call f (g, ϵ)-row-symmetric, if for all input matrices $\mathbf{A}$, for all permutations $\sigma \in S_m$, and for all $y \in A_{f,\epsilon}(\mathbf{\Sigma A})$, $g(y, \sigma) \in A_{f,\epsilon}(\mathbf{A})$. (Here, $\mathbf{\Sigma}$ is the row-permutation matrix corresponding to σ).*

We define(g, ϵ)-column-symmetric functions analogously.

Goldreich and Trevisan [12] showed simulations for graph property testers in the adjacency matrix model. We extend their result to general row- and column-symmetric functions.

A standard sampling algorithm computing a function over matrices queries one entry of the input matrix at a time. We call a sampling algorithm that queries a full row of the input matrix at a time a row-querying sampling algorithm. The query cost of such an algorithm is the number of full rows it queries. The following theorem shows that any sampling algorithm approximating a row-symmetric function can be simulated by a row-querying algorithm that chooses its row queries uniformly at random with replacement. An analogous theorem holds for (g, ϵ)-column symmetric functions. The proofs appear in the full version of the paper.

THEOREM 2. *Let Δ be a sampling protocol that (ϵ, δ) approximates a (g, ϵ)-row symmetric function $f : M_{m\times n}(\mathbb{R}) \to \mathcal{Y}$. Let q be the worst-case query cost of Δ and let q_E be its expected query cost. Then, there is a private-coin row-querying sampling protocol $\hat{\Delta}$ that $(\epsilon, 2\delta)$-approximates f and whose index distributions on all inputs are uniform with replacement. If g depends only on its first argument and $q \le \sqrt{2\delta n}$, then $\hat{\Delta}$ is also index-oblivious. The worst-case query cost of $\hat{\Delta}$ is at most $2q$ and its expected query cost is at most $2q_E$.*

5. REDUCTION FROM CLASSIFICATION TO SAMPLING

In this section we show that any sampling algorithm that approximates a function f derives a statistical classifier for the query distributions of an appropriately chosen family of inputs to f. We start by defining statistical classification and then describe the reduction.

Statistical classification Loosely speaking, statistical classification is defined as follows. A "black box" contains a distribution μ, which is guaranteed to be one of ℓ known distributions $\mu_1, \ldots, \mu_\ell$ on the same domain $\mathcal{B}$. (We do not consider here the "Bayesian" scenario, in which there is some prior distribution on $[\ell]$). A classifier is a randomized oracle algorithm that has to determine the identity of μ. Each oracle call produces a sample from μ. At the end of its execution, the classifier announces a guess for the identity of μ. This guess is required to be correct with probability at least $1-\delta$, for all possible choices of μ (the probability is over both the samples from μ and the coin tosses of the classifier).

The formal definition of a classifier is given in the full version of the paper and is briefly described next. A sequential classifier C for ℓ sequential distributions $\mu_1, \ldots, \mu_\ell$ on a domain $\mathcal{B}$ is a communication protocol between two players: Alice and a "referee". Alice gets as input an index $j \in [\ell]$. The referee gets no input, but is given a random string. Alice continuously sends samples from μ_j to the referee. The referee monitors the messages sent from Alice until she decides she has enough information to determine the identity of j, at which point she declares this decision and stops.

The step T at which the referee stops the protocol is called the "stopping time". T is a random variable depending on the choice of j, on the samples drawn from μ_j and on the referee's random string. The worst-case sample cost of the classifier is the maximum stopping time over all choices of j, all possible sample sequences drawn from μ_j and all values of the random string. The expected sample cost of C is the maximum expected stopping time, where the maximum is over the choice of j, and the expectation is over the samples from μ_j and the random string of the referee.

The reduction In the reduction described below, we show that given any sampling algorithm that approximates a function f and given any collection of inputs, on which f takes very "different" values, there is a classifier for the query distributions corresponding to these inputs. We start by defining what it means for two inputs to have "different" f-values:

Definition 5 (Disjoint inputs). *Let $f : \mathcal{X}^n \rightarrow \mathcal{Y}$ be a function with approximation notion $A_{f,\epsilon}$. Two inputs $\mathbf{x}, \mathbf{x}' \in \mathcal{X}^n$ are called ϵ-disjoint, if $A_{f,\epsilon}(\mathbf{x}) \cap A_{f,\epsilon}(\mathbf{x}') = \emptyset$.*

Theorem 3. *Let Δ be a private-coin sampling protocol that (ϵ, δ)-approximates a function $f : \mathcal{X}^n \rightarrow \mathcal{Y}$. Let q be the worst-case query cost of Δ and let q_E be its expected query cost. Let $\mathcal{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_\ell\}$ be any set of inputs to f, each two of which are ϵ-disjoint. Then, there exists a δ-error sequential classifier C for the query distributions $\mu_{\mathbf{x}_1}, \ldots, \mu_{\mathbf{x}_\ell}$, whose worst-case query cost is at most q and expected query cost is at most q_E. If Δ is index-oblivious, there exists a similar classifier for the answer distributions $\mu^A_{\mathbf{x}_1}, \ldots, \mu^A_{\mathbf{x}_\ell}$.*

Proof. The classifier C, given an input query distribution $\mu_{\mathbf{x}_j}$, simulates the sampling protocol Δ on $\mathbf{x}_j$. The role of Alice and Bob in the protocol Δ is simulated by Alice alone in C: Alice produces samples from the query distribution $\mu_{\mathbf{x}_j}$; this distribution is identical by definition to the distribution of (index-answer) pairs Alice and Bob produce in Δ when given the input $\mathbf{x}_j$. Since the random string used by the referee in Δ is independent of the random string used by Alice (recall that Δ is a private-coin protocol), the referee of C can simulate the referee of Δ.

In the end of the simulation, Δ gives some output. C decides that the input distribution is $\mu_{\mathbf{x}_j}$ if and only if the output of Δ belongs to $A_{f,\epsilon}(\mathbf{x}_j)$. This decision is well defined, since the sets $A_{f,\epsilon}(\mathbf{x}_1), \ldots, A_{f,\epsilon}(\mathbf{x}_\ell)$ are pairwise disjoint. Since Δ produces a value in $A_{f,\epsilon}(\mathbf{x}_j)$ with probability at least $1-\delta$, the classifier makes the right decision with the same probability.

When Δ is index-oblivious, C does not need the query indices for the simulation, and can thus use only the answer distributions. $\square$

6. CLASSIFICATION LOWER BOUND

In this section we present a new lower bound on the sample complexity of statistical classification in terms of the Jensen-Shannon divergence. We mention the implication to query complexity lower bounds of functions in the end of the section.

