

Learners that Use Little Information

Raef Bassily

BASSILY.1@OSU.EDU

*Department of Computer Science and Engineering
The Ohio State University
Columbus, OH*

Shay Moran

SHAYMORAN1@GMAIL.COM

*School of Mathematics
Institute for Advanced Study
Princeton, NJ*

Ido Nachum

IDON@TX.TECHNION.AC.IL

*Department of Mathematics
Technion-IIT
Haifa, Israel*

Jonathan Shafer

SHAFFERJO@BERKELEY.EDU

*Computer Science Division
University of California, Berkeley
Berkeley, CA*

Amir Yehudayoff

AMIR.YEHUDAYOFF@GMAIL.COM

*Department of Mathematics
Technion-IIT
Haifa, Israel*

Editor: ?

Abstract

We study learning algorithms that are restricted to using a small amount of information from their input sample. We introduce a category of learning algorithms we term *d-bit information learners*, which are algorithms whose output conveys at most d bits of information of their input. A central theme in this work is that such algorithms generalize.

We focus on the learning capacity of these algorithms, and prove sample complexity bounds with tight dependencies on the confidence and error parameters. We also observe connections with well studied notions such as sample compression schemes, Occam's razor, PAC-Bayes and differential privacy.

We discuss an approach that allows us to prove upper bounds on the amount of information that algorithms reveal about their inputs, and also provide a lower bound by showing a simple concept class for which every (possibly randomized) empirical risk minimizer must reveal a lot of information. On the other hand, we show that in the distribution-dependent setting every VC class has empirical risk minimizers that do not reveal a lot of information.

Keywords: PAC Learning, Information Theory, Compression, PAC-Bayes, Sample Compression Scheme, Occam’s Razor, Differential Privacy.

1. Introduction

The amount of information that an algorithm uses is a natural and important quantity to study. A central idea that this paper revolves around is that a learning algorithm that only uses a small amount of information from its input sample will generalize well. The amount of information used can be quantified by

$$I(\mathcal{A}(S); S)$$

which is the mutual information between the output of the algorithm $\mathcal{A}(S)$ and the input sample S . With this quantity we define a new class of learning algorithms termed *d-bit information learners*, which are learning algorithms in which the mutual information is at most d . This definition naturally combines notions from information theory and from learning theory (Littlestone and Warmuth, 1986; Moran and Yehudayoff, 2016). It also relates to privacy, because one can think of this definition as a bound on the information that an algorithm leaks or reveals about its potentially sensitive training data.

1.1 Results

The main contributions of the current work are as follows (for definitions see Section 2. For related work, see Section 1.2).

Low information yields generalization. Our work stems from the intuition that a learning algorithm that only uses a small amount of information from its input will generalize well. We formalize this intuition in Theorem 8, that roughly states that

$$\mathbb{P}[|\text{true error} - \text{empirical error}| > \varepsilon] = O\left(\frac{I(\mathcal{A}(S); S)}{m\varepsilon^2}\right),$$

where m is the number of examples in the input sample. We provide four different proofs of this statement, each of which emphasizes a different perspective of this phenomenon (see Section 4).

Sharpness of the sample complexity bound. Theorem 8 entails that to achieve an error of ε with confidence δ , it is sufficient to use

$$m = \Omega\left(\frac{I(\mathcal{A}(S); S)}{\varepsilon^2 \cdot \delta}\right)$$

examples. This differs from results in well-known settings such as learning hypothesis classes of finite VC dimension, where m only grows logarithmically with $1/\delta$. Nonetheless, we prove that this bound is sharp (Section 4.1). In particular, we show the existence of a learning problem and an $O(1)$ -bit information learner that has a true error of at least $\frac{1}{2}$ with probability of at least $\frac{1}{m}$, where m is the size of the input sample.

A lower bound for mutual information. In Section 5 we show that for the simple class of thresholds, every (possibly randomized) proper ERM must reveal at least

$$\Omega\left(\frac{\log \log N}{m^2}\right)$$

bits of information, where N is the size of the domain. This means that even in very simple settings, learning may not always be possible if we restrict the information used by the algorithm. However, this does not imply the non-existence of bounded information learners that are either non-consistent or non-proper, an issue we leave open for future work.

Upper bounds for mutual information. Section 6.1 provides a method for upper bounding the amount of information that algorithms reveal. We also define a generic learner $\mathcal{A}_{\mathcal{H}}$ for a concept class \mathcal{H} , and show that in a number of natural cases this algorithm conveys as little information as possible (up to some constant). This generic learner is proper and consistent (i.e. an ERM); it simply outputs a uniformly random hypothesis from the set of hypotheses that are consistent with the input sample. However, we show that in other simple cases, this algorithm has significantly higher mutual information than necessary.

The distribution-dependent setting. We also consider an alternative setting, in Section 6.3, in which the distribution over the domain is known to the learner. Here, for any concept class with finite VC-dimension d and for any distribution on the data domain there exists a learning algorithm that outputs with high probability an approximately correct function from the concept class, such that the mutual information between the input sample and the output is $O(d \log(m))$. In contrast with the abovementioned lower bound, the information here does not grow with the size of the domain.

Contrast with pure differential privacy. Corollary 12 provides a separation between differential privacy and bounded mutual information. For the class of point functions \mathcal{PF} , it is known that any pure differentially private algorithm that properly learns this class must require a number of examples that grows with the domain size (Beimel et. al., 2010). On the other hand, we show that the generic ERM learner $\mathcal{A}_{\mathcal{PF}}$ leaks at most 2 bits of information and properly learns this class with optimal PAC-learning sample complexity.

