Statistical Algorithms and a Lower Bound for Detecting Planted Cliques

고가을 · VITALY FELDMAN · ELENA GRIGORESCU · LEV REYZIN

IBM Almaden Research Center

E-mail: vitalyfeldman@ibm.com

[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, for example, 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 $O(n1/2-\delta)$ 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.

[KEYWORDS] *Statistical algorithm, Detecting Clique*

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 NP = RP). Here we present a different approach. We show that algorithms that 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 [Dempster et al. 1977], local search, MCMC optimization [Tanner and Wong 1987; Gelfand and Smith 1990], simulated annealing [Kirkpatrick et al. 1983; Cern ˇ y 1985], first- and second-order methods for linear/convex optimization ´ [Dunagan and Vempala 2008; Belloni et al. 2009], k-means, Principal Component Analysis, Independent Component Analysis, Na¨ıve 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 $\min_{z \in K} E_{x \sim D}[f(x, z)]$ 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 $\nabla_z E_x[f(x, z)] = E_x[\nabla_z f(x, z)]$ 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 $E_x[\nabla_z f(x, z)]$ would also suffice. 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] Probably Approximately Correct (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.

2. DEFINITION AND OVERVIEW

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, that is, for $x \in X$ such that D(x) = 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 Rn). 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, $E_{x\sim D}[f(x)] - E_{x\sim D}[f(x)] = E_{x\sim D}\left[\frac{D(x)}{D(x)} f(x)\right] - E_{x\sim D}[f(x)] = \langle D - D, f \rangle_D$. 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 D1, D2 and a reference distribution D, their pairwise correlation is defined as $\chi_D(D1, D2) = \langle D1 - 1, D2 - 1 \rangle_D$. When D1 = D2, the quantity $\langle D1 - 1, D1 - 1 \rangle_D$ is known as the $\chi^2(D1, D)$ 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 D relative to a distribution D. We denote it by ρ(D , D) and define it as follows: $\rho(D, D) = \frac{1}{|D|^2} \sum_{D1,D2 \in D} \chi_D(D1, D2) = \frac{1}{|D|^2} \sum_{D1,D2 \in D} \langle D1 - 1, D2 - 1 \rangle_D$. 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.

Recently, Zhang et al [11] have proposed an efficient image matching algorithm based on SURF. This approach has enormous advantages of less computations and short time-consuming. Moreover, the Random Sample Consensus algorithm is used to eliminate the false match and wrong match points. In this work, their proposed approach is developed in terms of the automated heterogeneous platoon and in dynamic environment.

The proof of Theorem 3.2 relies on the following lemma that translates a lower bound on SDA(Df , D, γ¯) into a lower bound on the number of queries that A needs to use. Its proof is based on ideas of Szor¨ enyi [2009] and Feldman [2012]. ´ LEMMA 3.3. Let X be a domain and Z be a search problem over a set of solutions F and a class of distributions D over X. Let A be a (deterministic) SQ algorithm for Z that uses at most q queries to VSTAT(1/(3 ¯γ)). For a distribution D, consider the execution of A on D in which to each query h of A, the oracle returns exactly ED[h], and let f denote the output. For a set of distributions Df ⊆ D\Z f and γ >¯ 0, let d = SDA(Df , D, γ¯). Let D+ be the set of all distributions in Df for which A successfully solves Z for all valid responses of VSTAT(1/(3 ¯γ)). Then q ≥ d · |D+|/|Df |. PROOF. Let h1, h2,..., hq be the queries asked by A when executed on D with the exact responses of the oracle. Let m = |D+|, and we denote the distributions in D+ by {D1, D2,..., Dm}. For every k ≤ q, let Ak be the set of all distributions Di such that E D [hk(x)] − E Di [hk(x)] > τi,k . = max 1 t , pi,k(1 − pi,k) t , where we use t to denote 1/(3 ¯γ) and pi,k to denote EDi [hk(x)]. To prove the desired bound, we first prove the following two claims:

3. PLANTED BICLIQUE AND DENSEST SUBGRAPH

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 ⊆ [n], let DS be the distribution over {0, 1}n with a planted set S. Let Sk denote the set of all (n k) subsets of [n] of size k and m = (n k). We index the elements of Sk in some arbitrary order as S1,..., Sm. For i ∈ [m], we use Di to denote DSi . We will also assume, whenever necessary, that k and n are larger than some fixed constant.