The lower bound is proved in two steps: first we show a lower bound on the error probability of the classifier (a.k.a. the "misclassification error") in terms of the Jensen-Shannon divergence among the distributions that are being classified. We then prove a decomposition property of the Jensen-Shannon divergence among i.i.d. distributions. The two steps together yield a lower bound on the sample cost of classification of i.i.d. distributions.

The following theorem gives a lower bound on the misclassification error of a classifier that uses q samples to classify sequential distributions $\mu_1, \ldots, \mu_\ell$ in terms of the Jensen-Shannon divergence among $\mu_{1,q}, \ldots, \mu_{\ell,q}$ (recall our definition of sequential distributions from Section 2).

Theorem 4. *Let λ be any distribution on $[\ell]$. The classification error δ of any classifier C of sample cost q for sequential distributions $\mu_1, \ldots, \mu_\ell$ on $\mathcal{B}$ satisfies:*

$$H_2(\delta) + \delta \log(\ell - 1) \geq H(\lambda) - JS_\lambda(\mu_{1,q}, \ldots, \mu_{\ell,q}).$$

Bounding $H_2(\delta)$ by $2\delta \log(1/\delta)$ gives an explicit bound: $\delta \geq \Omega(\frac{1}{\log \ell \cdot \log log \ell} \cdot (H(\lambda) - JS_\lambda(\mu_{1,q}, \ldots, \mu_{\ell,q})))$. This is an exponential improvement (in terms of ℓ) over the bound of Lin [17], who showed $\delta \geq \frac{1}{4(\ell-1)}(H(\lambda) - JS_\lambda(\mu_{1,q}, \ldots, \mu_{\ell,q}))^2$.

Proof. The proof is based on Fano's inequality (cf. [7]) from information theory, which gives a lower bound on the error probability of predicting the value of a random variable X from the observation of another random variable Y:

Theorem 5. *Let X and Y be two random variables on domains $\mathcal{X}$ and $\mathcal{Y}$ respectively. Let $g : \mathcal{Y} \rightarrow \mathcal{X}$ be a prediction function, and let $\delta = \Pr(g(Y) \neq X)$ be the prediction error probability. Then, $H_2(\delta) + \delta \log(|\mathcal{X}| - 1) \geq H(X \mid Y)$.*

Consider now the classifier C that uses at most q samples to classify $\mu_1, \ldots, \mu_\ell$. For each $d = 1, \ldots, \ell$, let $B_d \sim \mu_{d,q}$ denote the random variable corresponding to the first q samples drawn from μ_d.

If we select an index $d \in [\ell]$ according to λ and run the classifier with μ_d as input, we can recover d with probability at least $1 - \delta$. Therefore, if $D \sim \lambda$ is the random variable corresponding to the choice of d, the decision function of the referee in the classifier C induces a prediction function $g_C : \mathcal{B}^q \times \mathcal{R} \rightarrow [\ell]$, which predicts the value of D based on q samples drawn from μ_D and based on the random string of the referee. Thus, the prediction error, $\Pr(g_C(B_D, R) \neq D)$, is at most δ. Here, B_D denotes the random variable $\pi(D, B_1, \ldots, B_\ell)$, where $\pi(d, \mathbf{b}_1, \ldots, \mathbf{b}_\ell) = \mathbf{b}_d$. R is the random variable corresponding to the random string used by the classifier's referee. Using Fano's inequality, we have the following lower bound on δ:

$$H_2(\delta) + \delta \log(\ell - 1) \geq H(D \mid g_C(B_D, R)).$$

We now expand the conditional entropy, using the information theory properties mentioned in Proposition 1. By the definition of mutual information, $H(D \mid g_C(B_D, R)) = H(D) - I(D;\ g_C(B_D, R))$. By the data processing inequality, $I(D;\ g_C(B_D, R)) \leq I(D;\ B_D, R)$. By the chain rule for mutual information, $I(D;\ B_D, R) = I(D;\ R) + I(D;\ B_D \mid R)$. Since D and B_D are jointly independent of R, then $I(D;\ R) = 0$ and $I(D;\ B_D \mid R) = I(D;\ B_D)$. We conclude: $H_2(\delta) + \delta \log(\ell - 1) \ \geq \ H(D) - I(D;\ B_D)$. We complete the proof by noting that $I(D;\ B_D) = JS_\lambda(\mu_{1,q}, \ldots, \mu_{\ell,q})$ (Proposition 2). $\square$

Proposition 4. *Let $\mu_{1,q} = \nu_1^q, \ldots, \mu_{\ell,q} = \nu_\ell^q$ be i.i.d. distributions on $\mathcal{B}^q$. Then,*

$$JS_\lambda(\mu_{1,q}, \ldots, \mu_{\ell,q}) \ \leq \ q \cdot JS_\lambda(\nu_1, \ldots, \nu_\ell).$$

Proof. Let $B_j \sim \mu_{j,q}$ and $X_j \sim \nu_j$. Since $\mu_{j,q} = \nu_j^q$, $B_j = X_j^q$; that is, B_j consists of q independent copies of X_j. Also, let $D \sim \lambda$, $B_D = \pi(D, B_1, \ldots, B_\ell)$, and $X_D = \pi(D, X_1, \ldots, X_\ell)$, where $\pi(d, x_1, \ldots, x_\ell) = x_d$. From Proposition 2 we have that $JS_\lambda(\mu_{1,q}, \ldots, \mu_{\ell,q}) = I(D;\ B_D)$ and $JS_\lambda(\nu_1, \ldots, \nu_\ell) = I(D;\ X_D)$,

By the definition of mutual information, $I(D;\ B_D) = H(B_D) - H(B_D \mid D)$. By the subadditivity of entropy, $H(B_D) \leq q \cdot H(X_D)$. Note that conditioned on $D = d$, $B_D = X_d^q$. Therefore, $H(B_D \mid D) = \sum_d \lambda(d) H(X_d^q) = \sum_d \lambda(d) \cdot q \cdot H(X_d) = q \cdot H(X_D \mid D)$. The next to the last equality follows from the fact the joint entropy of independent random variables is the sum of their entropies. We thus have: $I(D;\ B_D) \ \leq \ q \cdot H(X_D) - q \cdot H(X_D \mid D) = q \cdot I(D;\ X_D)$. $\square$

We can now immediately derive the sample complexity lower bound for classification:

Theorem 6. *Let $\mu_1, \ldots, \mu_\ell$ be sequential i.i.d. distributions on $\mathcal{B}$, where $\mu_{j,t} = \nu_j^t$ for all $t \geq 1$. Let λ be any distribution on $[\ell]$. Then, the worst-case sample cost q of any δ-error classifier C for $\mu_1, \ldots, \mu_\ell$ satisfies:*

$$q \ \geq \ \frac{1}{JS_\lambda(\nu_1, \ldots, \nu_\ell)} \left(H(\lambda) - \delta \log(\ell - 1) - H_2(\delta)\right).$$

Choosing λ to be uniform on $[\ell]$ gives: $q \geq \Omega(\frac{\log \ell}{JS_\lambda(\nu_1, \ldots, \nu_\ell)} \cdot (1 - \delta))$.

