Electronic Colloquium on Computational Complexity, Revision 2 of Report No. 64 (2012)

Statistical Algorithms and a Lower Bound for Detecting Planted Cliques

Vitaly Feldman* Elena Grigorescu†‡ Lev Reyzin§‡

Santosh S. Vempala¶‡ Ying Xiao¶‡

Abstract

We introduce a framework for proving lower bounds on computational problems over distributions against algorithms that can be implemented using access to a *statistical query* oracle. For such algorithms, access to the input distribution is limited to obtaining an estimate of the expectation of any given function on a sample drawn randomly from the input distribution, rather than directly accessing samples. Most natural algorithms of interest in theory and in practice, e.g., moments-based methods, local search, standard iterative methods for convex optimization, MCMC and simulated annealing can be implemented in this framework. Our framework is based on, and generalizes, the statistical query model in learning theory [Kearns, 1998].

Our main application is a nearly optimal lower bound on the complexity of *any* statistical query algorithm for detecting planted bipartite clique distributions (or planted dense subgraph distributions) when the planted clique has size $O(n^{1/2-\delta})$ for any constant $\delta > 0$. The assumed hardness of variants of these problems has been used to prove hardness of several other problems and as a guarantee for security in cryptographic applications. Our lower bounds provide concrete evidence of hardness, thus supporting these assumptions.

[*] IBM Almaden Research Center.

[†] Department of Computer Science, Purdue University. Supported by the National Science Foundation under Grant #1019343 to the Computing Research Association for the CIFellows Project.

[‡] Research supported in part by NSF awards CCF-0915903 and CCF-1217793.

[§] Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago. Research supported by a Simons Postdoctoral Fellowship.

[¶] School of Computer Science, Georgia Institute of Technology. Research supported by NSF awards CCF-1217793 and EAGER-1415498.

ISSN 1433-8092

Contents

1 Introduction

We study the complexity of problems where the input consists of independent samples from an unknown distribution. Such problems are at the heart of machine learning and statistics (and their numerous applications) and also occur in many other contexts such as compressed sensing and cryptography. While several methods have been developed to estimate the sample complexity of such problems (e.g. VC dimension [Vapnik and Chervonenkis, 1971] and Rademacher complexity [Bartlett and Mendelson, 2002]), proving lower bounds on the computational complexity of these problems has been much more challenging. The traditional approach to proving lower bounds is via reductions and by finding distributions that can generate instances of some problem conjectured to be intractable (e.g., assuming $\mathsf{NP} \neq \mathsf{RP}$).

Here we present a different approach. We show that algorithms which access the unknown distribution only via a *statistical query (SQ) oracle* have high complexity, unconditionally. Most algorithmic approaches used in practice and in theory on a wide variety of problems can be implemented using only access to such an oracle; these include Expectation Maximization (EM) [Dempster et al., 1977], local search, MCMC optimization [Tanner and Wong, 1987, Gelfand and Smith, 1990], simulated annealing [Kirkpatrick et al., 1983, Černý, 1985], first and second order methods for linear/convex optimization, [Dunagan and Vempala, 2008, Belloni et al., 2009], k-means, Principal Component Analysis (PCA), Independent Component Analysis (ICA), Naïve Bayes, Neural Networks and many others (see [Chu et al., 2006] and [Blum et al., 2005] for proofs and many other examples). In fact, we are aware of only one algorithm that provably does not have a statistical query counterpart: Gaussian elimination for solving linear equations over a field (*e.g.* mod 2).

Informally, a statistical query oracle provides an estimate of the expected value of any given bounded real-valued function within some tolerance. Many popular algorithms rely only on the average value of various functions over random samples (commonly referred to as *empirical averages*). Standard Chernoff-Hoeffding bounds imply that the average value of a bounded function on the independent samples will be highly concentrated around the expectation on the unknown distribution (and, indeed in many cases the empirical average is used precisely to obtain an estimate of the expectation). As a result such algorithms can often be equivalently analyzed in our oracle-based model.

Our approach also allows proving lower bounds against algorithms that rely on a 1*-bit sampling* oracle, referred to as 1-bit sampling algorithms. This oracle provides the value of any Boolean function on a fresh random sample from the distribution. Many existing algorithms require only such limited access to random samples. Others can be implemented using such access to samples (possibly using a polynomially larger number of samples). For brevity, we refer to algorithms that rely on either of these types of oracles as *statistical algorithms*.

For example, many problems over distributions are solved using convex programs. Such a problem is typically formulated as finding an approximation to $\min_{z \in K} \mathbf{E}_{x \sim D}[f(x,z)]$ for some convex set K and functions $f(x,\cdot)$ that are convex in the second parameter z. A standard approach (both in theory and practice) to solve such a problem is to use a gradient descent-based technique. The gradient of the objective function is

$$\nabla_z \mathop{\mathbf{E}}_{x}[f(x,z)] = \mathop{\mathbf{E}}_{x}[\nabla_z f(x,z)]$$

and is usually estimated using the average value of $\nabla_z f(x,z)$ on (some of) the given random samples. However, standard analysis of gradient descent-based algorithms implies that a sufficiently accurate estimate of each of the coordinates of $\mathbf{E}_x[\nabla_z f(x,z)]$ would also suffice. Hence, for an objective function of the form above, gradient descent can be implemented using either of the above oracles (detailed analysis of such implementations can be found in a subsequent work [Feldman et al., 2015]).

The key motivation for our framework is the empirical observation that almost all algorithms that work on random samples are either already statistical in our sense or have natural statistical counterparts. Thus, lower bounds for statistical algorithms can be directly translated into lower bounds against a large number of existing approaches. We present the formal oracle-based definitions of statistical algorithms in Section 2.

Our model is based on the *statistical query learning* model [Kearns, 1998] defined as a restriction of Valiant's [1984] PAC learning model. The primary goal of the restriction was to simplify the design of noise-tolerant learning algorithms. As was shown by Kearns and others in subsequent works, almost all classes of functions that can be learned efficiently can also be efficiently learned in the SQ model. A notable and so far unique exception is the algorithm for learning parities, based on Gaussian elimination. As was already shown by Kearns [1998], parities require exponentially many queries to learn in the SQ model. Further, Blum et al. [1994] proved that the number of SQs required for weak learning (that is, for obtaining a non-negligible advantage over the random guessing) of a class of functions C over a fixed distribution D is characterized by a combinatorial parameter of C and D, referred to as SQ-DIM(C, D), the SQ dimension.

We consider SQ algorithms in the broader context of arbitrary computational problems over distributions. We also define an SQ oracle that strengthens the oracle introduced by Kearns [1998]. For any problem over distributions we define a parameter of the problem that lower bounds the complexity of solving the problem by any SQ algorithm in the same way that SQ-DIM lower bounds the complexity of learning in the SQ model. Our techniques for proving lower bounds are also based on methods developed for lower-bounding the complexity of SQ learning algorithms. However, as we will describe later, they depart from the known techniques in a number of significant ways that are necessary for our more general setting and our applications.

The 1-bit sampling oracle and its more general k-bit version was introduced by Ben-David and Dichterman [1998]. They showed that it is equivalent (up to polynomial factors) to the SQ oracle. Using our stronger SQ oracle we sharpen this equivalence. This sharper relationship is crucial for obtaining meaningful lower bounds against 1-bit sampling algorithms in our applications.

We demonstrate our techniques by applying them to the problems of detecting planted bipartite cliques and planted bipartite dense subgraphs. We now define these problems precisely and give some background.

Detecting Planted Cliques. In the planted clique problem, we are given a graph G whose edges are generated by starting with a random graph $G_{n,1/2}$, then "planting," i.e., adding edges to form a clique on k randomly chosen vertices. Jerrum [1992] and Kucera [1995] introduced the planted clique problem as a potentially easier variant of the classical problem of finding the largest clique in a random graph [Karp, 1979]. A random graph $G_{n,1/2}$ contains a clique of size $2\log n$ with high probability, and a simple greedy algorithm can find one of size $\log n$. Finding cliques of size $(2-\epsilon)\log n$ is a hard problem for any $\epsilon > 0$. Planting a larger clique should make it easier to find one. The problem of finding the smallest k for which the planted clique can be detected in polynomial time has attracted significant attention. For $k \geq c\sqrt{n \log n}$, simply picking vertices of large degrees suffices [Kucera, 1995]. Cliques of size $k = \Omega(\sqrt{n})$ can be found using spectral methods [Alon et al., 1998, McSherry, 2001, Coja-Oghlan, 2010], via SDPs [Feige and Krauthgamer, 2000], combinatorial methods [Feige and Ron, 2010, Dekel et al., 2011], nuclear norm minimization [Ames and Vavasis, 2011] and belief propagation [Deshpande and Montanari, 2013].

While there is no known polynomial-time algorithm that can detect cliques of size below the threshold of $\Omega(\sqrt{n})$, there is a quasipolynomial algorithm for any $k \geq 2\log n$: enumerate subsets of size $2\log n$; for each subset that forms a clique, take all common neighbors of the subset; one of these will be the planted clique. This is also the fastest known algorithm for any $k = O(n^{1/2-\delta})$, where $\delta > 0$.

Some evidence of the hardness of the problem was shown by Jerrum [1992] who proved that a specific approach using a Markov chain cannot be efficient for $k = o(\sqrt{n})$. Additional evidence of hardness is given in [Feige and Krauthgamer, 2003], where it is shown that Lovász-Schrijver SDP relaxations, which include the SDP used in [Feige and Krauthgamer, 2000], cannot be used to efficiently find cliques of size $k = o(\sqrt{n})$. Most recently, lower bounds against a constant level of the more powerful Sum-of-Squares SDP hierarchy were shown by Meka et al. [2015] and Deshpande and Montanari [2015]. The problem has been used to generate cryptographic primitives [Juels and Peinado, 2000], and as a hardness assumption in a large number of works (*e.g.* [Alon et al., 2007, Hazan and Krauthgamer, 2011, Minder and Vilenchik, 2009, Berthet and Rigollet, 2013, Dughmi, 2014]).

We focus on the bipartite planted clique problem, where a $(k \times k)$-biclique is planted in a random bipartite graph. A densest-subgraph version of the bipartite planted clique problem has been used as a hard problem for cryptographic applications [Applebaum et al., 2010]. The bipartite version can be easily seen to be at least as hard as the original version. At the same time all known bounds and algorithms for the k-clique problem can be easily adapted to the bipartite case (*e.g.* [Ames and Vavasis, 2011]). Therefore it is natural to expect that new upper bounds on the planted k-clique problem would also yield new upper bounds for the bipartite case.

The starting point of our investigation for this problem is the property of the planted k-biclique problem that it has an equivalent formulation as a problem over distributions defined as follows.

Problem 1.1. *Fix an integer k, $1 \leq k \leq n$, and a subset of k indices $S \subseteq \{1, 2, \ldots, n\}$. The input distribution D_S on vectors $x \in \{0,1\}^n$ is defined as follows: with probability $1 - (k/n)$, x is uniform over $\{0,1\}^n$; and with probability k/n, x is such that its k coordinates from S are set to 1, and the remaining coordinates are uniform in $\{0,1\}$. For an integer t, the* ***distributional planted k-biclique*** *problem with t samples is the problem of finding the unknown subset S using t samples drawn randomly from D_S.*

One can view samples $x_1, \ldots, x_t$ as adjacency vectors of the vertices of a bipartite graph as follows: the bipartite graph has n vertices on the right (with k marked as members of the clique) and t vertices on the left. Each of the t samples gives the adjacency vector of the corresponding vertex on the left. It is not hard to see that for $t = n$, conditioned on the event of getting exactly k samples with planted indices, we will get a random bipartite graph with a planted $(k \times k)$-biclique (we prove the equivalence formally in Appendix A).

One interesting approach for finding the planted clique was proposed by Frieze and Kannan [2008]. They gave a reduction from finding a planted clique in a random graph to finding a direction that maximizes a 2nd order tensor norm; this was extended to general r'th order tensor norm in [Brubaker and Vempala, 2009].Specifically, they show that maximizing the r'th moment (or the 2-norm of an r'th order tensor) allows one to recover planted cliques of size $\tilde{\Omega}(n^{1/r})$. A related approach is to maximize the 3rd or higher moment of the distribution given by the distributional planted clique problem. For this approach it is natural to consider the following type of optimization algorithm: start with some unit vector u, then estimate the gradient at u (via samples), move along that direction and return to the sphere; repeat to reach an approximate local maximum. Unfortunately, over the unit sphere, the expected r'th moment function can have (exponentially) many local maxima even for simple distributions. A more sophisticated approach [Kannan] is through Markov chains or simulated annealing; it attempts to sample unit vectors from a distribution on the sphere which is heavier on vectors that induce a higher moment, e.g., u is sampled with density proportional to $e^{f(u)}$ where $f(u)$ is the expected r'th moment along u. This could be implemented by a Markov chain with a Metropolis filter [Metropolis et al., 1953, Hastings, 1970] ensuring a proportional steady state distribution. If the Markov chain were to mix rapidly, that would give an efficient approximation algorithm because sampling from the steady state likely gives a vector of high moment. At each step, all one needs is to be able to estimate $f(u)$, which can be done by sampling from the input distribution.

As we will see presently, these approaches can be easily implemented in our framework and will have provably high complexity. For the distributional planted biclique problem, SQ algorithms need $n^{\Omega(\log n)}$ queries to detect planted bicliques of size $k < n^{\frac{1}{2}-\delta}$ for any $\delta > 0$. Even stronger exponential bounds apply for the more general problem of detecting planted dense subgraphs of the same size. These bounds match the known upper bounds. To describe these results precisely and discuss exactly what they mean for the complexity of these problems, we will need to define the models of statistical algorithms, the complexity measures we use, and our main tool for proving lower bounds, a notion of statistical dimension of a set of distributions. We do this in the next section. In Section 3 we prove our general lower bound results and in Section 5 we estimate the statistical dimension of detecting planted bicliques and dense subgraphs.

2 Definitions and Overview

Here we formally define statistical algorithms and the key notion of statistical dimension, and then describe the resulting lower bounds in detail.

2.1 Problems over Distributions

We begin by formally defining the class of problems addressed by our framework.

Definition 2.1 (Search problems over distributions)**.** *For a domain X, let $\mathcal{D}$ be a set of distributions over X, let $\mathcal{F}$ be a set called* solutions *and $\mathcal{Z} : \mathcal{D} \to 2^{\mathcal{F}}$ be a map from a distribution $D \in \mathcal{D}$ to a subset of solutions $\mathcal{Z}(D) \subseteq \mathcal{F}$ that are defined to be valid solutions for D. The* distributional search problem *$\mathcal{Z}$ over $\mathcal{D}$ and $\mathcal{F}$ using t samples is to find a valid solution $f \in \mathcal{Z}(D)$ given access (to an oracle or samples from) an unknown $D \in \mathcal{D}$.*

In some settings it is natural to parameterize the set of valid solutions by additional parameters, such as accuracy. The extension of the definition to such settings is immediate. An example of a distributional search problem is the distributional planted k-biclique we described in Definition 1.1. In this case the domain X is $\{0,1\}^n$, the set of input distributions is all the distributions with a planted k-biclique $\mathcal{D} = \{D_S \mid S \subset [n],\ |S| = k\}$ and the set of solutions is the set of all subsets of size k: $\mathcal{F} = \{S \mid S \subset [n],\ |S| = k\}$. For each D_S there is a single valid solution S. For a second example, we point the reader to the distributional MAX-XOR-SAT problem in Section 4.

We note that this definition also captures decision problems by having $\mathcal{F} = \{0,1\}$. A simple example of a decision problem over distributions that is relevant to our discussion is that of distinguishing a planted biclique distribution from the uniform distribution over $\{0,1\}^n$ which we denote by U. Here the set of input distributions is $\mathcal{D} = \{U\} \cup \{D_S \mid S \subset [n],\ |S| = k\}$. The only valid solution for a planted biclique distribution D_S is 1 and the only valid solution for U is 0. For a solution $f \in \mathcal{F}$, we denote by $\mathcal{Z}_f$ the set of distributions in $\mathcal{D}$ for which f is a valid solution.

It is important to note that the number of available random samples t can have a major influence on the complexity of the problem. First, for most problems there is a minimum t for which the problem is information-theoretically solvable. This value is often referred to as the *sample complexity* of the problem. But even for t which is larger than the sample complexity of the problem, having more samples can make the problem easier computationally. For example, in the context of attribute-efficient learning, there is a problem that is intractable with few samples (under cryptographic assumptions) but is easy to solve with a larger (but still polynomial) number of samples [Servedio, 2000]. Our distributional planted biclique problem exhibits the same phenomenon.

2.2 Statistical Algorithms

The statistical query learning model of Kearns [1998] is a restriction of the PAC model [Valiant, 1984]. It introduces an oracle that allows a learning algorithm to obtain an estimate of the expectation of any bounded function of an example. A query to such an oracle is referred to as *statistical query*. Kearns showed that many known PAC learning algorithms can be expressed as algorithms using statistical queries instead of random examples themselves. The main goal of Kearns' model was to give a simple way to design algorithms tolerant to random classification noise. Since the introduction of the model SQ algorithms have been given for many more learning tasks and the model itself found applications in a number of other contexts such as differential privacy [Blum et al., 2005, Kasiviswanathan et al., 2011], learning on massively parallel architectures [Chu et al., 2006] and evolvability [Feldman, 2008].

In the same spirit, for general search problems over a distribution, we define SQ algorithms as algorithms that do not see samples from the distribution but instead have access to a SQ oracle. The first SQ oracle we define is the natural generalization of the oracle defined by Kearns [1998] to samples from an arbitrary distribution.

Definition 2.2 (STAT oracle). *Let D be the input distribution over the domain X. For a* tolerance *parameter $\tau > 0$, $STAT(\tau)$ oracle is the oracle that for any query function $h : X \to [-1,1]$, returns a value*

$$v \in \left[\mathop{\mathbf{E}}_{x \sim D}[h(x)] - \tau,\ \mathop{\mathbf{E}}_{x \sim D}[h(x)] + \tau \right].$$

The general algorithmic techniques mentioned earlier can all be expressed as algorithms using STAT oracle instead of samples themselves, in most cases in a straightforward way. We would also like to note that in the PAC learning model some of the algorithms, such as the Perceptron algorithm, did not initially appear to fall into the SQ framework but SQ analogues were later found for all known learning techniques except Gaussian elimination (for specific examples, see [Kearns, 1998] and [Blum et al., 1998]). We expect the situation to be similar even in the broader context of search problems over distributions.

The most natural realization of STAT(τ) oracle is one that computes h on $O(1/\tau^2)$ random samples from D and returns their average. Chernoff's bound implies that the estimate is within the desired tolerance (with constant probability). However, if $h(x)$ is very biased (e.g. equal to 0 with high probability), it can be estimated with fewer samples. Our primary application requires a tight bound on the number of samples necessary to solve a problem over distributions. Therefore we define a stronger version of STAT oracle which tightly captures the accuracy of an estimate of the expectation given by random samples. More formally, for a Boolean query function $h : X \to \{0, 1\}$, VSTAT(t) can return any value v for which the Binomial distribution $B(t, v)$ (sum of t independent Bernoulli variables with bias v) is statistically close (for some constant distance) to $B(t, \mathbf{E}[h])$. See Sec. 3.3 for more details on this correspondence.