By applying this equation inductively we obtain, |Tj| ≤ 2j · |T0| j! · n2δj < 2j · (m− 1) j! · n2δj , where the last inequality holds, since |T0| ≤ m− 2 whenever n ≥ 2k + 1. For n larger than some fixed constant, k≥λ≥ j |Tλ| < k≥λ≥ j 2λ · (m− 1) λ! · n2δλ ≤ m− 1 n2δj k≥λ≥ j 2λ λ! · n2δ(λ− j) ≤ 3(m− 1) n2δj . By definition of λ0, |A| ≤ j≥λ0 |Tj| < 3(m−1)/n2δλ0 . In particular, if |A| ≥ 3(m−1)/n2 δ , then n2δλ0 < n2 δ or λ0 < . Now we can conclude that Si∈A |Dˆ S, DˆI | ≤ k\j=λ0 2j |Tj ∩ A|α ≤ 2λ0 |Tλ0 ∩ A| + k j=λ0+1 2j |Tj| α ≤ 2λ0 |Tλ0 ∩ A| + 2 · 2λ0+1|Tλ0+1| α < 2λ0+2|A|α ≤ 2 +1|A|α.
To derive the second-to-last inequality, we need to note that for every j ≥ 0, 2j |Tj| > 2(2j+1|Tj+1|) whenever n2δ ≥ 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 δ ≥ 1/ log n and k ≤ n1/2−δ , let Z the distributional planted kbiclique problem. Then, for any ≤ k, SDA(Z, 2 +1k2/n2, 1/(n k)) ≥ n2 δ/3. In addition, let D be the uniform distribution and denote the set of all planted distributions by D. Then, SDA(D, D, 2 +1k2/n2) ≥ n2 δ/3. PROOF. For every solution S ∈F, ZS = {DS}, and let DS = D\{DS}. Note that |DS| = (n k) − 1 and, therefore, |DS| ≥ (1 − 1/(n k))|D|. This means that we can use 1/(n k) as the solution set bound.

4. CONCLUSION

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 APBC(n, k1, k2)]. Given integers 1 ≤ k1, k2 ≤ n, consider the following distribution Gavg(n, k1, k2) on bipartite graphs on [n] × [n] vertices. Pick two random sets of k1 and k2 vertices each from left and right side, respectively, say, S1 and S2. Plant a bipartite clique on S1×S2 and add an edge between all other pairs of vertices with probability 1/2. The problem is to recover S1 and S2 given a random graph sampled from Gavg(n, k1, k2). We will refer to the distributional planted biclique problem with n samples as DPBC(n, k). Recall that in this problem, we are given n random and independent samples from distribution DS

over {0, 1}n for some unknown S ⊂ [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 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 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−

(k) . PROOF. We will think of the distribution Gavg(n, k , k) on graphs as a distribution on their respective adjacency matrices from {0, 1}n×n. Let A(n, k) be the algorithm that solves an instance of APBC(n, k , k). Given k and n, and access to n samples from DS for some set S of size k, we will design an algorithm that finds S by making O(nk) calls to the algorithm A(n, k) that solves an instance of APBC(n, k , k). Let M be the n × n binary matrix whose rows are the n samples from DS. First apply a random permutation π : [n] → [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 Gavg(n, k , k)).

REFERENCES:

[1] Mohammad Alfraheed, “An Approach for Features Matching Between Bilateral Images of Streo Vision System Applied for Automated Heterogeneous Platoon”, Open Journal, No. 2, May. 2018.

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

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

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

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

[6] P. Bartlett and S. Mendelson. 2002. Rademacher and gaussian Complexities: Risk Bounds and Structural Results. J. Mach. Learn. Res. 3 (2002), 463–482.

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

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

[9] Quentin Berthet and Philippe Rigollet. 2013. Complexity theoretic lower bounds for sparse principal component detection. In COLT. 1046–1066.

[10] Aditya Bhaskara, Moses Charikar, Eden Chlamtac, Uriel Feige, and Aravindan Vijayaraghavan. 2010. Detecting high log-densities: An o(n1/4) approximation for densest k-subgraph. In STOC. 201–210.

[11] A. Blum, C. Dwork, F. McSherry, and K. Nissim. 2005. Practical privacy: The SuLQ framework. In PODS. 128–138.

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

[13] Guy Bresler, David Gamarnik, and Devavrat Shah. 2014. Structure learning of antiferromagnetic Ising models. In NIPS. 2852–2860.

[14] S. Brubaker and S. Vempala. 2009. Random tensors and planted cliques. In Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques. Vol. 5687. 406–419.

[15] T. T. Cai, T. Liang, and A. Rakhlin. 2015. Computational and statistical boundaries for submatrix localization in a large noisy matrix. ArXiv E-prints (Feb. 2015).

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

[17] Amin Coja-Oghlan. 2010. Graph partitioning via adaptive spectral techniques. Combin. ProbabComput. 19, 2 (2010), 227–284.

[18] 2018050001, 구민준, S. Arya, D. M. Mount, “Approximate Nearest Neighbor Queries in Fixed Dimensions”, Open Journal, No. 1, May. 2018.

CONTRIBUTORS:

[1] 1, 구민준, LG Software Engineer
[2] 10, 임성수, The dean of the Kookin University

.