Using the reduction from classification to sampling, we obtain the following query complexity lower bound:

Theorem 7 (Main theorem). *Let Δ be any private-coin sampling algorithm that (ϵ, δ)-approximates a function $f : \mathcal{X}^n \to \mathcal{Y}$. Let q be the worst-case query cost of Δ. Let $\mathcal{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_\ell\}$ be any set of pairwise ϵ-disjoint inputs and let λ be any distribution over this set. If the query distributions of Δ on $\mathbf{x}_1, \ldots, \mathbf{x}_\ell$ are all i.i.d.– that is, $\mu_{\mathbf{x}_j,q} = \nu_{\mathbf{x}_j}^q$ for all $j \in [\ell]$ – then the following lower bound on q holds:*

$$q \ \geq \ \frac{1}{JS_\lambda(\nu_{\mathbf{x}_1}, \ldots, \nu_{\mathbf{x}_\ell})} \left(H(\lambda) - \delta \log(\ell - 1) - H_2(\delta)\right).$$

If Δ is index-oblivious, an analogous lower bound holds with the answer distributions replacing the query distributions.

7. APPLICATIONS

7.1 The election problem

The input for the election problem, EP, is a sequence $\mathbf{x} \in [m]^n$ of n votes to m parties. The vote frequency of party $i \in [m]$, $v_i(\mathbf{x})$, is the number votes party i gets in $\mathbf{x}$. $\mathrm{EP}(\mathbf{x})$ is the vote distribution $V_{\mathbf{x}} \stackrel{\text{def}}{=} (v_1(\mathbf{x})/n, \ldots, v_m(\mathbf{x})/n)$. We consider additive approximation w.r.t. statistical distance (i.e., half of the L_1 distance). That is, $\mu \in A_{\mathrm{EP},\epsilon}(\mathbf{x})$, iff $||\mu - V_{\mathbf{x}}|| < \epsilon$.

Theorem 8. *Let $0 < \epsilon < \frac{1}{96}$, $0 < \delta \leq \frac{1}{2}$, and $n \geq \Omega(m^2/\epsilon^4 \cdot 1/\delta)$. The query cost of any sampling algorithm that (ϵ, δ)-approximates* EP *is at least $\Omega(m/\epsilon^2)$.*

Proof. Since EP is ϵ-symmetric, we can assume it can be approximated by a private-coin sampling algorithm Δ, whose queries on any input are uniform with replacement. Moreover, our assumption about n implies that Δ never uses more than $O(\sqrt{n})$ queries. We can thus further assume Δ is index-oblivious. Note that since Δ makes queries which are uniform with replacement, the answer distribution on any input $\mathbf{x}$, $\mu_{\mathbf{x}}^{\mathrm{A}}$, is i.i.d. with a base distribution that exactly equals to $V_{\mathbf{x}}$ – the vote distribution corresponding to $\mathbf{x}$.

By Theorem 7, in order to get an $\Omega(m/\epsilon^2)$ lower bound on the query cost of Δ, all we need to do is find a family of pairwise disjoint inputs $\mathcal{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_\ell\}$ of size $\ell = 2^{\Omega(m)}$ and prove that for the uniform distribution λ on $[\ell]$, $JS_\lambda(V_{\mathbf{x}_1}, \ldots, V_{\mathbf{x}_\ell}) = O(\epsilon^2)$ (recall that $V_{\mathbf{x}_1}, \ldots, V_{\mathbf{x}_\ell}$ happen to coincide with the answer distributions of Δ on $\mathbf{x}_1, \ldots, \mathbf{x}_\ell$).

In order to construct the set $\mathcal{S}$, we use set designs. We associate with any family of subsets $\mathcal{F} = \mathcal{F}_1, \ldots, \mathcal{F}_\ell \subseteq [m]$, where $\forall j, |\mathcal{F}_j| = \frac{m}{2}$, a family of inputs to EP, $\mathcal{S}_{\mathcal{F}} = \{\mathbf{x}_1, \ldots, \mathbf{x}_\ell\}$. $\mathbf{x}_j$ is some input whose vote distribution is the following: $V_{\mathbf{x}_j}(i) = \frac{1+24\epsilon}{m}$ for $i \in \mathcal{F}_j$ and $V_{\mathbf{x}_j}(i) = \frac{1-24\epsilon}{m}$ for $i \notin \mathcal{F}_j$.

Let $\mathcal{F}$ be a $(\frac{m}{2}, \frac{11}{24}m)$-design of size $\ell = 2^{\Omega(m)}$, as given by Proposition 3. We first prove that every two inputs $\mathbf{x}_j, \mathbf{x}_{j'} \in \mathcal{S}_{\mathcal{F}}$ are ϵ-disjoint: $||V_{\mathbf{x}_j} - V_{\mathbf{x}_{j'}}|| \geq V_{\mathbf{x}_j}(\mathcal{F}_j \setminus \mathcal{F}_{j'}) - V_{\mathbf{x}_{j'}}(\mathcal{F}_j \setminus \mathcal{F}_{j'}) \geq \frac{m}{24} \cdot \frac{48\epsilon}{m} = 2\epsilon$.

Instead of bounding $JS_\lambda(V_{\mathbf{x}_1}, \ldots, V_{\mathbf{x}_\ell})$ directly, we will show $JS_\lambda(V_{\mathbf{x}_1}, \ldots, V_{\mathbf{x}_\ell}) \ \leq \ JS_{\lambda^*}(V_{\mathbf{y}_1}, \ldots, V_{\mathbf{y}_{\ell^*}})$, for some other family of inputs $\mathcal{S}^* = \{\mathbf{y}_1, \ldots, \mathbf{y}_{\ell^*}\}$, and then bound the latter. $\mathcal{S}^*$ is defined to be the family of inputs corresponding to the family $\mathcal{F}^*$ of all subsets of $[m]$ of size $\frac{m}{2}$ (thus, $\ell^* = \binom{m}{m/2}$). λ^* is the uniform distribution on $[\ell^*]$.