Definition 2.3 (VSTAT oracle)**.** *Let D be the input distribution over the domain X. For a* sample size *parameter $t > 0$, VSTAT(t) oracle is the oracle that for any query function $h : X \to [0, 1]$, returns a value $v \in [p - \tau, p + \tau]$, where $p = \mathbf{E}_{x\sim D}[h(x)]$ and $\tau = \max\left\{\frac{1}{t}, \sqrt{\frac{p(1-p)}{t}}\right\}$.*

Note that VSTAT(t) always returns the value of the expectation within $1/\sqrt{t}$. Therefore it is no weaker than STAT$(1/\sqrt{t})$ and no stronger than STAT$(1/t)$.

The STAT and VSTAT oracles we defined can return any value within the given tolerance and therefore can make adversarial choices. We also aim to prove lower bounds against algorithms that use a more benign, 1-bit sampling oracle[1]. The 1-bit sampling oracle gives the algorithm the true value of a Boolean query function on a randomly chosen sample. This oracle is a special case of the k-bit sampling oracle introduced by Ben-David and Dichterman [1998] who refer to it as the *weak Restricted Focus of Attention (wRFA)* model and is also equivalent to the Honest SQ oracle of Yang [2001]. Learning in this model has been studied in more recent work motivated by communication constraints on data processing in a distributed computing system. [Zhang et al., 2013, Steinhardt and Duchi, 2015, Steinhardt et al., 2016].

Definition 2.4 (1-STAT oracle)**.** *Let D be the input distribution over the domain X. The 1-STAT oracle is the oracle that given any function $h : X \to \{0, 1\}$, takes an independent random sample x from D and returns $h(x)$.*

Note that the 1-STAT oracle draws a fresh sample upon each time it is called. Without resampling each time, the answers of the 1-STAT oracle could be easily used to recover any sample bit-by-bit, making it equivalent to having access to random samples. Note that the 1-STAT oracle can be used to simulate VSTAT (with high probability) by taking the average of $O(t)$ replies of 1-STAT for the same function h. While it might seem that access to 1-STAT gives an algorithm more power than access to VSTAT we will show that t samples from 1-STAT can be simulated using access to VSTAT$(O(t))$. This will allow us to translate our lower bounds on SQ algorithms with access to VSTAT to lower bounds against 1-bit sampling algorithms.

2.3 Statistical Dimension

The main tool in our analysis is an information-theoretic bound on the complexity of statistical algorithms. Our definitions originate from the statistical query (SQ) dimension [Blum et al., 1994] used to characterize SQ learning algorithms. Roughly speaking, the SQ dimension corresponds to the number of nearly uncorrelated labeling functions in a class (see Section 6.1 for the details of the definition and the relationship to our bounds).

[1] In the STOC 2013 extended abstract, this oracle is also called the *unbiased* statistical oracle

We introduce a natural generalization and strengthening of this approach to search problems over arbitrary sets of distributions and prove lower bounds on the complexity of statistical algorithms based on the generalized notion. Our definition departs from SQ dimension in three aspects. (1) Our notion applies to any set of distributions; in the learning setting all known definitions of statistical dimension require fixing the distribution over the domain and only allow varying the labeling function. Such an extension was not known even in the context of PAC learning. (2) Instead of relying on a bound on pairwise correlations, our dimension relies on a bound on average correlations in a large set of distributions. This weaker condition allows us to derive tight bounds on the complexity of SQ algorithms for the planted k-biclique problem. (3) We show that our notion of dimension also gives lower bounds for the stronger VSTAT oracle (without incurring a quadratic loss in the parameter).

We now define our dimension formally. For two functions $f, g : X \to \mathbf{R}$ and a distribution D with probability density function $D(x)$, the inner product of f and g over D is defined as

$$\langle f, g\rangle_D \doteq \mathop{\mathbf{E}}_{x\sim D}[f(x)g(x)].$$

The norm of f over D is $\|f\|_D = \sqrt{\langle f, f\rangle_D}$. We remark that, by convention, the integral from the inner product is taken only over the support of D, i.e. for $x \in X$ such that $D(x) \neq 0$. Given a distribution D over X let $D(x)$ denote the probability density function of D relative to some fixed underlying measure over X (for example uniform distribution for discrete X or Lebesgue measure over $\mathbf{R}^n$). Our bound is based on the inner products between functions of the following form: $(D'(x) - D(x))/D(x)$ where D' and D are distributions over X. For this to be well-defined, we will only consider cases where $D(x) = 0$ implies $D'(x) = 0$, in which case $D'(x)/D(x)$ is treated as 1. To see why such functions are relevant to our discussion, note that for every real-valued function f over X,

$$\mathop{\mathbf{E}}_{x\sim D'}[f(x)] - \mathop{\mathbf{E}}_{x\sim D}[f(x)] = \mathop{\mathbf{E}}_{x\sim D}\left[\frac{D'(x)}{D(x)}f(x)\right] - \mathop{\mathbf{E}}_{x\sim D}[f(x)] = \left\langle \frac{D'-D}{D}, f\right\rangle_D .$$

This means that the inner product of any function f with $(D' - D)/D$ is equal to the difference of expectations of f under the two distributions. Analyzing this quantity for an arbitrary set of functions f was the high-level approach of statistical query lower bounds for learning. Here we depart from this approach, by defining a *pairwise correlation* of two distributions, independent of any specific query function. For two distributions D_1, D_2 and a reference distribution D, their pairwise correlation is defined as:

$$\chi_D(D_1, D_2) = \left|\left\langle \frac{D_1}{D} - 1, \frac{D_2}{D} - 1\right\rangle_D\right| .$$

When $D_1 = D_2$, the quantity $\langle \frac{D_1}{D} - 1, \frac{D_1}{D} - 1\rangle_D$ is known as the $\chi^2(D_1, D)$ distance and is widely used for hypothesis testing in statistics [Pearson, 1900].

A key notion for our statistical dimension is the *average correlation* of a set of distributions $\mathcal{D}'$ relative to a distribution D. We denote it by $\rho(\mathcal{D}', D)$ and define as follows:

$$\rho(\mathcal{D}', D) \doteq \frac{1}{|\mathcal{D}'|^2}\sum_{D_1,D_2\in\mathcal{D}'}\chi_D(D_1, D_2) = \frac{1}{|\mathcal{D}'|^2}\sum_{D_1,D_2\in\mathcal{D}'}\left|\left\langle \frac{D_1}{D} - 1, \frac{D_2}{D} - 1\right\rangle_D\right| .$$

Bounds on pairwise correlations easily imply bounds on the average correlation (see Lemma 3.10 for a proof). In Section 3.2 we describe a pairwise-correlation version of our bounds. It is sufficient for some applications and generalizes the statistical query dimension from learning theory (see Section 6.1 for the details). However, to obtain our nearly tight lower bounds for planted biclique, we will need to bound the average pairwise correlation directly, and with significantly better bounds than what is possible from pairwise correlations alone.

We are now ready to define the concept of statistical dimension. We first define the statistical dimension with average correlation of a set of distributions relative to some reference distribution. It captures the complexity of distinguishing distributions in $\mathcal{D}$ from D.

Definition 2.5. *For $\bar{\gamma} > 0$, domain X, a set of distributions $\mathcal{D}$ over X and a reference distribution D over X the **statistical dimension** of $\mathcal{D}$ relative to D with average correlation $\bar{\gamma}$ is defined to be the largest value d such that for any subset $\mathcal{D}' \subseteq \mathcal{D}$, where $|\mathcal{D}'| \geq |\mathcal{D}|/d$, $\rho(\mathcal{D}', D) \leq \bar{\gamma}$. We denote it by $\mathrm{SDA}(\mathcal{D}, D, \bar{\gamma})$.*

Intuitively, the definition says that any $1/d$ fraction of the set of distributions has low pairwise correlation; the largest such d is the statistical dimension.

For general search problems over distributions we define the statistical dimension by reducing it to the statistical dimension of some set of input distributions relative to some reference distribution.

Definition 2.6. *For $\bar{\gamma} > 0$, domain X, a search problem $\mathcal{Z}$ over a set of solutions $\mathcal{F}$ and a class of distributions $\mathcal{D}$ over X, let d be the largest value such that there exists a* reference *distribution D over X and a finite set of distributions $\mathcal{D}_D \subseteq \mathcal{D}$ with the following property: for any solution $f \in \mathcal{F}$ the set $\mathcal{D}_f = \mathcal{D}_D \setminus \mathcal{Z}_f$ is non-empty and $\mathrm{SDA}(\mathcal{D}_f, D, \bar{\gamma}) \geq d$. We define the **statistical dimension** with average correlation $\bar{\gamma}$ of $\mathcal{Z}$ to be d and denote it by $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma})$.*

The statistical dimension with average correlation $\bar{\gamma}$ of a search problem over distributions gives a lower bound on the complexity of any deterministic statistical algorithm for the problem that uses queries to $\mathrm{VSTAT}(1/(3\bar{\gamma}))$.

Theorem 2.7. *Let X be a domain and $\mathcal{Z}$ be a search problem over a set of solutions $\mathcal{F}$ and a class of distributions $\mathcal{D}$ over X. For $\bar{\gamma} > 0$ let $d = \mathrm{SDA}(\mathcal{Z}, \bar{\gamma})$. Any SQ algorithm requires at least d calls to $VSTAT(1/(3\bar{\gamma}))$ oracle to solve $\mathcal{Z}$.*

In Section 3.1 we give a refinement of SDA, by introducing a parameter which additionally bounds the size of the set $\mathcal{D}_f$ (and not just that it is non-empty). This refined notion allows us to extend the lower bound to randomized SQ algorithms. In Section 3.3 we use this refined notion of SDA to also show that (with high probability) one can simulate t samples of 1-STAT using $\mathrm{VSTAT}(O(t))$. This implies that lower bounds on SDA imply lower bounds on the number of queries required by any 1-bit sampling algorithm (Theorem 3.17).

In Section 6 we show that our bounds generalize and strengthen the known results for SQ learning that are based on SQ-DIM [Blum et al., 1994, Yang, 2005]. In the statement below, the statistical dimension $\mathrm{SDA}(\mathcal{C}, D', \bar{\gamma})$ uses the average pairwise correlation of Boolean functions from a set $\mathcal{C}$ relative to a distribution D' over a domain X', that is $\langle f_1, f_2 \rangle_{D'}$, where $f_1, f_2 \in \mathcal{C}$ (rather than distributions as in the definitions above). It is formally defined in Section 6 and is always at least as large as the statistical query dimension used in earlier work in learning theory.

Theorem 2.8. *Let $\mathcal{C}$ be a set of Boolean functions, D' be a distribution over X' and let $d = \mathrm{SDA}(\mathcal{C}, D', \bar{\gamma})$ for some $\bar{\gamma} > 0$. Then any SQ algorithm that, with probability at least $2/3$, learns $\mathcal{C}$ over D' with error $\epsilon < 1/2 - \sqrt{1/(3\bar{\gamma})}$ requires at least $d/3 - 1$ queries to $VSTAT(1/(3\bar{\gamma}))$.*

At a high level, our proof works as follows. The first step of the proof is a reduction from a decision problem in which the algorithm only needs to distinguish all the distributions in the set $\mathcal{D}_D$ (except those in $\mathcal{Z}_f$ for some f) from the reference distribution D. To distinguish between distributions the algorithm needs to ask a query g such that $\mathbf{E}_D[g]$ cannot be used as a response of VSTAT$(1/(3\bar{\gamma}))$ for $D' \in \mathcal{D}_f$. In the key component of the proof we show that if a query function g to VSTAT$(1/(3\bar{\gamma}))$ distinguishes between a distribution D and any distribution $D' \in \mathcal{D}'$, then $\mathcal{D}'$ must have average correlation of at least $\bar{\gamma}$ relative to D. The condition that for any $|\mathcal{D}'| \geq |\mathcal{D}_f|/d$, $\rho(\mathcal{D}', D) < \bar{\gamma}$ then immediately implies that at least d queries are required to distinguish any distribution in $\mathcal{D}_f$ from D. We remark that an immediate corollary of this proof technique is that the decision problem in which the algorithm needs to decide whether the input distribution is in $\mathcal{D}_f$ or is equal to the reference distribution D also has statistical dimension at least d. We elaborate on this in Theorem 3.7 where we give a simplified version of our lower bound for decision problems of this type.

2.4 Applications to the Planted Biclique Problem

We prove the following lower bound for the distributional planted biclique problem.

Theorem 2.9. *For any constant $\delta > 0$, any $k \leq n^{1/2-\delta}$ and $r > 0$, at least $n^{\Omega(\log r)}$ queries to VSTAT$(n^2/(rk^2))$ are required to solve the distributional planted k-biclique with probability at least $2/3$. In particular, no polynomial-time statistical algorithm can solve the problem using queries to VSTAT$(o(n^2/k^2))$ and any SQ algorithm requires $n^{\Omega(\log n)}$ queries to VSTAT$(n^{2-\delta}/k^2)$. This lower bound also applies to the problem of distinguishing any planted k-biclique distribution from the uniform distribution over $\{0,1\}^n$ (no planting).*

This bound is essentially tight. For every index in the planted set S, the probability that the corresponding bit of a randomly chosen point is set to 1 is $1/2 + k/(2n)$, whereas for every index not in S, this probability is $1/2$. Therefore using n queries to VSTAT$(16n^2/k^2)$ (i.e., of tolerance $k/4n$) it is easy to recover S. Indeed, this can be done by using the query functions $h_i(x) = x_i$, for each $i \in [n]$. So, the answers of the VSTAT oracle represent the expected value of the ith bit over the sample.

There is also a SQ algorithm that uses $n^{O(\log n)}$ queries to VSTAT$(25n/k)$ (corresponding to a significantly smaller number of samples) to find the planted set for any $k \geq \log n$. In fact, the same algorithm can be used for the standard planted clique problem that achieves complexity $n^{O(\log n)}$. We enumerate over subsets $T \subseteq [n]$ of $\log n$ indices and query VSTAT$(25n/k)$ with the function $g_T : \{0,1\}^n \to \{0,1\}$ defined as 1 if and only if the point has ones in all coordinates in T. Therefore, if the set T is included in the planted set then

$$\mathop{\mathbf{E}}_{D}[g_T] = \frac{k}{n} \cdot 1 + \left(1 - \frac{k}{n}\right) 2^{-\log n} \in \left[\frac{k}{n}, \frac{k+1}{n}\right].$$

With this expectation, VSTAT$(25n/k)$ has tolerance at most $\sqrt{k(k+1)/25n^2} \leq (k+1)/5n$ and will return at least $k/n - (k+1)/(5n) > 3k/(4n)$. If, on the other hand, at least one element of T is not from the planted set, then $\mathbf{E}_D[g_T] \leq k/(2n) + 1/n$ and VSTAT$(25n/k)$ will return at most $(k+2)/(2n) + (k+2)/(5n) < 3k/(4n)$. Thus, we will know all $(\log n)$-sized subsets of the planted set and hence the entire planted set. We remark that this algorithm demonstrates the difference between STAT and VSTAT oracles. Implementing this algorithm using the STAT oracle would require tolerance of $\Omega(k/n)$ which corresponds to $O(n^2/k^2)$ samples. This is the same tolerance as the polynomial-time degree-based algorithm needs (estimate degree of each vertex), so one cannot hope to have a superpolynomial lower bound against $\mathrm{STAT}(k/n)$.

To summarize, n samples directly correspond to having access to $\mathrm{VSTAT}(O(n))$. The discussion above shows that the distributional planted biclique problem can be solved in polynomial time when $k = \Omega(\sqrt{n})$. At the same time, Theorem 2.9 implies that for $k \leq n^{1/2-\delta}$, any SQ algorithm will require $n^{\Omega(\log n)}$ queries to $\mathrm{VSTAT}(n^{1+\delta})$.

We now turn to stating our bounds for 1-bit sampling algorithms.

Theorem 2.10. *For any constant $\delta > 0$ and any $k \leq n^{1/2-\delta}$, any 1-bit sampling algorithm that with probability at least $2/3$ can distinguish between the uniform distribution and any planted k-biclique distribution requires $\Omega(n^2/k^2)$ queries to 1-STAT.*

Each query of a 1-bit sampling algorithm uses a new sample from D. Therefore this bound implies that any algorithm that does not reuse samples will require $\Omega(n^2/k^2)$ samples. To place this bound in context, we note that it is easy to detect whether a biclique of size k has been planted using $\tilde{O}(n^2/k^2)$ samples (as before, to detect if a coordinate i is in the planted set we can compute the average of x_i on $\tilde{O}(n^2/k^2)$ samples). Of course, finding all coordinates in the set would require reusing samples (which 1-bit sampling algorithms cannot do). Note that $n^2/k^2 \leq n$ if and only if $k \geq \sqrt{n}$.

A closely related problem is the *planted densest subgraph* problem, where edges in the planted subset appear with higher probability than in the remaining graph. This is a variant of the densest k-subgraph problem, which itself is a natural generalization of k-clique that asks to recover the densest k-vertex subgraph of a given n-vertex graph [Feige, 2002, Khot, 2004, Bhaskara et al., 2010, 2012]. The conjectured hardness of its average case variant, the planted densest subgraph problem, has been used in public key encryption schemes [Applebaum et al., 2010] and in analyzing parameters specific to financial markets [Arora et al., 2010]. We define the following distributional version of this problem:

Problem 2.11. *Fix $0 < q < p \leq 1$. For $1 \leq k \leq n$, let $S \subseteq [n]$ be a set of k vertex indices and D_S be a distribution over $\{0,1\}^n$ such that when $x \sim D_S$, with probability $1 - (k/n)$ the entries of x are independently q-biased Bernoulli variables, and with probability k/n the k coordinates in S are independently chosen p-biased Bernoulli variables, and the rest are independently chosen q-biased Bernoulli variables. The **distributional** (p,q)**-planted densest** k**-subgraph** problem is to find the unknown subset S given access to samples from D_S.*

Our approach and lower bounds extend in a straightforward manner to this problem. In Section 5.2 we analyze this general setting and give lower bounds for all settings of p and q. Here we describe a special case of our lower bounds when $q = 1/2$ and $p = 1/2 + \alpha$. Our lower bound becomes exponential as α becomes (inverse-polynomially) close to 0. Specifically:

Corollary 2.12. *For any constant $\delta > 0$, any $k \leq n^{1/2-\delta}$, $\alpha > 0$, $\ell \leq \min\{k, 1/(4\alpha^2)\}$, at least $n^{\Omega(\ell)}$ queries to $VSTAT(n^2/(48\ell\alpha^2k^2))$ are required to solve the distributional $(1/2+\alpha, 1/2)$-planted densest k-subgraph with probability at least $2/3$.*

For example, consider the setting $k = \ell = n^{1/3}$ and $\alpha = n^{-1/4}$. It is not hard to see that for this setting the problem can be solved on a random bipartite graph with n vertices on both sides (in exponential time). Our lower bound for this setting implies that at least $n^{\Omega(n^{1/3})}$ queries to $\mathrm{VSTAT}(n^{3/2})$ will be required. Additional corollaries for the distributional (p,q)-planted densest k-subgraph can be found in Section 5.2.

