58th Annual IEEE Symposium on Foundations of Computer Science

A Time-Space Lower Bound for a Large Class of Learning Problems

Ran Raz
Department of Computer Science, Princeton University
Email: ran.raz.mail@gmail.com

***Abstract*—We prove a general memory-samples lower bound that applies for a large class of learning problems and shows that for every problem in that class, any learning algorithm requires either a memory of quadratic size or an exponential number of samples.**

Our result is stated in terms of the norm of the matrix that corresponds to the learning problem. Let X, A be two finite sets. A matrix $M : A \times X \to \{-1, 1\}$ corresponds to the following learning problem: An unknown element $x \in X$ was chosen uniformly at random. A learner tries to learn x from a stream of samples, $(a_1, b_1), (a_2, b_2) \ldots$, where for every i, $a_i \in A$ is chosen uniformly at random and $b_i = M(a_i, x)$.

Let $\sigma_{\max}$ be the largest singular value of M and note that always $\sigma_{\max} \le |A|^{\frac{1}{2}} \cdot |X|^{\frac{1}{2}}$. We show that if $\sigma_{\max} \le |A|^{\frac{1}{2}} \cdot |X|^{\frac{1}{2}-\varepsilon}$, then any learning algorithm for the corresponding learning problem requires either a memory of size at least $\Omega\left((\varepsilon n)^2\right)$ or at least $2^{\Omega(\varepsilon n)}$ samples, where $n = \log_2 |X|$.

As a special case, this gives a new proof for the memory-samples lower bound for parity learning [14].

I. Introduction

Several recent works studied the resources needed for learning, under memory constraints. The study was initiated by [17], [19], followed by several additional works (see in particular [14], [20], [9]). While the main motivation for studying this problem comes from learning theory, the problem is also relevant to computational complexity and cryptography [14], [20], [9].

Steinhardt, Valiant and Wager conjectured that any algorithm for learning parities of size n requires either a memory of size $\Omega(n^2)$ or an exponential number of samples. The conjecture was proven in [14]. In particular, this shows for the first time a learning problem that is infeasible under super-linear memory constraints. Building on [14], it was proved in [9] that even if the parity is known to be of sparsity ℓ, parity learning remains infeasible under memory constraints that are super-linear in n, as long as $\ell \ge \omega(\log n / \log\log n)$. Consequently, learning linear-size DNF Formulas, linear-size Decision Trees and logarithmic-size Juntas were all proved to be infeasible under super-linear memory constraints (by reductions from learning sparse parities) [9].

Our Results: In this work, we present a new technique for proving infeasibility of learning under super-linear memory constraints. The technique seems very general. In particular, our main result applies for a large class of learning problems and shows that for every problem in that class, any learning algorithm requires either a memory of quadratic size or an exponential number of samples.

Let X, A be two finite sets of size larger than 1 (where X represents the concept-class that we are trying to learn and A represents the set of possible samples).

Let $M : A \times X \to \{-1, 1\}$ be a matrix. The matrix M represents the following learning problem: An unknown element $x \in X$ was chosen uniformly at random. A learner tries to learn x from a stream of samples, $(a_1, b_1), (a_2, b_2) \ldots$, where for every i, $a_i \in A$ is chosen uniformly at random and $b_i = M(a_i, x)$.

Denote by $\sigma_{\max}(M)$ the largest singular value of M. Since the entries of M are in $\{-1, 1\}$, it is always the case that

$$\sigma_{\max}(M) \le |A|^{\frac{1}{2}} \cdot |X|^{\frac{1}{2}}.$$

Our main result, restated as Theorem 1, shows that if

$$\sigma_{\max}(M) \le |A|^{\frac{1}{2}} \cdot |X|^{\frac{1}{2}-\varepsilon}$$

(where $\varepsilon > 0$ is not necessarily a constant), then any learning algorithm for the learning problem represented by M requires either a memory of size at least $\Omega\left((\varepsilon n)^2\right)$ or at least $2^{\Omega(\varepsilon n)}$ samples, where $n = \log_2 |X|$.

For example, in the problem of parity learning, $|A| = |X| = 2^n$, and $M = H$ is Hadamard matrix. Hadamard matrix satisfies

$$\sigma_{\max}(H) = |A|^{\frac{1}{2}} \le |A|^{\frac{1}{2}} \cdot |X|^{\frac{1}{2}-\varepsilon},$$

where $\varepsilon = \frac{1}{2}$. Thus, as a special case of our main result, we obtain a new proof for the memory-samples lower bound for parity learning [14] (with a completely different set of techniques and slightly better constants).

Our result holds even if the learner has an exponentially small success probability (of $2^{-\Omega(\varepsilon n)}$).

We note that since it is always the case that $\sigma_{\max}(M) \ge \max\left\{|A|^{\frac{1}{2}}, |X|^{\frac{1}{2}}\right\}$, we have that εn is at most $\frac{1}{2} \cdot \min\{\log|A|, \log|X|\}$, and hence the bound on the memory in our main result, $\Omega\left((\varepsilon n)^2\right)$, is at most $\Omega\left(\min\left\{(\log|A|)^2, (\log|X|)^2\right\}\right)$.

0272-5428/17 $31.00 © 2017 IEEE
DOI 10.1109/FOCS.2017.73

As in [14], [9], we model the learning algorithm by a *branching program*. A branching program is the strongest and most general model to use in this context. Roughly speaking, the model allows a learner with infinite computational power, and bounds only the memory size of the learner and the number of samples used.

Our results are stated for a learner that learns x *exactly*. Nevertheless, it is not hard to see that in many interesting cases, the results hold also for weak learning, where the learner only needs to output a concept that *approximates* the correct one. This is true because if a learner is able to output a function that *approximates* a column of the matrix, the learner may as well output the column that approximates that function the best. The learner can compute the column that approximates that function the best because our lower bounds hold for a learner with an infinite computational power. If that column corresponds to the correct x (even with exponentially small success probability of $2^{-\Omega(\varepsilon n)}$), our results can be applied. In particular, this is the case for parity learning and for cases where the columns of the matrix are almost orthogonal.

Motivation and Discussion: Many previous works studied the resources needed for learning, under certain information, communication or memory constraints (see in particular [17], [19], [14], [20], [9] and the many references given there). A main message of some of these works is that for some learning problems, access to a relatively large memory is crucial. In other words, in some cases, learning is infeasible, due to memory constraints.

From the point of view of human learning, such results may help to explain the importance of memory in cognitive processes. From the point of view of machine learning, these results imply that a large class of learning algorithms cannot learn certain concept classes. In particular, this applies to any bounded-memory learning algorithm that considers the samples one by one. In addition, these works are related to computational complexity and have applications in cryptography.

Our work is the first to give a general criteria that applies for a large class of learning problems and shows infeasibility of learning under super-linear memory constraints. Our result is stated in terms of the operator norm (equivalently, the largest singular value) of the corresponding matrix. It resembles known general lower bounds for distributional communication complexity (see [11], Chapter 4, for a survey), unbounded-error communication complexity [5], [18], [16] and SQ-learning [7], [1], [10], [4].

There are known techniques to bound the operator norm of a matrix. For example, one can show that any matrix with low discrepancy has small operator norm. Thus, we obtain a memory-samples lower bound for the corresponding learning problem for every matrix with low discrepancy. This, in turn, relates our result once again to communication complexity, as the discrepancy method is one of the best-known general techniques for proving communication-complexity lower bounds.

Related Work: Independently of our work, Moshkovitz and Moshkovitz proved that if the matrix M has a (sufficiently strong) mixing property then any learning algorithm for the corresponding learning problem requires either a memory of size at least $1.25 \cdot n$ or at least $2^{\Omega(n)}$ samples [12].

For matrices M such that

$$\sigma_{\max}(M) \le |A|^{\frac{1}{2}} \cdot |X|^{\frac{1}{2}-\varepsilon},$$

their result implies that any learning algorithm for the corresponding learning problem requires either a memory of size at least $(5\varepsilon - 1.25) \cdot n$ or at least $2^{\Omega(n)}$ samples. (Since $\varepsilon \le 0.5$, we have that $(5\varepsilon - 1.25)$ is at most 1.25).