1.2 Related Work

Sample compression schemes. d -bit information learners resemble the notion of sample compression schemes (Littlestone and Warmuth, 1986). Sample compression schemes correspond to learning algorithms whose output hypothesis is determined by a small subsample of the input. For example, support vector machines output a separating hyperplane that is determined by a small number of support vectors.

Both sample compression schemes and information learners quantify (in different ways) the property of limited dependence between the output hypothesis and the input sample. It is therefore natural to ask how these two notions relate to each other.

It turns out that not every sample compression scheme of constant size also leaks a constant number of bits. Indeed, in Section 5 it is shown that there is no empirical risk minimizer (ERM) for thresholds that is an $O(1)$ -bits information learner.¹ On the other hand, there is an ERM for this class that is based on a sample compression scheme of size $O(1)$.

Occam’s razor. Theorem 8 extends the classical Occam’s razor generalization bound (Blumer et al., 1987), which states the following: Assume a fixed encoding of hypotheses in \mathcal{H} by bit strings. The *complexity* of a hypothesis is the bit-length of its encoding. A learning algorithm for \mathcal{H} is called an *Occam-algorithm with parameters c, α* if for every realizable sample of size m it produces a consistent hypothesis of complexity at most $n^c m^\alpha$, where n is the complexity of some hypothesis in \mathcal{H} that is consistent with the sample.

Theorem 1 (Blumer et al. 1987) *Let \mathcal{A} be an Occam-algorithm with parameters $c \geq 1$ and $0 \leq \alpha < 1$. Let \mathcal{D} be a realizable distribution, let $f \in \mathcal{H}$ be such that $\text{err}(f; \mathcal{D}) = 0$, and let n denote the complexity of f . Then,*

$$\Pr_{S \sim \mathcal{D}^m} [\text{err}(\mathcal{A}(S); \mathcal{D}) \geq \varepsilon] \leq \delta,$$

as long as m is at least $\Omega\left(\frac{\log(\frac{1}{\delta})}{\varepsilon} + \left(\frac{n^c}{\varepsilon}\right)^{1/(1-\alpha)}\right)$.

To relate Occam’s razor to Theorem 8, observe that an Occam-algorithm is in particular a $O(n^c m^\alpha)$ -bit information learner (since its output hypothesis is encoded by $O(n^c m^\alpha)$ bits), which implies that the probability of it outputting a function with true error more than ε is at most $O\left(\frac{n^c m^\alpha}{m \varepsilon^2}\right)$. The bound can be improved by standard confidence-boosting techniques (see Appendix I).

Mutual information for controlling bias in statistical analysis. The connection between mutual information and statistical bias has been recently studied in Russo and Zhou (2016) in the context of *adaptive* data analysis. In adaptive statistical analysis, the analyst conducts a sequence of analysis steps, where the choice and structure of each step depends adaptively on the outcomes of the previous ones. Some of the results of Russo and Zhou (2016) have been recently improved by Raginsky and Xu (2017).

Differential privacy and generalization. Differential privacy, introduced by Dwork et al. (2006), is a rigorous notion of privacy enabling a strong guarantee that data holders may provide to their sources. *Pure*² differential privacy implies a bound on mutual information (McGregor et al., 2010).

The role of differential privacy in controlling overfitting has been recently studied in several works (e.g. Dwork et al., 2015; Bassily et al., 2016; Rogers et al., 2016; Bassily et al., 2014). The authors of Bassily et al. (2016) provide a treatment of differential privacy as a notion of distributional stability, and a tight characterization of the generalization guarantees of differential privacy.

¹Here and below $O(\cdot)$, $\Omega(\cdot)$ and $\Theta(\cdot)$ mean up to some multiplicative universal constants.

²As opposed to a more relaxed notion known as *approximate* differential privacy (see Section 2.3 for a precise definition).

Max-information and approximate max-information: Dwork et. al. (2015) introduced and studied the notions of max-information – a stronger notion than mutual information – and its relaxation, approximate max-information.³ They showed that these notions imply generalization and that pure differentially private algorithms exhibit low (approximate) max-information. Rogers et. al. (2016) showed that approximate differentially private algorithms also have low approximate max-information, and that the notion of approximate max-information captures the generalization properties (albeit with slightly worse parameters) of differentially private algorithms (pure or approximate).

Connections to approximate differential privacy: De (2012) has shown that the relaxed notion of *approximate* differential privacy does not necessarily imply bounded mutual information. In McGregor et. al. (2010), it was also shown that if the dataset entries are independent, then approximate differential privacy implies a (weak) bound on the mutual information. Such a bound has an explicit dependence on the domain size, which restricts its applicability in general settings. Unlike the case of pure differential privacy, an exact characterization of the relationship between mutual information and approximate differential privacy algorithms is not fully known even when the dataset distribution is i.i.d.

Bun et. al. (2015) showed that the sample complexity of properly learning thresholds (in one dimension) under approximate differential privacy is $\Omega(\log^*(N))$, where N is the domain size. Hence, their result asserts the impossibility of this task for infinite domains. In this work, we show a result of a similar flavor (albeit of a weaker implication) for the class of bounded information learners. Specifically, for the problem of proper PAC-learning of thresholds over a domain of size N , we show that the mutual information of any proper learning algorithm (deterministic or randomized) that outputs a threshold that is *consistent* with the input sample is $\Omega(\log \log N)$. This result implies that there are no consistent proper bounded information learners for thresholds over infinite domains.