Relation to the planted k-biclique problem The upper and lower bounds we described for statistical algorithms match the state of the art for the average-case planted k-biclique and planted k-clique problems. Moreover, our lower bounds for the distributional versions of the planted k-biclique problem have implications for the hardness of the average-case planted k-biclique problem. An instance of the latter problem is a random $n \times n$ bipartite graph with a $k \times k$ biclique planted randomly. In Appendix A, we show that the average-case planted k-biclique is equivalent to our distributional planted k-biclique with n samples. Specifically, a single sample corresponds to the adjacency list of a vertex on the left, and n samples correspond to the adjacency matrix of the bipartite graph. By this equivalence, an algorithm that solves the average-case planted bipartite k-clique problem will also solve the distributional planted k-biclique with n samples. Our lower bounds for the distributional problem therefore imply that the planted k-biclique problem would require a non-statistical approach, i.e., one for which there is no statistical analogue.

2.5 Subsequent Work

In subsequent work, Feldman, Perkins and Vempala [2013] introduced a notion of statistical dimension that is based on the spectral norm of the correlation matrix of large sets of distributions. It is always at least as large as the average correlation-based dimension defined here and also leads to lower bounds on the complexity of SQ algorithms using VSTAT. Using this dimension they proved tight lower bounds on the complexity of statistical algorithms for planted k-SAT and Goldreich's pseudo-random generator. In addition, they described statistical algorithms based on power iteration with nearly matching upper bounds. Finally, they demonstrate that lower bounds against SQ algorithms can be used to derive concrete lower bounds for convex relaxations of the problem.

Feldman et al. [2013] have also extended the lower bounds against 1-STAT to lower bounds against the k-bit version of 1-STAT at the expense of factor 2^k blow-up in the number of queries. Steinhardt, Valiant and Wager [2016] gave a more direct approach for proving lower bounds against this oracle that is closely related to the techniques here and in Feldman et al. [2013]. They have further showed that statistical queries can be used to simulate the oracle that that extracts k bits from each sample in an interactive way (rather than at once).

Building on our approach, Feldman [2016] described new notions of statistical dimension and proved that they tightly characterize the SQ complexity of solving general search problems over distributions for both STAT and VSTAT oracle. He also simplified the analysis of VSTAT(t) by showing that it is equivalent (up to constant factors) to returning any value v such that $|\sqrt{v} - \sqrt{\mathbf{E}_D[h]}| \leq 1/\sqrt{t}$. Some additional recent applications of SQ lower bounds that are related to our work include learning of the Ising model [Bresler et al., 2014], convex optimization [Feldman et al., 2015] and distribution-independent PAC learning of lines over finite fields [Feldman, 2016].

The distributional planted k-biclique problem introduced here is a simple and natural problem that shows a remarkable property: information-theoretically it can be solved with many fewer samples than is necessary for any known efficient algorithm (and no efficient statistical algorithm exists). In particular, any algorithm that solves our problem with less than n samples will also solve the average-case k-biclique problem (that is at least as hard as the usual planted k-clique problem). In several more recent works, reductions from the planted clique problem were used to demonstrate a similar phenomenon in a number of important problems in statistics and machine learning [Berthet and Rigollet, 2013, Ma and Wu, 2013, Hajek et al., 2015, Gao et al., 2014, Wang et al., 2014, Cai et al., 2015].

3 Lower Bounds from Statistical Dimension

In this section we prove the general lower bounds. In later sections, we will compute the parameters in these bounds for specific problems of interest.

3.1 Lower Bounds for Statistical Query Algorithms

We start by proving Theorem 2.7 which is the basis of all our lower bounds. In fact, we will prove a stronger version of this theorem which also applies to randomized algorithms. For this version we need an additional parameter in the definition of SDA.

Definition 3.1. *For $\bar{\gamma} > 0$, $\eta > 0$, domain X and a search problem $\mathcal{Z}$ over a set of solutions $\mathcal{F}$ and a class of distributions $\mathcal{D}$ over X, let d be the largest value such that there exists a* reference *distribution D over X and a finite set of distributions $\mathcal{D}_D \subseteq \mathcal{D}$ with the following property: for any solution $f \in \mathcal{F}$ the set $\mathcal{D}_f = \mathcal{D}_D \setminus \mathcal{Z}_f$ has size at least $(1-\eta) \cdot |\mathcal{D}_D|$ and $\mathrm{SDA}(\mathcal{D}_f, D, \bar{\gamma}) \geq d$. We define the statistical dimension with average correlation $\bar{\gamma}$ and solution set bound η of $\mathcal{Z}$ to be d and denote it by $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma}, \eta)$.*

Note that for any $\eta < 1$, $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma}) \geq \mathrm{SDA}(\mathcal{Z}, \bar{\gamma}, \eta)$ and for $\eta = 1 - 1/|\mathcal{D}_D|$, we get $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma}) = \mathrm{SDA}(\mathcal{Z}, \bar{\gamma}, \eta)$, where $\mathcal{D}_D$ is the set of distributions that maximizes $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma})$.

Theorem 3.2. *Let X be a domain and $\mathcal{Z}$ be a search problem over a set of solutions $\mathcal{F}$ and a class of distributions $\mathcal{D}$ over X. For $\bar{\gamma} > 0$ and $\eta \in (0,1)$ let $d = \mathrm{SDA}(\mathcal{Z}, \bar{\gamma}, \eta)$. Any randomized SQ algorithm that solves $\mathcal{Z}$ with probability $\alpha > \eta$ requires at least $\frac{\alpha-\eta}{1-\eta} d$ calls to VSTAT$(1/(3\bar{\gamma}))$.*

Theorem 2.7 is obtained from Theorem 3.2 by setting $\alpha = 1$ and using any $1 - 1/|\mathcal{D}_D| \leq \eta < 1$. Further, for any $\eta < 1$, $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma}) \geq \mathrm{SDA}(\mathcal{Z}, \bar{\gamma}, \eta)$ and therefore for any $\eta < 1$, a bound on $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma}, \eta)$ can be used in Theorem 2.7 in place of bound on $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma})$. We now prove Theorem 3.2.

of Theorem 3.2. We prove our lower bound by exhibiting a distribution over inputs (which are distributions over X) for which every deterministic SQ algorithm that solves $\mathcal{Z}$ with probability α (over the choice of input) requires at least $(\alpha - \eta) \cdot d/(1-\eta)$ calls to VSTAT$(1/(3\bar{\gamma}))$. The claim of the theorem will then follow by Yao's minimax principle [Yao, 1977].

Using the notation of Definition 3.1, let D be the reference distribution and $\mathcal{D}_D$ be a set of distributions for which the value d is achieved. Let $\mathcal{A}$ be a deterministic SQ algorithm that uses q queries to VSTAT$(1/(3\bar{\gamma}))$ to solve $\mathcal{Z}$ with probability α over the random and uniform choice of a distribution from $\mathcal{D}_D$. Consider the execution of $\mathcal{A}$ in which to each query h of $\mathcal{A}$, the oracle returns exactly $\mathbf{E}_D[h]$ and let f denote the output. Let the set $\mathcal{D}_D^+ \subseteq \mathcal{D}_D$ be the set of distributions on which $\mathcal{A}$ is successful for all valid responses of VSTAT$(1/(3\bar{\gamma}))$. Let $\mathcal{D}^+ = \mathcal{D}_f \cap \mathcal{D}_D^+$ (recall that we defined $\mathcal{D}_f = \mathcal{D}_D \setminus \mathcal{Z}_f$). We observe that $\mathcal{D}^+ = \mathcal{D}_D^+ \setminus (\mathcal{D}_D \setminus \mathcal{D}_f)$ and therefore

$$|\mathcal{D}^+| \geq |\mathcal{D}_D^+| - |\mathcal{D}_D \setminus \mathcal{D}_f| \geq \alpha|\mathcal{D}_D| - |\mathcal{D}_D \setminus \mathcal{D}_f| = \frac{\alpha|\mathcal{D}_D| - |\mathcal{D}_D \setminus \mathcal{D}_f|}{|\mathcal{D}_D| - |\mathcal{D}_D \setminus \mathcal{D}_f|}|\mathcal{D}_f| \geq \frac{\alpha - \eta}{1-\eta}|\mathcal{D}_f|. \tag{1}$$

By the definition of $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma})$, it holds that $\mathrm{SDA}(\mathcal{D}_f, D, \bar{\gamma}) \geq d$. In Lemma 3.3 given below, we will show that under the conditions of this proof, $\mathrm{SDA}(\mathcal{D}_f, D, \bar{\gamma}) \geq d$ implies that $\mathcal{A}$ must use at least $q \geq d|\mathcal{D}^+|/|\mathcal{D}_f|$ queries. By inequality (1), $q \geq \frac{\alpha-\eta}{1-\eta} \cdot d$ giving the desired lower bound.

□

The proof of Theorem 3.2 relies on the following lemma that translates a lower bound on $\mathrm{SDA}(\mathcal{D}_f, D, \bar{\gamma})$ into a lower bound on the number of queries that $\mathcal{A}$ needs to use. Its proof is based on ideas from [Szörényi, 2009] and [Feldman, 2012].

Lemma 3.3. *Let X be a domain and $\mathcal{Z}$ be a search problem over a set of solutions $\mathcal{F}$ and a class of distributions $\mathcal{D}$ over X. Let $\mathcal{A}$ be a (deterministic) SQ algorithm for $\mathcal{Z}$ that uses at most q queries to VSTAT$(1/(3\bar{\gamma}))$. For a distribution D, consider the execution of $\mathcal{A}$ on D in which to each query h of $\mathcal{A}$, the oracle returns exactly $\mathbf{E}_D[h]$ and let f denote the output. For a set of distributions $\mathcal{D}_f \subseteq \mathcal{D} \setminus \mathcal{Z}_f$ and $\bar{\gamma} > 0$, let $d = \mathrm{SDA}(\mathcal{D}_f, D, \bar{\gamma})$. Let $\mathcal{D}^+$ be the set of all distributions in $\mathcal{D}_f$ for which $\mathcal{A}$ successfully solves $\mathcal{Z}$ for all valid responses of VSTAT$(1/(3\bar{\gamma}))$. Then $q \geq d \cdot |\mathcal{D}^+|/|\mathcal{D}_f|$.*

Proof. Let $h_1, h_2, \ldots, h_q$ be the queries asked by $\mathcal{A}$ when executed on D with the exact responses of the oracle. Let $m = |\mathcal{D}^+|$ and we denote the distributions in $\mathcal{D}^+$ by $\{D_1, D_2, \ldots, D_m\}$. For every $k \leq q$, let A_k be the set of all distributions D_i such that

$$\left| \mathop{\mathbf{E}}_{D}[h_k(x)] - \mathop{\mathbf{E}}_{D_i}[h_k(x)] \right| > \tau_{i,k} \doteq \max\left\{ \frac{1}{t}, \sqrt{\frac{p_{i,k}(1-p_{i,k})}{t}} \right\},$$

where we use t to denote $1/(3\bar{\gamma})$ and $p_{i,k}$ to denote $\mathbf{E}_{D_i}[h_k(x)]$. To prove the desired bound we first prove the following two claims:

1. $\sum_{k \leq q} |A_k| \geq m$;

2. for every $k \leq q$, $|A_k| \leq |\mathcal{D}_f|/d$.

Combining these two implies that $q|\mathcal{D}_f|/d \geq m$ or, equivalently, $q \geq d|\mathcal{D}^+|/|\mathcal{D}_f|$.

In the rest of the proof for conciseness we drop the subscript D from inner products and norms. To prove the first claim we assume, for the sake of contradiction, that there exists $D_i \notin \cup_{k \leq q} A_k$. Then for every $k \leq q$, $|\mathbf{E}_D[h_k(x)] - \mathbf{E}_{D_i}[h_k(x)]| \leq \tau_{i,k}$. This implies that $\mathbf{E}_D[h_k(x)]$ are within $\tau_{i,k}$ of $\mathbf{E}_{D_i}[h_k(x)]$. By the definition of VSTAT(t), this implies that the responses we used in our execution of $\mathcal{A}$ on D are also valid responses of VSTAT(t) when $\mathcal{A}$ is executed on D_i. The output of this execution is f and hence it must be a valid solution for D_i. This contradicts the definition of $\mathcal{D}^+$ since it is a subset of $\mathcal{D}_f \subseteq \mathcal{D} \setminus \mathcal{Z}_f$.

To prove the second claim, suppose that for some $k \in [d]$, $|A_k| > |\mathcal{D}_f|/d$. Let $p_k = \mathbf{E}_D[h_k(x)]$ and assume that $p_k \leq 1/2$ (when $p_k > 1/2$ we just replace h_k by $1 - h_k$ in the analysis below). First we note that:

$$\mathop{\mathbf{E}}_{D_i}[h_k(x)] - \mathop{\mathbf{E}}_{D}[h_k(x)] = \mathop{\mathbf{E}}_{D}\left[\frac{D_i(x)}{D(x)} h_k(x)\right] - \mathop{\mathbf{E}}_{D}[h_k(x)] = \left\langle h_k, \frac{D_i}{D} - 1 \right\rangle = p_{i,k} - p_k.$$

Let $\hat{D}_i(x) = \frac{D_i(x)}{D(x)} - 1$, (where the convention is that $\hat{D}_i(x) = 0$ if $D(x) = 0$). We will next show upper and lower bounds on the following quantity

$$\Phi = \left\langle h_k, \sum_{D_i \in A_k} \hat{D}_i \cdot \mathsf{sign}\langle h_k, \hat{D}_i \rangle \right\rangle.$$

By Cauchy-Schwartz we have that

$$\begin{aligned}\Phi^2 = \left\langle h_k, \sum_{D_i \in A_k} \hat{D}_i \cdot \mathsf{sign}\langle h_k, \hat{D}_i\rangle \right\rangle^2 &\leq \|h_k\|^2 \cdot \left\| \sum_{D_i \in A_k} \hat{D}_i \cdot \mathsf{sign}\langle h_k, \hat{D}_i\rangle \right\|^2 \\ &\leq \|h_k\|^2 \cdot \left(\sum_{D_i, D_j \in A_k} \left| \langle \hat{D}_i, \hat{D}_j \rangle \right| \right) \\ &\leq \|h_k\|^2 \cdot \rho(A_k, D) \cdot |A_k|^2. \qquad (2)\end{aligned}$$

We also have that

$$\begin{aligned}\Phi^2 = \left\langle h_k, \sum_{D_i \in A_k} \hat{D}_i \cdot \mathsf{sign}\langle h_k, \hat{D}_i\rangle \right\rangle^2 &= \left(\sum_{D_i \in A_k} \langle h_k, \hat{D}_i \rangle \cdot \mathsf{sign}\langle h_k, \hat{D}_i\rangle \right)^2 \\ &\geq \left(\sum_{D_i \in A_k} |p_{i,k} - p_k| \right)^2. \qquad (3)\end{aligned}$$

To evaluate the last term of this inequality we use the fact that $|p_{i,k} - p_k| \geq \tau_{i,k} = \max\{1/t, \sqrt{p_{i,k}(1-p_{i,k})/t}\}$. Next we use a simple fact (proved in Lemma 3.5 below) that $|p_{i,k} - p_k| \geq \max\{1/t, \sqrt{p_{i,k}(1-p_{i,k})/t}\}$ implies that $|p_k - p_{i,k}| \geq \sqrt{\frac{\min\{p_k, 1-p_k\}}{3t}}$ to obtain: For every $D_i \in A_k$,

$$|p_k - p_{i,k}| \geq \sqrt{\frac{\min\{p_k, 1-p_k\}}{3t}} = \sqrt{\frac{p_k}{3t}}. \qquad (4)$$

By substituting equation (4) into (3) we get that $\Phi^2 \geq \frac{p_k}{3t} \cdot |A_k|^2$.

We note that, h_k is a $[0,1]$-valued function and therefore $\|h_k\|^2 \leq p_k$. Substituting this into equation (2) we get that $\Phi^2 \leq p_k \cdot \rho(A_k, D) \cdot |A_k|^2$. By combining these two bounds on Φ^2 we obtain that $\rho(A_k, D) \geq 1/(3t) = \bar{\gamma}$ which contradicts the definition of SDA. □

Remark 3.4. *We remark that for algorithms using the STAT oracle, the proof can be simplified somewhat. For* $\tau = \sqrt{\bar{\gamma}}$,

$$\Phi^2 \geq \left(\sum_{D_i \in A_k} |p_{i,k} - p_k| \right)^2 \geq \tau^2 |A_k|^2$$

and the proof could be obtained by directly combining equations (2) and (3) to get a contradiction. This also eliminates the factor of 3 *in the bound and the assumption that queries are* $[0,1]$*-valued can be relaxed to* $[-1,1]$*-valued queries since it suffices that* $\|h_k\|^2 \leq 1$. *This leads to an identical lower bound on the number of queries for STAT*$(\sqrt{\bar{\gamma}})$ *in place of VSTAT*$(1/(3\bar{\gamma}))$.

We now prove a bound on the distance between any $p \in [0,1]$ and p' which is returned by VSTAT(t) on a query with expectation p in terms of p' that we used in the proof of Lemma 3.3.

Lemma 3.5. *For an integer t and any* $p \in [0,1]$, *let* $p' \in [0,1]$ *be such that* $|p' - p| \geq \tau = \max\left\{ \frac{1}{t}, \sqrt{\frac{p(1-p)}{t}} \right\}$. *Then* $|p' - p| \geq \sqrt{\frac{\min\{p', 1-p'\}}{3t}}$.

Proof. First note that our conditions and bounds do not change if we replace both p and p' with $1-p$ and $1-p'$, respectively. Therefore it is sufficient to prove the bound when $p \le 1/2$. We know that $|p'-p| \ge \tau = \max\{1/t, \sqrt{p(1-p)/t}\}$. If $p \ge 2p'/3$ then certainly

$$|p'-p| \ge \sqrt{\frac{p(1-p)}{t}} \ge \sqrt{\frac{\frac{2}{3}p' \cdot \frac{1}{2}}{t}} = \sqrt{\frac{p'}{3t}}.$$

Otherwise (when $p < 2p'/3$), $p'-p \ge p' - 2p'/3 = p'/3$. We also know that $|p-p'| \ge \tau \ge 1/t$ and therefore $|p-p'| \ge \sqrt{\frac{p'}{3t}}$. □

3.1.1 Decision Problems

For decision problems, our dimension and lower bounds can be simplified. We denote by $\mathcal{B}(\mathcal{D}, D)$ a decision problem in which the input distribution D' either equals D or belongs to $\mathcal{D}$ and the goal of the algorithm is to identify whether $D' = D$ or $D' \in \mathcal{D}$. For example, for the distributional planted k-biclique problem, the decision version is to determine whether the given input distribution corresponds to a planted k-biclique or to one with no planting (uniform distribution on $\{0,1\}^n$.

For the decision problem $\mathcal{B}(\mathcal{D}, D)$ our notion of dimension simplifies to the following.