Let $V_{\mathbf{x}_\lambda}$ and $V_{\mathbf{y}_{\lambda^*}}$ denote, respectively, the λ and λ^* average distributions of $V_{\mathbf{x}_1}, \ldots, V_{\mathbf{x}_\ell}$ and $V_{\mathbf{y}_1}, \ldots, V_{\mathbf{y}_{\ell^*}}$. Also, let $D \sim \lambda$, $D^* \sim \lambda^*$, $X_D \sim V_{\mathbf{x}_\lambda}$, and $Y_{D^*} \sim V_{\mathbf{y}_{\lambda^*}}$.

By Proposition 2, $JS_\lambda(V_{\mathbf{x}_1}, \ldots, V_{\mathbf{x}_\ell}) = I(D;\ X_D)$ and $JS_{\lambda^*}(V_{\mathbf{y}_1}, \ldots, V_{\mathbf{y}_{\ell^*}}) = I(D^*;\ Y_{D^*})$. $V_{\mathbf{x}_\lambda}$ can be written as the convex combination: $V_{\mathbf{x}_\lambda} = (1 - 24\epsilon) \cdot U_m + 24\epsilon \cdot F$, where U_m is the uniform distribution on $[m]$ and F is the distribution of a random element in a random set of $\mathcal{F}$. Formally, $F(i) = \frac{2f_i}{\ell m}$ for $i \in [m]$, where f_i is the number of sets in $\mathcal{F}$ that contain i. Similarly, $V_{\mathbf{y}_{\lambda^*}} = (1 - 24\epsilon) \cdot U_m + 24\epsilon \cdot F^*$, where F^* is the distribution of a random element in a random set in $\mathcal{F}^*$. Clearly, $F^* = U_m$, and therefore $V_{\mathbf{y}_{\lambda^*}} = U_m$.

$I(D;\ X_D) \ = \ H(X_D) - H(X_D \mid D) \ \leq \ H(Y_{D^*}) - H(X_D \mid D)$. The inequality follows from the fact Y_{D^*} is uniformly distributed on $[m]$ and thus has the maximum entropy among all distributions on $[m]$. By the definition of conditional entropy and the fact λ is independent of $V_{\mathbf{x}_j}$ for all j, $H(X_D \mid D) = \frac{1}{\ell}\sum_{j\in[\ell]} H(V_{\mathbf{x}_j})$. Similarly, $H(Y_{D^*} \mid D^*) = \frac{1}{\ell^*}\sum_{j\in[\ell^*]} H(V_{\mathbf{y}_j})$. Note that the entropy of $V_{\mathbf{x}_j}$ depends only on the size of $\mathcal{F}_j$ and not on $\mathcal{F}_j$ itself. Since all the subsets in $\mathcal{F}$ and in $\mathcal{F}^*$ are of equal size, it follows that: $H(X_D \mid D) = H(Y_{D^*} \mid D^*)$. Thus, $I(D;\, X_D) \le H(Y_{D^*}) - H(Y_{D^*} \mid D^*) = I(D^*;\, Y_{D^*})$.

It thus suffices to show that $JS_{\lambda^*}(V_{\mathbf{y}_1}, \ldots, V_{\mathbf{y}_{\ell^*}}) \le O(\epsilon^2)$. $JS_{\lambda^*}(V_{\mathbf{y}_1}, \ldots, V_{\mathbf{y}_{\ell^*}}) = \frac{1}{\ell^*}\sum_{j\in[\ell^*]} D_{KL}(V_{\mathbf{y}_j} \parallel V_{\mathbf{y}_{\lambda^*}})$, by definition. Recall that $V_{\mathbf{y}_{\lambda^*}} = U_m$. A simple calculation now shows that $\forall j \in [\ell^*]$, $D_{KL}(V_{\mathbf{y}_j} \parallel U_m) \le O(\epsilon^2)$. □

7.2 Low rank matrix approximation

THEOREM 9. *Let $0 < \epsilon < 1/\sqrt{24}$ and $\frac{2}{\min\{m,n\}} < \delta < \frac{1}{2}$. Any sampling algorithm Δ that (ϵ,δ)-approximates LRM_k requires $\Omega(m+n)$ queries.*

PROOF. We start by proving that LRM_k is ϵ-row and ϵ-column permutation commutative. Let $\mathbf{A}$ be any $m \times n$ matrix, let $\mathbf{\Sigma}$ be any row permutation matrix, and let $\mathbf{\Pi}$ be any column permutation matrix. Pick any $\mathbf{B}$ in the ϵ-approximation set of $\mathbf{\Sigma A \Pi}$. That is, $||\mathbf{\Sigma A\Pi} - \mathbf{B}||_F \le ||\mathbf{\Sigma A\Pi} - (\mathbf{\Sigma A\Pi})_k||_F + \epsilon\,||\mathbf{\Sigma A\Pi}||_F$. We need to show that $\mathbf{\Sigma}^{-1}\mathbf{B\Pi}^{-1}$ is in the ϵ-approximation set of $\mathbf{A}$.

Note that for any matrix $\mathbf{M}$, any row permutation matrix $\mathbf{U}$, and any column permutation matrix $\mathbf{V}$, (1) $\mathbf{UMV}$ has the same rank as $\mathbf{M}$; (2) $||\mathbf{UMV}||_F = ||\mathbf{M}||_F$; (3) $\mathbf{U}^{-1}$ and $\mathbf{V}^{-1}$ are also permutation matrices; and (4) $(\mathbf{UMV})_k = \mathbf{UM}_k\mathbf{V}$. It follows that $\mathbf{\Sigma}^{-1}\mathbf{B\Pi}^{-1}$ is a rank k matrix and

$$\begin{aligned}
||\mathbf{A} - \mathbf{\Sigma}^{-1}\mathbf{B\Pi}^{-1}||_F &= ||\mathbf{\Sigma A\Pi} - \mathbf{B}||_F \\
&\le ||\mathbf{\Sigma A\Pi} - (\mathbf{\Sigma A\Pi})_k||_F + \epsilon\,||\mathbf{\Sigma A\Pi}||_F \\
&= ||\mathbf{\Sigma A\Pi} - \mathbf{\Sigma A}_k\mathbf{\Pi}||_F + \epsilon\,||\mathbf{A}||_F \\
&= ||\mathbf{A} - \mathbf{A}_k||_F + \epsilon\,||\mathbf{A}||_F\,.
\end{aligned}$$

Let q be the worst-case query cost of the algorithm Δ. Since LRM_k is ϵ-row-permutation commutative, we can assume that Δ queries q full rows of $\mathbf{A}$, which are chosen uniformly at random with replacement. The base distribution of the query distribution $\mu_{\mathbf{A}}$ of Δ on $\mathbf{A}$ is a distribution $\nu_{\mathbf{A}}$ on $[m] \times \mathbb{R}^n$, which is uniform on the rows of $\mathbf{A}$. We will show that $q \ge \Omega(m)$ by constructing a set of pairwise disjoint inputs $\{\mathbf{A}_1, \ldots, \mathbf{A}_\ell\}$ of size $\ell = 2^{\Omega(k)}$ and prove that $JS_\lambda(\nu_{\mathbf{A}_1}, \ldots, \nu_{\mathbf{A}_\ell}) = O(k/m)$, where λ is the uniform distribution on $[\ell]$. The lower bound would then follow from Theorem 7. An equivalent argument shows that $q \ge \Omega(n)$.