2. Preliminaries

2.1 Learning

We start with some basic terminology from statistical learning (for a textbook see Shalev-Shwartz and Ben-David 2014). Let \mathcal{X} be a set called *the domain*, $\mathcal{Y} = \{0, 1\}$ be the *label-set*, and $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ be the *examples domain*. A *sample* $S = ((x_1, y_1), \dots, (x_m, y_m)) \in \mathcal{Z}^m$ is a sequence of examples. A function $h : \mathcal{X} \rightarrow \mathcal{Y}$ is called a *hypothesis* or a *concept*.

Let \mathcal{D} be a distribution over \mathcal{Z} . The *error* of a hypothesis h with respect to \mathcal{D} is defined by $\text{err}(h; \mathcal{D}) = \mathbb{E}_{(x,y) \sim \mathcal{D}} \mathbb{1}[h(x) \neq y]$. Let $S = ((x_1, y_1), \dots, (x_m, y_m))$ be a sample. The *empirical error* of h with respect to S is defined by $\widehat{\text{err}}(h; S) = \frac{1}{m} \sum_{i=1}^m \mathbb{1}[h(x_i) \neq y_i]$.

A *hypothesis class* \mathcal{H} is a set of hypotheses. A distribution \mathcal{D} is *realizable* by \mathcal{H} if there is $h \in \mathcal{H}$ with $\text{err}(h; \mathcal{D}) = 0$. A sample S is *realizable* by \mathcal{H} if there is $h \in \mathcal{H}$ with $\widehat{\text{err}}(h; S) = 0$.

³Unlike max-information, the relaxed notion of approximate max-information is not directly related to mutual information; that is, boundedness of one does not necessarily imply the same for the other.

A *learning algorithm*, or a *learner* is a (possibly randomized) algorithm \mathcal{A} that takes a sample S as input and outputs a hypothesis, denoted by $\mathcal{A}(S)$. We say that \mathcal{A} *learns*⁴ \mathcal{H} if for every $\varepsilon, \delta > 0$ there is a finite bound $m = m(\varepsilon, \delta)$ such that for every \mathcal{H} -realizable distribution \mathcal{D} ,

$$\Pr_{S \sim \mathcal{D}^m} [\text{err}(\mathcal{A}(S); \mathcal{D}) \geq \varepsilon] \leq \delta.$$

ε is called the *error* parameter, and δ the *confidence* parameter. \mathcal{A} is called *proper* if $\mathcal{A}(S) \in \mathcal{H}$ for every realizable S , and it is called *consistent* if $\widehat{\text{err}}(\mathcal{A}(S); S) = 0$ for every realizable S .

2.2 Information Theoretic Measures

Information theory studies the quantification and communication of information. In this work, we use the language of learning theory combined with information theory to define and study a new type of learning theoretic compression. Here are standard notions from information theory (for more background see the textbook [Cover and Thomas 2006](#)).

Let Y and Z be two discrete random variables. The entropy of Y measures the number of bits required to encode Y on average.

Definition 2 (Entropy) *The entropy of Y is defined as*

$$H(Y) = - \sum_y \Pr(Y = y) \log(\Pr(Y = y)),$$

where $\log = \log_2$ and by convention $0 \log 0 = 0$.

The mutual information between Y and Z is (roughly speaking) a measure for the amount of random bits Y and Z share on average. It is also a measure of their independence; for example $I(Y; Z) = 0$ iff Y and Z are independent.

Definition 3 (Mutual information) *The mutual information between Y and Z is defined to be*

$$I(Y; Z) = H(Y) + H(Z) - H(Y, Z).$$

The Kullback-Leibler divergence between two measures μ and ν is a useful measure for the “distance” between them (it is not a metric and may be infinite).

Definition 4 (KL-divergence) *The KL-divergence between two measures μ and ν on \mathcal{X} is*

$$\text{KL}(\mu || \nu) = \sum_x \mu(x) \log \frac{\mu(x)}{\nu(x)}$$

where $0 \log \frac{0}{0} = 0$.

Mutual information can be written as the following KL-divergence:

$$I(Y; Z) = \text{KL}(p_{Y,Z} || p_Y \cdot p_Z),$$

where $p_{Y,Z}$ is the joint distribution of the pair (Y, Z) , and $p_Y \cdot p_Z$ is the product of the marginals p_Y and p_Z .

⁴In this paper we focus on learning in the realizable case.

2.3 Differential Privacy

Differential privacy (Dwork et. al., 2006) is a standard notion for statistical data privacy. Despite the connotation perceived by the name, differential privacy is a distributional stability condition that is imposed on an algorithm performing analysis on a dataset. Algorithms satisfying this condition are known as *differentially private* algorithms. There is a vast literature on the properties of this class of algorithms and their design and structure (see e.g., Dwork and Roth, 2014, for an in-depth treatment).

Definition 5 (Differential privacy) *Let \mathcal{X}, \mathcal{Z} be two sets, and let $m \in \mathbb{N}$. Let $\alpha > 0, \beta \in [0, 1)$. An algorithm $\mathcal{A} : \mathcal{X}^m \rightarrow \mathcal{Z}$ is said to be (α, β) -differentially private if for all datasets $S, S' \in \mathcal{X}^m$ that differ in exactly one entry, and all measurable subsets $\mathcal{O} \subseteq \mathcal{Z}$, we have*

$$\mathbb{P}_{\mathcal{A}}[\mathcal{A}(S) \in \mathcal{O}] \leq e^{\alpha} \mathbb{P}_{\mathcal{A}}[\mathcal{A}(S') \in \mathcal{O}] + \beta$$

where the probability is taken over the random coins of \mathcal{A} .