Subsequent Work: Subsequent to the first appearance of our work [15], Moshkovitz and Moshkovitz [13] improved their initial result [12], and obtained a theorem that is very similar to ours. (Their result is stated in terms of a combinatorial mixing property, rather than matrix norm. The two notions are closely related (see in particular Corollary 5.1 and Note 5.1 in [2])).

In [6], Garg, Raz and Tal build on the current work and show that if k, ℓ, r are such that any submatrix of M of at least $2^{-k} \cdot |A|$ rows and at least $2^{-\ell} \cdot |X|$ columns, has a bias of at most 2^{-r}, then any learning algorithm for the learning problem corresponding to M requires either a memory of size at least $\Omega(k \cdot \ell)$, or at least $2^{\Omega(r)}$ samples. In particular, this shows that for a large class of learning problems, any learning algorithm requires either a memory of size at least $\Omega((\log|X|) \cdot (\log|A|))$ or an exponential number of samples, achieving a tight $\Omega((\log|X|) \cdot (\log|A|))$ lower bound on the size of the memory, rather than a bound of $\Omega\left(\min\left\{(\log|X|)^2, (\log|A|)^2\right\}\right)$ obtained in the current work. Moreover, the result implies all previous memory-samples lower bounds, as well as a number of new applications [6].

Independently of [6], Beame, Oveis Gharan and Yang also build on our current work and give a combinatorial property of a matrix M, that holds for a large class of matrices and implies that any learning algorithm for the corresponding learning problem requires either a memory of size $\Omega((\log|X|) \cdot (\log|A|))$ or an exponential number of samples (when $|A| \le |X|$) [3]. Their property is based on a measure of how matrices amplify the 2-norms of probability distributions that is more refined than the 2-norms of these matrices. They give several applications of their results.

The proofs of the main results in both of these works [6], [3] use, and build on, the techniques presented here.

II. Preliminaries

Denote by $\mathcal{U}_X : X \to \mathbb{R}^+$ the uniform distribution over X.

For a random variable Z and an event E, we denote by $\mathbb{P}_Z$ the distribution of the random variables Z, and we denote by $\mathbb{P}_{Z|E}$ the distribution of the random variable Z conditioned on the event E.

Viewing a Learning Problem as a Matrix: Let X, A be two finite sets of size larger than 1. Let $n = \log_2 |X|$.

Let $M : A \times X \to \{-1, 1\}$ be a matrix. The matrix M corresponds to the following learning problem: There is an unknown element $x \in X$ that was chosen uniformly at random. A learner tries to learn x from samples (a, b), where $a \in A$ is chosen uniformly at random and $b = M(a, x)$. That is, the learning algorithm is given a stream of samples, $(a_1, b_1), (a_2, b_2) \ldots$, where each a_t is uniformly distributed and for every t, $b_t = M(a_t, x)$.

Norms: For a function $f : X \to \mathbb{R}$, denote by $\|f\|_2$ the ℓ_2 norm of f, with respect to the uniform distribution over X, that is:

$$\|f\|_2 = \left(\mathop{\mathbf{E}}_{x \in_R X} \left[f(x)^2 \right] \right)^{1/2}.$$

For a function $f : A \to \mathbb{R}$, denote by $\|f\|_2$ the ℓ_2 norm of f, with respect to the uniform distribution over A, that is:

$$\|f\|_2 = \left(\mathop{\mathbf{E}}_{a \in_R A} \left[f(a)^2 \right] \right)^{1/2}.$$

The induced matrix norm on $M : A \times X \to \{-1, 1\}$ is defined by

$$\|M\|_2 = \sup_{f \neq 0} \frac{\|Mf\|_2}{\|f\|_2}.$$

We note that

$$\|M\|_2 = \sqrt{\tfrac{|X|}{|A|}} \cdot \sigma_{\max}(M),$$

where $\sigma_{\max}(M)$ denotes the largest singular value of M, and the factor $\sqrt{\frac{|X|}{|A|}}$ comes because in our definition of the ℓ_2 norm of functions $f : X \to \mathbb{R}$ and $f : A \to \mathbb{R}$ we use expectation rather than (the more common) sum.

While $\|M\|_2$ and $\sigma_{\max}(M)$ are equal, up to a normalization factor, it will be more convenient for us to work with $\|M\|_2$.

Branching Program for a Learning Problem: In the following definition, we model the learner for the learning problem that corresponds to the matrix M, by a *branching program.*

Definition II.1. Branching Program for a Learning Problem: *A branching program of length m and width d, for learning, is a directed (multi) graph with vertices arranged in $m + 1$ layers containing at most d vertices each. In the first layer, that we think of as layer 0, there is only one vertex, called the start vertex. A vertex of outdegree 0 is called a leaf. All vertices in the last layer are leaves (but there may be additional leaves). Every non-leaf vertex in the program has $2|A|$ outgoing edges, labeled by elements $(a, b) \in A \times \{-1, 1\}$, with exactly one edge labeled by each such (a, b), and all these edges going into vertices in the next layer. Each leaf v in the program is labeled by an element $\tilde{x}(v) \in X$, that we think of as the output of the program on that leaf.*

Computation-Path: *The samples $(a_1, b_1), \ldots, (a_m, b_m) \in A \times \{-1, 1\}$ that are given as input, define a computation-path in the branching program, by starting from the start vertex and following at step t the edge labeled by (a_t, b_t), until reaching a leaf. The program outputs the label $\tilde{x}(v)$ of the leaf v reached by the computation-path.*

Success Probability: *The success probability of the program is the probability that $\tilde{x} = x$, where $\tilde{x}$ is the element that the program outputs, and the probability is over $x, a_1, \ldots, a_m$ (where x is uniformly distributed over X and $a_1, \ldots, a_m$ are uniformly distributed over A, and for every t, $b_t = M(a_t, x)$).*

III. Overview of the Proof

Let B be a branching program for the learning problem that corresponds to the matrix M. Assume for a contradiction that B is of a relatively small length and a relatively small width.

We define the *truncated-path*, $\mathcal{T}$, to be the same as the computation-path of B, except that it sometimes stops before reaching a leaf. Roughly speaking, $\mathcal{T}$ stops before reaching a leaf if certain "bad" events, that make the analysis difficult, occur. Nevertheless, we show that the probability that $\mathcal{T}$ stops before reaching a leaf is negligible, so we can think of $\mathcal{T}$ as almost identical to the computation-path.

For a vertex v of B, we denote by E_v the event that $\mathcal{T}$ reaches the vertex v. For simplicity, we denote by $\Pr(v) = \Pr(E_v)$ the probability for E_v (where the probability is over $x, a_1, \ldots, a_m$), and we denote by $\mathbb{P}_{x|v} = \mathbb{P}_{x|E_v}$ the distribution of the random variable x conditioned on the event E_v. Similarly, for an edge e of the branching program B, let E_e be the event that $\mathcal{T}$ traverses the edge e. Denote, $\Pr(e) = \Pr(E_e)$, and $\mathbb{P}_{x|e} = \mathbb{P}_{x|E_e}$.

A vertex v of B is called *significant* if $\left\|\mathbb{P}_{x|v}\right\|_2$ is non-negligible. Roughly speaking, this means that conditioning on the event that $\mathcal{T}$ reaches the vertex v, a non-negligible amount of information is known about x. In order to guess x with a non-negligible success probability, $\mathcal{T}$ must reach a significant vertex. Lemma IV.1 shows that the probability that $\mathcal{T}$ reaches any significant vertex is negligible, and thus the main result follows.

To prove Lemma IV.1, we show that for every fixed significant vertex s, the probability that $\mathcal{T}$ reaches s is extremely small (smaller than one over the number of vertices in B). Hence, we can use a union bound to prove the lemma.

The proof that the probability that $\mathcal{T}$ reaches s is extremely small is the main part of our proof. To that end, we introduce the following functions to measure the progress made by the branching program towards reaching s.

Let L_i be the set of vertices v in layer-i of B, such that $\Pr(v) > 0$. Let Γ_i be the set of edges e from layer-$(i-1)$ of B to layer-i of B, such that $\Pr(e) > 0$. Let

$$\mathcal{Z}_i = \sum_{v \in L_i} \Pr(v) \cdot \langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s} \rangle^n,$$

$$\mathcal{Z}'_i = \sum_{e \in \Gamma_i} \Pr(e) \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s} \rangle^n,$$

where $\langle \cdot, \cdot \rangle$ denotes inner product. We think of $\mathcal{Z}_i, \mathcal{Z}'_i$ as measuring the progress made by the branching program, towards reaching a state with distribution similar to $\mathbb{P}_{x|s}$.

