Proceedings of Machine Learning Research vol 75:1–22, 2018 31st Annual Conference on Learning Theory

A Direct Sum Result for the Information Complexity of Learning

Ido Nachum IDON@TX.TECHNION.AC.IL
Department of Mathematics, Technion-IIT, Haifa, Israel.

Jonathan Shafer SHAFERJO@BERKELEY.EDU
Computer Science Division, University of California, Berkeley, CA.

Amir Yehudayoff AMIR.YEHUDAYOFF@GMAIL.COM
Department of Mathematics, Technion-IIT, Haifa, Israel.

Editors: Sebastien Bubeck, Vianney Perchet and Philippe Rigollet

Abstract

How many bits of information are required to PAC learn a class of hypotheses of VC dimension d? The mathematical setting we follow is that of Bassily et al., where the value of interest is the mutual information $\mathrm{I}(S;A(S))$ between the input sample S and the hypothesis outputted by the learning algorithm A. We introduce a class of functions of VC dimension d over the domain $\mathcal{X}$ with information complexity at least $\Omega\left(d \log\log \frac{|\mathcal{X}|}{d}\right)$ bits for any consistent and proper algorithm (deterministic or random). Bassily et al. proved a similar (but quantitatively weaker) result for the case $d = 1$.

The above result is in fact a special case of a more general phenomenon we explore. We define the notion of *information complexity* of a given class of functions $\mathcal{H}$. Intuitively, it is the minimum amount of information that an algorithm for $\mathcal{H}$ must retain about its input to ensure consistency and properness. We prove a direct sum result for information complexity in this context; roughly speaking, the information complexity sums when combining several classes.

Keywords: PAC Learning, Information Theory, VC Dimension, Direct Sum.

1. Introduction

A simple, interesting and central observation in computational learning theory suggests that *learning* and *information compression* are closely related tasks, and in some sense are equivalent: In order to compress a dataset, one needs to identify patterns or regularities that exist in the data, and leverage them to construct a representation that is more concise than the verbatim description of the data. Similarly, learning involves identifying patterns or regularities in the training data, usually for the purpose of making predictions about some future data that is expected to exhibit similar patterns.

This intuition has been formalized a number of times in various ways; for example, via sample compression schemes, Occam's razor, and minimum description length (see Section 1). The observation is fruitful not least because it enables borrowing tools from the study of compression, which are often combinatorial or information-theoretic in nature, and applying them to study statistical notions of learning. This paper presents a result pertaining to the *information complexity* of hypothesis classes, which is yet another formalization.

The information complexity view roughly goes as follows. Consider a learner that is given some training data, and will later need to make predictions about new data that it has not yet seen. Once the learner is successful in identifying the underlying patterns in the data, we expect it will be able to keep just a small amount of information which represents these patterns, discard the rest of the training data, and still be successful in making predictions about future instances. Thus, it is natural to ask how much information the learner retains from the training set.

This question was first introduced in Bassily et al. (2018), where it is formalized as follows (see Section 2 for definitions and notation). Let S be a random variable representing the i.i.d. training samples for a supervised learning algorithm and let h represent the hypothesis that the algorithm outputs on input S. Consider the mutual information $\mathrm{I}(S;h)$, which quantifies the amount of information that the algorithm retains about the input when making predictions. The authors show that

$$\Pr\left[|\text{true error} - \text{empirical error}| > \varepsilon\right] < O\left(\frac{\mathrm{I}(S;h)}{m\varepsilon^2}\right).$$

where m is the number of samples in the training set. Thus, if the mutual information grows slowly compared to m, say $\mathrm{I}(S;h) = o(m)$, then the algorithm generalizes well, meaning that it does not over-fit. If in addition the empirical error vanishes then the algorithm PAC learns. Similar results were also obtained by Xu and Raginsky (2017).

Again, this makes sense. If the mutual information does not grow a lot slower than m, then the algorithm is basically memorizing large portions of the training set, and therefore its output is unlikely to generalize to unseen instances. Conversely, if the mutual information is small compared to m then the algorithm cannot tailor its output to the noisy details of specific training instances, and so it cannot over-fit.

More generally, given a hypothesis class $\mathcal{H}$, we define the information complexity $\mathsf{IC}(\mathcal{H})$ to be the least amount of information that the output of an algorithm for $\mathcal{H}$ must retain to ensure consistency and properness. Here is a brief and rough overview of the formalism behind this notions (see Definition 9 for the full details). The *information cost* of a consistent and proper learning algorithm A for $\mathcal{H}$ is $\mathsf{IC}_A(\mathcal{H}) = \sup_p \mathrm{I}(S;A(S))$ where p is a distribution on inputs and I denotes mutual information (here the sample size m is fixed). In words, it is the maximum amount of information the output of A contains over all distributions on inputs. The *information complexity* of $\mathcal{H}$ is $\mathsf{IC}(\mathcal{H}) = \inf_A \mathsf{IC}_A(\mathcal{H})$.

This definition is the main object of study in this work. As the name suggests, the definition is inspired by similar notions in computational complexity theory, like the concept of information complexity in the field of communication complexity (see e.g. Braverman, 2012). Loosely speaking, algorithms have costs and the minimum cost for a given problem is the complexity of the problem.

The information complexity of a learning problem seems related to important properties of the problem, like the sample size needed to perform learning, and may yield new and useful learning paradigms, which aim at minimizing the amount of information algorithms use. It is also related to several standard notions in learning theory (see Section 1 below). We therefore set out to explore this measure and understand how it works.

One question that arises immediately inquires how, if at all, does information complexity relate to the Vapnik–Chervonenkis (VC) dimension, which is the standard measure of complexity in learning theory. VC dimension is important because a hypothesis class is PAC-learnable if and only if its VC dimension is finite, and furthermore, when learning is possible then the VC dimension determines the sample complexity (Vapnik and Chervonenkis, 1971; Blumer et al., 1989).

This work makes a first step towards understanding information complexity, and its relation to the VC dimension, by proving the following theorem.

Theorem 1 (Lower bound for VC classes) *There exists a family of hypothesis classes* $\{\mathcal{H}_{k,d} : k,d,n \in \mathbb{N}, k = d2^n\}$ *where* $\mathcal{H}_{k,d} \subseteq \{0,1\}^{[k]}$ *and* $\mathsf{VC}(\mathcal{H}_{k,d}) = d$ *such that*

$$\mathsf{IC}(\mathcal{H}_{k,d}) = \Omega\left(d \log\log\left(k/d\right)\right) = \Omega\left(d \log n\right).$$

This theorem shows that in some cases learners must retain or leak a large amount of information about their inputs, even for classes of low VC dimension. The theorem applies to both deterministic and randomized algorithms. A weaker variant of it, with $d = 1$, already appeared in Bassily et al. (2018). It also provides a separation between information complexity and sample compression schemes (as explained below).

The theorem is in fact an instance of a much more general direct-sum-type phenomenon. In a nutshell, direct sums in computational complexity theory refer to the behavior of the computational complexity when problems are combined. For example, how does the complexity of completing two tasks relate to the complexity of completing each of the tasks separately. It is a central questions that appears in boolean formula complexity (see e.g. Karchmer et al., 1995), communication complexity (see e.g. Section 4.1 in Kushilevitz and Nisan, 1997), and more.

Here we describe the statement in rough terms (for formal details see Section 4). Given two concept classes $\mathcal{H}_1 \subseteq \{0,1\}^{\mathcal{X}_1}$ and $\mathcal{H}_2 \subseteq \{0,1\}^{\mathcal{X}_2}$, define the product class $\mathcal{H}_1 \times \mathcal{H}_2$ as the class of functions over the disjoint union of $\mathcal{X}_1$ and $\mathcal{X}_2$ that are obtained by combining some $h_1 \in \mathcal{H}_1$ and $h_2 \in \mathcal{H}_2$. The main question we address is how does the information complexity of $\mathcal{H}_1 \times \mathcal{H}_2$ relate to that of $\mathcal{H}_1$ and $\mathcal{H}_2$. It is natural to conjecture that the information complexity sums; namely,

$$\mathsf{IC}(\mathcal{H}_1 \times \mathcal{H}_2) \approx \mathsf{IC}(\mathcal{H}_1) + \mathsf{IC}(\mathcal{H}_2).$$

We prove that this is indeed the case.

Related work

Connections between learning and compression have been studied extensively. The minimum description length principle developed by Rissanen and others is one important avenue (Rissanen, 1978; Grünwald, 2007), as is Solomonoff induction, which relates to compression via Kolmogorov complexity (Solomonoff 1964; Ming and Vitányi 1997; see also Hutter 2007). Below we discuss two seminal results relating learning and compression which we find particularly pertinent.