When $\beta = 0$, the condition is sometimes referred to as pure differential privacy (as opposed to approximate differential privacy when $\beta > 0$.)

The general form of differential privacy entails two parameters: α which is typically a small constant and β which in most applications is of the form $\beta = o(1/m)$.

Differential privacy has been shown to provide non-trivial generalization guarantees especially in the adaptive settings of statistical analyses (see e.g., Dwork et. al., 2015; Bassily et. al., 2016; Dwork et. al., 2015). In the context of (agnostic) PAC-learning, there has been a long line of work (e.g. Kasiviswanathan et. al., 2008; Beimel et. al., 2010, 2013; Feldman and Xiao, 2014; Bun et. al., 2015) that studied differentially private learning and the characterization of the sample complexity of private learning in several settings. However, the picture of differentially private learning is very far from complete and there are still so many open questions. Vadhan (2017) gives a good survey on the subject.

3. d -Bit Information Learners

Here we define learners that use little information from their input.⁵ We start by setting some notation. Let \mathcal{A} be a (possibly randomized) learning algorithm. For every sample $S \in (\mathcal{X} \times \{0, 1\})^m$, let $P_{h|S}(\cdot)$ denote the conditional distribution function of the output of the algorithm \mathcal{A} given that its input is S . When \mathcal{A} is deterministic, $P_{h|S}$ is a degenerate distribution. For a fixed distribution \mathcal{D} over examples and $m \in \mathbb{N}$, let $P_h(\cdot)$ denote the marginal distribution of the output of \mathcal{A} when it takes an input sample of size m drawn i.i.d. from \mathcal{D} , i.e. $P_h(f) = \mathbb{E}_{S \sim \mathcal{D}^m} [P_{h|S}(f)]$ for every function f .

⁵In this text we focus on Shannon's mutual information, but other notions of divergence may be interesting to investigate as well.

Definition 6 (Mutual information of an algorithm) We say that \mathcal{A} has mutual information of at most d bits (for sample size m) with respect to a distribution \mathcal{D} if

$$I(S; \mathcal{A}(S)) \leq d$$

where $S \sim \mathcal{D}^m$.

Definition 7 (d -bit information learner) A learning algorithm \mathcal{A} for \mathcal{H} is called a d -bit information learner if it has mutual information of at most d bits with respect to every realizable distribution (d can depend on the sample size).

3.1 Bounded Information Implies Generalization

The following theorem quantifies the generalization guarantees of d -bit information learners.

Theorem 8 Let \mathcal{A} be a learner that has mutual information of at most d bits with a distribution \mathcal{D} , and let $S \sim \mathcal{D}^m$. Then, for every $\varepsilon > 0$,

$$\mathbb{P}_{\mathcal{A}, S} [|\widehat{\text{err}}(\mathcal{A}(S); S) - \text{err}(\mathcal{A}(S); \mathcal{D})| > \varepsilon] < \frac{d+1}{2m\varepsilon^2 - 1}$$

where the probability is taken over the randomness in the sample S and the randomness of \mathcal{A} .

Theorem 8 states a simple and basic property, and is proved in section 4 (below we provide a proof sketch for deterministic algorithms).

In particular, if a class \mathcal{H} admits a $o(m) = d$ -bit information learner then the class is PAC learnable. Also, some of the proofs will go through for multi-class classification with every bounded loss function.

The fact that the sample complexity bound that follows from the theorem is sharp is proved in Section 4.1. We mention that the dependence on ε can be improved in the realizable case; if the algorithm always outputs a hypothesis with empirical error 0 then the bound on the right hand side can be replaced by $O\left(\frac{d+1}{m\varepsilon-1}\right)$. As in similar cases, the reason for this difference stems from the fact that estimating the bias of a coin up to an additive error ε requires $m \approx \frac{1}{\varepsilon^2}$ samples, but if the coin falls on heads with probability ε then the chance of seeing m tails in a row is $(1-\varepsilon)^m \approx e^{-m\varepsilon}$.

PROOF SKETCH FOR DETERMINISTIC ALGORITHMS

Here we sketch a proof of Theorem 8 for deterministic algorithms. When \mathcal{A} is deterministic, we have

$$I = I(S; \mathcal{A}(S)) = H(\mathcal{A}(S)).$$

Let P_h denote the distribution of $\mathcal{A}(S)$. Let \mathcal{H}_0 be the set of hypotheses f so that $P_h(f) \geq 2^{-I/\delta}$. By Markov's inequality, $P_h(\mathcal{H}_0) \geq 1 - \delta$. In addition, the size of \mathcal{H}_0 is at most $2^{I/\delta}$. So Chernoff's inequality and the union bound imply that for every $f \in \mathcal{H}_0$ the empirical error is close to the true error for $m \approx \frac{I}{\varepsilon^2\delta}$ (with probability at least $1 - \delta$).

4. Proofs that Bounded Information Implies Generalization

In this paper, we prove the statement in Theorem 8 via different approaches (some of the arguments are only sketched). We provide four different proofs of this statement, each of which highlights a different general idea.