Definition 3.6. *For $\bar{\gamma} > 0$, domain X and a decision problem $\mathcal{B}(\mathcal{D}, D)$, let $\mathrm{SDA}(\mathcal{B}(\mathcal{D}, D), \bar{\gamma})$ be defined as the largest value d such that there exists a finite set of distributions $\mathcal{D}_D \subseteq \mathcal{D}$ such that $\mathrm{SDA}(\mathcal{D}_D, D, \bar{\gamma}) = d$.*

Our technique gives the following lower bound for decision problems:

Theorem 3.7. *Let D be a distribution and $\mathcal{D}$ be a set of distributions over a domain X such that for some $\bar{\gamma}$, $\mathrm{SDA}(\mathcal{B}(\mathcal{D}, D), \bar{\gamma}) = d$. Any (randomized) SQ algorithm that solves $\mathcal{B}(\mathcal{D}, D)$ with success probability $\alpha > 1/2$ requires at least $(2\alpha - 1)d$ queries to $VSTAT(1/(3\bar{\gamma}))$.*

Proof. As before, we exhibit a hard distribution over input distributions for which every deterministic SQ algorithm that solves $\mathcal{B}(\mathcal{D}, D)$ with probability α (over the choice of input) requires at least $(2\alpha - 1)d$ queries to $\mathrm{VSTAT}(1/(3\bar{\gamma}))$. Let $\mathcal{D}_D$ be the set of distributions that witnesses the statistical dimension, namely, $\mathrm{SDA}(\mathcal{D}_D, D, \bar{\gamma}) = d$. Consider the following distribution over the input distribution D': D' equals D with probability 1/2 and D' equals a random uniform element of $\mathcal{D}_D$ with probability 1/2.

$\mathcal{A}$ has success probability $\alpha > 1/2$ and therefore, when executed on D with exact responses to queries, it must correctly identify D (say it outputs 0 in this case). We then define $\mathcal{D}^+ \subseteq \mathcal{D}_D$ as the set of distributions on which $\mathcal{A}$ is successful (that is outputs 1). The probability of success of $\mathcal{A}$ implies that $|\mathcal{D}^+| \ge (2\alpha - 1)|\mathcal{D}_D|$. Now, the set $\mathcal{D}_D$ is included in the set of distributions $\mathcal{D}$ for which 0 is not a valid solution. Therefore we can apply Lemma 3.3 with $\mathcal{D}_f = \mathcal{D}_D$ to obtain that the number of queries to $\mathrm{VSTAT}(1/(3\bar{\gamma}))$ is $q \ge (2\alpha - 1)d$. □

3.2 Statistical Dimension Based on Pairwise Correlations

In addition to SDA which is based on average correlation we introduce a simpler notion based on pairwise correlations. It is sufficient for some applications and is easy to relate to SQ-DIM used in learning (as we do in Section 6).

Definition 3.8. *We say that a set of m distributions $\mathcal{D} = \{D_1, \ldots, D_m\}$ over X is (γ, β)-correlated relative to a distribution D over X if:*

$$\left| \left\langle \frac{D_i}{D} - 1, \frac{D_j}{D} - 1 \right\rangle_D \right| \leq \begin{cases} \beta \text{ for } i = j \in [m] \\ \gamma \text{ for } i \neq j \in [m]. \end{cases}$$

Definition 3.9. *For $\gamma, \beta > 0$, domain X and a search problem $\mathcal{Z}$ over a set of solutions $\mathcal{F}$ and a class of distributions $\mathcal{D}$ over X, let m be the largest integer such that there exists a* reference *distribution D over X and a finite set of distributions $\mathcal{D}_D \subseteq \mathcal{D}$ such that for any solution $f \in \mathcal{F}$, $\mathcal{D}_f = \mathcal{D}_D \backslash \mathcal{Z}_f$ is (γ, β)-correlated relative to D and $|\mathcal{D}_f| \geq m$. We define the* ***statistical dimension*** *with pairwise correlations (γ, β) of $\mathcal{Z}$ to be m and denote it by $\mathrm{SD}(\mathcal{Z}, \gamma, \beta)$.*

For decision problems $\mathrm{SD}(\mathcal{B}(\mathcal{D}, D), \gamma, \beta)$ is defined as the largest integer m such that there exists a set of distributions $\mathcal{D}_D \subseteq \mathcal{D}$ of size m that is (γ, β)-correlated relative to D.

It is easy to bound SDA of any (γ, β)-correlated set of distributions.

Lemma 3.10. *Let $\mathcal{D} = \{D_1, D_2, \ldots, D_m\}$ be a (γ, β)-correlated set of distributions relative to a distribution D. Then for every $\gamma' > 0$, $\mathrm{SDA}(\mathcal{D}, D, \gamma + \gamma') \geq \frac{m\gamma'}{\beta - \gamma}$.*

Proof. Take $d = m\gamma'/(\beta - \gamma)$; we will prove that $\mathrm{SDA}(\mathcal{D}, D, \gamma + \gamma') \geq d$. Consider a set of distributions $\mathcal{D}' \subseteq \mathcal{D}$, where $|\mathcal{D}'| \geq |\mathcal{D}|/d \geq m/d = (\beta - \gamma)/\gamma'$:

$$\begin{aligned} \rho(\mathcal{D}', D) &= \frac{1}{|\mathcal{D}'|^2} \sum_{D_1, D_2 \in \mathcal{D}'} \left| \left\langle \frac{D_1}{D} - 1, \frac{D_2}{D} - 1 \right\rangle_D \right| \\ &\leq \frac{1}{|\mathcal{D}'|^2} \left(|\mathcal{D}'|\beta + (|\mathcal{D}'|^2 - |\mathcal{D}'|)\gamma \right) \\ &\leq \gamma + \frac{\beta - \gamma}{|\mathcal{D}'|} \\ &\leq \gamma + \gamma' \end{aligned}$$

□

As an immediate corollary we obtain a bound on SDA of a search or decision problem from a bound on SD.

Corollary 3.11. *Let X be a domain and $\mathcal{Z}$ be a search or decision problem over a set of solutions $\mathcal{F}$ and a class of distributions $\mathcal{D}$ over X. For $\gamma, \beta > 0$, let $m = \mathrm{SD}(\mathcal{Z}, \gamma, \beta)$. Then for every $\gamma' > 0$, $\mathrm{SDA}(\mathcal{Z}, \gamma + \gamma') \geq \frac{m\gamma'}{\beta - \gamma}$.*

We now apply Theorem 2.7 to obtain the following lower bound on SQ algorithms in terms of SD.

Corollary 3.12. *Let X be a domain and $\mathcal{Z}$ be a search or decision problem over a set of solutions $\mathcal{F}$ and a class of distributions $\mathcal{D}$ over X. For $\gamma, \beta > 0$, let $m = \mathrm{SD}(\mathcal{Z}, \gamma, \beta)$. For any $\gamma' > 0$, any SQ algorithm requires at least $m\gamma'/(\beta - \gamma)$ queries to the $STAT(\sqrt{\gamma + \gamma'})$ or $VSTAT(1/(3(\gamma + \gamma')))$ oracle to solve $\mathcal{Z}$.*

In this corollary if, for example, $\mathrm{SD}(\mathcal{Z}, \gamma = \frac{m^{-2/3}}{2}, \beta = 1) \geq m$ then at least $m^{1/3}/2$ queries to $\mathrm{VSTAT}(m^{2/3}/3)$ or $\mathrm{STAT}(m^{-1/3})$ oracle are required to solve the problem.

3.3 Lower Bounds for 1-bit Sampling Algorithms

Next we address lower bounds on algorithms that use the 1-STAT oracle. We recall that the 1-STAT oracle returns the value of a function on a single randomly chosen point. To estimate the expectation of a function, an algorithm can simply query this oracle multiple times with the same function and average the results.

We note that responses of 1-STAT do not have the room for the possibly adversarial deviation afforded by the tolerance of the STAT and VSTAT oracles. The ability to use these slight deviations in a coordinated way is used crucially in our lower bounds against VSTAT and in all known lower bounds for SQ learning algorithms. While it is possible to derive lower bounds against 1-bit sampling algorithms using m queries from lower bounds against algorithms that use $O(m)$ queries to STAT$(1/m)$ [Ben-David and Dichterman, 1998], such lower bound will not suffice for our main application. It would only imply the trivial lower bound of $\Omega(n/k)$ queries to 1-STAT for the planted k-biclique problem. Proving tighter lower bounds against 1-bit sampling algorithms directly is harder and indeed lower bounds for the equivalent Honest SQ learning model required a substantially more involved argument than lower bounds for the regular SQ model [Yang, 2005].

Our lower bounds for 1-bit sampling algorithms rely on a direct simulation of the 1-STAT oracle using the VSTAT oracle. This simulation allows us to derive lower bounds against 1-bit sampling algorithms from Theorem 3.2. We also provide a reverse simulation of VSTAT oracle using 1-STAT oracle.

Theorem 3.13. *Let $\mathcal{Z}$ be a search problem and let $\mathcal{A}$ be a (possibly randomized)* 1*-bit sampling algorithm that solves $\mathcal{Z}$ with probability at least α using m samples from 1-STAT. For any $\delta \in$* $(0, 1/4]$*, there exists a SQ algorithm $\mathcal{A}'$ that uses at most m queries to VSTAT*(m/δ^2) *and solves $\mathcal{Z}$ with probability at least $\alpha - \delta$.*

Our proof relies on a simple simulation. Given query $h_1 : X \to \{0, 1\}$ from $\mathcal{A}$ to 1-STAT, we make the same query h_1 to VSTAT(t) for $t = m/\delta^2$. Let p'_1 be the response. We flip a coin with bias p'_1 (that is one that outputs 1 with probability p'_1 and 0 with probability $1 - p'_1$) and return it to the algorithm. We do the same for the remaining $m - 1$ queries which we denote by $h_2, h_3, \ldots, h_m$. We then prove that the true m samples of 1-STAT and our simulated coin flips are statistically close by upper bounding the expected ratio of their density functions (which is equal to the χ^2 divergence plus 1) . This implies that the success probability of the simulated algorithm is not much worse than that of the 1-bit sampling algorithm.

In our proof we will, for simplicity and without loss of generality, assume that VSTAT(t) always outputs a value in the interval $[1/t, 1-1/t]$. We can always replace a value v returned by VSTAT(t) by v' which is the closest value to v in the above interval. It is easy to see that if v is a valid answer of VSTAT(t) then so is v'.

We will need the following lemmas for our proof. The first one bounds the total variation distance between two distributions in terms of the expected ratio of probability density functions.

Lemma 3.14. *Let D_1 and D_2 be two distribution over a domain X of finite*[2] *size such that $D_2(x)$ is non-vanishing. Denote the total variation distance between D_1 and D_2 by $\Delta_{TV}(D_1, D_2)$. Then $\Delta_{TV}(D_1, D_2) \leq \sqrt{\rho}/2$, where $\rho = \mathbf{E}_{D_1}\left[\frac{D_1(x)}{D_2(x)}\right] - 1$.*

[2] This assumption is simply for convenience of notation. It holds in our applications.

Proof. The key observation is that the χ^2-divergence between D_1 and D_2 is exactly the expected ratio minus 1.

$$\rho = \mathop{\mathbf{E}}_{D_1}\left[\frac{D_1(x)}{D_2(x)}\right] - 1 = \mathop{\mathbf{E}}_{D_2}\left[\frac{D_1^2(x)}{D_2^2(x)}\right] - 1 = \mathop{\mathbf{E}}_{D_2}\left[\frac{D_1^2(x)}{D_2^2(x)} - 2\frac{D_1(x)}{D_2(x)} + 1\right] = \mathop{\mathbf{E}}_{D_2}\left[\left(\frac{D_1(x)}{D_2(x)} - 1\right)^2\right].$$

By Jensen's inequality this implies that

$$\mathop{\mathbf{E}}_{D_2}\left[\left|\frac{D_1(x)}{D_2(x)} - 1\right|\right] \le \sqrt{\mathop{\mathbf{E}}_{D_2}\left[\left(\frac{D_1(x)}{D_2(x)} - 1\right)^2\right]} = \sqrt{\rho}\ .$$

Finally,

$$\Delta_{TV}(D_1, D_2) = \frac{1}{2}\sum_{x\in X}|D_1(x) - D_2(x)| = \frac{1}{2}\mathop{\mathbf{E}}_{D_2}\left[\left|\frac{D_1(x)}{D_2(x)} - 1\right|\right] \le \frac{\sqrt{\rho}}{2}.$$

□

The second lemma is that if p' is an answer of VSTAT(t) for a query h, such that $\mathbf{E}_D[h] = p$, then the expected ratio of density functions of Bernoulli random variables with biases p and p', denoted $B(p)$ and $B(p')$, is small.

Lemma 3.15. *For an integer t and $p \in [0,1]$ let $p' \in [1/t, 1-1/t]$ such that $|p'-p| \le \max\left\{\frac{1}{t}, \sqrt{\frac{p(1-p)}{t}}\right\}$. Then*

$$\mathop{\mathbf{E}}_{b\sim B(p)}\left(\frac{\mathbf{Pr}[B(p)=b]}{\mathbf{Pr}[B(p')=b]}\right) \le 1 + \frac{3}{t}.$$

Proof. If $b = 1$, the ratio is p/p' and when $b = 0$, then it is $(1-p)/(1-p')$. Thus, the expected ratio is

$$\frac{p^2}{p'} + \frac{(1-p)^2}{1-p'} = 1 + \frac{(p-p')^2}{p'(1-p')}\ .$$

We can assume without loss of generality that $p' \le 1/2$.

Now if $p \le 3p'$ then $p(1-p) \le 3p'(1-p')$. Otherwise (when, $p > 3p'$), we know that $p \ge 3p' \ge 3/t$. This implies that $p - p' \ge 2p/3 \ge 2/t$. This means that

$$\frac{2p}{3} \le p - p' \le \sqrt{\frac{p(1-p)}{t}} \le \sqrt{\frac{p}{t}}.$$

This can only be true when $p \le (3/2)^2/t = 9/(4t)$, contradicting our assumption that $p \ge 3/t$. This implies that

$$\max\left\{\frac{1}{t}, \sqrt{\frac{p(1-p)}{t}}\right\} \le \max\left\{\frac{1}{t}, \sqrt{\frac{3p'(1-p')}{t}}\right\} \le \sqrt{\frac{3p'(1-p')}{t}}\ .$$

By using this bound in the ratio equation we get that

$$1 + \frac{(p-p')^2}{p'(1-p')} \le 1 + \frac{\frac{3p'(1-p')}{t}}{p'(1-p')} \le 1 + \frac{3}{t}\ .$$

□

We can now complete the proof of Theorem 3.13.

of Theorem 3.13. We simulate $\mathcal{A}$ using VSTAT(t) as described above. We now prove that for any algorithm the total variation distance between the true answers of 1-STAT and the simulated distribution is at most δ. Formally, let R denote the set of all outcomes of $\mathcal{A}$'s random bits and for $r \in R$, let $\mathcal{A}^r$ denote the execution of $\mathcal{A}$ when its random bits are set to r. let $\Pi_{\mathcal{A}}$ denote the distribution over the m bits obtained by the algorithm $\mathcal{A}$ when it is run with 1-STAT oracle. Similarly, let $\Pi'_{\mathcal{A}}$ denote the distribution over $\{0,1\}^m$ obtained by running the algorithm $\mathcal{A}$ simulated using VSTAT(t) as above. By definition, $\Pi_{\mathcal{A}} = \mathbf{E}_{r\in R}\, \Pi_{\mathcal{A}^r}$ and similarly $\Pi'_{\mathcal{A}} = \mathbf{E}_{r\in R}\, \Pi'_{\mathcal{A}^r}$. This implies that,

$$\Delta_{TV}(\Pi_{\mathcal{A}}, \Pi'_{\mathcal{A}}) \le \mathop{\mathbf{E}}_{r\in R}\left[\Delta_{TV}(\Pi_{\mathcal{A}^r}, \Pi'_{\mathcal{A}^r})\right] \le \max_{r\in R} \Delta_{TV}(\Pi_{\mathcal{A}^r}, \Pi'_{\mathcal{A}^r}).$$

The algorithm $\mathcal{A}^r$ is deterministic and it is therefore sufficient to prove the bound on total variation distance for deterministic algorithms. For conciseness we assume henceforth that $\mathcal{A}$ is deterministic.

For any $i \in [m]$ let $\Pi_{\mathcal{A}_i}$ denote the probability distribution on the first i samples of $\mathcal{A}$ executed with 1-STAT. For $j \le i$ let z^j denote the first j bits of z. Let $\Pi_{\mathcal{A}_i}(z \mid z^{i-1})$ denote the probability that the first i samples of $\mathcal{A}$ executed with 1-STAT oracle are equal to z conditioned on the probability that the first $i-1$ samples are equal to z^{i-1}. We define $\Pi'_{\mathcal{A}_i}(z)$ and $\Pi'_{\mathcal{A}_i}(z \mid z^{i-1})$ analogously. We also denote by h_z the query that $\mathcal{A}$ asks after getting z as the response to first i samples and let $p_z = \mathbf{E}_D[h_z]$. Let p'_z denote the response of VSTAT(t) on h_z.

For $i \in [m]$ and any $z \in \{0,1\}^i$, $\Pi_{\mathcal{A}_i}(z \mid z^{i-1}) = \mathbf{Pr}[B(p_{z^{i-1}}) = z_i]$ and hence $\Pi_{\mathcal{A}_i}(z) = \Pi_{\mathcal{A}_{i-1}}(z^{i-1})\,\mathbf{Pr}[B(p_{z^{i-1}}) = z_i]$. Similarly, $\Pi'_{\mathcal{A}_i}(z \mid z^{i-1}) = \mathbf{Pr}[B(p'_{z^{i-1}}) = z_i]$ and $\Pi'_{\mathcal{A}_i}(z) = \Pi'_{\mathcal{A}_{i-1}}(z^{i-1})\,\mathbf{Pr}[B(p'_{z^{i-1}}) = z_i]$. This implies that:

$$\begin{aligned}\mathop{\mathbf{E}}_{z\sim\Pi_{\mathcal{A}_i}}\left[\frac{\Pi_{\mathcal{A}_i}(z)}{\Pi'_{\mathcal{A}_i}(z)}\right] &= \mathop{\mathbf{E}}_{z\sim\Pi_{\mathcal{A}_i}}\left[\frac{\Pi_{\mathcal{A}_{i-1}}(z^{i-1})\,\mathbf{Pr}[B(p_{z^{i-1}}) = z_i]}{\Pi'_{\mathcal{A}_{i-1}}(z^{i-1})\,\mathbf{Pr}[B(p'_{z^{i-1}}) = z_i]}\right] \\ &= \mathop{\mathbf{E}}_{y\sim\Pi_{\mathcal{A}_{i-1}}}\left[\frac{\Pi_{\mathcal{A}_{i-1}(y)}}{\Pi'_{\mathcal{A}_{i-1}(y)}}\cdot \mathop{\mathbf{E}}_{b\sim B(p_y)}\left[\frac{\mathbf{Pr}[B(p_y) = b]}{\mathbf{Pr}[B(p'_y) = b]}\right]\right] \ .\end{aligned}$$

Now by Lemma 3.15, this implies that for any z of length $i \in [m]$,

$$\mathop{\mathbf{E}}_{z\sim\Pi_{\mathcal{A}_i}}\left[\frac{\Pi_{\mathcal{A}_i}(z)}{\Pi'_{\mathcal{A}_i}(z)}\right] \le \mathop{\mathbf{E}}_{y\sim\Pi_{\mathcal{A}_{i-1}}}\left[\frac{\Pi_{\mathcal{A}_{i-1}(y)}}{\Pi'_{\mathcal{A}_{i-1}(y)}}\right]\cdot\left(1 + \frac{2}{t}\right) \ .$$

Applying this iteratively we obtain that

$$\mathop{\mathbf{E}}_{z\sim\Pi_{\mathcal{A}}}\left[\frac{\Pi_{\mathcal{A}}(z)}{\Pi'_{\mathcal{A}}(z)}\right] \le \left(1 + \frac{3}{t}\right)^m \le e^{3m/t} \ .$$

By our definition, $t = m/\delta^2 \ge 16m$. Therefore, $3m/t \le 1/5$ and hence $e^{3m/t} \le 1 + 4m/t$. By Lemma 3.14, we get that $\Delta_{TV}(\Pi_{\mathcal{A}}, \Pi'_{\mathcal{A}}) \le \sqrt{(1 + 4m/t - 1)/2} = \sqrt{m/t} = \delta$. This implies that the success probability of $\mathcal{A}$ using the simulated oracle is at least $\alpha - \delta$.