Sample compression schemes. The concept of information complexity was first conceived as an attempt to generalize sample compression schemes, which constitute a well-known connection between learning and compression. Sample compression schemes are a class of learning algorithms whose output hypothesis is determined by a small subsample of the input. The classic example is support vector machines, which output a separating hyperplane that is determined by a small number of support vectors. Littlestone and Warmuth (1986) introduced sample compression schemes, and showed that every sample compression scheme is a PAC learner. Recently, Moran and Yehudayoff (2016) resolved a longstanding open question and showed that the converse also holds: every hypothesis class of finite VC dimension has a compression scheme. Together, these two results show one way in which learning and compression are equivalent. The present work extends Bassily et al.

(2018) in showing that learning with small mutual information and sample compression schemes are different.

Occam's razor. Another classic connection between learning and compression was provided by Blumer et al. (1987). They assume some fixed encoding of the hypotheses in a class $\mathcal{H}$, and define the complexity of a hypothesis to be the length of its encoding. Roughly, they show that if an algorithm always outputs a consistent and relatively simple hypothesis then the algorithm generalizes. More explicitly, they show a bound on the sample complexity under the condition that the output hypothesis is of some given complexity. Since information complexity is a lower bound on the entropy of the output hypothesis, which itself is a lower bound on the length of its encoding, the lower bound for information complexity proved herein carries over also to the setting of Occam's razor.

Differential privacy. The mutual information $\mathrm{I}(S;A(S))$ can also be thought of as the amount of information the learning algorithm reveals about its input, which calls to mind the setting of differential privacy. First introduced by Dwork et al. (2006), differential privacy is a principled notion of privacy that has recently been studied extensively because it provides strong privacy guarantees to data sources in a rigorous way. It also plays a roll in controlling overfitting, as several recent works have shown (e.g. Dwork et al., 2015; Bassily et al., 2016; Rogers et al., 2016; Bassily et al., 2014). Specifically, Bassily et al. (2016) provides an analysis of differential privacy as a form of distributional stability, and provides a tight characterization of the generalization guarantees that differential privacy entails. The ideas presented in this work may carry over to show direct sums in differential privacy.

Proof Outline

Here we provide an outline of the proof's structure. The proof consists of four parts.

In section 3, we consider the class of thresholds over the domain $\mathcal{X}$. We improve upon the lower bound in Theorem 5.1 of Bassily et al. (2018), and show that for every sample size m and every consistent and proper learning algorithm, there exists a distribution and a threshold function such that $\mathrm{I}(S;A(S)) = \Omega(\log\log|\mathcal{X}|)$. This corresponds to the case $d = 1$ in Theorem 1.

In section 4, we define the direct sum of classes of functions $\mathcal{H}_1,\ldots,\mathcal{H}_d$. Every function in the new class is the concatenation of d functions, one from each class. In particular, we are interested in the direct sum of d classes of threshold functions. The new class has VC dimension d and is denoted $\mathcal{T}_{k,d}$.

Then, we prove that (under certain conditions) the information complexity of the direct sum of d classes is roughly the sum of their information complexities. This is the main technical contribution of this work. It harnesses Sion's generalization of von Neumann's minimax theorem in a somewhat surprising way; instead of considering the space of distributions over the domain of interest, we need to move to the space of distributions over distributions. Finally, in section 5, we verify that the relevant conditions hold for $\mathcal{T}_{k,d}$ and conclude our proof.

2. Preliminaries

STANDARD NOTATION

Notation 2.1 *Let $\mathcal{X}$ be a set. The support of a probability function $p : \mathcal{X} \to [0,1]$ is the subset of elements of $\mathcal{X}$ that have a positive probability,* $\mathrm{supp}(p) = \{x \in \mathcal{X} : p(x) > 0\}$. *We use $\Delta(\mathcal{X})$ to*

denote the set of all probability mass functions over $\mathcal{X}$ that have a finite support. If p is a probability function, we use p^m to denote the probability function over $\mathcal{X}^m$ that corresponds to the probability of performing m i.i.d. samples from $\mathcal{X}$ according to p: $p^m((x_1,\cdots,x_m)) = \Pi_{i=1}^m p(x_i)$.

INFORMATION THEORY

Definition 2 *Let $\mathcal{X}$ be a countable set, and let X be a random variable over $\mathcal{X}$ with probability mass function p such that $p(x) = \Pr(X = x)$. The entropy of X is*[1] $\mathrm{H}(X) = \sum_{x\in\mathcal{X}} p(x)\log\frac{1}{p(x)}$.

Definition 3 *Let X and Y be random variables over countable sets $\mathcal{X}$ and $\mathcal{Y}$ respectively. The mutual information between X and Y is $\mathrm{I}(X;Y) = H(X) + H(Y) - H(X,Y)$.*

See the textbook Cover and Thomas (2006) for additional basic definitions and results from information theory which are used throughout this paper.

LEARNING THEORY

Part I of Shalev-Shwartz and Ben-David (2014) provides an excellent comprehensive introduction to computational learning theory. Following are some basic definitions.

Definition 4 *Let $\mathcal{X}$ and $\mathcal{Y}$ be sets. $\mathcal{H}$ is called a class of hypotheses if $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$. $\mathcal{S} = \mathcal{X}\times\mathcal{Y}$ is called the sample space. A realizable sample for $\mathcal{H}$ of size m is*

$$S = \big((x_1,y_1),\ldots,(x_m,y_m)\big) \in \mathcal{S}^m$$

such that there exists $h \in \mathcal{H}$ satisfying $y_i = h(x_i)$ for all $i \in [m]$.

Definition 5 *A learning algorithm for $\mathcal{H}$ with sample size m is a (possibly randomized) algorithm that takes a realizable sample $S = ((x_1,y_1),\ldots,(x_m,y_m))$ for $\mathcal{H}$ as input, and returns a function $h : \mathcal{X} \to \mathcal{Y}$ as output. We say that the learning algorithm is consistent if the output h always satisfies $y_i = h(x_i)$ for all $i \in [m]$. We say the algorithm is proper if it outputs members of $\mathcal{H}$.*

Definition 6 *Let $p \in \Delta(\mathcal{S})$. We say that p is realizable by $\mathcal{H}$ or equivalently that p is consistent with $\mathcal{H}$ if there exists $h \in \mathcal{H}$ such that for all $(x,y) \in \mathrm{supp}(p)$ it holds that $y = h(x)$. We use $\Delta_{\mathcal{H}}$ to denote the set of all distributions in $\Delta(\mathcal{S})$ that are consistent with $\mathcal{H}$.*

Definition 7 *We say that $\mathcal{H} \subseteq \{0,1\}^{\mathcal{X}}$ shatters some finite set $C \subseteq \mathcal{X}$ if*[2] *$\mathcal{H}_C = \{0,1\}^C$. The VC dimension of $\mathcal{H}$ denoted $\mathsf{VC}(\mathcal{H})$ is the maximal size of a set $C \subseteq \mathcal{X}$ such that $\mathcal{H}$ shatters C. If $\mathcal{H}$ can shatter sets of arbitrary size, we say that the VC dimension is ∞.*

INFORMATION COMPLEXITY

Definition 8 *Let A be a learning algorithm for $\mathcal{H}$ with sample size m, and let h be the output of A when executed with input S. We say that A retains at most d bits of information from S if*

$$\forall p \in \Delta_{\mathcal{H}} \quad \mathop{\mathrm{I}}_{S\sim p^m}(S;h) \le d.$$

[1] $\log(x)$ is a shorthand for $\log_2(x)$, and we use the convention that $0\log\frac{1}{0} = 0$.
[2] $\mathcal{H}_C$ is the restriction of $\mathcal{H}$ to C.

Definition 9 *Let $\mathcal{A}_m$ denote the set of (possibly randomized) consistent and proper learning algorithms for $\mathcal{H}$ over samples of size m. The information complexity of $\mathcal{H}$ for samples of size m is*

$$\mathsf{IC}_m(\mathcal{H}) = \inf_{A\in\mathcal{A}_m} \sup_{p\in\Delta_{\mathcal{H}}} \mathop{\mathrm{I}}_{S\sim p^m}(A(S);S)\,.$$

The information complexity of $\mathcal{H}$ is $\mathsf{IC}(\mathcal{H}) = \sup_{m\in\mathbb{N}} \mathsf{IC}_m(\mathcal{H})$.

Conceptually, the information complexity is the minimal amount of information that an algorithm must retain in order to be consistent and proper.

3. The Lower Bound for Thresholds

We start with the case $d = 1$, and with the class of thresholds.

Definition 10 *The class of threshold functions of size 2^n is denoted $\mathcal{T}_n$ and defined as follows: Let $\mathcal{X} = [2^n]$ and $\mathcal{Y} = \{0,1\}$. The set $\mathcal{T}_n \subseteq \mathcal{Y}^{\mathcal{X}}$ consists of all monotone increasing functions; that is, $\mathcal{T}_n = \{f_k\}_{k\in[2^n+1]}$ where*

$$f_k(x) = \begin{cases} 0 & x < k \\ 1 & x \geq k \end{cases}$$

Following is our main result for thresholds.