We show that each $\mathcal{Z}_i$ may only be negligibly larger than $\mathcal{Z}_{i-1}$. Hence, $\mathcal{Z}_i$ is negligible for every i. On the other hand, if s is in layer-i then $\mathcal{Z}_i$ is at least $\Pr(s) \cdot \langle \mathbb{P}_{x|s}, \mathbb{P}_{x|s} \rangle^n$. Thus, $\Pr(s) \cdot \langle \mathbb{P}_{x|s}, \mathbb{P}_{x|s} \rangle^n$ must be negligible. Since s is significant, $\langle \mathbb{P}_{x|s}, \mathbb{P}_{x|s} \rangle$ is non-negligible and hence $\Pr(s)$ must be negligible.

The proof that $\mathcal{Z}_i$ may only be negligibly larger than $\mathcal{Z}_{i-1}$ is done in two steps: Claim IV.12 shows by a simple convexity argument that $\mathcal{Z}_i \leq \mathcal{Z}'_i$. The hard part, that is done in Claim IV.10 and Claim IV.11, is to prove that $\mathcal{Z}'_i$ may only be negligibly larger than $\mathcal{Z}_{i-1}$.

For this proof, we define for every vertex v, the set of edges $\Gamma_{out}(v)$ that are going out of v, such that $\Pr(e) > 0$. Claim IV.10 shows that for every vertex v,

$$\sum_{e \in \Gamma_{out}(v)} \Pr(e) \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s} \rangle^n$$

may only be negligibly higher than

$$\Pr(v) \cdot \langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s} \rangle^n.$$

For the proof of Claim IV.10, which is the hardest proof in the paper, we consider the function $\mathbb{P}_{x|v} \cdot \mathbb{P}_{x|s}$. We first show how to bound $\left\|\mathbb{P}_{x|v} \cdot \mathbb{P}_{x|s}\right\|_2$. We then consider two cases: If $\left\|\mathbb{P}_{x|v} \cdot \mathbb{P}_{x|s}\right\|_1$ is negligible, then $\langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s} \rangle^n$ is negligible and doesn't contribute much, and we show that for every $e \in \Gamma_{out}(v)$, $\langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s} \rangle^n$ is also negligible and doesn't contribute much. If $\left\|\mathbb{P}_{x|v} \cdot \mathbb{P}_{x|s}\right\|_1$ is non-negligible, we use the bound on $\left\|\mathbb{P}_{x|v} \cdot \mathbb{P}_{x|s}\right\|_2$ and the bound on $\|M\|_2$ to show that for almost all edges $e \in \Gamma_{out}(v)$, we have that $\langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s} \rangle^n$ is very close to $\langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s} \rangle^n$. Only an exponentially small fraction of edges are "bad" and give a significantly larger $\langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s} \rangle^n$.

The reason that in the definitions of $\mathcal{Z}_i$ and $\mathcal{Z}'_i$ we raised $\langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s} \rangle$ and $\langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s} \rangle$ to the power of n is that this is the largest power for which the contribution of the "bad" edges is still small (as their fraction is exponentially small).

This outline oversimplifies many details. Let us briefly mention two of them. First, it is not so easy to bound $\left\|\mathbb{P}_{x|v} \cdot \mathbb{P}_{x|s}\right\|_2$. We do that by bounding $\left\|\mathbb{P}_{x|s}\right\|_2$ and $\left\|\mathbb{P}_{x|v}\right\|_\infty$. In order to bound $\left\|\mathbb{P}_{x|s}\right\|_2$, we force $\mathcal{T}$ to stop whenever it reaches a significant vertex (and thus we are able to bound $\left\|\mathbb{P}_{x|v}\right\|_2$ for every vertex reached by $\mathcal{T}$). In order to bound $\left\|\mathbb{P}_{x|v}\right\|_\infty$, we force $\mathcal{T}$ to stop whenever $\mathbb{P}_{x|v}(x)$ is large, which allows us to consider only the "bounded" part of $\mathbb{P}_{x|v}$. (This is related to the technique of *flattening* a distribution that was used in [8]). Second, some edges are so "bad" that their contribution to $\mathcal{Z}'_i$ is huge so they cannot be ignored. We force $\mathcal{T}$ to stop before traversing any such edge. (This is related to an idea that was used in [9] of analyzing separately paths that traverse "bad" edges). We show that the total probability that $\mathcal{T}$ stops before reaching a leaf is negligible.

IV. Main Result

Theorem 1. *Let X, A be two finite sets. Let $n = \log_2 |X|$. Let $M : A \times X \to \{-1, 1\}$ be a matrix, such that $\|M\|_2 \leq 2^{\gamma n}$ (where $\gamma < 1$ is not necessarily a constant, but $(1-\gamma) \cdot n \to \infty$ (when $n \to \infty$)).*

For any constant $c' < \frac{1}{3}$, there exists a (sufficiently small) constant $\epsilon' > 0$, such that the following holds: Let $c = c' \cdot (1-\gamma)^2$, and let $\epsilon = \epsilon' \cdot (1-\gamma)$. Let B be a branching program of length at most $2^{\epsilon n}$ and width at most 2^{cn^2} for the learning problem that corresponds to the matrix M. Then, the success probability of B is at most $O(2^{-\epsilon n})$.

Remark:: We note that it is always the case that $\|M\|_2 \leq 2^n$, and that the condition $\|M\|_2 \leq 2^{\gamma n}$ is equivalent to $\sigma_{\max}(M) \leq |A|^{\frac{1}{2}} \cdot |X|^{\frac{1}{2}-\varepsilon}$, where $\varepsilon = 1 - \gamma$.

Proof:

Let $0 < \delta' < \frac{1}{3}$ and $0 < \beta' < 2$ be constants (to be optimized later on), such that, $\beta' + 6\delta' < 2$. Let

$\epsilon' > 0$ be a sufficiently small constant (chosen to be sufficiently small after δ', β' were chosen). In particular, we will assume that $10\epsilon' < 2 - \beta' - 6\delta'$. Let

$$\delta = \delta' \cdot (1 - \gamma),$$

$$\beta = \beta' \cdot (1 - \gamma),$$

$$\epsilon = \epsilon' \cdot (1 - \gamma).$$

Thus,

$$10\epsilon < 2 \cdot (1 - \gamma) - \beta - 6\delta.$$

Let B be a branching program of length $m = 2^{\epsilon n}$ and width $d = 2^{cn^2}$ for the learning problem that corresponds to the matrix M. We will show that the success probability of B is at most $O(2^{-\epsilon n})$. We will assume that n is sufficiently large (chosen to be sufficiently large after $\delta', \beta', \epsilon'$ were chosen). This is justified because we only need to prove that the success probability of B is at most $O(2^{-\epsilon n})$ (so n can be assumed to be sufficiently large because of the big O).

A. The Truncated-Path and Additional Definitions and Notation

We will define the **truncated-path**, $\mathcal{T}$, to be the same as the computation-path of B, except that it sometimes stops before reaching a leaf. Formally, we define $\mathcal{T}$, together with several other definitions and notations, by induction on the layers of the branching program B.

Assume that we already defined the truncated-path $\mathcal{T}$, up to layer-i of B. For a vertex v in layer-i of B, let E_v be the event that $\mathcal{T}$ reaches the vertex v. For simplicity, we denote by $\Pr(v) = \Pr(E_v)$ the probability for E_v (where the probability is over $x, a_1, \ldots, a_m$), and we denote by $\mathbb{P}_{x|v} = \mathbb{P}_{x|E_v}$ the distribution of the random variable x conditioned on the event E_v.

There will be three cases in which the truncated-path $\mathcal{T}$ stops on a non-leaf v:

1) If v is a, so called, significant vertex, where the ℓ_2 norm of $\mathbb{P}_{x|v}$ is non-negligible. (Intuitively, this means that conditioned on the event that $\mathcal{T}$ reaches v, a non-negligible amount of information is known about x).
2) If $\mathbb{P}_{x|v}(x)$ is non-negligible. (Intuitively, this means that conditioned on the event that $\mathcal{T}$ reaches v, the correct element x could have been guessed with a non-negligible probability).
3) If $(M \cdot \mathbb{P}_{x|v})(a_{i+1})$ is non-negligible. (Intuitively, this means that $\mathcal{T}$ is about to traverse a "bad" edge, which is traversed with a non-negligibly higher or lower probability than other edges).