The first proof is based on an information theoretic lemma, which roughly states that if the KL-divergence between two measures μ and ν is small then $\mu(E)$ is not much larger than $\nu(E)$ for every event E . The nature of this proof strongly resembles the proof of the PAC-Bayes bounds (Shalev-Shwartz and Ben-David, 2014), and indeed a close variant of the theorem can be derived from these standard bounds as well (see proof IV). The second proof is based on a method to efficiently “de-correlate” two random variables in terms of their mutual information; roughly speaking, this implies that an algorithm of low mutual information can only generate a small number of hypotheses and hence does not overfit. The third proof highlights an important connection between low mutual information and the stability of a learning algorithm. The last proof uses the PAC-Bayes framework. See appendices A.1, A.2, A.3 and A.4 for the proofs.

4.1 The Sample Complexity Bound is Sharp

Standard bounds on the sample complexity of learning hypotheses classes of VC dimension d imply that to achieve a fixed confidence $\delta \leq 1/2$ one must use at least

$$m = \Omega\left(\frac{d + \log(1/\delta)}{\varepsilon^2}\right)$$

examples in the non-realizable case (see e.g., Shalev-Shwartz and Ben-David, 2014, theorem 6.7), and this bound is sharp.

In contrast, Theorem 8 above states that achieving confidence δ requires

$$m = \Omega\left(\frac{d}{\varepsilon^2} \cdot \frac{1}{\delta}\right)$$

examples, where in this case d is the bound on $I(\mathcal{A}(S); S)$. A natural question to ask is whether this sample complexity bound is also sharp. We now show that indeed it is.

To see that the bound is tight for d and ε , consider the case where $\mathcal{X} = [d]$ and $\mathcal{H} = \{0, 1\}^{\mathcal{X}}$. For any learner \mathcal{A} it holds that

$$I(\mathcal{A}(S); S) \leq H(\mathcal{A}(S)) \leq \log |\mathcal{H}| = d$$

However, the VC dimension of \mathcal{H} is also d . Because the bound for VC dimension is always sharp and it equals the bound from Theorem 8, it follows that that bound is also sharp in this case.

To see that the bound is sharp in δ as well, follow appendix B.

5. A Lower Bound on Information

In this section we show that any proper consistent learner for the class of thresholds cannot use only little information with respect to all realizable distributions \mathcal{D} . Namely, we find for every such algorithm \mathcal{A} a realizable distribution \mathcal{D} so that $I(S; \mathcal{A}(S))$ is large.

Let $\mathcal{X} = [2^n]$ and let $\mathcal{T} \subseteq \{0, 1\}^{\mathcal{X}}$ be the set of all thresholds; that is $\mathcal{T} = \{f_k\}_{k \in [2^n]}$ where

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

Theorem 9 *For any consistent and proper learning algorithm \mathcal{A} for \mathcal{T} with sample size m there exists a realizable distribution $\mathcal{D} = \mathcal{D}(\mathcal{A})$ so that*

$$I(S; \mathcal{A}(S)) = \Omega\left(\frac{\log n}{m^2}\right) = \Omega\left(\frac{\log \log |\mathcal{X}|}{m^2}\right),$$

where $S \sim \mathcal{D}^m$.

The high-level approach is to identify in \mathcal{A} a rich enough structure and use it to define the distribution \mathcal{D} . Part of the difficulty in implementing this approach is that we need to argue on a general algorithm, with no specific structure. A different aspect of the difficulty in defining \mathcal{D} stems from that we can not adaptively construct \mathcal{D} , we must choose it and then the algorithm gets to see many samples from it. Full details are in appendix C.

6. Approaches for Proving Upper Bounds on Information

Here we give a simple and generic information learner that provides sharp upper bounds for some basic concept classes. For example, we get an ERM that is a $O(\log \log N)$ -bit information learner for the class of thresholds over $[N]$; this bound is indeed tight given the lower bound in Section 5.

Definition 10 (Generic information learner) *The generic algorithm $\mathcal{A}_{\mathcal{H}}$ for a hypothesis class \mathcal{H} acts as follows. Given a realizable sample S , the algorithm first finds the set of all hypotheses in \mathcal{H} that are consistent with S , and then it simply outputs a uniformly random hypothesis from that set.*

This algorithm is well defined for finite \mathcal{H} and it is proper and consistent. Specifically, if \mathcal{H} has finite VC-dimension then $\mathcal{A}_{\mathcal{H}}$ PAC-learns the class with the standard sample complexity of VC classes (when there are no information constraints).

6.1 A Method for Upper Bounding Information

The following simple lemma provides a useful method for upper bounding mutual information. Let $\mathcal{H} \subseteq \{0, 1\}^{\mathcal{X}}$ and $m \in \mathbb{N}$. Let $\mathcal{A} : (\mathcal{X} \times \{0, 1\})^m \rightarrow \mathcal{H}$ be an algorithm that takes a sample S of m examples and outputs a hypothesis $h = \mathcal{A}(S) \in \mathcal{H}$. Recall that $P_{h|S}$ denotes the distribution of

$\mathcal{A}(S)$ when the input sample is S , and that $P_h(\cdot) = \mathbb{E}_{S \sim \mathcal{D}^m} [P_{h|S}(\cdot)]$ denotes the marginal distribution of the output of \mathcal{A} with respect to some fixed input distribution \mathcal{D} .