Theorem 11 (Lower bound for thresholds) *For any (possibly randomized) proper and consistent learning algorithm A for $\mathcal{T}_n$ with sample size $m \geq 2$ there exists a distribution $p \in \Delta_{\mathcal{T}_n}$ such that*

$$\mathop{\mathrm{I}}_{S\sim p^m}(A(S);S) = \Omega(\log n) = \Omega(\log\log|\mathcal{X}|).$$

This theorem (which is proved in Appendix A.1) is an improvement upon Theorem 5.1 from Bassily et al. (2018). The proof of Theorem 11 follows the same outline as the proof of Bassily et al. (2018). Their proof, however, was based upon conditioning on an event of small probability $O(\frac{1}{m^2})$, so they arrived at a quantitatively weaker bound of $\Omega(\frac{\log n}{m^2})$.

We are able to remove this dependence by using the following simple, yet useful and general observation: Permuting the ordered set of samples does not increase the mutual information. In other words, a learning algorithm that aims at minimizing the information cost should not use the order in which the examples appeared.

Lemma 12 *Let $\mathcal{H}$ be a class of hypotheses and A be a learning algorithm that accepts samples of size m. Define A' to be $A'(S) = A(\Sigma(S))$ where Σ is a random permutation chosen uniformly from all permutations on m elements (independently of the input sample and the random coins of A). Then for all $p \in \Delta_{\mathcal{H}}$,*

$$\mathop{\mathrm{I}}_{S\sim p^m}(A(S);S) \geq \mathop{\mathrm{I}}_{S\sim p^m}\left(A'(S);S\right).$$

Proof [Lemma 12].

$$\mathrm{H}(A(S)\,|\,S) \overset{(a)}{=} \mathrm{H}(A(\Sigma(S))\,|\,\Sigma(S)) \overset{(b)}{=} \mathrm{H}(A(\Sigma(S))\,|\,\Sigma(S),S,\Sigma) \leq \mathrm{H}(A(\Sigma(S))\,|\,S)$$

where (a) holds because S and $\Sigma(S)$ have the same distribution, and (b) holds because $A(\Sigma(S))$ and (S,Σ) are independent conditioned on $\Sigma(S)$. Therefore,

$$\begin{aligned}\mathop{\mathrm{I}}_{S\sim p^m}(A(S);S) &= \mathrm{H}(A(S)) - \mathrm{H}(A(S)\,|\,S) \\ &\geq \mathrm{H}(A(\Sigma(S))) - \mathrm{H}(A(\Sigma(S))\,|\,S) \\ &= \mathop{\mathrm{I}}_{S\sim p^m}(A(\Sigma(S));S)\end{aligned}$$

as desired. ■

4. The Direct Sum

Our strategy for proving the lower bound in Theorem 1 is to use a reduction to the lower bound for thresholds. The main step in the proof is a direct sum result for information complexity (as was discussed in the introduction). We know that the information complexity of thresholds $\mathcal{T}_n$ is high. Our goal is to prove that this implies that the information complexity of the class of functions that have d thresholds is d times as large as that of $\mathcal{T}_n$.

The statement we prove can be more generally stated using the following terminology.

Definition 13 *Let*

$$\mathcal{H}_1 \subseteq \mathcal{Y}_1^{\mathcal{X}_1}, \ldots, \mathcal{H}_d \subseteq \mathcal{Y}_d^{\mathcal{X}_d}$$

be classes of hypotheses, where $\mathcal{X}_1,\ldots,\mathcal{X}_d$ are disjoint sets. The d-fold product of $\mathcal{H}_1,\ldots,\mathcal{H}_d$ is defined by:

$$\mathcal{X} = \bigcup_{i=1}^{d} \mathcal{X}_i, \quad \mathcal{Y} = \bigcup_{i=1}^{d} \mathcal{Y}_i,$$

and

$$\mathcal{H} = \mathcal{H}_1 \times \mathcal{H}_2 \times \cdots \times \mathcal{H}_d = \{(h_1,\ldots,h_d) : \forall i \;\; h_i \in \mathcal{H}_i\} \subseteq \mathcal{Y}^{\mathcal{X}}$$

such that if $h = (h_1,\ldots,h_d)$ and $x \in \mathcal{X}$ then $h(x) = h_i(x)$ where i is the integer for which $x \in \mathcal{X}_i$.

We are interested in the behavior of the VC dimension with respect to the product of classes.

Lemma 14 $\mathsf{VC}(\mathcal{H}_1 \times \mathcal{H}_2 \times \cdots \times \mathcal{H}_d) = \sum_{j=1}^{d} \mathsf{VC}(\mathcal{H}_i)$.

We now state the direct sum result for information complexity. Namely, that the information complexity of the product class roughly equals the sum of the information complexities. We do this using the two theorems below (all further proofs are deferred to the appendices).

The first theorem enable us to treat the mutual information on each summand separately.

Theorem 15 (Direct sum I) *Assume that*

- *$\mathcal{H}$ is the product of $\mathcal{H}_1,\ldots,\mathcal{H}_d$.*
- *A is a (possibly randomized) learning algorithm for $\mathcal{H}$.*
- *$p \in \Delta_{\mathcal{H}}$ and $S \sim p^m$.*

- $A(S) = (h_1, \ldots, h_d)$.
- *For each* $i \in [d]$,
  - S_i *is the subsample of* S *of examples from* $\mathcal{X}_i \times \mathcal{Y}_i$.
  - $M_i = |S_i|$ *is the number of examples in* S_i.

Then

$$\mathop{\mathrm{I}}_{S\sim p^m}(A(S);S) \geq \sum_{i=1}^{d} \mathop{\mathrm{I}}_{S\sim p^m}(h_i;S_i|M_i)$$

Here is a simple illustration of the usefulness of the theorem. Consider a learning algorithm A for the product class $\mathcal{H}$. It gets as input a list of examples S, and it knows to which $\mathcal{H}_i$ each example corresponds to. When applying the chain rule on $\mathrm{I}(A(S);S)$, we see e.g. that when analyzing the information relevant to $\mathcal{H}_2$ we need to condition on examples for $\mathcal{H}_1$. This in turn implies that we are not dealing with a single algorithm for $\mathcal{H}_2$, but with a family of algorithms that implicitly depend on S. The theorem, however, shows that all of this can be ignored without a significant price. The only thing that matters is how many examples that correspond to $\mathcal{H}_2$ there are.

To complete the proof, we now need to find a hard distribution for the product class, one that yields large information complexity. A natural idea would be to use the hard distributions of the individual classes. However, this approach fails, since there is no single hard distribution; it follows from Bassily et al. (2018) for the case of thresholds that for every algorithm there is a hard distribution but for every distribution there is also an algorithm that retains little information ($O(1)$).

To solve this problem, we need to replace the notion of a hard distribution, by the notion of a hard distribution on distributions. Indeed, we show that for thresholds (and in fact more generally) there is a single distribution on distributions that for every algorithm yields high information cost (on average). To this end, we use the spirit of von Neumann minimax theorem (Theorem 25) and write the following (we actually need Sion's minimax theorem, Theorem 26):

$$\min_{A} \max_{D\in\Delta(\Delta_{\mathcal{H}})} f(A,D) = \max_{D\in\Delta(\Delta_{\mathcal{H}})} \min_{A} f(A,D)$$

where $f(A,D) = \mathop{\mathbb{E}}_{p\sim D} \mathop{\mathrm{I}}_{S\sim p^m}(A_m(S);S)$. This enables us to find a distribution over the space of distributions that is hard for all algorithms. This rational can be useful in other contexts where the minimax theorem doesn't hold; although the minimax theorem does not apply for distributions versus algorithms in this context, it does hold for distributions over distributions versus algorithms.

We still have one more technical difficulty to handle: Each hard distribution is tailored for a specific m, and we do not know m in advance. To address this issue, we need to consider the setting where the number of samples m is also random, as in the following lemma.

Definition 16 *Let* $M \subseteq \mathbb{N}$ *and let* $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$ *be a class of hypotheses. A learning algorithm for* $\mathcal{H}$ *that accepts samples of sizes in* M *is a vector* $(A_m)_{m\in M}$ *such that each* A_m *is a learning algorithm for* $\mathcal{H}$ *that accepts samples of size* m*. We say that* $(A_m)_{m\in M}$ *is consistent and proper if all the algorithms in the vector are consistent and proper.*

The combination of the minimax idea together with a randomized sample size is summarized in the following lemma.