Next, we describe these three cases more formally.

Significant Vertices: We say that a vertex v in layer-i of B is **significant** if

$$\left\| \mathbb{P}_{x|v} \right\|_2 > 2^{\delta n} \cdot 2^{-n}.$$

(Recall that $|X| = 2^n$ and hence $\left\| \mathbb{P}_{x|v} \right\|_2$ is always at least 2^{-n}).

Significant Values: Even if v is not significant, $\mathbb{P}_{x|v}$ may have relatively large values. For a vertex v in layer-i of B, denote by $\text{Sig}(v)$ the set of all $x' \in X$, such that,

$$\mathbb{P}_{x|v}(x') > 2^{2(\delta+\epsilon)\cdot n} \cdot 2^{-n}.$$

Bad Edges: For a vertex v in layer-i of B, denote by $\text{Bad}(v)$ the set of all $\alpha \in A$, such that,

$$\left| (M \cdot \mathbb{P}_{x|v})(\alpha) \right| \geq 2^{(\delta+\gamma+\epsilon)\cdot n} \cdot 2^{-n}.$$

(Intuitively, $\left| (M \cdot \mathbb{P}_{x|v})(\alpha) \right|$ is the advantage in predicting $M(\alpha, x)$ on the known row α and the unknown column x).

The Truncated-Path $\mathcal{T}$: We define $\mathcal{T}$ by induction on the layers of the branching program B. Assume that we already defined $\mathcal{T}$ until it reaches a vertex v in layer-i of B. The path $\mathcal{T}$ stops on v if (at least) one of the following occurs:

1) v is significant.
2) $x \in \text{Sig}(v)$.
3) $a_{i+1} \in \text{Bad}(v)$.
4) v is a leaf.

Otherwise, $\mathcal{T}$ proceeds by following the edge labeled by (a_{i+1}, b_{i+1}) (same as the computational-path).

B. Proof of Theorem 1

Since $\mathcal{T}$ follows the computation-path of B, except that it sometimes stops before reaching a leaf, the success probability of B is bounded (from above) by the probability that $\mathcal{T}$ stops before reaching a leaf, plus the probability that $\mathcal{T}$ reaches a leaf v and $\tilde{x}(v) = x$.

The main lemma needed for the proof of Theorem 1 is Lemma IV.1 that shows that the probability that $\mathcal{T}$ reaches a significant vertex is at most $O(2^{-\epsilon n})$.

Lemma IV.1. *The probability that $\mathcal{T}$ reaches a significant vertex is at most $O(2^{-\epsilon n})$.*

Lemma IV.1 is proved in Section IV-C. We will now show how the proof of Theorem 1 follows from that lemma.

Lemma IV.1 shows that the probability that $\mathcal{T}$ stops on a non-leaf vertex, because of the first reason (i.e., that the vertex is significant), is small. The next two lemmas imply that the probabilities that $\mathcal{T}$ stops on a non-leaf vertex, because of the second and third reasons, are also small.

Claim IV.2. *If v is a non-significant vertex of B then*

$$\Pr_x[x \in \mathit{Sig}(v) \mid E_v] \leq 2^{-2\epsilon n}.$$

Proof: Since v is not significant,

$$\mathop{\mathbf{E}}_{x' \sim \mathbb{P}_{x|v}} \left[\mathbb{P}_{x|v}(x')\right] = \sum_{x' \in X} \left[\mathbb{P}_{x|v}(x')^2\right] =$$

$$2^n \cdot \mathop{\mathbf{E}}_{x' \in_R X} \left[\mathbb{P}_{x|v}(x')^2\right] \leq 2^{2\delta n} \cdot 2^{-n}.$$

Hence, by Markov's inequality,

$$\Pr_{x' \sim \mathbb{P}_{x|v}} \left[\mathbb{P}_{x|v}(x') > 2^{2\epsilon n} \cdot 2^{2\delta n} \cdot 2^{-n}\right] \leq 2^{-2\epsilon n}.$$

Since conditioned on E_v, the distribution of x is $\mathbb{P}_{x|v}$, we obtain

$$\Pr_x[x \in \mathrm{Sig}(v) \mid E_v] =$$

$$\Pr_x \left[\left(\mathbb{P}_{x|v}(x) > 2^{2\epsilon n} \cdot 2^{2\delta n} \cdot 2^{-n}\right) \,\middle|\, E_v \right] \leq 2^{-2\epsilon n}.$$

■

Claim IV.3. *If v is a non-significant vertex of B then*

$$\Pr_{a_{i+1}} [a_{i+1} \in \mathit{Bad}(v)] \leq 2^{-2\epsilon n}.$$

Proof: Since v is not significant,

$$\mathop{\mathbf{E}}_{\alpha \in_R A} \left[|(M \cdot \mathbb{P}_{x|v})(\alpha)|^2\right] = \left\|M \cdot \mathbb{P}_{x|v}\right\|_2^2$$

$$\leq \|M\|_2^2 \cdot \left\|\mathbb{P}_{x|v}\right\|_2^2 \leq 2^{2\gamma n} \cdot 2^{2\delta n} \cdot 2^{-2n}.$$

Hence, by Markov's inequality,

$$\Pr_{\alpha \in_R A} [\alpha \in \mathrm{Bad}(v)] =$$

$$\Pr_{\alpha \in_R A} \left[\left|(M \cdot \mathbb{P}_{x|v})(\alpha)\right| \geq 2^{(\delta+\gamma+\epsilon)\cdot n} \cdot 2^{-n}\right] =$$

$$\Pr_{\alpha \in_R A} \left[\left|(M \cdot \mathbb{P}_{x|v})(\alpha)\right|^2 \geq 2^{2\epsilon n} \cdot 2^{2\gamma n} \cdot 2^{2\delta n} \cdot 2^{-2n}\right]$$

$$\leq 2^{-2\epsilon n}.$$

The claim follows since a_{i+1} is uniformly distributed over A. ■

We can now use Lemma IV.1, Claim IV.2 and Claim IV.3 to prove that the probability that $\mathcal{T}$ stops before reaching a leaf is at most $O(2^{-\epsilon n})$. Lemma IV.1 shows that the probability that $\mathcal{T}$ reaches a significant vertex and hence stops because of the first reason, is at most $O(2^{-\epsilon n})$. Assuming that $\mathcal{T}$ doesn't reach any significant vertex (in which case it would have stopped because of the first reason), Claim IV.2 shows that in each step, the probability that $\mathcal{T}$ stops because of the second reason, is at most $2^{-2\epsilon n}$. Taking a union bound over the $m = 2^{\epsilon n}$ steps, the total probability that $\mathcal{T}$ stops because of the second reason, is at most $2^{-\epsilon n}$. In the same way, assuming that $\mathcal{T}$ doesn't reach any significant vertex (in which case it would have stopped because of the first reason), Claim IV.3 shows that in each step, the probability that $\mathcal{T}$ stops because of the third reason, is at most $2^{-2\epsilon n}$. Again, taking a union bound over the $2^{\epsilon n}$ steps, the total probability that $\mathcal{T}$ stops because of the third reason, is at most $2^{-\epsilon n}$. Thus, the total probability that $\mathcal{T}$ stops (for any reason) before reaching a leaf is at most $O(2^{-\epsilon n})$.

Recall that if $\mathcal{T}$ doesn't stop before reaching a leaf, it just follows the computation-path of B. Recall also that by Lemma IV.1, the probability that $\mathcal{T}$ reaches a significant leaf is at most $O(2^{-\epsilon n})$. Thus, to bound (from above) the success probability of B by $O(2^{-\epsilon n})$, it remains to bound the probability that $\mathcal{T}$ reaches a non-significant leaf v and $\tilde{x}(v) = x$. Claim IV.4 shows that for any non-significant leaf v, conditioned on the event that $\mathcal{T}$ reaches v, the probability for $\tilde{x}(v) = x$ is at most $2^{-\epsilon n}$, which completes the proof of Theorem 1.