We use again set designs to construct the set of inputs. Given a family $\mathcal{F} = \mathcal{F}_1, \ldots, \mathcal{F}_\ell$ of subsets of $[2k]$ of size k each, we define a corresponding family of input matrices $\mathcal{S}_{\mathcal{F}}\{\mathbf{A}_1, \ldots, \mathbf{A}_\ell\}$. All the entries of $\mathbf{A}_j$ are 0, except for the entries (i,i), $1 \le i \le 2k$, $i \in \mathcal{F}_j$, that are 1.

Let $\mathcal{F}$ be a $(k, \frac{11}{12}k)$-design of size $\ell = 2^{\Omega(k)}$. We first prove that for all $j \ne j' \in [\ell]$, $\mathbf{A}_j$ and $\mathbf{A}_{j'}$ are ϵ-disjoint. Note that both $\mathbf{A}_j$ and $\mathbf{A}_{j'}$ are of rank k, and therefore in order to be ϵ-disjoint it suffices that $||\mathbf{A}_j - \mathbf{A}_{j'}||_F \ge \epsilon(||\mathbf{A}_j||_F + ||\mathbf{A}_{j'}||_F)$. Indeed, $||\mathbf{A}_j||_F = ||\mathbf{A}_{j'}||_F = \sqrt{k}$, while $||\mathbf{A}_j - \mathbf{A}_{j'}||_F^2 = |\mathcal{F}_j \setminus \mathcal{F}_{j'}| + |\mathcal{F}_{j'} \setminus \mathcal{F}_j| \ge k/6$. The assertion follows now from the fact $\epsilon \le 1/\sqrt{24}$.

Let $\mathbf{0}, \mathbf{e}_1, \ldots, \mathbf{e}_n$ denote, respectively, the all-zero and the standard unit vectors in $\mathbb{R}^n$. Recall that the base distribution $\nu_{\mathbf{A}_j}$ is uniform on rows of $\mathbf{A}_j$. Thus, $\nu_{\mathbf{A}_j}$ is uniform on the following set of pairs: all pairs $(i, \mathbf{e}_i)$, for which $1 \le i \le 2k$, $i \in \mathcal{F}_j$, all the pairs $(i, \mathbf{0})$, $1 \le i \le 2k$, $i \notin \mathcal{F}_j$, and all the pairs $(i, \mathbf{0})$, $i > 2k$.

A similar argument to the one done in the proof of Theorem 8 shows that $JS_\lambda(\nu_{\mathbf{A}_1}, \ldots, \nu_{\mathbf{A}_\ell}) \le JS_{\lambda^*}(\nu_{\mathbf{B}_1}, \ldots, \nu_{\mathbf{B}_{\ell^*}})$, where $\mathbf{B}_1, \ldots, \mathbf{B}_{\ell^*}$ are the inputs corresponding to the family of subsets $\mathcal{F}^*$, which consists of all the $\ell^* = \binom{2k}{k}$ subsets of $[2k]$ of size k, and λ^* is the uniform distribution on $[\ell^*]$. It thus suffices to show $JS_{\lambda^*}(\nu_{\mathbf{B}_1}, \ldots, \nu_{\mathbf{B}_{\ell^*}}) \le O(k/m)$.

Note that $\nu_{\mathbf{B}_{\lambda^*}}$, the average distribution of $\nu_{\mathbf{B}_1}, \ldots, \nu_{\mathbf{B}_{\ell^*}}$, assigns probability $1/2m$ to all the pairs of the form $(i, \mathbf{e}_i)$, $i \in [2k]$, probability $1/2m$ to all the pairs $(i, \mathbf{0})$, $i \in [2k]$, and probability $1/m$ to all the pairs $(i, \mathbf{0})$, $i > 2k$. A simple calculation now shows that $\forall j \in [\ell^*]$, $D_{KL}(\nu_{\mathbf{B}_j} \parallel \nu_{\mathbf{B}_{\lambda^*}}) = 2k\frac{1}{m}\log\frac{(1/m)}{(1/2m)} = 2k/m$. The lower bound follows. □

For an input matrix $\mathbf{A}$, we denote by $P_{\mathbf{A}}$ the weight distribution of its rows. We show that even if the algorithm is given the *exact* weight distribution as advice, it still requires $\Omega(k/\epsilon^2)$ row queries.

THEOREM 10. *Let $0 < \epsilon < 1/\sqrt{24}$ and $0 < \delta < 1$. Any private coins row querying sampling algorithm that (ϵ,δ)-approximates LRM_k and whose index distribution on $\mathbf{A}$ is i.i.d. with base distribution $P_{\mathbf{A}}$, requires $\Omega(\min\{k/\epsilon^2, m\})$ queries.*

PROOF. Since the algorithm's index distribution on $\mathbf{A}$ is i.i.d. with base distribution $P_{\mathbf{A}}$, then also its query distribution on $\mathbf{A}$ is i.i.d. with some base distribution $\nu_{\mathbf{A}}$ on $[m] \times \mathbb{R}^n$.

We use again Theorem 7. For that, we construct a set of pairwise disjoint input $\mathcal{S} = \{\mathbf{A}_1, \ldots, \mathbf{A}_\ell\}$ of size $\ell = 2^{\Omega(k)}$ and prove that $JS_\lambda(\nu_{\mathbf{A}_1}, \ldots, \nu_{\mathbf{A}_\ell}) \le O(\max(\epsilon^2, k/m))$, where λ is the uniform distribution on $[\ell]$.