□

We now combine Theorems 3.7 and 3.13 to obtain the following lower bound for decision problems.

Theorem 3.16. *Let X be a domain and D be a distribution over X and $\mathcal{D}$ be a set of distributions over X. For $\bar{\gamma} > 0$, let $d = \mathrm{SDA}(\mathcal{B}(\mathcal{D}, D), \bar{\gamma})$. Any 1-bit sampling algorithm that solves $\mathcal{B}(\mathcal{D}, D)$ with probability α requires at least m queries to 1-STAT for*

$$m = \min\left\{\frac{d(2\alpha-1)}{2}, \frac{(2\alpha+1)^2}{48\bar{\gamma}}\right\} .$$

In particular, any algorithm with success probability of at least $2/3$ requires at least $\min\{d/6, 1/(432\bar{\gamma})\}$ queries to 1-STAT.

Proof. Assuming the existence of a 1-bit sampling algorithm using less than m queries, we apply Theorem 3.13 for $\delta = (2\alpha-1)/4$ to simulate the algorithm using VSTAT. The bound on m ensures that the resulting algorithm uses less than $d(2\alpha-1)/2$ queries to $\mathrm{VSTAT}(\frac{1}{3\bar{\gamma}})$ and has success probability of at least $\alpha - \delta = (2\alpha+1)/4$. By substituting these parameters into Theorem 3.7 we obtain a contradiction. □

For general search problems this leads to the following lower bound.

Theorem 3.17. *Let X be a domain and $\mathcal{Z}$ be a search problem over a set of solutions $\mathcal{F}$ and a class of distributions $\mathcal{D}$ over X. For $\bar{\gamma} > 0$ and $\eta \in (0,1)$, let $d = \mathrm{SDA}(\mathcal{Z}, \bar{\gamma}, \eta)$. Any (possibly randomized) 1-bit sampling algorithm that solves $\mathcal{Z}$ with probability α requires at least m calls to 1-STAT for*

$$m = \min\left\{\frac{d(\alpha-\eta)}{2(1-\eta)}, \frac{(\alpha-\eta)^2}{12\bar{\gamma}}\right\} .$$

In particular, if $\eta \leq 1/6$ then any algorithm with success probability of at least $2/3$ requires at least $\min\{d/4, 1/(48\bar{\gamma})\}$ queries to 1-STAT.

To conclude, we formally state a simple reduction in the other direction, namely that $\mathrm{VSTAT}(t)$ oracle can be simulated using the 1-STAT oracle. It has been observed that, given a Boolean query function h one can obtain an estimate of $\mathbf{E}_D[h]$ using $t = O(\log(1/\delta)/\tau^2)$ 1-bit samples which with probability at least $1-\delta$ will be within τ of $\mathbf{E}_D[h]$ [Ben-David and Dichterman, 1998]. Using the multiplicative Chernoff bound, it is not hard to see that $O(t\log(1/\delta))$ samples are sufficient to estimate $p = \mathbf{E}_D[h]$ within tolerance guaranteed by $\mathrm{VSTAT}(t)$. In addition, we will show how to use 1-STAT oracle to estimate the expectation of real-valued queries.

Theorem 3.18. *Let $t, q > 0$ be any integers and $\delta > 0$. There exists an algorithm $\mathcal{A}'$ that for any input distribution D and any algorithm $\mathcal{A}$ that asks at most q queries to VSTAT, with probability at least $1-\delta$, provides valid for $\mathrm{VSTAT}(t)$ answers to all the queries of $\mathcal{A}$. $\mathcal{A}'$ uses $O(qt \cdot \log(q/\delta))$ queries to 1-STAT for the same input distribution D.*

Proof. For every query $h : X \to [0,1]$ of $\mathcal{A}$, the algorithm $\mathcal{A}'$ estimates $p = \mathbf{E}_D[h]$ as follows. To generate a random Bernoulli variable with bias p, $\mathcal{B}$ draws $\theta \in [0,1]$ randomly and uniformly and defines: $h_\theta(x) = 1$ if $h(x) \leq \theta$ and $h_\theta(x) = 0$ otherwise. It then makes the query h_θ to 1-STAT. Observe that

$$\mathop{\mathbf{Pr}}_{\theta, x\sim D}[h_\theta(x) = 1] = \mathop{\mathbf{E}}_{x\sim D}[\mathop{\mathbf{Pr}}_{\theta}[h_\theta(x) = 1]] = \mathop{\mathbf{E}}_{x\sim D}[h(x)] = p.$$

The algorithm $\mathcal{B}$ repeats this m times (each time choosing a new random θ) and then answers the query h with the mean of the obtained samples (for m to be defined later). We denote the mean by v.

Assuming that $p \le 1/2$, multiplicative Chernoff bounds imply that

$$\mathbf{Pr}\left[|v-p| \ge \sqrt{p(1-p)/t}\right] \le 2e^{-3mp/(p(1-p)t)} \le 2e^{-6m/t}.$$

The bound in the case of $p > 1/2$ follows from the symmetric argument.

Choosing $m = 6t \cdot \ln(2q/\delta)$ ensures that $\mathbf{Pr}[|v-p| \ge \sqrt{p(1-p)/t}] \le \delta/q$. This implies that q arbitrary queries for VSTAT(t) can be answered correctly with probability at least $1-\delta$ using $6qt \cdot \ln(2q/\delta)$ queries to 1-STAT. □

4 Warm-up: MAX-XOR-SAT

In this section, we demonstrate our techniques on a warm-up problem, MAX-XOR-SAT. For this problem, it is sufficient to use pairwise correlations, rather than average correlations.

For $\epsilon \ge 0$, the ϵ-approximate **MAX-XOR-SAT** problem is defined as follows. Given samples from some unknown distribution D over XOR clauses on n variables, find an assignment that maximizes up to additive error ϵ the probability a random clause drawn from D is satisfied.

In the worst case, it is known that MAX-XOR-SAT is NP-hard to approximate to within $1/2-\delta$ for any constant δ [Håstad, 2001]. In practice, local search algorithms such as WalkSat [Selman et al., 1995] are commonly applied as heuristics for maximum satisfiability problems. We give strong evidence that the distributional version of MAX-XOR-SAT is hard for algorithms that locally seek to improve an assignment by flipping variables as to satisfy more clauses, giving some theoretical justification for the observations of [Selman et al., 1995]. Moreover, our proof even applies to the case when there exists an assignment that satisfies all the clauses generated by the target distribution.

The bound we obtain can be viewed as a restatement of the known lower bound for learning parities using statistical query algorithms (indeed, the problem of learning parities is a special case of our distributional MAX-XOR-SAT).

To formalize the search problem, we will denote by $C = \{0,1\}^n$ the set of XOR clauses in n variables, such that for $c \in C$, if for $i \in [n]$ we have $c_i = 1$ then the ith variable appears in c, and otherwise it does not; for simplicity, no variables are negated in the clauses. Let $A = \{0,1\}^n$ denote the set of possible assignments to the variables. We will say that the assignment $a \in A$ satisfies the clause $c \in C$ if $a \cdot c = 1$ (where $a \cdot c$ denotes the inner product modulo 2).

Let $\mathcal{D}$ be the set of distributions over clauses in C. For a distribution $D \in \mathcal{D}$ and an assignment $a \in A$, let $f_D(a) = \mathbf{E}_{c\sim D}[a \cdot c]$ be the fraction of clauses that a satisfies under D. For $D \in \mathcal{D}$ let $M_D = \max_{a\in A} f_D(a)$. The MAX-XOR-SAT problem asks to find $a \in A$ that maximizes $f_D(a)$, given samples from an unknown distribution D.

We are now ready to formalize the search problem that we are interested in, using the notation above and that of Definition 2.1.

Problem 4.1. *(ϵ-approximate MAX-XOR-SAT) Let $X = C = \{0,1\}^n$ (the set of clauses), $\mathcal{D}$ be the set of distributions over X, $\mathcal{F} = A = \{0,1\}^n$ (the set of assignments). Let $\mathcal{Z} : \mathcal{D} \to 2^{\mathcal{F}}$ be defined as $\mathcal{Z}(D) = \{a \in A \mid f_D(a) \ge M_D - \epsilon\}$.*

Theorem 4.2. *For any $\delta > 0$, any SQ algorithm requires at least $2^{n/3} - 1$ queries to STAT$(2^{-n/3})$ to solve $\left(\frac{1}{2} - \delta\right)$-approximate MAX-XOR-SAT.*

We will first determine the statistical dimension of our search problem. This will immediately imply Theorem 4.2 using Corollary 3.11 (by choosing $\gamma' = 2^{-n/3}$).

Lemma 4.3. *For a $\delta > 0$, let $\mathcal{Z}$ denote the $(1/2-\delta)$-approximate MAX-XOR-SAT. Then the statistical dimension of $\mathcal{Z}$ with pairwise correlation $(\gamma, \beta) = (0, 1)$ is $\mathrm{SD}(\mathcal{Z}, 0, 1) \geq 2^n - 1$.*

Proof. We verify the properties of Definition 3.9.

Let the reference distribution $D = U_C$, the uniform distribution over $C = \{0,1\}^n$. For $a \in A = \{0,1\}^n = \mathcal{F}$, let $D_a \in \mathcal{D}$ be the uniform distribution over $c \in C$ such that $a \cdot c = 1$. Let $\mathcal{D}_D = \{D_a \mid \ a \in A\}$, so $|\mathcal{D}_D| = 2^n$.

For $a, b \in A$, we have

$$f_{D_a}(b) = \mathop{\mathbf{E}}_{c \sim D_a}[b \cdot c] = \frac{1}{2^{n-1}} \left(\sum_{c \in C \ \mid \ a \cdot c = 1, b \cdot c = 1} 1 \right).$$

Note that if $b = a$ then $f_{D_a}(b) = 1$, and if $b \neq a$, $f_{D_a}(b) = 1/2$ (indeed, $|\{c \in C \mid a \cdot c = 1, b \cdot c = 1\}| = 2^n/4$ since it is the size of two intersecting affine subspaces in $\{0,1\}^n$).

Therefore, for $a \in A$ and $\epsilon = 1/2 - \delta > 0$, the set of solutions is

$$\mathcal{Z}(D_a) = \{b \in A \mid f_{D_a}(b) \geq 1/2 + \delta\} = \{a\},$$

and so $\mathcal{Z}_a = \{D_a\}$.

To conclude the proof we will show that for any assignment $a \in \mathcal{F} = A$ the set $\mathcal{D}_a = \mathcal{D}_D \setminus \{D_a\}$ of distributions is $(0,1)$-correlated (see Definition 3.8).

Note that for $a \in A$, and $c \in C$,

$$\left(\frac{D_a}{D} - 1\right)(c) = \begin{cases} -1 \text{ if } c \cdot a = 0 \\ 1 \text{ if } c \cdot a = 1. \end{cases}$$

In other words, $\left(\frac{D_a}{D} - 1\right)(c) = -(-1)^{a \cdot c}$. A well-known (and easy to verify) property of $\{-1,1\}$-valued parity functions is that they are $(0,1)$-correlated over the uniform distribution. That is, for $a, b \in A$

$$\left| \left\langle \frac{D_a}{D} - 1, \frac{D_b}{D} - 1 \right\rangle_D \right| \leq \begin{cases} 0 \text{ for } a = b \\ 1 \text{ for } a \neq b. \end{cases}$$

□

5 Planted Biclique and Densest Subgraph

5.1 Statistical Dimension of Planted Biclique

We now prove the lower bound claimed in Theorem 2.9 on the problem of detecting a planted k-biclique in the given distribution on vectors from $\{0,1\}^n$ as defined above.

Throughout this section we will use the following notation. For a subset $S \subseteq [n]$, let D_S be the distribution over $\{0,1\}^n$ with a planted set S. Let $\mathcal{S}_k$ denote the set of all $\binom{n}{k}$ subsets of $[n]$ of size k and $m = \binom{n}{k}$. We index the elements of $\mathcal{S}_k$ in some arbitrary order as $S_1, \ldots, S_m$. For $i \in [m]$, we use D_i to denote D_{S_i}. We will also assume, whenever necessary, that k and n are larger than some fixed constant.

The reference distribution in our lower bounds will be the uniform distribution over $\{0,1\}^n$ and let $\hat{D}_S$ denote $D_S/D - 1$. In order to apply our lower bounds based on statistical dimension with average correlation we now prove that for the planted biclique problem average correlations of large sets of distributions must be small. We start with a lemma that bounds the correlation of two planted biclique distributions relative to the reference distribution D as a function of the overlap between the planted sets:

Lemma 5.1. *For $i, j \in [m]$,*

$$\rho_D(D_i, D_j) = \left| \left\langle \hat{D}_i, \hat{D}_j \right\rangle_D \right| \leq \frac{2^\lambda k^2}{n^2},$$

where $\lambda = |S_i \cap S_j|$.

Proof. For the distribution D_i, we consider the probability $D_i(x)$ of generating the vector x. Then,

$$D_i(x) = \begin{cases} (\frac{n-k}{n})\frac{1}{2^n} + (\frac{k}{n})\frac{1}{2^{n-k}} & \text{if } \forall s \in S_i, x_s = 1 \\ (\frac{n-k}{n})\frac{1}{2^n} & \text{otherwise.} \end{cases}$$

Now we compute the vector $\hat{D}_i = \frac{D_i}{D} - 1$:

$$\frac{D_i}{D} - 1 = \begin{cases} \frac{k2^k}{n} - \frac{k}{n} & \text{if } \forall s \in S_i, x_s = 1 \\ -\frac{k}{n} & \text{otherwise.} \end{cases}$$

We then bound the inner product:

$$\begin{aligned} \left\langle \hat{D}_i, \hat{D}_j \right\rangle_D &\leq \frac{2^{n-2k+\lambda}}{2^n}\left(\frac{k2^k}{n} - \frac{k}{n}\right)^2 + 2\left(\frac{2^{n-k}}{2^n} - \frac{2^{n-2k+\lambda}}{2^n}\right)\left(\frac{k2^k}{n} - \frac{k}{n}\right)\left(-\frac{k}{n}\right) + \left(-\frac{k}{n}\right)^2 \\ &\leq \frac{2^\lambda k^2}{n^2}, \end{aligned}$$

which holds when $k \geq 3$. We also note that $\left\langle \hat{D}_i, \hat{D}_j \right\rangle_D \geq 0$. □

We now give a bound on the average correlation of any $\hat{D}_S$ with a large number of distinct biclique distributions.

Lemma 5.2. *Let $\delta \geq 1/\log n$ and $k \leq n^{1/2-\delta}$. For any integer $\ell \leq k$, $S \in \mathcal{S}_k$ and any set $A \subseteq \mathcal{S}_k$ where $|A| \geq 3(m-1)/n^{2\ell\delta}$,*

$$\frac{1}{|A|}\sum_{S_i \in A} \left| \langle \hat{D}_S, \hat{D}_i \rangle \right| < 2^{\ell+1}\frac{k^2}{n^2}.$$

Proof. In this proof we first show that if the total number of sets in A is large then most of sets in A have a small overlap with S. We then use the bound on the overlap of most sets to obtain a bound on the average correlation of D_S with distributions for sets in A.

Formally, we let $\alpha = \frac{k^2}{n^2}$ and using Lemma 5.1 get the bound $|\langle \hat{D}_i, \hat{D}_j \rangle| \leq 2^{|S_i \cap S_j|}\alpha$. Summing over $S_i \in A$,

$$\sum_{S_i \in A} \left| \langle \hat{D}_S, \hat{D}_i \rangle \right| \leq \sum_{S_i \in A} 2^{|S \cap S_i|}\alpha.$$

For any set $A \subseteq \mathcal{S}_k$ of size t this bound is maximized when the sets of A include S, then all sets that intersect S in $k-1$ indices, then all sets that intersect S in $k-2$ indices and so on until the size bound t is exhausted. We can therefore assume without loss of generality that A is defined in precisely this way.

Let $T_\lambda = \{S_i \mid |S \cap S_i| = \lambda\}$ denote the subset of all k-subsets that intersect with S in exactly λ indices. Let λ_0 be the smallest λ for which $A \cap T_\lambda$ is non-empty. We first observe that for any $1 \leq j \leq k-1$,

$$\begin{aligned} \frac{|T_j|}{|T_{j+1}|} = \frac{\binom{k}{j}\binom{n-k}{k-j}}{\binom{k}{j+1}\binom{n-k}{k-j-1}} = \frac{(j+1)(n-2k+j+1)}{(k-j)^2} \geq \frac{(j+1)(n-2k)}{(k-j)^2} \geq \\ \geq \frac{(j+1)(n-2n^{1/2-\delta})}{n^{1-2\delta}} \geq \frac{(j+1)(1-2n^{-1/2-\delta})}{n^{-2\delta}} \geq \frac{(j+1)n^{2\delta}}{2}. \end{aligned} \tag{5}$$

By applying this equation inductively we obtain,

$$|T_j| \leq \frac{2^j \cdot |T_0|}{j! \cdot n^{2\delta j}} < \frac{2^j \cdot (m-1)}{j! \cdot n^{2\delta j}}$$

where the last inequality holds since $|T_0| \leq m-2$ whenever $n \geq 2k+1$. For n larger than some fixed constant

$$\sum_{k \geq \lambda \geq j} |T_\lambda| < \sum_{k \geq \lambda \geq j} \frac{2^\lambda \cdot (m-1)}{\lambda! \cdot n^{2\delta\lambda}} \leq \frac{m-1}{n^{2\delta j}} \sum_{k \geq \lambda \geq j} \frac{2^\lambda}{\lambda! \cdot n^{2\delta(\lambda - j)}} \leq \frac{3(m-1)}{n^{2\delta j}} \ .$$

By definition of λ_0, $|A| \leq \sum_{j \geq \lambda_0} |T_j| < 3(m-1)/n^{2\delta\lambda_0}$. In particular, if $|A| \geq 3(m-1)/n^{2\ell\delta}$ then $n^{2\delta\lambda_0} < n^{2\ell\delta}$ or $\lambda_0 < \ell$. Now we can conclude that

$$\begin{aligned} \sum_{S_i \in A} \left| \langle \hat{D}_S, \hat{D}_i \rangle \right| &\leq \sum_{j=\lambda_0}^{k} 2^j |T_j \cap A| \alpha \\ &\leq \left(2^{\lambda_0} |T_{\lambda_0} \cap A| + \sum_{j=\lambda_0+1}^{k} 2^j |T_j| \right) \alpha \\ &\leq \left(2^{\lambda_0} |T_{\lambda_0} \cap A| + 2 \cdot 2^{\lambda_0+1} |T_{\lambda_0+1}| \right) \alpha \\ &< 2^{\lambda_0+2} |A| \alpha \leq 2^{\ell+1} |A| \alpha. \end{aligned}$$

To derive the second to last inequality we need to note that for every $j \geq 0$, $2^j|T_j| > 2(2^{j+1}|T_{j+1}|)$ whenever $n^{2\delta} \geq 4$. We can therefore telescope the sum. □

We can now bound the statistical dimension (with average correlation) of the planted k-biclique problem.