Claim IV.4. *If v is a non-significant leaf of B then*

$$\Pr[\tilde{x}(v) = x \mid E_v] \leq 2^{-\epsilon n}.$$

Proof: Since v is not significant,

$$\mathop{\mathbf{E}}_{x' \in_R X} \left[\mathbb{P}_{x|v}(x')^2\right] \leq 2^{2\delta n} \cdot 2^{-2n}.$$

Hence, for every $x' \in X$,

$$\Pr[x = x' \mid E_v] = \mathbb{P}_{x|v}(x') \leq 2^{\delta n} \cdot 2^{-n/2}$$

$$= 2^{-n \cdot \left(\frac{1}{2} - \delta\right)} < 2^{-\epsilon n}$$

(since $\epsilon < \frac{1}{2} - \delta$). In particular,

$$\Pr[\tilde{x}(v) = x \mid E_v] < 2^{-\epsilon n}.$$

■

This completes the proof of Theorem 1. ■

C. Proof of Lemma IV.1

Proof: We need to prove that the probability that $\mathcal{T}$ reaches any significant vertex is at most $O(2^{-\epsilon n})$. Let s be a significant vertex of B. We will bound from above the probability that $\mathcal{T}$ reaches s, and then use a union bound over all significant vertices of B. Interestingly, the upper bound on the width of B is used only in the union bound.

The Distributions $\mathbb{P}_{x|v}$ and $\mathbb{P}_{x|e}$: Recall that for a vertex v of B, we denote by E_v the event that $\mathcal{T}$ reaches the vertex v. For simplicity, we denote by $\Pr(v) = \Pr(E_v)$ the probability for E_v (where the probability is over $x, a_1, \ldots, a_m$), and we denote by $\mathbb{P}_{x|v} = \mathbb{P}_{x|E_v}$ the distribution of the random variable x conditioned on the event E_v.

Similarly, for an edge e of the branching program B, let E_e be the event that $\mathcal{T}$ traverses the edge e. Denote, $\Pr(e) = \Pr(E_e)$ (where the probability is over $x, a_1, \ldots, a_m$), and $\mathbb{P}_{x|e} = \mathbb{P}_{x|E_e}$.

Claim IV.5. *For any edge $e = (v, u)$ of B, labeled by (a, b), such that $\Pr(e) > 0$, for any $x' \in X$,*

$$\mathbb{P}_{x|e}(x') =$$

$$\begin{array}{cccccc} 0 & \textit{if} & x' \in \textit{Sig}(v) & \textit{or} & M(a,x') \neq b \\ \mathbb{P}_{x|v}(x') \cdot c_e^{-1} & \textit{if} & x' \notin \textit{Sig}(v) & \textit{and} & M(a,x') = b \end{array}$$

where c_e is a normalization factor that satisfies,

$$c_e > \tfrac{1}{2} - 2 \cdot 2^{-2\epsilon n}.$$

Proof: Let $e = (v, u)$ be an edge of B, labeled by (a, b), and such that $\Pr(e) > 0$. Since $\Pr(e) > 0$, the vertex v is not significant (as otherwise $\mathcal{T}$ always stops on v and hence $\Pr(e) = 0$). Also, since $\Pr(e) > 0$, we know that $a \notin \text{Bad}(v)$ (as otherwise $\mathcal{T}$ never traverses e and hence $\Pr(e) = 0$).

If $\mathcal{T}$ reaches v, it traverses the edge e if and only if: $x \notin \text{Sig}(v)$ (as otherwise $\mathcal{T}$ stops on v) and $M(a, x) = b$ and $a_{i+1} = a$. Therefore, for any $x' \in X$,

$$\mathbb{P}_{x|e}(x') =$$

$$\begin{array}{cccccc} 0 & \text{if} & x' \in \text{Sig}(v) & \text{or} & M(a,x') \neq b \\ \mathbb{P}_{x|v}(x') \cdot c_e^{-1} & \text{if} & x' \notin \text{Sig}(v) & \text{and} & M(a,x') = b \end{array}$$

where c_e is a normalization factor, given by

$$c_e = \sum_{\{x' \,:\, x' \notin \text{Sig}(v) \,\wedge\, M(a,x')=b\}} \mathbb{P}_{x|v}(x')$$

$$= \Pr_x[(x \notin \text{Sig}(v)) \wedge (M(a,x) = b) \mid E_v].$$

Since v is not significant, by Claim IV.2,

$$\Pr_x[x \in \text{Sig}(v) \mid E_v] \leq 2^{-2\epsilon n}.$$

Since $a \notin \text{Bad}(v)$,

$$\left| \Pr_x[M(a,x) = 1 \mid E_v] - \Pr_x[M(a,x) = -1 \mid E_v] \right|$$

$$= \left| (M \cdot \mathbb{P}_{x|v})(a) \right| \leq 2^{(\delta+\gamma+\epsilon)\cdot n} \cdot 2^{-n}.$$

and hence

$$\Pr_x[M(a,x) \neq b \mid E_v] \leq \tfrac{1}{2} + 2^{(\delta+\gamma+\epsilon)\cdot n} \cdot 2^{-n}.$$

Hence, by the union bound,

$$c_e = \Pr_x[(x \notin \text{Sig}(v)) \wedge (M(a,x) = b) \mid E_v]$$

$$\geq \tfrac{1}{2} - 2^{(\delta+\gamma+\epsilon)\cdot n} \cdot 2^{-n} - 2^{-2\epsilon n} > \tfrac{1}{2} - 2 \cdot 2^{-2\epsilon n}$$

(where the last inequality follows since $3\epsilon < 1 - \delta - \gamma$). ■

Bounding the Norm of $\mathbb{P}_{x|s}$: We will show that $\left\| \mathbb{P}_{x|s} \right\|_2$ cannot be too large. Towards this, we will first prove that for every edge e of B that is traversed by $\mathcal{T}$ with probability larger than zero, $\left\| \mathbb{P}_{x|e} \right\|_2$ cannot be too large.

Claim IV.6. *For any edge e of B, such that $\Pr(e) > 0$,*

$$\left\| \mathbb{P}_{x|e} \right\|_2 \leq 4 \cdot 2^{\delta n} \cdot 2^{-n}.$$

Proof: Let $e = (v, u)$ be an edge of B, labeled by (a, b), and such that $\Pr(e) > 0$. Since $\Pr(e) > 0$, the vertex v is not significant (as otherwise $\mathcal{T}$ always stops on v and hence $\Pr(e) = 0$). Thus,

$$\left\| \mathbb{P}_{x|v} \right\|_2 \leq 2^{\delta n} \cdot 2^{-n}.$$

By Claim IV.5, for any $x' \in X$,

$$\mathbb{P}_{x|e}(x') =$$

$$\begin{array}{cccccc} 0 & \text{if} & x' \in \text{Sig}(v) & \text{or} & M(a,x') \neq b \\ \mathbb{P}_{x|v}(x') \cdot c_e^{-1} & \text{if} & x' \notin \text{Sig}(v) & \text{and} & M(a,x') = b \end{array}$$

where c_e satisfies,

$$c_e > \tfrac{1}{2} - 2 \cdot 2^{-2\epsilon n} > \tfrac{1}{4}$$

(where the last inequality holds because we assume that n is sufficiently large).