Lemma 17 *Let $c \in \mathbb{R}$, $M \subseteq \mathbb{N}$, $|M| < \infty$, $\mu \in \Delta(M)$ and let $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$ be a class of hypotheses. Assume that for any consistent and proper learning algorithm $(A_m)_{m\in M}$ for $\mathcal{H}$ there exists $p \in \Delta_{\mathcal{H}}$ such that*

$$\mathop{\mathbb{E}}_{m\sim\mu} \mathop{\mathrm{I}}_{S\sim p^m}(A_m(S);S) \geq c$$

Then there exists $D \in \Delta(\Delta_{\mathcal{H}})$ such that for any consistent and proper learning algorithm $(A_m)_{m\in M}$ it holds that

$$\mathop{\mathbb{E}}_{p\sim D} \mathop{\mathbb{E}}_{m\sim\mu} \mathop{\mathrm{I}}_{S\sim p^m}(A_m(S);S) \geq c$$

The second direct sum theorem assumes that the consequent of Lemma 17 holds, i.e. that for a specific distribution on m (the number of examples received), each class has a hard distribution on distributions such that the expected mutual information over all those distributions is high for all algorithms.

Theorem 18 (Direct sum II) *Let $m,d \in \mathbb{N}$ and $T = [\frac{m}{2}, \frac{3m}{2}] \cap \mathbb{N}$. Let $\mu \in \Delta(T)$ be the distribution*[3] $\mathrm{Bin}(dm, \frac{1}{d})$ *conditioned on the event E that the integer sampled is in T. Assume that for each $i \in [d]$:*

1. *$\mathcal{H}_i \subseteq \mathcal{Y}_i^{\mathcal{X}_i}$ is a class of hypotheses.*
2. *There exists $c_i \in \mathbb{R}$ and $D_i \in \Delta(\Delta_{\mathcal{H}_i})$ such that for every (possibly randomized) consistent and proper learning algorithm $(A_t)_{t\in T}$ for $\mathcal{H}_i$, it holds that*

$$\mathop{\mathbb{E}}_{p\sim D_i} \mathop{\mathbb{E}}_{t\sim\mu} \mathop{\mathrm{I}}_{S\sim p^t}(A_t(S);S) \geq c_i$$

Finally, let $\mathcal{H}$ be the product of $\mathcal{H}_1,\ldots,\mathcal{H}_d$. Then for every (possibly randomized) consistent and proper learning algorithm A for $\mathcal{H}$ that accepts samples of size dm there exists a distribution $p \in \Delta_{\mathcal{H}}$ such that

$$\mathop{\mathrm{I}}_{S\sim p^{dm}}(A(S);S) \geq \alpha \sum_{i=1}^{d} c_i$$

where $\alpha = 1 - 2e^{-\frac{m}{2d}}$.

5. The Lower Bound for VC Classes

To prove Theorem 1 we need to verify that condition 2 of Theorem 18 is true for the product of d thresholds.

Definition 19 *Let $\mathcal{T}_{n,d}$ be the d-fold product of $\mathcal{T}_n$. The domain of the functions in $\mathcal{T}_{n,d}$ is $[d \cdot 2^n]$. Each function in $\mathcal{T}_{n,d}$ is of the form:*

$$f_{k_1,\ldots k_d}(x) = \begin{cases} 0 & x < k_{i(x)} \\ 1 & x \geq k_{i(x)} \end{cases} \qquad i(x) = \left\lceil \frac{x}{2^d} \right\rceil$$

where $k_j \in [(j-1)\cdot 2^n + 1, j \cdot 2^n + 1]$ for each $j \in [d]$. (See Figure 1 for a graphical illustration of this definition.)

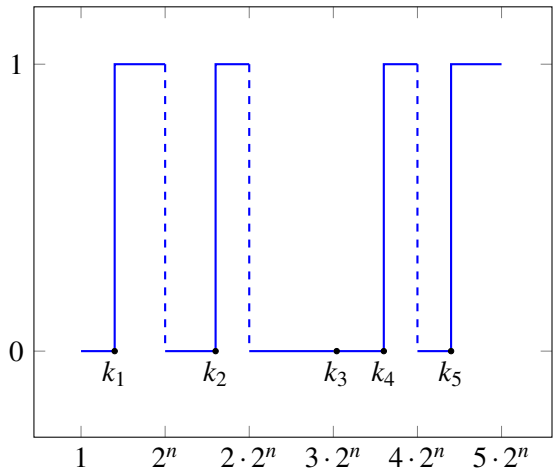

Figure 1: Example of a function in $\mathcal{T}_{n,5}$.

Lemma 14 implies:

Corollary 20 *The VC dimension of $\mathcal{T}_{n,d}$ is d.*

It thus remains to prove the following lemma, which is stronger than Theorem 11.

Lemma 21 *Let $a,b,n \in \mathbb{N}$, $2 \leq a \leq b$, $M = [a,b] \cap \mathbb{N}$, let $(A_m)_{m \in M}$ be a (possibly randomized) consistent and proper learning algorithm for $\mathcal{T}_n$, and let $\mu \in \Delta(M)$. Then there exist a distribution $p \in \Delta_{\mathcal{T}_n}$ such that*

$$\mathop{\mathbb{E}}_{m \sim \mu} \mathop{\mathrm{I}}_{S \sim p^m} (S; A_m(S)) = \Omega\left(\left(\frac{a}{b}\right)^2 \log n\right).$$

Finally, we can prove the theorem for VC classes.

Proof [Theorem 1]. Fix some $m \geq 2d$ and take μ as in Theorem 18. Apply Lemma 21 with $a = \frac{m}{2}$ and $b = \frac{3m}{2}$, which together with Lemma 17 entails that there exists $D \in \Delta(\Delta_{\mathcal{T}_n})$ such that for any $(A_m)_{m \in M}$ for $\mathcal{T}_n$,

$$\mathop{\mathbb{E}}_{p \sim D} \mathop{\mathbb{E}}_{m \sim \mu} \mathop{\mathrm{I}}_{S \sim p^m} (A_m(S); S) \geq \Omega(\log n).$$

Theorem 1 now follows directly from applying Theorem 18 to the class $\mathcal{T}_{n,d}$. ■

6. Discussion and Directions for Further Work

A direct continuation of the current line of research would be to extend the lower bound for VC classes to PAC learners that are not necessarily proper or consistent. Note that the lower bound for thresholds does not hold for the case of randomized, consistent, non-proper algorithms. Consider the algorithm for thresholds that outputs a hypothesis h as follows. For any (x,y) in the training

[3] $\mathrm{Bin}(k,p)$ is the binomial distribution with k i.i.d. trials each of which has probability of success p.

sample, $h(x) = y$. For any x that did not appear in the sample, $h(x)$ is sampled uniformly from $\{0,1\}$. This algorithm has mutual information that does not grow with the size of the domain (it is $O(m)$). This is not too meaningful, as this algorithm is not a PAC learner. But it illustrates that the lower bound breaks somewhere, and it would be worthwhile to identify exactly how far the assumptions can be pushed before it breaks.

A different and interesting direction is to prove upper bounds on information complexity. First, we would like to understand whether the lower bound presented here is sharp. Better yet: Can we provide explicit general constructions for learning algorithms that obtain the information complexity, i.e., retain the minimal amount of information possible? Following the theorem of Bassily et al. (2018) stating that compression entails learning, this would yield a novel class of learning algorithms for all hypothesis classes in which the information complexity is $o(m)$ – a strong result that might even have practical applications.

Lastly, and perhaps most interestingly, one may also consider the converse of that theorem: Is there a sense in which low information complexity is a necessary condition for learnability? Are the concepts equivalent?

References

Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pages 464–473. IEEE, 2014.

Raef Bassily, Kobbi Nissim, Adam Smith, Thomas Steinke, Uri Stemmer, and Jonathan Ullman. Algorithmic stability for adaptive data analysis. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 1046–1059. ACM, 2016.

Raef Bassily, Shay Moran, Ido Nachum, Jonathan Shafer, and Amir Yehudayoff. Learners that use little information. In *Proceedings of the 29th international conference on algorithmic learning theory*. To appear, 2018.

Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Occam's razor. *Information processing letters*, 24(6):377–380, 1987.

Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965, 1989.

Mark Braverman. Interactive information complexity. In *In Proceedings of the 44th annual ACM Symposium on Theory of Computing, STOC12*, 2012.

Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2ed edition, 2006.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pages 265–284. Springer, 2006.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Leon Roth. Preserving statistical validity in adaptive data analysis. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 117–126. ACM, 2015.

Peter D Grünwald. *The minimum description length principle*. MIT press, 2007.

Marcus Hutter. Universal algorithmic intelligence: A mathematical top down approach. In *Artificial General Intelligence*, pages 227–290. Springer, 2007.

Mauricio Karchmer, Ran Raz, and Avi Wigderson. Super-logarithmic depth lower bounds via the direct sum in communication complexity. *Computational Complexity*, 5(3-4):191–204, 1995.

Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, 1997.

Nick Littlestone and Manfred Warmuth. Relating data compression and learnability. Technical report, Technical report, University of California, Santa Cruz, 1986.