Theorem 5.3. *For $\delta \geq 1/\log n$ and $k \leq n^{1/2-\delta}$ let $\mathcal{Z}$ the distributional planted k-biclique problem. Then for any $\ell \leq k$, $\mathrm{SDA}(\mathcal{Z}, 2^{\ell+1}k^2/n^2, 1/\binom{n}{k}) \geq n^{2\ell\delta}/3$. In addition, let D be the uniform distribution and denote the set of all planted distributions by $\mathcal{D}$. Then, $\mathrm{SDA}(\mathcal{D}, D, 2^{\ell+1}k^2/n^2) \geq n^{2\ell\delta}/3$.*

Proof. For every solution $S \in \mathcal{F}$, $\mathcal{Z}_S = \{D_S\}$ and let $\mathcal{D}_S = \mathcal{D} \setminus \{D_S\}$. Note that $|\mathcal{D}_S| = \binom{n}{k} - 1$ and therefore $|\mathcal{D}_S| \geq (1 - 1/\binom{n}{k})|\mathcal{D}|$. This means that we can use $1/\binom{n}{k}$ as the solution set bound.

Let $\mathcal{D}'$ be a set of distributions $\mathcal{D}' \subseteq \mathcal{D}_S$ such that $|\mathcal{D}'| \geq 3(m-1)/n^{2\ell\delta}$. Then by Lemma 5.2, for every $S_i \in \mathcal{D}'$,

$$\frac{1}{|\mathcal{D}'|} \sum_{S_j \in \mathcal{D}'} \left| \langle \hat{D}_i, \hat{D}_j \rangle \right| < 2^{\ell+1} \frac{k^2}{n^2}.$$

In particular, $\rho(\mathcal{D}', D) < 2^{\ell+1}\frac{k^2}{n^2}$. By the definition of SDA (Definition 3.1), this means that $\mathrm{SDA}(\mathcal{Z}, 2^{\ell+1}k^2/n^2, 1/\binom{n}{k}) \geq n^{2\ell\delta}/3$.

The second claim holds by exactly the same argument since $|\mathcal{D}'| \geq m/d$ implies $|\mathcal{D}'| \geq (m-1)/d$. □

For a positive r we choose $\ell = \log r - 1$. Our lower bound for the planted bi-clique problem stated in Theorem 2.9 follows from substituting the bound $\mathrm{SDA}(\mathcal{Z}, rk^2/n^2, 1/\binom{n}{k}) \geq n^{2(\log(r)-1)\delta}/3$ into Theorem 3.2 (with $\eta = 1/\binom{n}{k}$ and $\delta = 2/3$). In addition, by Theorem 3.7 used with $\alpha = 1/2 + 1/t$, we obtain hardness of the decision version of the problem for randomized SQ algorithms which also implies Theorem 2.9.

Theorem 5.4. *For any constant $\delta > 0$, any $k \leq n^{1/2-\delta}$ and $r > 0$, let D be the uniform distribution over $\{0,1\}^n$ and $\mathcal{D}$ be the set of all planted k-biclique distributions. For some $t = n^{\Omega(\log r)}$, any randomized SQ algorithm that solves the decision problem $\mathcal{B}(\mathcal{D}, D)$ with probability $1/2 + 1/t$ requires t queries to VSTAT$(n^2/(rk^2))$.*

Theorem 3.13 used with $\delta = 1/9$ implies that an algorithm that uses m queries to 1-STAT and has success probability $2/3$ gives an algorithm that uses m queries to $\mathrm{VSTAT}(81m)$ and has success probability $2/3 - 1/9 = 5/9$. For some $m = \Omega(n^2/k^2)$, Theorem 5.4 applied with $r = \Omega(1)$ implies that such algorithm cannot exist. This implies the lower bound for 1-bit sampling algorithms stated in Theorem 2.10.

5.2 Generalized Planted Densest Subgraph

We will now show lower bounds on detecting a (p,q)-planted densest subgraph, a generalization of the distributional planted biclique problem we defined in Definition 2.11. Note that $p = 1, q = 1/2$ is precisely the distributional planted k-biclique problem. For this generalized problem, we will take D, the reference distribution, to be that of n independent Bernoulli variables with bias q.

Before we give our results for this problem, we have to fix some further notation: for $x \in \{0,1\}^n$, we define $\|x\|_1 = \sum x_i$ (*i.e.* the number of 1's in x); similarly for $\|\overline{x}\|_1 = \sum 1 - x_i$ (the number of 0's in x). We will denote the restriction of a set by subscripting so that x_S is x restricted to the subset $S \subseteq [n]$. We use $\bar{S}$ to denote the complement of S in the current ground set.

First, we give a computation of the correlation. This is a generalized version of Lemma 5.1.

Lemma 5.5. *Fix $0 < q \leq p \leq 1$ and let $\Delta_{pq} = 1 + \frac{(p-q)^2}{q(1-q)}$. For $i, j \in [m]$,*

$$\left\langle \hat{D}_i, \hat{D}_j \right\rangle_D = \left(\Delta_{pq}^{\lambda} - 1 \right) \frac{k^2}{n^2},$$

where $\lambda = |S_i \cap S_j|$.

Proof. For any x, we have $D(x) = q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1}$. For $D_i(x)$:

$$\begin{aligned} D_i(x) &= \mathbf{Pr}[x|\text{planted}]\,\mathbf{Pr}[\text{planted}] + \mathbf{Pr}[x|\text{not planted}]\,\mathbf{Pr}[\text{not planted}] \\ &= \frac{k}{n} p^{\|x_{S_i}\|_1}(1-p)^{\|\overline{x_{S_i}}\|_1} q^{\left\|x_{\bar{S_i}}\right\|_1}(1-q)^{\left\|\overline{x_{\bar{S_i}}}\right\|_1} + \left(1 - \frac{k}{n}\right) q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1}. \end{aligned}$$

For $D_i(x)/D(x) - 1$, we have:

$$\begin{aligned} \frac{D_i(x)}{D(x)} - 1 &= \frac{k}{n} \cdot \frac{p^{\|x_{S_i}\|_1}(1-p)^{\|\overline{x_{S_i}}\|_1} q^{\left\|x_{\bar{S_i}}\right\|_1}(1-q)^{\left\|\overline{x_{\bar{S_i}}}\right\|_1}}{q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1}} - \frac{k}{n} \\ &= \frac{k}{n}\left(\frac{p}{q}\right)^{\|x_{S_i}\|_1}\left(\frac{1-p}{1-q}\right)^{\|\overline{x_{S_i}}\|_1} - \frac{k}{n}. \end{aligned}$$

Now, for S_j where $|S_i \cap S_j| = \lambda$, we want to compute:

$$\left\langle \hat{D}_i, \hat{D}_j \right\rangle_D = \left(\frac{k}{n}\right)^2 \sum_{x \in \{0,1\}^n} q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1} \left[\left(\frac{p}{q}\right)^{\|x_{S_i}\|_1}\left(\frac{1-p}{1-q}\right)^{\|\overline{x_{S_i}}\|_1} - 1\right]\left[\left(\frac{p}{q}\right)^{\left\|x_{S_j}\right\|_1}\left(\frac{1-p}{1-q}\right)^{\left\|\overline{x_{S_j}}\right\|_1} - 1\right].$$

There are three types of terms in the product in the summand. We deal with all these terms by repeated applications of the Binomial theorem. The first term illustrates this approach:

$$\sum_{x \in \{0,1\}^n} q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1} = (q + (1-q))^n = 1.$$

The second type of term is given by:

$$\begin{aligned} &\sum_{x \in \{0,1\}^n} q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1}\left(\frac{p}{q}\right)^{\|x_{S_i}\|_1}\left(\frac{1-p}{1-q}\right)^{\|\overline{x_{S_i}}\|_1} \\ &\quad = \sum_{x \in \{0,1\}^n} q^{\left\|x_{\bar{S_i}}\right\|_1}(1-q)^{\left\|\overline{x_{\bar{S_i}}}\right\|_1} p^{\|x_{S_i}\|_1}(1-p)^{\|\overline{x_{S_i}}\|_1} \\ &\quad = \sum_{y \in \{0,1\}^{|S_i|}} p^{\|y\|_1}(1-p)^{\|\overline{y}\|_1} \sum_{z \in \{0,1\}^{|\bar{S_i}|}} p^{\|z\|_1}(1-p)^{\|\overline{z}\|_1} \\ &\quad = 1. \end{aligned}$$

The third type of term is more complicated – using the above trick, we can restrict x to $T = S_i \cup S_j$ because the sum taken over the remaining x_i yields 1.

$$\begin{aligned} &\sum_{x \in \{0,1\}^n} q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1}\left[\left(\frac{p}{q}\right)^{\|x_{S_i}\|_1}\left(\frac{1-p}{1-q}\right)^{\|\overline{x_{S_i}}\|_1}\right]\left[\left(\frac{p}{q}\right)^{\left\|x_{S_j}\right\|_1}\left(\frac{1-p}{1-q}\right)^{\left\|\overline{x_{S_j}}\right\|_1}\right] \\ &= \sum_{x \in \{0,1\}^{|T|}} q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1}\left[\left(\frac{p}{q}\right)^{\|x_{S_i}\|_1}\left(\frac{1-p}{1-q}\right)^{\|\overline{x_{S_i}}\|_1}\right]\left[\left(\frac{p}{q}\right)^{\left\|x_{S_j}\right\|_1}\left(\frac{1-p}{1-q}\right)^{\left\|\overline{x_{S_j}}\right\|_1}\right]. \end{aligned}$$

Similarly, we can sum x over coordinates in $S_i \setminus S_j$ and $S_j \setminus S_i$. Hence, the sum simplifies:

$$\begin{aligned}
&\sum_{x\in\{0,1\}^n} q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1}\left[\left(\frac{p}{q}\right)^{\|x_{S_i}\|_1}\left(\frac{1-p}{1-q}\right)^{\|\overline{x_{S_i}}\|_1}\right]\left[\left(\frac{p}{q}\right)^{\left\|x_{S_j}\right\|_1}\left(\frac{1-p}{1-q}\right)^{\left\|\overline{x_{S_j}}\right\|_1}\right]\\
&= \sum_{x\in\{0,1\}^{\left|S_i\cap S_j\right|}} q^{\|x\|_1}(1-q)^{\|\overline{x}\|_1}\left[\left(\frac{p}{q}\right)^{\|x\|_1}\left(\frac{1-p}{1-q}\right)^{\|\overline{x}\|_1}\right]^2\\
&= \sum_{x\in\{0,1\}^{\left|S_i\cap S_j\right|}} \left(\frac{p^2}{q}\right)^{\|x\|_1}\left(\frac{(1-p)^2}{1-q}\right)^{\|\overline{x}\|_1}\\
&= \left(\frac{p^2}{q}+\frac{(1-p)^2}{1-q}\right)^{\lambda}\\
&= \Delta_{pq}^{\lambda}.
\end{aligned}$$

Combining these three calculations yields:

$$\left\langle \hat{D}_i, \hat{D}_j \right\rangle_D = \left(\frac{k}{n}\right)^2\left(\Delta_{pq}^{\lambda}-1\right)$$

□

Next, in analogy with Lemma 5.2, we give a bound on average correlation for sufficiently many distributions.

Lemma 5.6. *Fix $0 < q < p \le 1$ and let $\Delta_{pq} = 1 + \frac{(p-q)^2}{q(1-q)}$. For $\delta > 0$ and $k \le n^{1/2-\delta}$, if $n^{2\delta} \ge 8\Delta_{pq}$ then for any integer $\ell \le k$, $S \in \mathcal{S}_k$ and $A \subseteq \mathcal{S}_k$ of size at least $2(m-1)/n^{2\ell\delta}$,*

$$\frac{1}{|A|}\sum_{S_i\in A}\left|\left\langle \hat{D}_S, \hat{D}_i\right\rangle\right| < \frac{2k^2}{n^2}\left(\Delta_{pq}^{\ell}-1\right).$$

Proof. We proceed as in the proof of Lemma 5.2. Recall that $T_\lambda = \{S_i \mid |S \cap S_i| = \lambda\}$ denotes the subset of all k-subsets that intersect with S in exactly λ indices. Let λ_0 be the smallest λ for which $A \cap T_\lambda$ is non-empty. As before, we obtain that $\lambda_0 < \ell$.

We now bound the average correlation with $\hat{D}_S$ as follows:

$$\begin{aligned}
\sum_{S_i\in A}\left|\langle \hat{D}_S, \hat{D}_i\rangle\right| &\le \sum_{j=\lambda_0}^{k}\frac{k^2}{n^2}\cdot\left(\Delta_{pq}^{j}-1\right)|T_j\cap A|\\
&\le \frac{k^2}{n^2}\cdot\left(|T_{\lambda_0}\cap A|\left(\Delta_{pq}^{\lambda_0}-1\right)+\sum_{j=\lambda_0+1}^{k}|T_j|(\Delta_{pq}^{j}-1)\right).
\end{aligned}$$

To bound the sum

$$\sum_{j=\lambda_0+1}^{k}(\Delta_{pq}^{j}-1)\,|T_j|$$

it suffices to show that it is geometrically decreasing as:

$$(\Delta_{pq}^{j}-1)\left|T_j\right| \geq 2\cdot(\Delta_{pq}^{j+1}-1)\left|T_{j+1}\right|.$$

We first note that $\Delta_{pq} > 1$ and therefore for $j \geq 1$,

$$\frac{\Delta_{pq}^{j+1}-1}{\Delta_{pq}^{j}-1} \leq \Delta_{pq}+1 < 2\Delta_{pq}.$$

From equation (5) in the proof of Lemma 5.2 and our assumption on Δ_{pq} we obtain the necessary property:

$$\frac{|T_j|}{|T_{j+1}|} \geq \frac{(j+1)n^{2\delta}}{2} \geq 4\cdot\Delta_{pq} > \frac{2(\Delta_{pq}^{j+1}-1)}{\Delta_{pq}^{j}-1}.$$

To conclude,

$$\begin{aligned}\sum_{S_i\in A}\left|\langle\hat{D}_S,\hat{D}_i\rangle\right| &\leq \frac{k^2}{n^2}\cdot\left(|T_{\lambda_0}\cap A|\left(\Delta_{pq}^{\lambda_0}-1\right)+\sum_{j=\lambda_0+1}^{k}|T_j|(\Delta_{pq}^{j}-1)\right)\\ &\leq \frac{k^2}{n^2}\cdot\left(|T_{\lambda_0}\cap A|\left(\Delta_{pq}^{\lambda_0}-1\right)+2\cdot|T_{\lambda_0+1}|(\Delta_{pq}^{\lambda_0+1}-1)\right)\\ &\leq 2\cdot\frac{k^2}{n^2}\cdot|A|\left(\Delta_{pq}^{\lambda_0+1}-1\right)\\ &\leq 2\cdot\frac{k^2}{n^2}\cdot|A|\left(\Delta_{pq}^{\ell}-1\right).\end{aligned}$$

□

From here the bound on statistical dimension SDA of detecting the (p,q)-planted densest subgraph now follows in the same way as in Theorem 5.3.

Theorem 5.7. *Fix $0 < q < p \leq 1$. For $\delta > 0$ and $k \leq n^{1/2-\delta}$ let $\mathcal{Z}$ be the distributional (p,q)-planted densest k-subgraph problem. Then for any $\ell \leq k$,*

$$\mathrm{SDA}\left(\mathcal{Z},\frac{2k^2}{n^2}\left(\Delta_{pq}^{\ell}-1\right),\frac{1}{\binom{n}{k}}\right) \geq n^{2\ell\delta}/2.$$

provided that $n^{2\delta} \geq 8\Delta_{pq}$.

This SDA bound yields lower bounds for the VSTAT oracle:

Corollary 5.8. *Fix $0 < q < p \leq 1$ and let $\Delta_{pq} = 1+\frac{(p-q)^2}{q(1-q)}$. For any constant $\delta > 0$, any $k \leq n^{1/2-\delta}$, $\ell \leq k$, at least $n^{\Omega(\ell)}$ queries to VSTAT$(n^2/(6k^2(\Delta_{pq}^{\ell}-1)))$ are required to solve the distributional (p,q)-planted densest k-subgraph problem with probability at least $2/3$ provided that $n^{2\delta} \geq 8\Delta_{pq}$.*

Similarly, by Theorem 3.7, the same lower bound applies to the decision version of the problem.

One is often interested in the case when $q = 1/2$ and $p = 1/2 + \alpha$ (the classical planted densest k-subgraph problem). In this setting $\Delta_{pq} = 1 + 4\alpha^2$ and $\Delta_{pq}^{\ell} - 1 \leq e^{4\alpha^2 \ell} - 1 \leq 8\alpha^2 \ell$ whenever $\ell \leq 1/(4\alpha^2)$. This gives a lower bound of $n^{\Omega(\ell)}$ against $\mathrm{VSTAT}(n^2/(48\ell\alpha^2 k^2))$ as stated in Corollary 2.12.

Finally, we give an example of a corollary for the 1-STAT oracle.