Thus,

$$\left\| \mathbb{P}_{x|e} \right\|_2 \leq c_e^{-1} \cdot \left\| \mathbb{P}_{x|v} \right\|_2 \leq 4 \cdot 2^{\delta n} \cdot 2^{-n}$$

■

Claim IV.7.

$$\left\| \mathbb{P}_{x|s} \right\|_2 \leq 4 \cdot 2^{\delta n} \cdot 2^{-n}.$$

Proof: Let $\Gamma_{in}(s)$ be the set of all edges e of B, that are going into s, such that $\Pr(e) > 0$. Note that

$$\sum_{e \in \Gamma_{in}(s)} \Pr(e) = \Pr(s).$$

By the law of total probability, for every $x' \in X$,

$$\mathbb{P}_{x|s}(x') = \sum_{e \in \Gamma_{in}(s)} \tfrac{\Pr(e)}{\Pr(s)} \cdot \mathbb{P}_{x|e}(x'),$$

and hence by Jensen's inequality,

$$\mathbb{P}_{x|s}(x')^2 \leq \sum_{e \in \Gamma_{in}(s)} \tfrac{\Pr(e)}{\Pr(s)} \cdot \mathbb{P}_{x|e}(x')^2.$$

Summing over $x' \in X$, we obtain,

$$\left\| \mathbb{P}_{x|s} \right\|_2^2 \leq \sum_{e \in \Gamma_{in}(s)} \tfrac{\Pr(e)}{\Pr(s)} \cdot \left\| \mathbb{P}_{x|e} \right\|_2^2.$$

By Claim IV.6, for any $e \in \Gamma_{in}(s)$,

$$\left\| \mathbb{P}_{x|e} \right\|_2^2 \leq \left(4 \cdot 2^{\delta n} \cdot 2^{-n} \right)^2.$$

Hence,

$$\left\| \mathbb{P}_{x|s} \right\|_2^2 \leq \left(4 \cdot 2^{\delta n} \cdot 2^{-n} \right)^2.$$

■

Similarity to a Target Distribution: For two functions $f, g : X \to \mathbb{R}^+$, define

$$\langle f, g\rangle = \mathop{\mathbf{E}}_{z \in_R X}[f(z) \cdot g(z)].$$

We think of $\langle f, g\rangle$ as a measure for the similarity between a function f and a target function g. Typically f, g will be distributions.

Claim IV.8.

$$\langle \mathbb{P}_{x|s}, \mathbb{P}_{x|s}\rangle > 2^{2\delta n} \cdot 2^{-2n}.$$

Proof: Since s is significant,

$$\langle \mathbb{P}_{x|s}, \mathbb{P}_{x|s}\rangle = \left\|\mathbb{P}_{x|s}\right\|_2^2 > 2^{2\delta n} \cdot 2^{-2n}.$$

■

Claim IV.9.

$$\langle \mathcal{U}_X, \mathbb{P}_{x|s}\rangle = 2^{-2n},$$

where $\mathcal{U}_X$ is the uniform distribution over X.

Proof: Since $\mathbb{P}_{x|s}$ is a distribution,

$$\langle \mathcal{U}_X, \mathbb{P}_{x|s}\rangle = 2^{-2n} \cdot \sum_{z \in X} \mathbb{P}_{x|s}(z) = 2^{-2n}.$$

■

Measuring the Progress: For $i \in \{0, \ldots, m\}$, let L_i be the set of vertices v in layer-i of B, such that $\Pr(v) > 0$. For $i \in \{1, \ldots, m\}$, let Γ_i be the set of edges e from layer-$(i-1)$ of B to layer-i of B, such that $\Pr(e) > 0$. Recall that β was fixed at the beginning of the proof of Theorem 1. For $i \in \{0, \ldots, m\}$, let

$$\mathcal{Z}_i = \sum_{v \in L_i} \Pr(v) \cdot \langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s}\rangle^{\beta n}.$$

For $i \in \{1, \ldots, m\}$, let

$$\mathcal{Z}'_i = \sum_{e \in \Gamma_i} \Pr(e) \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle^{\beta n}.$$

We think of $\mathcal{Z}_i, \mathcal{Z}'_i$ as measuring the progress made by the branching program, towards reaching a state with distribution similar to $\mathbb{P}_{x|s}$.

For a vertex v of B, let $\Gamma_{out}(v)$ be the set of all edges e of B, that are going out of v, such that $\Pr(e) > 0$. Note that

$$\sum_{e \in \Gamma_{out}(v)} \Pr(e) \le \Pr(v).$$

(We don't always have an equality here, since sometimes $\mathcal{T}$ stops on v).

The next four claims show that the progress made by the branching program is slow.

Claim IV.10. *For every vertex v of B, such that $\Pr(v) > 0$,*

$$\sum_{e \in \Gamma_{out}(v)} \tfrac{\Pr(e)}{\Pr(v)} \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle^{\beta n} \le$$

$$\langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s}\rangle^{\beta n} \cdot \left(1 + 2^{-1.9 \cdot \epsilon n}\right) + \left(2^{-2n+2}\right)^{\beta n}.$$

Proof: If v is significant or v is a leaf, then $\mathcal{T}$ always stops on v and hence $\Gamma_{out}(v)$ is empty and thus the left hand side is equal to zero and the right hand side is positive, so the claim follows trivially. Thus, we can assume that v is not significant and is not a leaf.

Define $P : X \to \mathbb{R}^+$ as follows. For any $x' \in X$,

$$P(x') = \begin{cases} 0 & \text{if} \quad x' \in \mathrm{Sig}(v) \\ \mathbb{P}_{x|v}(x') & \text{if} \quad x' \notin \mathrm{Sig}(v) \end{cases}$$

Note that by the definition of $\mathrm{Sig}(v)$, for any $x' \in X$,

$$P(x') \le 2^{2(\delta+\epsilon)\cdot n} \cdot 2^{-n}. \tag{1}$$

Define $f : X \to \mathbb{R}^+$ as follows. For any $x' \in X$,

$$f(x') = P(x') \cdot \mathbb{P}_{x|s}(x').$$

By Claim IV.7 and Equation (1),

$$\|f\|_2 \le 2^{2(\delta+\epsilon)\cdot n} \cdot 2^{-n} \cdot \left\|\mathbb{P}_{x|s}\right\|_2 \le$$

$$2^{2(\delta+\epsilon)\cdot n} \cdot 2^{-n} \cdot 4 \cdot 2^{\delta n} \cdot 2^{-n} = 2^{(3\delta+2\epsilon)\cdot n+2} \cdot 2^{-2n}. \tag{2}$$

By Claim IV.5, for any edge $e \in \Gamma_{out}(v)$, labeled by (a, b), for any $x' \in X$,

$$\mathbb{P}_{x|e}(x') = \begin{cases} 0 & \text{if} \quad M(a, x') \ne b \\ P(x') \cdot c_e^{-1} & \text{if} \quad M(a, x') = b \end{cases}$$

where c_e satisfies,

$$c_e > \tfrac{1}{2} - 2 \cdot 2^{-2\epsilon n}.$$

Therefore, for any edge $e \in \Gamma_{out}(v)$, labeled by (a, b), for any $x' \in X$,

$$\mathbb{P}_{x|e}(x') \cdot \mathbb{P}_{x|s}(x') =$$

$$\begin{cases} 0 & \text{if} \quad M(a, x') \ne b \\ f(x') \cdot c_e^{-1} & \text{if} \quad M(a, x') = b \end{cases}$$

and hence, denoting

$$F = \sum_{x' \in X} f(x'),$$

we have

$$\langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle = \mathop{\mathbf{E}}_{x' \in_R X}[\mathbb{P}_{x|e}(x') \cdot \mathbb{P}_{x|s}(x')] =$$

$$c_e^{-1} \cdot 2^{-n} \cdot \sum_{\{x' : M(a,x') = b\}} f(x') = c_e^{-1} \cdot 2^{-n} \cdot \tfrac{F + b \cdot (M \cdot f)(a)}{2}$$

$$\le (2 \cdot c_e)^{-1} \cdot 2^{-n} \cdot (F + |(M \cdot f)(a)|)$$

$$< 2^{-n} \cdot (F + |(M \cdot f)(a)|) \cdot \left(1 + 2^{-2\epsilon n + 3}\right) \tag{3}$$

(where the last inequality holds by the bound that we have on c_e, because we assume that n is sufficiently large).

We will now consider two cases:

Case I: $F \le 2^{-n}$: In this case, we bound $|(M \cdot f)(a)| \le F$ (since f is non-negative and the entries of M are in $\{-1, 1\}$) and $(1 + 2^{-2\epsilon n+3}) < 2$ (since we assume that n is sufficiently large) and obtain for any edge $e \in \Gamma_{out}(v)$,

$$\langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s} \rangle < 4 \cdot 2^{-2n}.$$

Since $\sum_{e \in \Gamma_{out}(v)} \frac{\Pr(e)}{\Pr(v)} \le 1$, Claim IV.10 follows, as the left hand side of the claim is smaller than the second term on the right hand side.