Li Ming and Paul Vitányi. *An introduction to Kolmogorov complexity and its applications*. Springer Heidelberg, 1997.

Shay Moran and Amir Yehudayoff. Sample compression schemes for VC classes. *Journal of the ACM (JACM)*, 63(3):21, 2016.

Jorma Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978.

Ryan Rogers, Aaron Roth, Adam Smith, and Om Thakkar. Max-information, differential privacy, and post-selection hypothesis testing. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 487–494. IEEE, 2016.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Maurice Sion. On general minimax theorems. *Pacific journal of mathematics*, 8(1):171–176, 1958.

Ray J Solomonoff. A formal theory of inductive inference, part I. *Information and control*, 7(1): 1–22, 1964.

Vladimir N Vapnik and Alexey Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Measures of Complexity*, 16(2):11, 1971.

John Von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.

John Von Neumann and Oskar Morgenstern. Theory of games and economic behavior. 1944.

Aolin Xu and Maxim Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. In *Advances in Neural Information Processing Systems*, pages 2521–2530, 2017.

Appendix A. Proofs

A.1. Lower Bound for Thresholds

We start with two lemmas from Bassily et al. (2018).

Lemma 22 *Let $Q \in \mathrm{Mat}_{2^n \times 2^n}(\Delta([2^n]))$, i.e. Q is a $2^n \times 2^n$ matrix where each cell contains a probability function over $[2^n]$. Furthermore, assume that Q is symmetric and that it has the property that*

$$\forall i,j: \ \mathrm{supp}\, Q_{ij} \subseteq [\min\{i,j\}, \max\{i,j\}] \tag{ii}$$

Then Q contains a row with $n+1$ distributions $p_1,\ldots,p_{n+1}$ such that there exist pairwise disjoint sets $S_1,\ldots,S_{n+1} \subseteq [2^n]$ satisfying

$$\forall i \in [n+1]: \ p_i(S_i) \geq \frac{1}{2} \tag{iii}$$

(And hence, from symmetry, it also contains such a column.)

Lemma 23 *Let Ω be a finite sample space (a finite set), let $p_1,\ldots,p_n$ be probability distributions over Ω, and let $A_1,\ldots,A_n \subseteq \Omega$ be pairwise disjoint events such that*

$$\forall i \in [n]: \ p_i(A_i) \geq \frac{1}{2} \tag{v}$$

Let U be a random variable distributed uniformly over $[n]$, and let W be

$$W \sim p_i, i \sim U$$

Namely, W is a random variable over Ω that results from sampling an index $i \in [n]$ according to U and then sampling an element of Ω according to p_i and assigning that element to W.

Then it holds that

$$\mathrm{I}(U;W) = \Omega(\log n)$$

Now, we can prove the desired lower bound for probabilistic algorithms using lemma 12.

Proof [Theorem 11]. Let A be a consistent learning algorithm for $\mathcal{T}_n$, and let A' be as in Lemma 12. Let

$$M \in \mathrm{Mat}_{2^n \times 2^n}(\Delta([2^n]))$$

be a matrix such that for all $i \neq j$, M_{ij} is a probability function such that $M_{ij}(k)$ is the probability that A' will output hypothesis f_k for an input sample of the form

$$S_{ij} = \Big(\underbrace{(1,0),\ldots,(1,0)}_{\text{m-2}}, (\min\{i,j\},0), (\max\{i,j\},1)\Big)$$

and for all i, M_{ii} is the degenerate distribution that assigns probability 1 to i. Notice that because A' is indifferent to the order of the examples in the input sample, M_{ij} actually equals the probability functions of the output for any permutation of S_{ij}.

M is symmetric and because A' is consistent it follows that M satisfies property (ii), and hence by lemma 22 M contains a row r with probabilities $p_1,\ldots,p_{n+1}$ for which there are pairwise disjoint sets $A_1,\ldots,A_{n+1} \subseteq [2^n]$ such that for all i, $p_i(A_i) \geq \frac{1}{2}$. Note that at least $\frac{n}{2}$ of these probabilities on row r are located above the diagonal, or else, from symmetry of M, at least $\frac{n}{2}$ of them are located above the diagonal on column r. Thus, we assume w.l.o.g. that probabilities $p_1,\ldots,p_{\frac{n}{2}}$ are located above the diagonal on row r in cells $(r,k_1),\ldots,(r,k_{\frac{n}{2}})$ (the symmetric case can be handled very similarly).

We use the following probability p over realizable samples of length m, where U_K is the uniform distribution over $\{k_1,\ldots,k_{\frac{n}{2}}\}$:

$$p(w)=\left(1-\frac{1}{m}\right)\mathbf{1}_{w=1}(w)+\frac{1}{2m}\mathbf{1}_{w=r}(w)+\frac{1}{2m}U_K(w)$$

Consider the event in which the generated sample S is any permutation of

$$S_{rk_i}=\Big(\underbrace{(1,0),\ldots,(1,0)}_{\text{m-2}},(r,0),(k_i,1)\Big)$$

for some k_i, and let E be an indicator random variable of this event. p satisfies

$$p(E=1)\geq\left(1-\frac{1}{m}\right)^{m-2}\cdot\left(\frac{1}{2m}\right)^2\cdot m(m-1)\geq\frac{1}{16e}$$

and

$$p(S_{rk_1})=p(S_{rk_2})=\cdots=p(S_{rk_{\frac{n}{2}}})$$

Let h be a random variable denoting the output of A when the input sample S is distributed according to p. We have the following chain of inequalities

$$\begin{aligned}\mathrm{I}(S;h)&\overset{(a)}{\geq}\mathrm{I}(S;h|E)\overset{(b)}{=}p(E=1)\cdot\mathrm{I}(S;h|E=1)+p(E=0)\cdot\mathrm{I}(S;h|E=0)\overset{(c)}{\geq}\\&\overset{(c)}{\geq}p(E=1)\cdot\mathrm{I}(S;h|E=1)\geq\frac{1}{16e}\cdot\mathrm{I}(S;h|E=1)\overset{(d)}{\geq}\\&\overset{(d)}{\geq}\frac{1}{16e}\cdot\mathrm{I}(U(S);h|E=1)\overset{(e)}{=}\Omega(\log n)\end{aligned}$$

which is justified as follows:

(a) Notice that $E\perp h|S$ because once we saw the actual sample, we know with certainty whether event E occurred or not (formally, $\mathrm{I}(E;h|S)\leq\mathrm{H}(E|S)=0\implies E\perp h|S$). Thus, this inequality follows from claim B.2.2 in the appendix.

(b) Definition of conditional mutual information.

(c) Positivity of mutual information.

(d) Here $U(\cdot)$ is any mapping that satisfies $\sigma(S_{rk_i})\mapsto i$ for all i and all permutations σ. The inequality then follows from the data processing inequality.

(e) Given that $E=1$, $U(S)$ is the uniform distribution on $[\frac{n}{2}]$. Furthermore, h is the result of sampling a hypothesis according to the distribution p_i, where i is the value of $U(S)$. Lastly, our choice of $p_1,\ldots,p_{\frac{n}{2}}$ ensured that there exist pairwise disjoint sets $A_1,\ldots,A_{\frac{n}{2}}$ such that $p_i(A_i)\geq\frac{1}{2}$ for all i, and so the lower bound follows from lemma 23.

Thus, we have shown that for every consistent learning algorithm for $\mathcal{T}_n$ that accepts samples of size m there exists a distribution $p\in\Delta_{\mathcal{T}_n}$ such that

$$\underset{S\sim p^m}{\mathrm{I}}(S;h)=\Omega(\log\log|\mathcal{X}|)$$

as desired. ■

A.2. Direct Sum

Proof [Lemma 14]. Let $\mathcal{H}$ be the product class, $k = \mathsf{VC}(\mathcal{H})$ and $k_i = \mathsf{VC}(\mathcal{H}_i)$ for all i. To see that $k \geq \sum k_i$, take sets $R_i \subset \mathcal{X}_i$ for all i such that $|R_i| = k_i$ and $\mathcal{H}_i$ shatters R_i (such sets exist because $k_i = \mathsf{VC}(\mathcal{H}_i)$). Now note that $\mathcal{H}$ shatters $\bigcup_{i=1}^{d} R_i$.