Corollary 5.9. *For constants $c, \delta > 0$, density $p = 1/2 + 1/n^c$, and $k \leq n^{1/2-\delta}$, Let D be the uniform distribution over $\{0,1\}^n$ and $\mathcal{D}$ be the set of all $(p, 1/2)$-planted densest k-subgraph distributions. Any (randomized) 1-bit sampling algorithm that solves the decision problem $\mathcal{B}(\mathcal{D}, D)$ with probability at least $2/3$, requires $\Omega((n^{2+2c})/k^2)$ queries to 1-STAT.*

Proof. By the argument above with $\alpha = 1/n^c$, $\mathrm{SDA}(\mathcal{B}(\mathcal{D}, D), 16k^2\ell/n^{2+2c}) \geq n^{2\ell\delta}/2$. For $\ell = 4/2\delta$ we obtain that $\mathrm{SDA}(\mathcal{B}(\mathcal{D}, D), 64k^2/n^{2+2c}) \geq n^4/2$. By applying Theorem 3.16 for success probability 2/3, we obtain a lower bound of

$$m = \min\left\{\frac{d/3}{2}, \frac{(4/3+1)^2}{48\bar{\gamma}}\right\} = \min\left\{\frac{n^4}{12}, \frac{49/9}{48} \cdot \frac{n^{2+2c}}{64k^2}\right\} = \Omega\left(\frac{n^{2+2c}}{k^2}\right) .$$

samples to 1-STAT. □

6 Applications to Statistical Query Learning

We will now use Corollary 3.12 to demonstrate that our results generalize the notion of statistical query dimension in learning theory and the statistical query lower bounds based on SQ-DIM. We then show that our lower bounds imply stronger and more general lower bounds in the context of learning.

We start with a few relevant definitions. In an instance of a PAC learning problem, the learner has access to random examples of an unknown boolean function $c : X' \to \{-1, 1\}$ from a set of Boolean functions $\mathcal{C}$. A random example is a pair including a point and its label $(x', c(x'))$ such that x' is drawn randomly from a distribution D', which might or might not be known to the learning algorithm (whenever necessary, we use $'$ to distinguish variables from the identically named ones in the context of general search problems). Specifically, for a target function $c \in C$ and distribution D' over X' we denote by D_c over $X = X' \times \{-1, 1\}$, where $D_c(x', c(x')) = D'(x')$ and $D_c(x', -c(x')) = 0$.

For $\epsilon > 0$, the goal of an ϵ-accurate learning algorithm is to find, with high probability, a Boolean hypothesis h for which $\mathbf{Pr}_{x' \sim D'}[h(x') \neq c(x')] \leq \epsilon$. A statistical query learning algorithm [Kearns, 1998] has access to the STAT oracle for the input distribution D_c in place of random examples.

6.1 Relationship to SQ-DIM

Blum et al. [1994] defined the *statistical query dimension* or SQ-DIM of a set of functions $\mathcal{C}$ and distribution D' over X' as follows (we present a simplification and strengthening due to Yang [2005]).

Definition 6.1 ([Blum et al., 1994]). *For a concept class $\mathcal{C}$ and distribution D', SQ-DIM$(\mathcal{C}, D') = d'$ if d' is the largest value for which there exist d' functions $c_1, c_2, \ldots, c_{d'} \in \mathcal{C}$ such that for every $i \neq j$, $|\langle c_i, c_j \rangle_{D'}| \leq 1/d'$.*

We first observe that correlations of Boolean functions relative to a distribution D' are equivalent to correlations of corresponding distributions over examples relative to some reference distribution. Namely, let the reference distribution D be the distribution for which for every $(x', \ell) \in X$, $D(x', \ell) = D'(x')/2$. This is the distribution in which points are distributed according to D' and labels are random unbiased coin flips. We denote it by $D' \times \{1/2, 1/2\}$.

Lemma 6.2. *For a distribution D' and any Boolean functions c, c_1 and c_2, For all $x' \in X'$, $\frac{D_c(x',\ell)}{D(x',\ell)} - 1 = \ell \cdot c(x')$ and*

$$\left\langle \frac{D_{c_1}}{D} - 1, \frac{D_{c_2}}{D} - 1 \right\rangle_D = \langle c_1, c_2 \rangle_{D'}.$$

Proof. We first note that the definition of D ensures that $D(x', \ell)$ is non-vanishing only when $D'(x')$ is non-vanishing and hence the function $\left(\frac{D_c}{D} - 1\right)$ is well-defined for any Boolean $c \in \mathcal{C}$. For every $c \in \mathcal{C}$, we have

$$\frac{D_c(x', c(x'))}{D(x', c(x'))} - 1 = 2 - 1 = 1 \quad \text{and} \quad \frac{D_c(x', -c(x'))}{D(x', -c(x'))} - 1 = 0 - 1 = -1.$$

Therefore, $\frac{D_c(x',\ell)}{D(x',\ell)} - 1 = \ell \cdot c(x')$. This implies that for any $c_1, c_2 \in \mathcal{C}$,

$$\left\langle \frac{D_{c_1}}{D} - 1, \frac{D_{c_2}}{D} - 1 \right\rangle_D = \mathop{\mathbf{E}}_{(x',\ell)\sim D}[\ell \cdot c_1(x') \cdot \ell \cdot c_2(x')] = \mathop{\mathbf{E}}_{D'}[c_1(x') \cdot c_2(x')] = \langle c_1, c_2 \rangle_{D'}.$$

□

The direct implication of this is that if SQ-DIM$(\mathcal{C}, D') = d'$ then there exist d' distributions over examples that are $(1/d', 1)$-correlated relative to D. In particular, the decision problem of distinguishing example distributions from D has large statistical dimension with pairwise correlations. We state this formally:

Theorem 6.3. *For a concept class $\mathcal{C}$ and distribution D' over X let $d' = SQ\text{-}DIM(\mathcal{C}, D')$. Then for $\mathcal{D}_\mathcal{C} = \{D_c \mid c \in \mathcal{C}\}$ and $D = D' \times \{1/2, 1/2\}$, $\mathrm{SD}(\mathcal{B}(\mathcal{D}_\mathcal{C}, D), 1/d', 1) \geq d'$.*

Blum et al. [1994] proved that if a class of functions is learnable using only a polynomial number of statistical queries of inverse polynomial tolerance then its statistical query dimension is polynomial. Yang [2005] strengthened their result and proved the following bound (see [Szörényi, 2009] for a simpler proof).

Theorem 6.4 ([Yang, 2005])**.** *Let $\mathcal{C}$ be a class of functions and D' be a distribution over X' and let $d' = SQ\text{-}DIM(\mathcal{C}, D')$. Any SQ algorithm that learns $\mathcal{C}$ over D' with error $\epsilon < 1/2 - 1/(2d'^{1/3})$ requires at least $d'^{1/3}/2 - 1$ queries to $STAT(1/d'^{1/3})$.*

In this result, the distribution D' is fixed and known to the learner (such learning is referred to as *distribution-specific*) and it can be used to lower bound the complexity of learning $\mathcal{C}$ even in a weak sense. Specifically, when the learning algorithm is only required to output a hypothesis h' such that $\mathbf{Pr}_{x'\sim D'}[h'(x') \neq c(x')] \leq 1/2 - \gamma'$ for some inverse polynomial γ'. It is well-known that weak learning of functions from $\mathcal{C}$ implies ability to distinguish examples of any function in $\mathcal{C}$ from points labeled randomly. This implies that we can apply our lower bound for decision problems to obtain a lower bound for weak learning that is essentially the same as the result of Yang [2005].

Corollary 6.5. *Let $\mathcal{C}$ be a class of functions and D' be a distribution over X' and let $d' =$ SQ-DIM$(\mathcal{C}, D')$. Any SQ algorithm that learns $\mathcal{C}$ over D' with error $\epsilon < 1/2 - 1/d'^{1/3}$ requires at least $2d'^{1/3} - 1$ queries to STAT$(1/d'^{1/3})$.*

Proof. Let $\mathcal{D}_{\mathcal{C}} = \{D_c \mid c \in \mathcal{C}\}$ and $D = D' \times \{1/2, 1/2\}$. We convert the weak learning algorithm into the algorithm for $\mathcal{B}(\mathcal{D}_{\mathcal{C}}, D)$ as follows. Run the weak learning algorithm. Given hypothesis h estimate the prediction error within $d'^{-1/3}/2$ by using the query $\phi(x', \ell) = h(x') \cdot \ell$ with tolerance $d'^{-1/3}$. If the answer to the query is $> d'^{-1/3}$ output 1 (meaning the input distribution is in $\mathcal{D}_{\mathcal{C}}$), otherwise output 0 (meaning that the input distribution is D). Note that $\mathbf{E}_D[\phi] = 0$ and therefore this algorithm will always output 0 on D. Further, if the input distribution is D_c and $\mathbf{Pr}_{D'}[h(x') \neq c(x')] < 1/2 - 1/d'^{1/3}$ then

$$\mathop{\mathbf{E}}_{(x',\ell)\sim D_c}[\phi(x', \ell)] = \mathop{\mathbf{E}}_{(x',\ell)\sim D_c}[h(x') \cdot \ell] = \mathop{\mathbf{E}}_{x'\sim D'}[h(x') \cdot c(x')] = 1 - 2 \mathop{\mathbf{Pr}}_{x'\sim D'}[h(x') \neq c(x')] > 2/d'^{1/3}.$$

Therefore in this case the answer to the query will be $> 2/d'^{1/3} - 1/d'^{1/3} = 1/d'^{1/3}$ and the algorithm will output 1.

By Theorem 6.3, $\mathrm{SD}(\mathcal{B}(\mathcal{D}_{\mathcal{C}}, D), 1/d', 1) \geq d'$. We can now apply the lower bound in Corollary 3.12 with $\gamma' = d'^{-2/3}/2$ to obtain that our algorithm must use $2d'^{1/3}$ queries to $\mathrm{STAT}(d'^{-1/3})$ to solve the problem. Our algorithm used one more query than the learning algorithm (of the same tolerance) which gives the stated lower bound. □

This corollary implies that lower bounds based on SQ-DIM are a special case of our lower bounds. One can also similarly show that the lower bounds based on the statistical query dimension of Feldman [2012] that characterizes learning to high accuracy are also a special case of our lower bounds.

6.2 Lower Bounds for 1-bit Sampling Oracle

We now show how our results can be used to obtain lower bounds against 1-bit sampling algorithms based on SQ-DIM. Such lower bounds have been previously proved by Yang [2005] who referred to his model as Honest SQ model (apparently unaware of the connection to the model in [Ben-David and Dichterman, 1998]). In the Honest SQ model, the learner has access to an HSQ oracle. A query to HSQ oracle is a function $\phi : X' \times \{-1, 1\} \to \{-1, 1\}$ and a sample size $t > 0$. The oracle draws $x'_1, \ldots, x'_t \sim D'$, and returns the value $\frac{1}{t}\sum_{i=1}^{t} \phi(x', c(x'))$. The total sample complexity of an algorithm is the sum of the sample sizes it passes to HSQ.

We note that using 1-STAT is equivalent to a call to HSQ with sample size 1. Also 1-STAT can simulate estimation of queries from a larger number of samples in a straightforward way while obtaining the same total sample complexity. Therefore HSQ is equivalent to the 1-STAT oracle.

Using Lemma 3.10 with $\gamma' = 1/\sqrt{d'}$ to convert a bound on pairwise correlations to a bound on average correlation, we can obtain that $\mathrm{SDA}(\mathcal{B}(\mathcal{D}_{\mathcal{C}}, D), 1/\sqrt{d'} + 1/d', 1) \geq \sqrt{d'}/(1 - 1/d')$. Plugging this bound into Theorem 3.16, we can derive sample complexity bounds on 1-bit sampling algorithms for learning in the same way as in the proof of Corollary 6.5.

Corollary 6.6. *Let $\mathcal{C}$ be a class of functions, D' be a distribution over X', $d' =$ SQ-DIM$(\mathcal{C}, D')$ and $\epsilon = 1/2 - 1/d'^{1/4}$. Then any 1-bit sampling algorithm that, with probability at least $2/3$, ϵ-accurately learns $\mathcal{C}$ over D' requires $\Omega(\sqrt{d'})$ queries to 1-STAT.*

This lower bound is similar to the result of Yang [2005] who shows a bound of $\Omega(d'/\log d')$ using a stronger $1/d'^3$ upper bound on correlations (and a substantially more involved proof). Note that the inverse of the maximum pairwise correlation is usually much lower than the number of functions. Therefore our result will give a stronger lower bound in most cases.

6.3 New Lower Bound for Learning

We now briefly describe a version of our lower bound for weak distribution-specific learning. It is stronger than known SQ-DIM-based bounds in several ways. First, it explicitly decouples the tolerance (or number of samples) from the number of queries. This is particularly relevant for *attribute-efficient* learning that is learning when the dimension is high but the target function depends on few variables (see [Feldman, 2014] for more details on SQ learning in this setting). Second, it captures sample complexity in a tighter way by going to average correlations and proving lower bounds against VSTAT. Lower bounds against VSTAT also imply tighter lower bounds for 1-STAT and, via the reductions in [Feldman et al., 2013], against stronger oracles.

We now give versions of our main definitions specialized to the case of distribution-specific PAC learning. Although the target distribution is fixed, by varying the concept by which examples are labeled, we effectively generate a large set of different distributions as before. The average correlation can be defined directly for a set of functions $\mathcal{C}'$ relative to a distribution D':

$$\rho(\mathcal{C}', D') \doteq \frac{1}{|\mathcal{C}'|^2} \sum_{c_1, c_2 \in \mathcal{C}'} |\langle c_1, c_2 \rangle_{D'}| \,.$$

Definition 6.7. *For $\bar{\gamma} > 0$, a distribution D' over domain X' and a set of Boolean functions $\mathcal{C}$ over X' the **statistical dimension** of $\mathcal{C}$ over D' with average correlation $\bar{\gamma}$ is defined to be the largest integer d for which there exists a finite set of functions $\mathcal{C}_{\bar{\gamma}} \subseteq \mathcal{C}$ such that for any subset $\mathcal{C}' \subseteq \mathcal{C}_{\bar{\gamma}}$, where $|\mathcal{C}'| \geq \mathcal{C}_{\bar{\gamma}}/d$, $\rho(\mathcal{C}', D') \leq \bar{\gamma}$. We denote it by* $\mathrm{SDA}(\mathcal{C}, D', \bar{\gamma})$.

Using Theorem 3.7 and the reduction in Corollary 6.5 imply Theorem 2.8.

Acknowledgments

We thank Benny Applebaum, Avrim Blum, Uri Feige, Ravi Kannan, Michael Kearns, Robi Krauthgamer, Moni Naor, Jan Vondrak, and Avi Wigderson for insightful comments and helpful discussions.

References

N. Alon, A. Andoni, T. Kaufman, K. Matulef, R. Rubinfeld, and N. Xie. Testing k-wise and almost k-wise independence. In *STOC*, pages 496–505, 2007.

Noga Alon, Michael Krivelevich, and Benny Sudakov. Finding a large hidden clique in a random graph. In *SODA*, pages 594–598, 1998.

Brendan P. W. Ames and Stephen A. Vavasis. Nuclear norm minimization for the planted clique and biclique problems. *Math. Program.*, 129(1):69–89, 2011.

Benny Applebaum, Boaz Barak, and Avi Wigderson. Public-key cryptography from different assumptions. In *STOC*, pages 171–180, 2010.

Sanjeev Arora, Boaz Barak, Markus Brunnermeier, and Rong Ge. Computational complexity and information asymmetry in financial products (extended abstract). In *ICS*, pages 49–65, 2010.

P. Bartlett and S. Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

Alexandre Belloni, Robert M. Freund, and Santosh Vempala. An efficient rescaled perceptron algorithm for conic systems. *Math. Oper. Res.*, 34(3):621–641, 2009.

Shai Ben-David and Eli Dichterman. Learning with restricted focus of attention. *J. Comput. Syst. Sci.*, 56(3):277–298, 1998.

Quentin Berthet and Philippe Rigollet. Complexity theoretic lower bounds for sparse principal component detection. In *COLT*, pages 1046–1066, 2013.

Aditya Bhaskara, Moses Charikar, Eden Chlamtac, Uriel Feige, and Aravindan Vijayaraghavan. Detecting high log-densities: an $o(n^{1/4})$ approximation for densest k-subgraph. In *STOC*, pages 201–210, 2010.

Aditya Bhaskara, Moses Charikar, Aravindan Vijayaraghavan, Venkatesan Guruswami, and Yuan Zhou. Polynomial integrality gaps for strong sdp relaxations of densest k-subgraph. In *SODA*, pages 388–405, 2012.

A. Blum, C. Dwork, F. McSherry, and K. Nissim. Practical privacy: the SuLQ framework. In *PODS*, pages 128–138, 2005.

Avrim Blum, Merrick L. Furst, Jeffrey C. Jackson, Michael J. Kearns, Yishay Mansour, and Steven Rudich. Weakly learning dnf and characterizing statistical query learning using fourier analysis. In *STOC*, pages 253–262, 1994.

Avrim Blum, Alan M. Frieze, Ravi Kannan, and Santosh Vempala. A polynomial-time algorithm for learning noisy linear threshold functions. *Algorithmica*, 22(1/2):35–52, 1998.

Guy Bresler, David Gamarnik, and Devavrat Shah. Structure learning of antiferromagnetic ising models. In *NIPS*, pages 2852–2860, 2014.

S. Brubaker and S. Vempala. Random tensors and planted cliques. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, volume 5687, pages 406–419. 2009.

T. T. Cai, T. Liang, and A. Rakhlin. Computational and Statistical Boundaries for Submatrix Localization in a Large Noisy Matrix. *ArXiv e-prints*, February 2015.

C. Chu, S. Kim, Y. Lin, Y. Yu, G. Bradski, A. Ng, and K. Olukotun. Map-reduce for machine learning on multicore. In *NIPS*, pages 281–288, 2006.

Amin Coja-Oghlan. Graph partitioning via adaptive spectral techniques. *Combinatorics, Probability & Computing*, 19(2):227–284, 2010.

Y. Dekel, O. Gurel-Gurevich, and Y. Peres. Finding hidden cliques in linear time with high probability. In *ANALCO*, pages 67–75, 2011.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

Yash Deshpande and Andrea Montanari. Finding hidden cliques of size \sqrt{N/e} in nearly linear time. *CoRR*, abs/1304.7047, 2013.

Yash Deshpande and Andrea Montanari. Improved sum-of-squares lower bounds for hidden clique and hidden submatrix problems. In *COLT*, pages 523–562, 2015. URL `http://jmlr.org/proceedings/papers/v40/Deshpande15.html`.

Shaddin Dughmi. On the hardness of signaling. In *FOCS*, pages 354–363, 2014.

John Dunagan and Santosh Vempala. A simple polynomial-time rescaling algorithm for solving linear programs. *Math. Program.*, 114(1):101–114, 2008.

U. Feige and R. Krauthgamer. Finding and certifying a large hidden clique in a semirandom graph. *Random Struct. Algorithms*, 16(2):195–208, 2000.

U. Feige and D. Ron. Finding hidden cliques in linear time. In *AofA*, pages 189–204, 2010.

Uriel Feige. Relations between average case complexity and approximation complexity. In *IEEE Conference on Computational Complexity*, page 5, 2002.

Uriel Feige and Robert Krauthgamer. The probable value of the Lovász–Schrijver relaxations for maximum independent set. *SICOMP*, 32(2):345–370, 2003.

V. Feldman. Evolvability from learning algorithms. In *STOC*, pages 619–628, 2008.

V. Feldman. A complete characterization of statistical query learning with applications to evolvability. *Journal of Computer System Sciences*, 78(5):1444–1459, 2012.

Vitaly Feldman. Open problem: The statistical query complexity of learning sparse halfspaces. In *COLT*, pages 1283–1289, 2014.