Case II: $F \ge 2^{-n}$: For every $a \in A$, define

$$t(a) = \left(\frac{(M \cdot f)(a)}{F} \right)^2.$$

By Equation (3),

$$\langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s} \rangle^{\beta n} < \left(2^{-n} \cdot F\right)^{\beta n} \cdot \left(1 + \sqrt{t(a)}\right)^{\beta n} \cdot \left(1 + 2^{-2\epsilon n+3}\right)^{\beta n}. \tag{4}$$

Note that by the definitions of P and f,

$$2^{-n} \cdot F = \mathop{\mathbf{E}}_{x' \in_R X}[f(x')] = \langle P, \mathbb{P}_{x|s} \rangle \le \langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s} \rangle.$$

Note also that for every $a \in A$, there is at most one edge $e_{(a,1)} \in \Gamma_{out}(v)$, labeled by $(a, 1)$, and at most one edge $e_{(a,-1)} \in \Gamma_{out}(v)$, labeled by $(a, -1)$, and we have

$$\frac{\Pr(e_{(a,1)})}{\Pr(v)} + \frac{\Pr(e_{(a,-1)})}{\Pr(v)} \le \frac{1}{|A|},$$

since $\frac{1}{|A|}$ is the probability that the next sample read by the program is a. Thus, summing over all $e \in \Gamma_{out}(v)$, by Equation (4),

$$\sum_{e \in \Gamma_{out}(v)} \frac{\Pr(e)}{\Pr(v)} \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s} \rangle^{\beta n} < \langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s} \rangle^{\beta n} \cdot \mathop{\mathbf{E}}_{a \in_R A} \left[\left(1 + \sqrt{t(a)}\right)^{\beta n} \right] \cdot \left(1 + 2^{-2\epsilon n+3}\right)^{\beta n}. \tag{5}$$

It remains to bound

$$\mathop{\mathbf{E}}_{a \in_R A} \left[\left(1 + \sqrt{t(a)}\right)^{\beta n} \right], \tag{6}$$

where for every $a \in A$, $0 \le t(a) \le 1$, under the constraint

$$\mathop{\mathbf{E}}_{a \in_R A}[t(a)] \le 2^{-(2-2\gamma-6\delta-4\epsilon)\cdot n+4}, \tag{7}$$

which follows since, by Equation (2) and the assumption $F \ge 2^{-n}$, we have

$$\mathop{\mathbf{E}}_{a \in_R A}[t(a)] = \mathop{\mathbf{E}}_{a \in_R A} \left[\left(\frac{(M \cdot f)(a)}{F} \right)^2 \right] = \frac{\|M \cdot f\|_2^2}{F^2} \le \frac{\|M\|_2^2 \cdot \|f\|_2^2}{F^2} \le \left(\frac{2^{\gamma n} \cdot 2^{(3\delta+2\epsilon)\cdot n+2} \cdot 2^{-2n}}{2^{-n}} \right)^2 = 2^{-(2-2\gamma-6\delta-4\epsilon)\cdot n+4}.$$

We will bound the expectation in Equation (6), by splitting the expectation into two sums

$$\mathop{\mathbf{E}}_{a \in_R A} \left[\left(1 + \sqrt{t(a)}\right)^{\beta n} \right] = \frac{1}{|A|} \cdot \sum_{a \,:\, t(a) \le \frac{1}{(\beta n-2)^2}} \left(1 + \sqrt{t(a)}\right)^{\beta n} + \frac{1}{|A|} \cdot \sum_{a \,:\, t(a) > \frac{1}{(\beta n-2)^2}} \left(1 + \sqrt{t(a)}\right)^{\beta n}. \tag{8}$$

To bound the first sum in Equation (8), we note that in the range $0 \le t \le \frac{1}{(\beta n-2)^2}$, the function $g(t) = \left(1 + \sqrt{t}\right)^{\beta n}$ is concave (since its second derivative is negative). Hence, by Equation (7) and by the monotonicity of g,

$$\begin{aligned} \frac{1}{|A|} \cdot \sum_{a \,:\, t(a) \le \frac{1}{(\beta n-2)^2}} \left(1 + \sqrt{t(a)}\right)^{\beta n} &\le \left(1 + \sqrt{2^{-(2-2\gamma-6\delta-4\epsilon)\cdot n+4}}\right)^{\beta n} \\ &= \left(1 + 2^{-(1-\gamma-3\delta-2\epsilon)\cdot n+2}\right)^{\beta n} \\ &< 1 + 2^{-(1-\gamma-3\delta-3\epsilon)\cdot n} \\ &< 1 + 2^{-2\epsilon n} \end{aligned} \tag{9}$$

(where the last two inequalities hold because we assume that n is sufficiently large and $5\epsilon < 1 - \gamma - 3\delta$).

To bound the second sum in Equation (8), we note that by Equation (7) and Markov's inequality,

$$\begin{aligned} \mathop{\Pr}_{a \in_R A} \left[t(a) > \tfrac{1}{(\beta n-2)^2} \right] &\le 2^{-(2-2\gamma-6\delta-4\epsilon)\cdot n+4} \cdot (\beta n - 2)^2 \\ &< 2^{-(2-2\gamma-6\delta-5\epsilon)\cdot n} \end{aligned}$$

(where the last inequality holds because we assume that n is sufficiently large), and since for every $a \in A$, we have $t(a) \le 1$,

$$\begin{aligned} \frac{1}{|A|} \cdot \sum_{a \,:\, t(a) > \frac{1}{(\beta n-2)^2}} \left(1 + \sqrt{t(a)}\right)^{\beta n} &< 2^{-(2-2\gamma-6\delta-5\epsilon)\cdot n} \cdot 2^{\beta n} \\ &= 2^{-(2-2\gamma-\beta-6\delta-5\epsilon)\cdot n} \\ &< 2^{-5\epsilon n} \end{aligned} \tag{10}$$

(where the last inequality holds because $10\epsilon < 2-2\gamma-\beta-6\delta$).

Substituting Equation (9) and Equation (10) into Equation (8) and substituting this into Equation (5), we obtain

$$\sum_{e\in\Gamma_{out}(v)} \tfrac{\Pr(e)}{\Pr(v)} \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle^{\beta n}$$

$$< \langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s}\rangle^{\beta n} \cdot \left(1+2^{-2\epsilon n+3}\right)^{\beta n+1}$$

$$< \langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s}\rangle^{\beta n} \cdot \left(1+2^{-1.9\cdot\epsilon n}\right)$$

(where the last inequality holds because we assume that n is sufficiently large).

This completes the proof of Claim IV.10. ■

Claim IV.11. *For every $i \in \{1,\ldots,m\}$,*

$$\mathcal{Z}'_i \leq \mathcal{Z}_{i-1} \cdot \left(1+2^{-1.9\cdot\epsilon n}\right) + \left(2^{-2n+2}\right)^{\beta n}.$$

Proof: By Claim IV.10,

$$\mathcal{Z}'_i = \sum_{e\in\Gamma_i} \Pr(e) \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle^{\beta n}$$

$$= \sum_{v\in L_{i-1}} \Pr(v) \cdot \sum_{e\in\Gamma_{out}(v)} \tfrac{\Pr(e)}{\Pr(v)} \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle^{\beta n}$$

$$\leq \sum_{v\in L_{i-1}} \Pr(v) \cdot$$

$$\left(\langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s}\rangle^{\beta n} \cdot \left(1+2^{-1.9\cdot\epsilon n}\right) + \left(2^{-2n+2}\right)^{\beta n}\right)$$

$$= \mathcal{Z}_{i-1} \cdot \left(1+2^{-1.9\cdot\epsilon n}\right) + \sum_{v\in L_{i-1}} \Pr(v) \cdot \left(2^{-2n+2}\right)^{\beta n}$$

$$\leq \mathcal{Z}_{i-1} \cdot \left(1+2^{-1.9\cdot\epsilon n}\right) + \left(2^{-2n+2}\right)^{\beta n}$$

■

Claim IV.12. *For every $i \in \{1,\ldots,m\}$,*

$$\mathcal{Z}_i \leq \mathcal{Z}'_i.$$

Proof: For any $v \in L_i$, let $\Gamma_{in}(v)$ be the set of all edges $e \in \Gamma_i$, that are going into v. Note that

$$\sum_{e\in\Gamma_{in}(v)} \Pr(e) = \Pr(v).$$

By the law of total probability, for every $v \in L_i$ and every $x' \in X$,

$$\mathbb{P}_{x|v}(x') = \sum_{e\in\Gamma_{in}(v)} \tfrac{\Pr(e)}{\Pr(v)} \cdot \mathbb{P}_{x|e}(x'),$$

and hence

$$\langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s}\rangle = \sum_{e\in\Gamma_{in}(v)} \tfrac{\Pr(e)}{\Pr(v)} \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle.$$