Lemma 11 *For every distribution \mathcal{Q} over \mathcal{H} , we have*

$$I(S; \mathcal{A}(S)) = \mathbb{E}_{S \sim \mathcal{D}^m} [\text{KL}(P_{h|S} \parallel P_h)] \leq \max_{S \in \text{Supp}(\mathcal{D}^m)} \text{KL}(P_{h|S} \parallel \mathcal{Q}).$$

Proof Observe that for any distribution \mathcal{Q} over \mathcal{H} ,

$$\begin{aligned} I(S; \mathcal{A}(S)) &= \mathbb{E}_{S \sim \mathcal{D}^m} [\text{KL}(P_{h|S} \parallel \mathcal{Q})] - \text{KL}(P_h \parallel \mathcal{Q}) \\ &\leq \mathbb{E}_{S \sim \mathcal{D}^m} [\text{KL}(P_{h|S} \parallel \mathcal{Q})] \\ &\leq \max_{S \in \text{Supp}(\mathcal{D}^m)} \text{KL}(P_{h|S} \parallel \mathcal{Q}) \end{aligned}$$

(if the KL-divergence is infinite, the upper bound trivially follows). ■

In appendix D, we use the above technique to upper bound the mutual information for the class of thresholds and point functions. Specifically: for every sample size, the generic algorithm $\mathcal{A} = \mathcal{A}_{\mathcal{T}}$ is a $(\log \log(N) + O(1))$ -bit information learner; for every realizable distribution \mathcal{D} , and for every sample size $m \leq N/2$, the generic algorithm $\mathcal{A} = \mathcal{A}_{\mathcal{P}\mathcal{F}}$ has at most 2 bits of mutual information with respect to \mathcal{D} .

SEPARATION BETWEEN DIFFERENTIAL PRIVACY AND BOUNDED INFORMATION

The above result, together with known properties of VC classes and (Beimel et. al., 2010, Corollary 1), implies a separation between the family of proper d -bit information learners and the family of pure differentially private proper learners.

Corollary 12 *There is a proper 2-bit information learner for the class of point functions over $[N]$ with sample size $O\left(\frac{\log(1/\delta)}{\varepsilon}\right)$. On the other hand, for any $\alpha > 0$, any α -differentially private algorithm that properly learns point functions over $[N]$ requires $\Omega\left(\frac{\log N + \log(1/\delta)}{\varepsilon^\alpha}\right)$ examples.*

6.2 The Generic Learner is Far from Optimal

We have seen that the generic algorithm provides sharp upper bounds on the mutual information for some cases. However, there are also some simple settings in which it reveals a lot more information than is necessary. Take the following class

$$\mathcal{H} = \{1_{x=i} : 1 < i \leq N\} \cup \{1_{x>1}\}$$

over the space $\mathcal{X} = [N]$ and the distribution with $P(x = 1) = 1 - \frac{1}{m}$ and $P(x = i) = \frac{1}{m(N-1)}$ for $i \neq 1$. The generic algorithm satisfies: $I(S; \mathcal{A}(S)) > \frac{\log N}{2e} - 1$. (full calculation in the appendix E).

In comparison, let us define a deterministic ERM with low information for this class. Define E to be the event where the sample is consistent with $1_{x>1}$. When E occurs, the algorithm outputs $1_{x>1}$. Otherwise, the algorithm outputs $1_{x=k}$ where k is the minimal integer such that $1_{x=k}$ is consistent with the sample. This algorithm satisfies:

$$I(S; \mathcal{A}(S)) = H(\mathcal{A}(S)) \leq H(\mathcal{A}(S), 1_E) = H(1_E) + H(\mathcal{A}(S)|1_E) \leq 1 + \log(m + 2).$$

6.3 The Distribution-Dependent Setting

As was shown in Section 5, there are classes of VC dimension 1 (thresholds on a domain of size N) for which every proper consistent learner must leak at least $\Omega(\log \log N)$ bits of information on some realizable distributions.

Here, we consider the distribution-dependent setting; that is, we assume the learner knows the marginal distribution $\mathcal{D}_{\mathcal{X}}$ on inputs (but it does not know the target concept). We show that in this setting every VC class can be learned with relatively low information.

Theorem 13 *Given the size m of the input sample, a distribution $\mathcal{D}_{\mathcal{X}}$ over \mathcal{X} and $\mathcal{H} \subset \{0, 1\}^{\mathcal{X}}$ with VC-dimension d , there exists a consistent, proper, and deterministic learner with $O(d \log(m+1))$ -bits of information (for \mathcal{H} -realizable samples).*

The proof of the theorem appears in appendix F. Now, we discuss a somewhat surprising phenomenon. In a nutshell, the theorem says that for every distribution there is a deterministic algorithm with small entropy. It is tempting to “conclude” using von Neumann’s minimax theorem that this implies that there is a randomized algorithm that for every distribution has small information. This “conclusion” however is false, as the threshold example shows.

Acknowledgments

AY was supported by ISF Grant No. 1162/15.

Bibliography

E. A. Arutyunyan. Bounds for the exponent of the probability of error for a semicontinuous memoryless channel. *Problems of Information Transmission* 4(4), pages 29-40, 1968.

R. Bassily, K. Nissim, A. Smith, U. Stemmer, and J. Ullman. Algorithmic stability for adaptive data analysis. In *STOC*, pages 1046–1059, 2016.

R. Bassily, A. Smith, and A. Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *FOCS*, pages 464–473, 2014.

A. Beimel, S. P. Kasiviswanathan, and K. Nissim. Bounds on the sample complexity for private learning and private data release. In *TCC*, pages 437–454, 2010.