Vitaly Feldman. A general characterization of the statistical query complexity. *CoRR*, abs/1608.02198, 2016. URL `http://arxiv.org/abs/1608.02198`.

Vitaly Feldman, Will Perkins, and Santosh Vempala. On the complexity of random satisfiability problems with planted solutions. *CoRR*, abs/1311.4821, 2013. Extended abstract in STOC 2015.

Vitaly Feldman, Cristobal Guzman, and Santosh Vempala. Statistical query algorithms for stochastic convex optimization. *CoRR*, abs/1512.09170, 2015. URL `http://arxiv.org/abs/1512.09170`.

Alan M. Frieze and Ravi Kannan. A new approach to the planted clique problem. In *FSTTCS*, pages 187–198, 2008.

C. Gao, Z. Ma, and H. H. Zhou. Sparse CCA: Adaptive Estimation and Computational Barriers. *ArXiv e-prints*, September 2014.

A. E. Gelfand and A. F. M. Smith. Sampling based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85:398–409, 1990.

Bruce E. Hajek, Yihong Wu, and Jiaming Xu. Computational lower bounds for community detection on random graphs. In *COLT*, pages 899–928, 2015. URL `http://jmlr.org/proceedings/papers/v40/Hajek15.html`.

Johan Håstad. Some optimal inapproximability results. *J. ACM*, 48:798–859, July 2001. ISSN 0004-5411.

W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

Elad Hazan and Robert Krauthgamer. How hard is it to approximate the best nash equilibrium? *SIAM J. Comput.*, 40(1):79–91, 2011.

Mark Jerrum. Large cliques elude the metropolis process. *Random Struct. Algorithms*, 3(4):347–360, 1992.

Ari Juels and Marcus Peinado. Hiding cliques for cryptographic security. *Des. Codes Cryptography*, 20(3):269–280, 2000.

Ravi Kannan. personal communication.

R. Karp. Probabilistic analysis of graph-theoretic algorithms. In *Proceedings of Computer Science and Statistics 12th Annual Symposium on the Interface*, page 173, 1979.

Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM J. Comput.*, 40(3):793–826, June 2011.

M. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45(6): 983–1006, 1998.

Subhash Khot. Ruling out ptas for graph min-bisection, densest subgraph and bipartite clique. In *FOCS*, pages 136–145, 2004.

Scott Kirkpatrick, D. Gelatt Jr., and Mario P. Vecchi. Optimization by simmulated annealing. *Science*, 220(4598):671–680, 1983.

Ludek Kucera. Expected complexity of graph partitioning problems. *Discrete Applied Mathematics*, 57(2-3):193–212, 1995.

Zongming Ma and Yihong Wu. Computational barriers in minimax submatrix detection. *CoRR*, abs/1309.5914, 2013. URL `http://arxiv.org/abs/1309.5914`.

F. McSherry. Spectral partitioning of random graphs. In *FOCS*, pages 529–537, 2001.

R. Meka, A. Potechin, and A. Wigderson. Sum-of-squares lower bounds for planted clique. In *STOC*, pages 87–96, 2015.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1092, 1953.

L. Minder and D. Vilenchik. Small clique detection and approximate nash equilibria. 5687:673–685, 2009.

K. Pearson. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine, Series 5*, 50(302):157–175, 1900.

Bart Selman, Henry Kautz, and Bram Cohen. Local search strategies for satisfiability testing. In *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 521–532, 1995.

R. Servedio. Computational sample complexity and attribute-efficient learning. *Journal of Computer and System Sciences*, 60(1):161–178, 2000.

J. Steinhardt, G. Valiant, and S. Wager. Memory, communication, and statistical queries. In *COLT*, pages 1490–1516, 2016.

Jacob Steinhardt and John C. Duchi. Minimax rates for memory-bounded sparse linear regression. In *COLT*, pages 1564–1587, 2015. URL `http://jmlr.org/proceedings/papers/v40/Steinhardt15.html`.

Balázs Szörényi. Characterizing statistical query learning: Simplified notions and proofs. In *ALT*, pages 186–200, 2009.

M Tanner and W Wong. The calculation of posterior distributions by data augmentation (with discussion). *Journal of the American Statistical Association*, 82:528–550, 1987.

Leslie G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, 1984.

V. Vapnik and A. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probab. and its Applications*, 16(2):264–280, 1971.

V. Černý. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51, January 1985. ISSN 0022-3239.

T. Wang, Q. Berthet, and R. J. Samworth. Statistical and computational trade-offs in estimation of sparse principal components. *ArXiv e-prints*, August 2014.

Ke Yang. On learning correlated boolean functions using statistical queries. In *ALT*, pages 59–76, 2001.

Ke Yang. New lower bounds for statistical query learning. *J. Comput. Syst. Sci.*, 70(4):485–509, 2005.

Andrew Yao. Probabilistic computations: Toward a unified measure of complexity. In *FOCS*, pages 222–227, 1977.

Yuchen Zhang, John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. Information-theoretic lower bounds for distributed statistical estimation with communication constraints. In *NIPS*, pages 2328–2336, 2013. URL `http://papers.nips.cc/paper/4902-information-theoretic-lower-bounds-for-distributed-statistical-estimation-with-communic`

A Average-case vs Distributional Planted Bipartite Clique

In this section we show the equivalence between the average-case planted biclique problem (where a single graph is chosen randomly) and the distributional biclique problem (where a bipartite graph is obtained from independent samples over $\{0,1\}^n$). The primary issue is that in the distributional biclique problem the biclique does not necessarily have the same size on the left side of vertices as it does on the right side. We show that this is easy to fix by producing planted bicliques of smaller size on one of the sides. We do this by replacing vertices of the graph with randomly connected ones. We now describe the reductions more formally.

Definition A.1. *[Average-case planted biclique* $\mathrm{APBC}(n, k_1, k_2)$*] Given integers* $1 \le k_1, k_2 \le n$*, consider the following distribution* $\mathcal{D}_{avg}(n, k_1, k_2)$ *on bipartite graphs on* $[n] \times [n]$ *vertices. Pick two random sets of* k_1 *and* k_2 *vertices each from left and right side, respectively, say* S_1 *and* S_2*. Plant a bipartite clique on* $S_1 \times S_2$ *and add an edge between all other pairs of vertices with probability* $1/2$*. The problem is to recover* S_1 *and* S_2 *given a random graph sampled from* $\mathcal{D}_{avg}(n, k_1, k_2)$*.*

We will refer to the distributional biclique problem with n samples as $\mathrm{DPBC}(n, k)$. Recall that in this problem we are given n random and independent samples from distribution D_S over $\{0,1\}^n$ for some unknown $S \subset [n]$ of size k (see Definition 1.1). The goal is to recover S.

Theorem A.2. *Suppose that there is an algorithm that solves* $\mathrm{APBC}(n, k', k')$ *in time* $T'(n, k')$ *and outputs the correct answer with probability* $p'(n, k')$*. Then there exists an algorithm that solves* $\mathrm{DPBC}(n, k)$ *in time* $T(n, k) = O(nkT'(n, k/2))$ *and outputs the correct answer with probability* $p(n, k) = p'(n, k/2) - n2^{-\Omega(k)}$*.*

Proof. We will think of the distribution $\mathcal{D}_{avg}(n, k', k')$ on graphs as a distribution on their respective adjacency matrices from $\{0,1\}^{n\times n}$. Let $\mathcal{A}(n, k')$ be the algorithm that solves an instance of $\mathrm{APBC}(n, k', k')$. Given k and n, and access to n samples from D_S for some set S of size k, we will design an algorithm that finds S by making $O(nk)$ calls to the algorithm $\mathcal{A}(n, k')$ that solves an instance of $\mathrm{APBC}(n, k', k')$.

Let M be the $n \times n$ binary matrix whose rows are the n samples from D_S. First apply a random permutation $\pi : [n] \to [n]$ to the columns of M to obtain M' (this will ensure that the planted set is uniformly distributed among the n coordinates, which is necessary in order to obtain instances distributed according to $\mathcal{D}_{avg}(n, k', k')$).

In what follows we will denote by $k' \times k$ a biclique with k' vertices on the left and k vertices on the right. Note that M' has a $k' \times k$ planted biclique for some k' that is distributed according to the binomial distribution $B(n, k/n)$. We denote the vertices on the left side of this biclique by L. By a multiplicative Chernoff bound, $\mathbf{Pr}[k/2 \le k' \le 2k] \ge 1 - 2e^{-k/8}$. From now on we will condition on this event occurring.

We first suppose that $k \le k' \le 2k$. We aim at obtaining instances of $\mathrm{APBC}(n, k, k)$ but recall that the left side of the planted bliclique has size $k' \ge k$. To reduce the size of the left side of the planted biclique to k we will be replacing the vertices on the left side by randomly connected ones, one-by-one in a random order. That is, start with $M'_0 = M'$. To obtain M'_{t+1}, we choose a random and uniform row of M'_t that was not previously picked and replace it with a random and uniform $\{0,1\}^n$ vector. This gives a sequence of random matrices: $M'_1, M'_2, \ldots, M'_n$. Clearly, M'_0 has a planted biclique of size $k' \times k$ and M'_n does not have a planted biclique (or, equivalently, has a $0 \times k$ biclique). A single step reduces the size of the left side of the biclique by at most 1.

Therefore for some i^*, M'_{i^*} has a $k \times k$ biclique. We denote the left side of this biclique by L^*. It is also easy to see that for every step i, conditioned on M'_i containing k'' out of vertices in L, M'_i is distributed exactly according to $\mathcal{D}_{avg}(n, k'', k)$. We now condition on the event that for all i, M'_i does not contain a $k \times k$ biclique such that its right side is different from $\pi(S)$. It is not hard to see that this event happens with probability at least $1 - n2^{-\Omega(k)}$.

To recover S, we run $\mathcal{A}(n, k)$ on all the matrices M'_i. Let $L_i \times S_i$ be the biclique that $\mathcal{A}$ outputs on M'_i. We verify that $L_i \times S_i$ is a $k \times k$ biclique in M'_i. If so we output $\pi^{-1}(S_i)$. Note that when executed on M'_{i^*}, with probability $p'(n, k)$ this procedure will return $L^* \times \pi(S)$. In this case we will return exactly S. Further, by our conditioning, if the output of $\mathcal{A}$ is a $k \times k$ biclique then its right side must be $\pi(S)$.

We can now assume that $k/2 \leq k' < k$. We aim at obtaining instances of $\mathrm{APBC}(n, k', k')$. To achieve this we reduce the size of the right side of the planted biclique to k' in the same way as we reduced the size of the left side above: we will be replacing the vertices on the right side by randomly connected ones, one-by-one in a random order. As before we start with $M'_0 = M'$. To obtain M'_{t+1}, we choose a random and uniform column of M'_t that was not previously picked and replace it with a random and uniform $\{0, 1\}^n$ vector. This gives a sequence of random matrices: $M'_1, M'_2, \ldots, M'_n$. We know that for some i^*, M'_{i^*} has a $k' \times k'$ biclique. We denote the right side of this biclique by S^*. We now condition on the event that for all i, M'_i does not contain a $k' \times k'$ biclique such that its left side is different from L. It is not hard to see that this event happens with probability at least $1 - n2^{-\Omega(k')} = 1 - n2^{-\Omega(k)}$.

Assume for now that we know k'. To recover S, we run $\mathcal{A}(n, k')$ on all the matrices M'_i. Let $L_i \times S_i$ be the biclique that $\mathcal{A}$ outputs on M'_i. We verify that $L_i \times S_i$ is a $k' \times k'$ biclique in M'_i. If so we let S' be the set of all vertices on the right side connected to each vertex in L_i (in the original graph after the permutation). If $|S'| = k$, we output $\pi^{-1}(S')$. Note that when executed on M'_{i*}, with probability $p'(n, k')$, this procedure will return $L \times S^*$. Further, by our conditioning if the output of $\mathcal{A}$ is a $k' \times k'$ biclique then its left side must be L. All vertices in $\pi(S)$ are connected to L. The probability that any other vertex on the right side of M' is connected to all vertices in L is at most $n \cdot 2^{-k}$. Hence, conditioned on this event not occurring, we will recover exactly S.

To address the fact that k' is not known, for each value of $k_1 = k - 1, k - 2, \ldots, k/2$, we run the algorithm under the assumption that $k' = k_1$ and stop once the algorithm has found a $k' \times k'$ biclique. If $k_1 > k'$ then, by our conditioning on none of M'_i containing a $k' \times k'$ biclique such that its left side is different from L there cannot exist a $k_1 \times k_1$ biclique in the graph. Therefore the algorithm will not output anything until $k_1 = k'$ at which point our analysis above applies.

To analyze the success probability and running time we can assume for simplicity that it is harder to find smaller planted bicliques than larger ones, and so for $k_1 \in [k/2, k]$, $T'(n, k_1) \leq T'(n, k/2)$ and $p'(n, k') \leq p'(n, k/2)$. Therefore the running time of our algorithm is $T(n, k) = O(nkT'(n, k/2))$, and its success probability is $p(n, k) = p'(n, k/2) - n2^{-\Omega(k)}$. □

We now prove the converse of Theorem A.2.

Theorem A.3. *Suppose that there is an algorithm that solves* $\mathrm{DPBC}(n, k)$ *that runs in time* $T(n, k)$ *and outputs the correct answer with probability* $p(n, k)$*. Then there exists an algorithm that solves* $\mathrm{APBC}(n, k', k')$ *in time* $T'(n, k') = O(nk'T(n, k'/2))$ *and outputs the planted biclique with probability* $p'(n, k') \geq p(n, k'/2) - n2^{\Omega(-k')}$.

Proof. Let $\mathcal{A}(n, k)$ denote the algorithm for solving $\mathrm{DPBC}(n, k)$, which, for any $S \subseteq [n]$ of size k, takes n samples chosen according to D_S and outputs the planted set S with probability $p(n, k)$. We will construct an algorithm for $\mathrm{APBC}(n, k', k')$ that takes as input an adjacency matrix M chosen randomly according to $\mathcal{D}_{avg}(n, k', k')$, (as in Definition A.1) and outputs a biclique $S_1 \times S_2$ of size $k' \times k'$. Note that, with probability $1 - n2^{-\Omega(k')}$, the set $S_1 \times S_2$ is the unique $k' \times k'$ biclique in M and we will condition on this event.

We first observe that an instance of $\mathrm{DPBC}(n, k)$ can be equivalently thought of as follows: first pick an arbitrary set S of k vertices on the right side of the graph; then pick ℓ according to $B(n, k/n)$; pick a random subset S' of ℓ vertices on the left side of the graph; make $S' \times S$ a biclique and connect all the other pairs of vertices randomly and independently with probability $1/2$. The probability that $\ell \in [5k/6, 6k/5]$ is at least $1 - 2^{-\Omega(k)}$ and therefore the probability that $\mathcal{A}(n, k)$ succeeds conditioned on this event is at least $p(n, k) - 2^{-\Omega(k)}$. This implies that there exists $\ell_k \in [5k/6, 6k/5]$ such that conditioned on $\ell = \ell_k$, the probability that $\mathcal{A}(n, k)$ succeeds is at least $p(n, k) - 2^{-\Omega(k)}$ (we note that ℓ_k might depend on S).

Let M be the adjacency matrix of the given instance of $\mathrm{APBC}(n, k', k')$. For each column of M (corresponding to a vertex on the right side), with probability $3/4$ we replace it with a random and uniform vector from $\{0, 1\}^n$ and let M' denote the obtained adjacency matrix. We denote by S the subset of S_2 containing vertices that were not replaced by randomly connected vertices. Let $k = |S|$. With probability at least $1 - 2^{-\Omega(k)}$, $k \in [3k'/5, 5k'/6]$ and we will condition on this event.

We aim at obtaining an instance of $\mathrm{DPBC}(n, k)$ in which exactly ℓ_k vertices on the left are connected to all vertices in the planted set S. By the argument above, we know that we can assume that $\ell_k \in [5k/6, 6k/5] \subseteq [k'/2, k']$ and $\mathcal{A}$ succeeds with probability at least $p(n, k) - 2^{-\Omega(k)}$ given an instance in which exactly ℓ_k vertices on the left are connected to all vertices in S.

M' has a $k' \times k$ biclique so to reduce the size of the left side of the planted biclique to ℓ_k we will be replacing the vertices on the left side by randomly connected ones, one-by-one in a random order as in the proof of Theorem A.2. This gives a sequence of random matrices: $M' = M'_0, M'_1, M'_2, \ldots, M'_n$. For every i, Let S'_i denote the subset of vertices in S_1 that were not replaced by a randomly connected vertex in M'_i. It is also easy to see that for every i, conditioned on $|S'_i| = k''$, M'_i is distributed exactly as n samples from D_S in which k'' samples were chosen to be connected to all vertices in S.

To recover S_1 and S_2, we run $\mathcal{A}(n, k)$ on all the matrices M'_i (where, we assume for now that k is known). Let $L_i \times R_i$ be the biclique that $\mathcal{A}$ outputs on M'_i. Let S^*_1 be the set of all (left side) vertices connected in the original input graph to all vertices in R_i and S^*_2 be the set of all (right side) vertices in the input graph connected to all vertices in S^*_1. If $|S^*_1| = |S^*_2| = k'$ then we output the biclique $S^*_1 \times S^*_2$. Otherwise we go to the next step (if none of the steps produces a biclique the algorithm fails). We first note that, unless the algorithm fails, it outputs a $k' \times k'$ biclique in the input graph which, by our conditioning, can only be the true planted biclique. Further, there exists i^* such that $|S'_{i^*}| = \ell_k$. M'_{i^*} is distributed as n samples from D_S in which ℓ_k samples were chosen to be connected to all vertices in S. Therefore by our conditioning, with probability at least $p(n, k) - 2^{-\Omega(k)}$, $\mathcal{A}(n, k)$ will output S. The vertices in S_1 are connected to all vertices in S and with probability at least $1 - n2^{-k}$ no other vertex in the original graph is. The vertices in S_2 are connected to all vertices in S_1 and, by our conditioning no other vertex is. Therefore, with probability at least $p(n, k) - n2^{-\Omega(k)}$ the algorithm will produce the true $k' \times k'$ biclique.

To address the fact that k is not known, for each value of $k_1 = \lceil 5k'/6 \rceil, \ldots, \lfloor 2k'/3 \rfloor$, we run the algorithm under the assumption that $k = k_1$ and stop once the algorithm has found a $k' \times k'$ biclique. The algorithm can only output the true planted biclique and therefore this will not reduce the success probability.

As before, to analyze the success probability and running time we assume for simplicity that it is harder to find smaller planted sets than larger ones, and so for $k_1 \in [\lfloor 2k'/3 \rfloor, \lceil 5k'/6 \rceil]$, $T(n, k_1) \leq T(n, k'/2)$ and $p(n, k) \leq p(n, k'/2)$. Therefore the running time of our algorithm is $T'(n, k') = O(nk'T(n, k'/2))$, and its success probability is $p'(n, k') \geq p(n, k'/2) - n2^{-\Omega(k')}$. $\square$

ECCC ISSN 1433-8092

http://eccc.hpi-web.de