To see that $k \leq \sum k_i$, assume for contradiction that $\mathcal{H}$ shatters a set R of size strictly more than $\sum k$. Then there exists j such that $|\mathcal{X}_j \cap R| > k_j$. The assumption entails that $\mathcal{H}_j$ shatters $\mathcal{X}_j \cap R$, a contradiction. ■

Proof [Theorem 15]. Denote $M = (M_1, \ldots, M_d)$. Then

$$\mathrm{I}(A(S);S) \overset{(a)}{=} \mathrm{I}(A(S);S,M) \overset{(b)}{\geq} \mathrm{I}(A(S);S \,|\, M) \overset{(c)}{\geq} \mathrm{I}(A(S);S_1,\ldots,S_d \,|\, M) \overset{(b)}{=} \sum_{i=1}^{d} \mathrm{I}(A(S);S_i \,|\, M, S_{<i})$$

$$= \sum_{i=1}^{d} \mathrm{I}(h_1,\ldots,h_d;S_i \,|\, M, S_{<i}) \overset{(b)}{\geq} \sum_{i=1}^{d} \mathrm{I}(h_i;S_i \,|\, M, S_{<i}) \overset{(d)}{\geq} \sum_{i=1}^{d} \mathrm{I}(h_i;S_i \,|\, M) \overset{(e)}{\geq} \sum_{i=1}^{d} \mathrm{I}(h_i;S_i \,|\, M_i)$$

where the steps are justified as follows:

(a) (S,M) and S are functions of each other.

(b) The chain rule for mutual information.

(c) $S_1,\ldots,S_d$ is a function of S (data processing inequality).

(d) Follows from claim B.2.1 because $S_i \perp S_{<i} \,|\, M_i$.

(e) Again from claim B.2.1, because $S_i \perp M_{\neq i} \,|\, M_i$.

And the proof is complete. ■

Proof [Theorem 18]. Let D be the distribution on distributions that results from sampling a distribution p_i from $D_i = D_{i_\mu}$ for each $i \in [d]$, and then taking the average of these distributions. Formally, $D \in \Delta(\Delta_{\mathcal{H}})$ is defined as follows:

$$D(p) = \begin{cases} \prod_{i=1}^{d} D_i(p_i) & p = \frac{1}{d}\sum_{i=1}^{d} p_i \text{ s.t. } \forall i : p_i \in \Delta_{\mathcal{H}_i} \\ 0 & \text{otherwise} \end{cases}$$

Taking expectation on both sides of Lemma 15, we have that

$$\mathop{\mathbb{E}}_{p\sim D} \mathop{\mathrm{I}}_{S\sim p^{dm}}(A(S);S) \geq \mathop{\mathbb{E}}_{p\sim D} \sum_{i=1}^{d} \mathop{\mathrm{I}}_{S\sim p^{dm}}(h_i;S_i|M_i) = \sum_{i=1}^{d} \mathop{\mathbb{E}}_{p\sim D} \mathop{\mathrm{I}}_{S\sim p^{dm}}(h_i;S_i|M_i)$$

$$= \sum_{i=1}^{d} \mathop{\mathbb{E}}_{p_1\sim D_1} \cdots \mathop{\mathbb{E}}_{p_d\sim D_d} \mathop{\mathrm{I}}_{S\sim\left(\frac{1}{d}\sum_{j=1}^{d} p_j\right)^{dm}}(h_i;S_i|M_i) = \sum_{i=1}^{d} \mathop{\mathbb{E}}_{p_{\neq i}\sim D_{\neq i}} \mathop{\mathbb{E}}_{p_i\sim D_i} \mathop{\mathrm{I}}_{S\sim\left(\frac{1}{d}\sum_{j=1}^{d} p_j\right)^{dm}}(h_i;S_i|M_i)$$

where $\mathop{\mathbb{E}}_{p_{\neq i}\sim D_{\neq i}}$ is a shorthand notation for

$$\mathop{\mathbb{E}}_{p_1\sim D_1} \cdots \mathop{\mathbb{E}}_{p_{i-1}\sim D_{i-1}} \mathop{\mathbb{E}}_{p_{i+1}\sim D_{i+1}} \cdots \mathop{\mathbb{E}}_{p_d\sim D_d}$$

Next, we bound the innermost expectation on p_i for any fixed vector of distributions $p_{\neq i}$. Let E_i be the event in which $M_i \in T$. Then

$$\mathop{\mathbb{E}}_{p_i\sim D_i} \mathop{\mathrm{I}}_{S\sim\left(\frac{1}{d}\sum_{j=1}^d p_j\right)^{dm}}(h_i;S_i|M_i) \overset{(a)}{=} \mathop{\mathbb{E}}_{p_i\sim D_i} \mathop{\mathrm{I}}_{S\sim\left(\frac{1}{d}\sum_{j=1}^d p_j\right)^{dm}}(h_i;S_i|M_i,1_{E_i})$$

$$\geq \mathop{\mathbb{E}}_{p_i\sim D_i}\Pr(1_{E_i}=1) \mathop{\mathrm{I}}_{S\sim\left(\frac{1}{d}\sum_{j=1}^d p_j\right)^{dm}}(h_i;S_i|M_i,1_{E_i}=1) \overset{(b)}{\geq} \alpha \mathop{\mathbb{E}}_{p_i\sim D_i} \mathop{\mathrm{I}}_{S\sim\left(\frac{1}{d}\sum_{j=1}^d p_j\right)^{dm}}(h_i;S_i|M_i,1_{E_i}=1)$$

$$= \alpha \mathop{\mathbb{E}}_{p_i\sim D_i}\mathop{\mathbb{E}}_{m_i\sim M_i|1_{E_i}=1} \mathop{\mathrm{I}}_{S\sim\left(\frac{1}{d}\sum_{j=1}^d p_j\right)^{dm}}(h_i;S_i|M_i=m_i) \overset{(c)}{=} \alpha \mathop{\mathbb{E}}_{p_i\sim D_i}\mathop{\mathbb{E}}_{m_i\sim\mu} \mathop{\mathrm{I}}_{S\sim\left(\frac{1}{d}\sum_{j=1}^d p_j\right)^{dm}}(h_i;S_i|M_i=m_i) \overset{(d)}{\geq} \alpha c_i$$

which is justified as follows:

(a) M_i and $(M_i, 1_{E_i})$ are functions of each other.

(b) $\Pr(1_{E_i}=1) \geq \alpha$, from Claim B.1.

(c) $\mu = (M_i|1_{E_i}=1)$

(d) From assumption 2. Note: A takes an input sample S of which S_i is just a subsample, and outputs a vector of hypotheses of which h_i is just one component. However, we may ignore these other outputs, and we may regard the other input subsamples S_j for $j \neq i$ as random coins used by A. Thus, for the sake of this analysis A is viewed as a randomized learning algorithm that takes $S_i \sim p_i^{m_i}$ as input and produces h_i as output.

Thus, we have $\mathop{\mathbb{E}}_{p\sim D} \mathop{\mathrm{I}}_{S\sim p^{dm}}(A(S);S) \geq \alpha\sum_{i=1}^d c_i$. This entails that there exists a distribution $p \in \mathrm{supp}(D)$ such that $\mathop{\mathrm{I}}_{S\sim p^{dm}}(A(S);S) \geq \alpha\sum_{i=1}^d c_i$ as desired. ■

A.3. Proofs for Section 5

Proof [Lemma 17] For each $m \in M$, let $\mathcal{A}_m$ be the set of consistent learning algorithms for $\mathcal{H}$ that accept samples of size m, and let

$$\mathcal{A} = \Pi_{m\in M}\mathcal{A}_m$$

Notice that

$$\begin{aligned} c &\leq \inf_{A\in\mathcal{A}}\sup_{p\in\Delta_{\mathcal{H}}}\mathop{\mathbb{E}}_{m\sim\mu}\mathop{\mathrm{I}}_{S\sim p^m}(A_m(S);S) \\ &\leq \inf_{A\in\mathcal{A}}\sup_{D\in\Delta(\Delta_{\mathcal{H}})}\mathop{\mathbb{E}}_{p\sim D}\mathop{\mathbb{E}}_{m\sim\mu}\mathop{\mathrm{I}}_{S\sim p^m}(A_m(S);S) \end{aligned}$$

where the first inequality follows from the assumption and the second holds because $\Delta(\Delta_{\mathcal{H}})$ contains all the degenerate distributions that assign probability 1 to a single distribution in $\Delta_{\mathcal{H}}$. We now choose topologies in which the assumptions of Sion's theorem (theorem 26) are satisfied:

- $\mathcal{A}$ is convex, and it is compact in $\mathbb{R}^k$ for a finite k. Every randomized algorithm can be identified with a conditional probability function $p(h|s)$. For each realizable sample with length in M, the algorithm assigns a point in $\Delta(\mathcal{H})$ (not to be confused with $\Delta_{\mathcal{H}}$). Thus, the set $\mathcal{A}$ of all algorithms is the product of t simplices, each of finite dimension $|\mathcal{H}|$, where t is the number of such realizable samples. We conclude that $\mathcal{A}$ is a compact and convex subset of $\mathbb{R}^k$, for $k = t \cdot |\mathcal{H}|$. It will be convenient to view $\mathbb{R}^k$ as the metric space induced by the ℓ_1 norm.