- A. Beimel, K. Nissim, and U. Stemmer. Characterizing the sample complexity of private learners. In *ITCS*, pages 1–10, 2013.
- A. Blumer, A. Ehrenfeucht, D. Haussler, and M. Warmuth. Occam’s razor. *IPL* 24 377-380, 1987.
- M. Braverman and A. Garg. Public vs private coin in bounded-round information. In *ICALP* (1), pages 502–513, 2014.
- M. Bun, K. Nissim, U. Stemmer, and S. P. Vadhan. Differentially private release and learning of threshold functions. In *FOCS*, pages 17–20, 2015.
- T. Cover and J. A. Thomas. *Elements of information theory*. Wiley-Interscience New York, 2006.
- A. De. Lower bounds in differential privacy. In *TCC*, pages 321–338, 2012.
- C. Dwork, V. Feldman, M. Hardt, T. Pitassi, O. Reingold, and A. Roth. Generalization in adaptive data analysis and holdout reuse. In *NIPS*, 2015.
- C. Dwork, V. Feldman, M. Hardt, T. Pitassi, O. Reingold, and Aaron Roth. Preserving statistical validity in adaptive data analysis. In *STOC*, pages 117–126, 2015.
- C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, pages 265–284, 2006.
- C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.
- V. Feldman and D. Xiao. Sample complexity bounds on differentially private learning via communication complexity. In *COLT*, pages 1000–1019, 2014.
- P. Harsha, R. Jain, D. McAllester, and J. Radhakrishnan. The Communication Complexity of Correlation. *IEEE Trans. Information Theory* 56(1), pages 438-449, 2010.
- David Haussler. Sphere packing numbers for subsets of the boolean n -cube with bounded Vapnik-Chervonenkis dimension. *J. Comb. Theory, Ser. A*, 69(2), pages 217–232, 1995.
- S. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith. What can we learn privately? In *FOCS*, pages 531–540, 2008.
- N. Littlestone and M. Warmuth. Relating data compression and learnability. *Unpublished*, 1986.
- D. McAllester. PAC-bayesian model averaging. *Machine Learning Journal*, 5:5–21, 2003.
- A. McGregor, I. Mironov, T. Pitassi, O. Reingold, K. Talwar, and S. Vadhan. The limits of two-party differential privacy. In *FOCS*, pages 81–90, 2010.
- S. Moran and A. Yehudayoff. Sample compression schemes for VC classes. *JACM* 63 (3), 2016.
- M. Raginsky and A. Xu. Information-theoretic analysis of generalization capability of learning algorithms. *arXiv:1705.07809*, 2017.

- R. Rogers, A. Roth, A. Smith, and O. Thakkar. Max-information, differential privacy, and post-selection hypothesis testing. In *FOCS*, pages 487–494, 2016.
- D. Russo and J. Zhou. Controlling bias in adaptive data analysis using information theory. In *AISTATS*, pages 1232–1240, 2016.
- S. Shalev-Shwartz and S. Ben-David. Understanding machine learning: From theory to algorithms. *Cambridge University Press*, 2014.
- S. Vadhan. The complexity of differential privacy. *Tutorials on the Foundations of Cryptography*, pages 347–450, 2017.

Appendix A. Four proofs of Theorem 8

A.1 Proof I: Mutual Information and Independence

The first proof of that we present uses the following lemma, which allows to control a distribution μ by a distribution ν as long as it is close to it in KL-divergence. The proof technique is similar to a classical technique in Shannon’s information theory, e.g., in [Arutyunyan \(1968\)](#).

Lemma 14 *Let μ and ν be probability distributions on a finite set \mathcal{X} and let $E \subseteq \mathcal{X}$. Then,*

$$\mu(E) \leq \frac{\text{KL}(\mu||\nu) + 1}{\log(1/\nu(E))}.$$

The lemma enables us to compare between events of small probability: if $\nu(E)$ is small then $\mu(E)$ is also small, as long as $\text{KL}(\mu||\nu)$ is not very large.

The bound given by the lemma above is tight, as the following example shows. Let $\mathcal{X} = [2n]$ and let $E = [n]$. For each $x \in \mathcal{X}$, let

$$\mu(x) = \frac{1}{2n}$$

and let

$$\nu(x) = \begin{cases} 1/n^2 & x \in E, \\ (n-1)/n^2 & x \notin E. \end{cases}$$

Thus, $\mu(E) = \frac{1}{2}$ and $\nu(E) = \frac{1}{n}$. But on the other hand

$$\text{KL}(\mu||\nu) = \frac{1}{2} \log\left(\frac{n}{2}\right) + \frac{1}{2} \log\left(\frac{1}{2}\right) + o(1),$$

so

$$\frac{1}{2} = \mu(E) \geq \lim_{n \rightarrow \infty} \frac{\text{KL}(\mu||\nu) + 1}{\log(1/\nu(E))} = \frac{1}{2}.$$

Similar examples can be given when $\text{KL}(\mu||\nu)$ is constant.

We now change the setting to allow it to apply more naturally to d -bit information learners.

Lemma 15 *Let μ be a distribution on the space $\mathcal{X} \times \mathcal{Y}$ and let E be an event that satisfies*

$$\mu_X(E_y) < \alpha$$

for all $y \in \mathcal{Y}$, where μ_X is the marginal distribution of X and $E_y = \{x : (x, y) \in E\}$ is a fiber of E over y . Then

$$\mu(E) \leq \frac{I(X; Y) + 1}{\log(1/\alpha)}$$

The lemma enables us to bound the probability of an event E , if we have a bound on the probability of its fibers over y measured with the marginal distribution of X .