The construction of the family of disjoint inputs is similar to the one used in the proof of Theorem 9. Let $r \stackrel{\text{def}}{=} \max\{48\epsilon^2 m, 2(k-1)\}$. For simplicity, we assume $2(k-1)$ divides r. Given a family $\mathcal{F} = \mathcal{F}_1, \ldots, \mathcal{F}_\ell$ of subsets of $[2(k-1)]$, each of size $k-1$, we define the following corresponding family of inputs $\mathbf{A}_1, \ldots, \mathbf{A}_\ell$. All the entries of $\mathbf{A_j}$ are 0, except for the following ones, which are 1: all the entries $(i, 2k)$ for $i > r$, and all the entries $(i,i), (i+2(k-1), i), (i+4(k-1), i), \ldots, (i+(r-2(k-1)), i)$ for $i \in \mathcal{F}_j$.

Assume $\mathcal{F}$ is a $(k-1, \frac{11}{12}(k-1))$ design of size $\ell = 2^{\Omega(k)}$ (Proposition 3). We first show that each pair of inputs in the family $\mathbf{A}_j, \mathbf{A}_{j'}$, $j \ne j' \in [\ell]$, are ϵ-disjoint. Note that $\mathbf{A}_j$ and $\mathbf{A}_{j'}$ are of rank k; thus, it suffices to prove that $||\mathbf{A}_j - \mathbf{A}_{j'}||_F \ge \epsilon\left(||\mathbf{A_j}||_F + ||\mathbf{A}_{j'}||_F\right)$: $||\mathbf{A}_j - \mathbf{A}_{j'}||_F^2 = \frac{r}{2(k-1)} \cdot (|\mathcal{F}_j \setminus \mathcal{F}_{j'}| + |\mathcal{F}_{j'} \setminus \mathcal{F}_j|) \ge \frac{r}{2(k-1)} \cdot \frac{k-1}{6} = \frac{r}{12} \ge 4\epsilon^2 m$, while $||\mathbf{A}_j||_F = ||\mathbf{A}_{j'}||_F \le \sqrt{m}$.

Since all the non-zero rows of $\mathbf{A}_j$ have the same weight, $\nu_{\mathbf{A}_j}$ is uniform on the set of pairs $(i + 2d(k-1), \mathbf{e}_i)$, for $i \in \mathcal{F}_j, d = 0, \ldots, r/(2(k-1)) - 1$ and $(i, \mathbf{e}_{2k})$, $i > r$. Let $s \stackrel{\text{def}}{=} m - r/2$ be the size of this set.

Our next goal is to bound $JS_\lambda(\nu_{\mathbf{A}_1}, \ldots, \mathbf{A}_\ell)$. Similar to the proof of Theorem 8, we can show $JS_\lambda(\nu_{\mathbf{A}_1}, \ldots, \nu_{\mathbf{A}_\ell}) \le JS_{\lambda^*}(\nu_{\mathbf{B}_1}, \ldots, \nu_{\mathbf{B}_{\ell^*}})$, where $\mathbf{B}_1, \ldots, \mathbf{B}_{\ell^*}$ are the $\ell^* = \binom{2(k-1)}{k-1}$ inputs corresponding to the family of all subsets of $[2(k-1)]$ of size $k-1$, and λ^* is the uniform distribution on $[\ell^*]$. Let $\nu_{\mathbf{B}_{\lambda^*}}$ be the average distribution of $\nu_{\mathbf{B}_1}, \ldots, \nu_{\mathbf{B}_{\ell^*}}$ w.r.t. λ^*. It is easy to verify that $\nu_{\mathbf{B}_{\lambda^*}}$ assigns probability $1/2s$ to all the pairs of the form $(i + 2d(k-1), \mathbf{e}_i)$, $i \in [2(k-1)], d = 0, \ldots, r/(2(k-1))$ and probability $1/s$ to all the pairs of the form $(i, \mathbf{e}_{2k})$, $i > r$. Therefore, for each $j \in [\ell^*]$,

$D_{KL}(\nu_{\mathbf{B}_j} \parallel \nu_{\mathbf{B}_{\lambda^*}}) = \frac{r}{2} \cdot \frac{1}{s} \log \frac{(1/s)}{(1/2s)} = r/2s$. Note that $2s = 2(m - r/2) \geq m$. Therefore, $JS_\lambda(\nu_{\mathbf{B}_1}, \ldots, \nu_{\mathbf{B}^*_\ell}) \leq r/m \leq O(\max(\epsilon^2, k/m))$. The lower bound follows. □

Since an algorithm that queries single entries of the matrix can be simulated by an algorithm that queries full rows of the matrix with the same number of queries, we obtain the following:

COROLLARY 1. *For any $0 < \epsilon < 1/\sqrt{24}$ and any $0 < \delta < 1$, any sampling algorithm that (ϵ, δ)-approximates* LRM_k*, that queries single entries, and whose queries on input $\mathbf{A}$ are in a set of rows that are chosen independently according to $P_{\mathbf{A}}$, requires $\Omega(\min\{k/\epsilon^2, m\})$ queries.*

7.3 Matrix reconstruction

In the following $P_{\mathbf{A}}$ and $Q_{\mathbf{A}}$ denote the weight distributions of the rows and columns of a matrix $\mathbf{A}$.

THEOREM 11. *Let $0 < \epsilon < 1/\sqrt{24}$ and $\frac{2}{mn} < \delta < \frac{1}{2}$. Any sampling algorithm Δ that (ϵ, δ)-approximates* MR_F *requires $\Omega(mn)$ queries.*

PROOF SKETCH. We first note that MR_F is ϵ-permutation commutative. It follows that we can assume, WLOG, that Δ is a private-coin sampling algorithm which queries single entries of $\mathbf{A}$ chosen uniformly at random with replacemet.

The family of ϵ-disjoint inputs is constructed, as usual, using set designs. This time we pick a $(\frac{mn}{2}, \frac{11}{24}mn)$-design $\mathcal{F}_1, \ldots, \mathcal{F}_\ell \subseteq [mn]$ of size $\ell = 2^{\Omega(mn)}$. We think of the set of input matrices as vectors (we span the rows of the matrix one after the other to form an mn-dimensional vector). The input matrix corresponding to $\mathcal{F}_j$ is the matrix formed from the 0-1 characteristic vector of the set $\mathcal{F}_j$. It is now standard to verify that each two inputs in the family are ϵ-disjoint, as long as $\epsilon < 1/\sqrt{24}$, and that the Jensen-Shannon divergence of the corresponding query distributions is $O(1)$. Applying Theorem 7 completes the proof. □

THEOREM 12. *Let $0 < \epsilon < 1/\sqrt{288}$ and $\frac{2}{\min\{n,m\}} < \delta < \frac{1}{2}$. Any sampling algorithm Δ that (ϵ, δ)-approximates* MR_2 *requires $\Omega(m + n)$ queries. Any private-coin row-querying algorithm whose index distribution on $\mathbf{A}$ is i.i.d. with base distribution $P_{\mathbf{A}}$ requires $\Omega(m)$ row queries. Any private-coin column-querying algorithm whose index distribution on $\mathbf{A}$ is i.i.d. with base distribution $Q_{\mathbf{A}}$ requires $\Omega(n)$ column queries.*

PROOF SKETCH. It is easy to verify that MR_2 is ϵ-row permutation commutative. We can thus assume that the algorithm Δ is private-coin row-querying and that its index distribution on every input is uniform with replacemet.