- $\Delta(\Delta_{\mathcal{H}})$ is convex. This is immediate, seeing that if $D_1, D_2 \in \Delta(\Delta_{\mathcal{H}})$, then $|\text{supp}(D_1)|, |\text{supp}(D_2)| < \infty$ and therefore
$$\left|\text{supp}\Big(\lambda D_1 + (1-\lambda)D_2\Big)\right| < \infty$$
Topologically, we view $\Delta(\Delta_{\mathcal{H}})$ as a metric space with the metric induced by the ℓ_1 norm.

- The function $f(A,D) = \mathop{\mathbb{E}}_{p\sim D}\mathop{\mathbb{E}}_{m\sim\mu}\mathop{\mathrm{I}}_{S\sim p^m}(A(S);S)$ is continuous with respect to the product topology induced on the domain. We view the domain as the metric space induced by the ℓ_1-norm product metric (which induces the product topology). Fix some $D_0 \in \Delta(\Delta_{\mathcal{H}})$, $A_0 = p_0(h|s) \in \mathcal{A}$ and $\varepsilon > 0$. We will find a value $\delta > 0$ such that
$$\|(A,D) - (A_0,D_0)\|_1 < \delta \implies |f(A,D) - f(A_0,D_0)| \le \varepsilon$$
Consider $g : \mathcal{A} \times \Delta_{\mathcal{H}} \to \mathbb{R}$ as follows:
$$g(A,q) = \mathop{\mathbb{E}}_{m\sim\mu}\mathop{\mathrm{I}}_{S\sim q^m}(A(S);S) = \mathop{\mathbb{E}}_{m\sim\mu}\sum_{s\in\mathcal{S}^m} q^m(s)\sum_{h\in\mathcal{H}} p(h|s)\log\frac{p(h|s)}{\sum_{s\in\mathcal{S}^m} p(h|s)q^m(s)}$$
clearly, g is continuous with respect to (p,q), and because $\mathcal{A}\times\Delta_{\mathcal{H}}$ is compact, g is uniformly continuous (per the Heine–Cantor theorem). Take $\delta' > 0$ such that
$$\|(A_1,q_1) - (A_2,q_2)\|_1 < \delta' \implies |g(A_1,q_1) - g(A_2,q_2)| < \frac{\varepsilon}{2}$$
Now, taking $\delta = \min\{\delta', \frac{\varepsilon}{2\log|\mathcal{H}|}\}$ we obtain
$$|f(A,D) - f(A_0,D_0)| \le |f(A,D) - f(A_0,D)| + |f(A_0,D) - f(A_0,D_0)| =$$
$$= \left|\mathop{\mathbb{E}}_{p\sim D}(g(A,p) - g(A_0,p))\right| + \left|\sum_p (D(p) - D_0(p))\, g(A,p)\right| \le$$
$$\le \mathop{\mathbb{E}}_{p\sim D}|g(A,p) - g(A_0,p)| + \log|\mathcal{H}|\sum_p |D(p) - D_0(p)| \le$$
$$\le \frac{\varepsilon}{2} + \log|\mathcal{H}| \cdot \frac{\varepsilon}{2\log|\mathcal{H}|} \le \varepsilon$$
as desired.

- The function $f(A,D) = \mathop{\mathbb{E}}_{p\sim D}\mathop{\mathbb{E}}_{m\sim\mu}\mathop{\mathrm{I}}_{S\sim p^m}(A(S);S)$ is convex-concave

– f is convex in A (for fixed D). This follows from Lemma 27 where we take X to be S and Y to be $A(S)$. We can identify the set of algorithms with the set of conditional probabilities $p(y|x)$. The lemma tells us that for each p in $\mathrm{supp}(D)$, the mutual information is convex, which entails that the expectation is also convex.
– f is concave in D (for fixed A). In fact f is linear in D, from the linearity of expectation.

Thus, the assumptions for Sion's minimax theorem hold, and we obtain that

$$\inf_{A\in\mathcal{A}} \sup_{D\in\Delta(\Delta_{\mathcal{H}})} \mathop{\mathbb{E}}_{p\sim D} \mathop{\mathbb{E}}_{m\sim\mu} \mathop{\mathrm{I}}_{S\sim p^m}(A_m(S);S) = \sup_{D\in\Delta(\Delta_{\mathcal{H}})} \inf_{A\in\mathcal{A}} \mathop{\mathbb{E}}_{p\sim D} \mathop{\mathbb{E}}_{m\sim\mu} \mathop{\mathrm{I}}_{S\sim p^m}(A_m(S);S)$$

as desired. ■

Proof [Lemma 21] We define L to be a consistent randomized learning algorithm for $\mathcal{T}_n$ that accepts samples of size 2 as follows. Let E denote the event in which L receives a sample S of the form $\big((i,0),(j,1)\big)$ or $\big((j,1),(i,0)\big)$, i.e. S contains precisely one example i that is labeled with 0 and one example j that is labeled 1.

- If E occurs, then L samples an integer m from μ, samples a permutation σ uniformly from all permutations on m elements, and returns the hypothesis

$$A_m\Big(\sigma\big(\underbrace{(i,0),\ldots,(i,0)}_{\text{m-1}},(j,1)\big)\Big)$$

- Otherwise, L returns some arbitrary hypothesis that is consistent with S.

from the proof of Lemma 11, there exists a distribution $q\in\Delta_{\mathcal{T}_n}$ such that

$$\mathop{\mathrm{I}}_{S\sim q^2}(S;L(S)\,|\,1_E=1)=\Omega(\log n)$$

and we can assume without loss of generality that $(S|1_E=1)$ is such that the value of i is fixed and j is distributed uniformly over some set of size $\Omega(n)$. We use f to denote the mapping

$$(j,m,\sigma)\longmapsto\sigma\big(\underbrace{(i,0),\ldots,(i,0)}_{\text{m-1}},(j,1)\big)$$

J to denote the value of j in S, U_J for the uniform distribution on the set of values for j, $U_{\Sigma,M}$ for the uniform distribution on the M orderings of a sample of this form with M elements, and we use B to denote a bit indicating whether j appeared first or second in S. We now may write

$$\begin{aligned}
\mathop{\mathrm{I}}_{S\sim q^2}(S;L(S)\,|\,1_E=1) &= \mathop{\mathrm{I}}_{S\sim q^2}(J,B;L(S)\,|\,1_E=1)\\
&\le \mathop{\mathrm{I}}_{S\sim q^2}(J;L(S)\,|\,1_E=1)+1\\
&= \mathop{\mathrm{I}}_{S\sim q^2\,|\,1_E=1}(J;L(S))+1\\
&= \mathop{\mathrm{I}}_{\substack{M\sim\mu\\ J\sim U_J\\ \Sigma\sim U_{\Sigma,M}}}(J;A_M(f(J,M,\Sigma)))+1
\end{aligned}$$

where the inequality follows from the chain rule. It holds that

$$\begin{aligned}
\underset{\substack{M\sim\mu\\ J\sim U_J\\ \Sigma\sim U_{\Sigma,M}}}{\mathrm{I}}(J;A_M(f(J,M,\Sigma))) &\overset{(a)}{\leq} \mathrm{I}(J;A_M(f(J,M,\Sigma))\,|\,M)\\
&\overset{(b)}{\leq} \mathrm{I}(J,M,\Sigma;A_M(f(J,M,\Sigma))\,|\,M)\\
&\overset{(c)}{=} \mathrm{I}(f(J,M,\Sigma);A_M(f(J,M,\Sigma))\,|\,M)\\
&= \mathop{\mathbb{E}}_{m\sim\mu} \underset{\substack{J\sim U_J\\ \Sigma\sim U_{\Sigma,m}}}{\mathrm{I}}(f(J,M,\Sigma);A_M(f(J,M,\Sigma))\,|\,M=m)
\end{aligned}$$

which is justified by:

(a) From Lemma B.2.1, because $J\perp M$.

(b) The data processing inequality.

(c) Because f is a bijection (data processing inequality).