Thus, by Jensen's inequality,

$$\langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s}\rangle^{\beta n} \leq \sum_{e\in\Gamma_{in}(v)} \tfrac{\Pr(e)}{\Pr(v)} \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle^{\beta n}.$$

Summing over all $v \in L_i$, we get

$$\mathcal{Z}_i = \sum_{v\in L_i} \Pr(v) \cdot \langle \mathbb{P}_{x|v}, \mathbb{P}_{x|s}\rangle^{\beta n}$$

$$\leq \sum_{v\in L_i} \Pr(v) \cdot \sum_{e\in\Gamma_{in}(v)} \tfrac{\Pr(e)}{\Pr(v)} \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle^{\beta n}$$

$$= \sum_{e\in\Gamma_i} \Pr(e) \cdot \langle \mathbb{P}_{x|e}, \mathbb{P}_{x|s}\rangle^{\beta n} = \mathcal{Z}'_i.$$

■

Claim IV.13. *For every $i \in \{1,\ldots,m\}$,*

$$\mathcal{Z}_i \leq 2^{2(\beta+\epsilon)n} \cdot 2^{-2\beta n^2}.$$

Proof: By Claim IV.9, $\mathcal{Z}_0 = \left(2^{-2n}\right)^{\beta n}$. By Claim IV.11 and Claim IV.12, for every $i \in \{1,\ldots,m\}$,

$$\mathcal{Z}_i \leq \mathcal{Z}_{i-1} \cdot \left(1+2^{-1.9\cdot\epsilon n}\right) + \left(2^{-2n+2}\right)^{\beta n}.$$

Hence, for every $i \in \{1,\ldots,m\}$,

$$\mathcal{Z}_i \leq \left(2^{-2n+2}\right)^{\beta n} \cdot m \cdot \left(1+2^{-1.9\cdot\epsilon n}\right)^{m}.$$

Since $m = 2^{\epsilon n}$,

$$\mathcal{Z}_i \leq 2^{-2\beta n^2} \cdot 2^{2\beta n} \cdot 2^{\epsilon n} \cdot 2 \leq 2^{-2\beta n^2} \cdot 2^{2(\beta+\epsilon)n}.$$

■

Proof of Lemma IV.1: We can now complete the proof of Lemma IV.1. Assume that s is in layer-i of B. By Claim IV.8,

$$\mathcal{Z}_i \geq \Pr(s) \cdot \langle \mathbb{P}_{x|s}, \mathbb{P}_{x|s}\rangle^{\beta n} > \Pr(s) \cdot \left(2^{2\delta n} \cdot 2^{-2n}\right)^{\beta n}$$

$$= \Pr(s) \cdot 2^{2\delta\beta n^2} \cdot 2^{-2\beta n^2}.$$

On the other hand, by Claim IV.13,

$$\mathcal{Z}_i \leq 2^{2(\beta+\epsilon)n} \cdot 2^{-2\beta n^2}.$$

Thus,

$$\Pr(s) \leq 2^{2(\beta+\epsilon)n} \cdot 2^{-2\delta\beta n^2}.$$

We will fix β' to be any constant smaller than 1 and δ' to be any constant smaller than $\frac{1}{6}$ (note that the requirement $\beta' + 6\delta' < 2$ is satisfied and recall that $\delta = \delta' \cdot (1-\gamma)$ and $\beta = \beta' \cdot (1-\gamma)$), to obtain

$$\Pr(s) \leq 2^{-\tilde{c}\cdot(1-\gamma)^2\cdot n^2},$$

for any constant $\tilde{c} < \frac{1}{3}$ (where we assumed that n is sufficiently large).

Taking a union bound over at most $2^{\epsilon n} \cdot 2^{cn^2}$ significant vertices of B, we conclude that the probability that $\mathcal{T}$ reaches any sifnificant vertex is at most $2^{-\Omega\left((1-\gamma)^2\cdot n^2\right)}$ (as $c < c' \cdot (1-\gamma)^2$, where c' is a constant smaller than $\frac{1}{3}$). Since we assume that n is sufficiently large, $2^{-\Omega\left((1-\gamma)^2\cdot n^2\right)}$ is certainly at most $2^{-\epsilon n}$. ■

Acknowledgment

Research supported by the Simons Collaboration on Algorithms and Geometry and by the National Science Foundation grants No. CCF-1412958 and CCF-1714779.

References

[1] Avrim Blum, Merrick L. Furst, Jeffrey C. Jackson, Michael J. Kearns, Yishay Mansour, Steven Rudich: *Weakly learning DNF and characterizing statistical query learning using Fourier analysis.* STOC 1994: 253-262

[2] Yonatan Bilu, Nathan Linial: *Lifts, Discrepancy and Nearly Optimal Spectral Gap.* Combinatorica 26(5): 495-519 (2006)

[3] Paul Beame, Shayan Oveis Gharan, Xin Yang: *Time-Space Tradeoffs for Learning from Small Test Spaces: Learning Low Degree Polynomial Functions.* Electronic Colloquium on Computational Complexity (ECCC) 24: 120 (2017)

[4] Vitaly Feldman: *A complete characterization of statistical query learning with applications to evolvability.* J. Comput. Syst. Sci. 78(5): 1444-1459 (2012)

[5] Jürgen Forster: *A linear lower bound on the unbounded error probabilistic communication complexity.* J. Comput. Syst. Sci. 65(4): 612-625 (2002)

[6] Sumegha Garg, Ran Raz, Avishay Tal: *Extractor-Based Time-Space Lower Bounds for Learning.* Electronic Colloquium on Computational Complexity (ECCC) 24: 121 (2017)

[7] Michael J. Kearns: *Efficient Noise-Tolerant Learning from Statistical Queries.* J. ACM 45(6): 983-1006 (1998)

[8] Gillat Kol, Ran Raz: *Interactive channel capacity.* STOC 2013: 715-724

[9] Gillat Kol, Ran Raz, Avishay Tal: *Time-Space Hardness of Learning Sparse Parities.* STOC 2017: 1067-1080

[10] Adam R. Klivans, Alexander A. Sherstov: *Unconditional lower bounds for learning intersections of halfspaces.* Machine Learning 69(2-3): 97-114 (2007)

[11] Troy Lee, Adi Shraibman: *Lower Bounds in Communication Complexity.* Foundations and Trends in Theoretical Computer Science 3(4): 263-398 (2009)

[12] Dana Moshkovitz, Michal Moshkovitz: *Mixing Implies Lower Bounds for Space Bounded Learning.* Proceedings of the 2017 Conference on Learning Theory, PMLR 65:1516-1566, 2017. Also in: Electronic Colloquium on Computational Complexity (ECCC) 24: 17 (2017)

[13] Dana Moshkovitz, Michal Moshkovitz: *Mixing Implies Strong Lower Bounds for Space Bounded Learning.* Electronic Colloquium on Computational Complexity (ECCC) 24: 116 (2017)

[14] Ran Raz: *Fast Learning Requires Good Memory: A Time-Space Lower Bound for Parity Learning.* FOCS 2016: 266-275

[15] Ran Raz: *A Time-Space Lower Bound for a Large Class of Learning Problems.* Electronic Colloquium on Computational Complexity (ECCC) 24: 20 (2017)

[16] Alexander A. Razborov, Alexander A. Sherstov: *The Sign-Rank of AC0.* SIAM J. Comput. 39(5): 1833-1855 (2010)

[17] Ohad Shamir: *Fundamental Limits of Online and Distributed Algorithms for Statistical Learning and Estimation.* NIPS 2014: 163-171

[18] Alexander A. Sherstov: *The unbounded-error communication complexity of symmetric functions.* Combinatorica 31(5): 583-614 (2011)

[19] Jacob Steinhardt, Gregory Valiant, Stefan Wager: *Memory, Communication, and Statistical Queries.* COLT 2016: 1490-1516

[20] Gregory Valiant, Paul Valiant: *Information Theoretically Secure Databases.* Electronic Colloquium on Computational Complexity (ECCC) 23: 78 (2016)