To construct a family of $2^{\Omega(m)}$ ϵ-disjoint inputs, we use an $(\frac{m}{2}, \frac{11}{24}m)$-design $\mathcal{F}_1, \ldots, \mathcal{F}_\ell \subseteq [m]$. We associate with $\mathcal{F}_j$ an input matrix, which is all-zero except for its first column which is the characteristic vector of $\mathcal{F}_j$. It is easy to verify that for $\epsilon < 1/\sqrt{24}$, each two such inputs are ϵ-disjoint and that the Jensen-Shannon divergence of the corresponding query distributions is $O(1)$. An $\Omega(m)$ lower bound then follows from Theorem 7.

MR_2 is also ϵ-column permutation commutative, so we can now assume that Δ is private-coin column-querying and that its index distribution is uniform with replacemet on any input. We construct a family of $2^{\Omega(n)}$ ϵ-disjoint inputs as follows: let $\mathcal{F}_1, \ldots, \mathcal{F}_\ell \subseteq [n]$ be a $(\frac{n}{2}, \frac{11}{24}n)$-design. We associate with $\mathcal{F}_j$ an input matrix $\mathbf{A}_j$, all of whose rows are identical and equal to the characteristic vector of $\mathcal{F}_j$ multiplied by the scalar $\frac{1}{\sqrt{mn}}$. First note that for each j, $||\mathbf{A}_j||_F^2 = m \cdot \frac{n}{2} \cdot \frac{1}{mn} = \frac{1}{2}$. For two distinct j, j', all the rows of $\mathbf{B} \stackrel{\text{def}}{=} \mathbf{A}_j - \mathbf{A}_{j'}$ are identical and equal to the following:

$$\mathbf{B}_{a,b} = \begin{cases} 0 & \text{if } b \in \mathcal{F}_j \cap \mathcal{F}_{j'} \text{ or } b \notin \mathcal{F}_j \cup \mathcal{F}_{j'} \\ \frac{1}{\sqrt{mn}} & \text{if } b \in \mathcal{F}_j \setminus \mathcal{F}_{j'} \\ -\frac{1}{\sqrt{mn}} & \text{if } b \in \mathcal{F}_{j'} \setminus \mathcal{F}_j \end{cases}$$

Take $\mathbf{x} \in \mathbb{R}^n$ to be the following unit vector:

$$\mathbf{x}_b = \begin{cases} -\frac{1}{\sqrt{n}} & \text{if } b \in \mathcal{F}_{j'} \setminus \mathcal{F}_j \\ \frac{1}{\sqrt{n}} & \text{Otherwise} \end{cases}$$

Then, $\forall a \in [m]$, $(\mathbf{Bx})_a^2 = \frac{1}{n} \cdot \frac{1}{mn} \cdot (|\mathcal{F}_j \setminus \mathcal{F}_{j'}| + |\mathcal{F}_{j'} \setminus \mathcal{F}_j|)^2 \geq \frac{1}{mn^2} \cdot (\frac{n}{12})^2 = \frac{1}{144m}$. Therefore, $||\mathbf{B}||_2^2 \geq \frac{1}{144}$. It follows that $||\mathbf{A}_j - \mathbf{A}_{j'}||_2 \geq \epsilon(||\mathbf{A}_j||_F + ||\mathbf{A}_{j'}||_F)$ (implying $\mathbf{A}_j$ and $\mathbf{A}_{j'}$ are ϵ-disjoint) as long as $\epsilon \leq 1/\sqrt{288}$. It is now easy to verify that the Jensen-Shannon divergence of uniform column distributions corresponding to this family of inputs is $O(1)$. An $\Omega(n)$ lower bound now follows from Theorem 7.

Both constructions derive similar lower bounds for row-querying algorithms that query according to $P_{\mathbf{A}}$ and to column-querying algorithms that query according to $Q_{\mathbf{A}}$. In the first construction, the weight of all the non-zero rows in each input matrix is 1, and therefore $P_{\mathbf{A}}$ simply selects uniformly at random from the set of non-zero rows. A simple calculation shows that the Jensen-Shannon divergence of the corresponding family of query distributions in this case too is $O(1)$. Similarly, in the second construction, the weights of all non-zero columns are identical and equal to $\sqrt{\frac{m}{n}}$, implying an algorithm that uses $Q_{\mathbf{A}}$ picks a non-zero column uniformly at random. Again, a simple calculation shows that in this case too the Jensen-Shannon divergence is $O(1)$. □

8. EXPECTED QUERY COMPLEXITY

In this section we prove a lower bound on the expected query complexity via the KL divergence. The lower bound is based on the reduction from statistical classification and on a lower bound on the expected sample cost of two-class classification. The latter is a well-known result from statistics called the *optimality of the sequential probability ratio test* (cf. [21]):

THEOREM 13. *Let μ_0, μ_1 be i.i.d. sequential distributions on a domain $\mathcal{B}$ with base distributions ν_0, ν_1. Then, the expected sample cost q_E of any deterministic δ-error classifier for μ_0, μ_1 satisfies:*

$$q_E > \frac{(1-\delta) \cdot \log\left(\frac{1}{\delta} - 1\right)}{\min\{D_{KL}(\nu_0 \parallel \nu_1), D_{KL}(\nu_1 \parallel \nu_0)\}}.$$

The proof appears in [21] and also in the full version of this paper. The above lower bound holds only for deterministic classifiers, but can easily be extended to randomized classifiers. Using now the reduction from classification to sampling, we deduce the following lower bound on the expected query cost of index-oblivious sampling algorithms:

THEOREM 14. *Let Δ be an index-oblivious sampling algorithm that (ϵ, δ)-approximates a function $f : \mathcal{X}^n \to \mathcal{Y}$.*

Let $\mathbf{x}$ and $\mathbf{y}$ be any pair of ϵ-disjoint inputs, such that the answer distributions of Δ on $\mathbf{x}$ and $\mathbf{y}$ are i.i.d. with base distributions $\nu_{\mathbf{x}}$ and $\nu_{\mathbf{y}}$. Then, the expected query cost q_E of Δ satisfies:

$$q_E \;>\; \frac{(1-4\delta)\cdot\log\left(\frac{1}{4\delta}-1\right)}{4\cdot\min\left\{D_{KL}(\nu_{\mathbf{x}}\parallel\nu_{\mathbf{y}}), D_{KL}(\nu_{\mathbf{y}}\parallel\nu_{\mathbf{x}})\right\}}.$$

The theorem is meaningful only for index-oblivious algorithms, since the KL divergence between the query distributions (as opposed to the answer distributions) of any two different inputs is always infinite. The bound is yet useful for symmetric functions, for which index-oblivious algorithms are the best possible.