Recall that q is defined as follows, for $t=2$:

$$q(w)=\left(1-\frac{1}{t}\right)\mathbf{1}_{w=1}(w)+\frac{1}{2t}\mathbf{1}_{w=i}(w)+\frac{1}{2t}U_J(w)$$

Let p be the same distribution, but with $t=b$. Then:

$$\mathop{\mathbb{E}}_{m\sim\mu} \underset{\substack{J\sim U_J\\ \Sigma\sim U_{\Sigma,m}}}{\mathrm{I}}(f(J,M,\Sigma);A_m(f(J,M,\Sigma))\,|\,M=m)$$

$$\begin{aligned}
&\overset{(a)}{=} \mathop{\mathbb{E}}_{m\sim\mu} \underset{S\sim p^m}{\mathrm{I}}(S;A_m(S)\,|\,M=m,1_{E_m}=1)\\
&\overset{(b)}{\leq} \mathop{\mathbb{E}}_{m\sim\mu} \frac{1}{\Pr(1_{E_m}=1)} \underset{S\sim p^m}{\mathrm{I}}(S;A_m(S)\,|\,M=m,1_{E_m})\\
&\leq \left(\max_m \frac{1}{\Pr(1_{E_m}=1)}\right) \mathop{\mathbb{E}}_{m\sim\mu} \underset{S\sim p^m}{\mathrm{I}}(S;A_m(S)\,|\,M=m,1_{E_m})\\
&\overset{(c)}{\leq} 16e\left(\frac{b}{a}\right)^2 \mathop{\mathbb{E}}_{m\sim\mu} \underset{S\sim p^m}{\mathrm{I}}(S;A_m(S)\,|\,M=m,1_{E_m})\\
&\overset{(d)}{\leq} 16e\left(\frac{b}{a}\right)^2 \left(\mathop{\mathbb{E}}_{m\sim\mu} \underset{S\sim p^m}{\mathrm{I}}(S;A_m(S)\,|\,M=m)+1\right)
\end{aligned}$$

where E_m is the event that the sample S is a permutation of

$$\Big(\underbrace{(1,0),\ldots,(1,0)}_{\text{m-2}},(i,0),(j,1)\Big)$$

for some $i<j$, and the justifications are:

(a) $(S|1_{E_m}=1)$ and $f(J,M,\Sigma)$ have the same distribution.

(b) From the definition of conditional mutual information.

(c) From the construction of p, we have for $m \in [a,b]$ that

$$\Pr(1_{E_m}=1) = \left(1-\frac{1}{b}\right)^{m-2}\left(\frac{1}{2b}\right)^2 m(m-1) \geq$$

$$\geq \left(1-\frac{1}{b}\right)^{b-2}\left(\frac{1}{2b}\right)^2 a(a-1) \geq \frac{1}{16e}\left(\frac{a}{b}\right)^2$$

(d) The chain rule for mutual information.

Finally, chaining all the above inequalities together yields

$$\mathop{\mathbb{E}}_{m\sim\mu} \mathop{\mathrm{I}}_{S\sim p^m}(S;A_m(S)) \geq \left(\frac{a}{b}\right)^2 \frac{\Omega(\log n)-1}{16e} - 1 = \Omega\left(\left(\frac{a}{b}\right)^2 \log n\right)$$

as desired. ■

Appendix B. Miscellaneous

Claim B.1 *Assume dm integers are sampled i.i.d. from the uniform distribution on $[d]$, and let Z_i denote the number of times the integer $i \in [d]$ was sampled. Then*

$$\Pr\left(\frac{m}{2} \leq Z_i \leq \frac{3m}{2}\right) \geq 1 - 2e^{-\frac{m}{2d}}$$

Proof [Claim B.1] Let X_t be an indicator denoting whether the t-th integer sampled was i.

$$\begin{aligned}
\Pr\left(\frac{m}{2} \leq Z_i \leq \frac{3m}{2}\right) &= 1 - \Pr\left(\left|\sum_{t=1}^{dm} X_t - \mathbb{E}\sum_{t=1}^{dm} X_t\right| > \frac{m}{2}\right) \\
&= 1 - \Pr\left(\left|\sum_{t=1}^{dm} X_t - m\right| > \frac{m}{2}\right) \\
&= 1 - \Pr\left(\left|\frac{1}{dm}\sum_{t=1}^{dm} X_t - \frac{1}{d}\right| > \frac{1}{2d}\right)
\end{aligned}$$

And from Hoeffding's inequality (lemma B.6 in Shalev-Shwartz and Ben-David, 2014)

$$\Pr\left(\left|\frac{1}{dm}\sum_{t=1}^{dm} X_t - \frac{1}{d}\right| > \frac{1}{2d}\right) \leq 2e^{-2dm\left(\frac{1}{2d}\right)^2} = 2e^{-\frac{m}{2d}}$$

as desired. ■ ■

Claim B.2 *Let X,Y,Z be random variables.*

1. *If* $X \perp Z$ *then* $\mathrm{I}(X;Y) \leq \mathrm{I}(X;Y|Z)$.
2. *If* $X \perp Z|Y$ *then* $\mathrm{I}(X;Y) \geq \mathrm{I}(X;Y|Z)$.

Proof [Claim B.2]
For 1:

$$\begin{aligned}\mathrm{I}(X;Y|Z) &= \mathrm{H}(X|Z) - \mathrm{H}(X|Y,Z) \\ &\overset{(*)}{=} \mathrm{H}(X) - \mathrm{H}(X|Y,Z) \\ &\geq \mathrm{H}(X) - \mathrm{H}(X|Y) \\ &= \mathrm{I}(X;Y)\end{aligned}$$

For 2:

$$\begin{aligned}\mathrm{I}(X;Y|Z) &= \mathrm{H}(X|Z) - \mathrm{H}(X|Y,Z) \\ &\leq \mathrm{H}(X) - \mathrm{H}(X|Y,Z) \\ &\overset{(*)}{=} \mathrm{H}(X) - \mathrm{H}(X|Y) \\ &= \mathrm{I}(X;Y)\end{aligned}$$

where $(*)$ follow from the assumptions. ■ ■

Lemma 24 *Let* X, Y *be random variables and let*

$$\mathrm{I}(X;Y) = \sum_{x,y} p_{XY}(x,y) \log \frac{p_{XY}(x,y)}{p_X(x)p_Y(y)} =$$

$$= \sum_{S^+} f(x,y) + \sum_{S^-} f(x,y)$$

be their mutual information, where

$$f(x,y) = p_{XY}(x,y) \log \frac{p_{XY}(x,y)}{p_X(x)p_Y(y)}$$

$$S^+ = \{(x,y) : f(x,y) \geq 0\};\ S^- = \{(x,y) : f(x,y) < 0\}$$

Then

$$\sum_{S^-} f(x,y) \geq -1$$

Proof [Lemma 24].

$$\sum_{S^-} f(x,y) = \sum_{S^-} p_{XY}(x,y) \log \frac{p_{XY}(x,y)}{p_X(x)p_Y(y)} \geq \left(\sum_{S^-} p_{XY}(x,y) \right) \log \frac{\sum_{S^-} p_{XY}(x,y)}{\sum_{S^-} p_X(x)p_Y(y)} =$$

$$= p_{XY}(S^-) \log \frac{p_{XY}(S^-)}{\sum_{S^-} p_X(x)p_Y(y)} = p_{XY}(S^-) \left(\log p_{XY}(S^-) \ - \ \log \sum_{S^-} p_X(x)p_Y(y) \right) \geq$$

$$\geq p_{XY}(S^-)\log p_{XY}(S^-) \geq \min_{x\in[0,1]} x\log x = -\frac{1}{\mathrm{e}} \geq -1$$

Where the the first inequality is the log-sum inequality, and the second inequality holds because $\sum_{S^-} p_X(x)p_Y(y) \leq 1$. ■

Theorem 25 (Minimax, Von Neumann 1928; Von Neumann and Morgenstern 1944) *Let $X \subseteq \mathbb{R}^n$, $Y \subseteq \mathbb{R}^m$ be compact convex sets. If $f : X \times Y \to \mathbb{R}$ is a continuous function that is convex-concave, i.e.,*

- *$f(\cdot, y) : X \to \mathbb{R}$ is convex for fixed $y \in Y$, and*
- *$f(x, \cdot) : Y \to \mathbb{R}$ is concave for fixed $x \in X$*

then

$$\min_{x\in X}\max_{y\in Y} f(x,y) = \max_{y\in Y}\min_{x\in X} f(x,y)$$

Theorem 26 (Minimax, Sion 1958) *Let X, Y be convex sets, one of which is compact. If $f : X \times Y \to \mathbb{R}$ is quasi-convex-concave, i.e.,*

- *$f(\cdot, y) : X \to \mathbb{R}$ is quasi-convex for fixed $y \in Y$, and*
- *$f(x, \cdot) : Y \to \mathbb{R}$ is quasi-concave for fixed $x \in X$*

and f is upper-semi-continuous–lower-semi-continuous, i.e.,

- *$f(\cdot, y) : X \to \mathbb{R}$ is upper-semi-continuous for fixed $y \in Y$, and*
- *$f(x, \cdot) : Y \to \mathbb{R}$ is lower-semi-continuous for fixed $x \in X$*

then

$$\sup_{x\in X}\inf_{y\in Y} f(x,y) = \sup_{y\in Y}\inf_{x\in X} f(x,y)$$

Lemma 27 (Theorem 2.7.4 in Cover and Thomas 2006) *Let $(X,Y) \sim p(x,y) = p(x)p(y|x)$. The mutual information $\mathrm{I}(X;Y)$ is a concave function of $p(x)$ for fixed $p(y|x)$ and a convex function of $p(y|x)$ for fixed $p(x)$.*