We next demonstrate the technique by an optimal lower bound on the expected query complexity of the mean. Given $x_1, \ldots, x_n \in [0,1]$, the average of $O\left(1/\epsilon^2 \cdot \log(1/\delta)\right)$ uniformly chosen samples is within ϵ of $\frac{1}{n}\sum_i x_i$ with probability at least $1-\delta$ (Chernoff-Hoeffding bound). Canetti *et al.* [6] proved that this bound is tight w.r.t. worst-case query complexity. Radhakrishnan and Ta-Shma [20] (implicitly) extended to the expected query complexity. We provide an elementary proof for the latter (albeit, we need to restrict ϵ and δ more than [20] do):

THEOREM 15. *Let $\Omega\left(\sqrt{\frac{\log(1/\delta)}{\sqrt{\delta^3 n}}}\right) < \epsilon \le \frac{1}{4}$, and let $\sqrt[3]{\frac{1}{2n}} < \delta < \frac{1}{16}$. Then, the expected query cost q of any sampling algorithm that (ϵ,δ)-approximates the mean is $\Omega\left(\frac{1}{\epsilon^2}\cdot\log\frac{1}{\delta}\right)$.*

PROOF. Since the mean is a symmetric function, we can assume the algorithm is index-oblivious and chooses its query positions uniformly at random with replacement. Consider the two inputs $\mathbf{x}$ and $\mathbf{y}$. $\mathbf{x}$ has $(\frac{1}{2}+\epsilon)n$ 0's and rest are 1's; $\mathbf{y}$ has $(\frac{1}{2}-\epsilon)n$ 0's and the rest are 1's. Since the mean of $\mathbf{x}$ is $1/2-\epsilon$ and the mean of $\mathbf{y}$ is $1/2+\epsilon$, $\mathbf{x}$ and $\mathbf{y}$ are ϵ-disjoint. Note that the answer distribution on $\mathbf{x}$, $\nu_{\mathbf{x}}$, is a Bernoulli distribution with probability of success $\frac{1}{2}-\epsilon$. Similarly, the answer distribution on $\mathbf{y}$, $\nu_{\mathbf{y}}$, is a Bernoulli distribution with probability of success $\frac{1}{2}+\epsilon$. A simple calculation shows that $D_{KL}(\nu_{\mathbf{x}}\parallel\nu_{\mathbf{y}}) = O(\epsilon^2)$. The lower bound follows from Theorem 14. $\square$

9. ACKNOWLEDGMENTS

I thank T.S. Jayram and D. Sivakumar for their help with the simulation lemmas for symmetric functions. I thank Ran El-Yaniv for pointing out references in the information theory and statistics literature.

10. REFERENCES

[1] D. Achlioptas and F. McSherry. Fast computation of low rank matrix approximations. In *Proc. 33rd STOC*, pages 611–618, 2001.

[2] Y. Azar, A. Fiat, A. R. Karlin, F. McSherry, and J. Saia. Spectral analysis of data. In *Proc. 33rd STOC*, pages 619–626, 2001.

[3] Z. Bar-Yossef. *The Complexity of Massive Data Set Computations*. PhD thesis, Computer Science Division, U.C. Berkeley, 2002.

[4] Z. Bar-Yossef, R. Kumar, and D. Sivakumar. Sampling algorithms: Lower bounds and applications. In *Proc. 33rd STOC*, pages 266–275, 2001.

[5] T. Batu, L. Fortnow, R. Rubinfeld, W. D. Smith, and P. White. Testing that distributions are close. In *Proc. 41st FOCS*, pages 259–269, 2000.

[6] R. Canetti, G. Even, and O. Goldreich. Lower bounds for sampling algorithms for estimating the average. *Information Processing Letters*, 53:17–25, 1995.

[7] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.

[8] P. Drineas, A. M. Frieze, R. Kannan, S. Vempala, and V. Vinay. Clustering in large graphs and matrices. In *Proc. 10th SODA*, pages 291–299, 1999.

[9] P. Drineas and R. Kannan. Pass efficient algorithm for approximating large matrices. Manuscript, 2002.

[10] P. Drineas, I. Kerenidis, and P. Raghavan. Competitive recommendation systems. In *Proc. 34th STOC*, pages 82–90, 2002.

[11] A. M. Frieze, R. Kannan, and S. Vempala. Fast Monte-Carlo algorithms for finding low-rank approximations. In *Proc. 39th FOCS*, pages 370–378, 1998.

[12] O. Goldreich and L. Trevisan. Three theorems regarding testing graph properties. In *Proc. 42nd FOCS*, pages 460–469, 2001.

[13] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, third edition, 1996.

[14] M. Gutman. Asymptotically optimal classification for multiple tests with empirically observed statistics. *IEEE Transactions on Information Theory*, 35(2):401–408, 1989.

[15] T. Kailath. The divergence and Bhattacharyya distance measures in signal selection. *IEEE Transactions on Communication Technology*, COM-15(1):52–60, 1967.

[16] S. Kullback and A. Leibler. On information and sufficiency. *IEEE Transactions on Information Theory*, 22:79–86, 1951.

[17] J. Lin. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

[18] I. Newman. Private vs. common random bits in communication complexity. *Information Processing Letters*, 39:67–71, 1991.

[19] N. Nisan and A. Wigderson. Hardness vs. randomness. *JCSS*, 49(2):149–167, 1994.

[20] J. Radhakrishnan and A. Ta-Shma. Bounds for dispersers, extractors, and depth-two superconcentrators. *SIAM J. on Discr. Math.*, 13(1):2–24, 2000.

[21] D. Siegmund. *Sequential Analysis - Tests and Confidence Intervals*. Springer-Verlag, 1985.

[22] G. T. Toussaint. On some measures of information and their application to pattern recognition. In *Proc. Conference on Measures of Information and Their Applications*, pages 21–28, 1974.

[23] J. Ziv. On classification with empirically observed statistics and universal data compression. *IEEE Transactions on Information Theory*, 34:278–286, 1988.