This lemma can be thought of as a generalization of the extremal case where X and Y are independent (i.e. $I(X, Y) = 0$). In this case, the lemma corresponds to the following geometric statement in the plane: if the width of every x -parallel fiber of a shape is at most α and its height is bounded by 1 then its area is also at most α . The bound given by the above lemma is $\frac{1}{\log(1/\alpha)}$, which is weaker, but the lemma applies more generally when $I(X, Y) > 0$. In fact, the bound is tight when the two variables are highly dependent; e.g. $X = Y$ and $X \sim U([n])$. In this case, the probability of the diagonal E is 1, while $I(X; Y) = \log(n)$ and $\alpha = 1/n$. So indeed $1 = \mu(E) \approx \frac{\log(n)+1}{\log(n)}$.

We now use this lemma to prove the theorem.

Proof [Proof of Theorem 8] Let μ be the distribution on pairs (S, h) where S is chosen i.i.d. from \mathcal{D} and h is the output of the algorithm given S . Let E be the event of error; that is,

$$E = \left\{ (S, h) : \left| \text{err}(h; \mathcal{D}) - \widehat{\text{err}}(h; S) \right| > \varepsilon \right\}.$$

Using Chernoff's inequality, for each h ,

$$\mu_S(E_h) \leq 2 \cdot \exp(-2m\varepsilon^2),$$

where E_h is the fiber of E over function h .

Lemma 15 implies

$$\mu(E) \leq \frac{I(S; \mathcal{A}(S)) + 1}{2m\varepsilon^2 - 1},$$

■

We now prove lemmas 14 and 15.

Proof [Proof of Lemma 14]

$$\begin{aligned}
 \text{KL}(\mu||\nu) &= -\mu(E) \sum_{x \in E} \frac{\mu(x)}{\mu(E)} \cdot \log\left(\frac{\nu(x)}{\mu(x)}\right) - \mu(E^c) \sum_{x \in E^c} \frac{\mu(x)}{\mu(E^c)} \cdot \log\left(\frac{\nu(x)}{\mu(x)}\right) \\
 &\stackrel{(a)}{\geq} -\mu(E) \cdot \log\left(\sum_{x \in E} \frac{\mu(x)}{\mu(E)} \frac{\nu(x)}{\mu(x)}\right) - \mu(E^c) \cdot \log\left(\sum_{x \in E^c} \frac{\mu(x)}{\mu(E^c)} \frac{\nu(x)}{\mu(x)}\right) \\
 &= -\mu(E) \cdot \log\left(\frac{\nu(E)}{\mu(E)}\right) - \mu(E^c) \cdot \log\left(\frac{\nu(E^c)}{\mu(E^c)}\right) \\
 &\stackrel{(b)}{\geq} -\mu(E) \cdot \log(\nu(E)) - \mu(E^c) \cdot \log(\nu(E^c)) - 1 \\
 &\geq -\mu(E) \cdot \log(\nu(E)) - 1,
 \end{aligned}$$

where (a) follows by convexity, and (b) holds since the binary entropy is at most one. ■

Proof [Proof of Lemma 15]

By Lemma 14, for each y ,

$$\mu_{X|Y=y}(E_y) \leq \frac{\text{KL}(\mu_{X|Y=y}||\mu_X) + 1}{\log(1/\mu_X(E_y))} \leq \frac{\text{KL}(\mu_{X|Y=y}||\mu_X) + 1}{\log(1/\alpha)}.$$

Taking expectation over y yields

$$\mu(E) \leq \frac{I(X;Y) + 1}{\log(1/\alpha)}.$$
■

A.2 Proof II: De-correlating

The second proof we present allows to “de-correlate” two random variables in terms of the mutual information. The lemma follows from [Harsha et. al. \(2010\)](#) and [Braverman and Garg \(2014\)](#).

Lemma 16 *Let μ be a probability distribution over $\mathcal{X} \times \mathcal{Y}$, where \mathcal{X} and \mathcal{Y} are finite sets. Let (X, Y) be chosen according to μ . Then there exists a random variable Z such that:*

1. Z is independent of X .
2. Y is a deterministic function of (X, Z) .
3. $H(Y|Z) \leq I(X, Y) + \log(I(X, Y) + 1) + O(1)$.

The lemma can be interpreted as follows. Think of X as sampled from some unknown process, and of Y as the output of a randomized algorithm on input X . Think of Z as the random coins of the algorithm. The lemma says that there is a way to sample Z , before seeing X , in a way that preserves functionality (namely, so that (X, Y) are correctly distributed) and so that for an average Z , the randomized algorithm outputs only a small number of Y 's.

Theorem 8 now follows similarly to the proof sketch for deterministic algorithms in section 3.1, since conditioned on Z , the algorithm uses only a small number of outputs (on average).

A.3 Proof III: Stability

The third approach for proving the connection between generalization and bounded information is based on stability. The parameters obtained from this proof are slightly different from those given in the statement of Theorem 8. For a d -bit information learner \mathcal{A} , we prove that

$$\mathbb{E}_{\mathcal{A}, S} [\text{err}(\mathcal{A}(S); \mathcal{D}) - \widehat{\text{err}}(\mathcal{A}(S); S)] < \sqrt{\frac{d}{m}} \quad (1)$$

where S is the input sample of m examples drawn i.i.d. from \mathcal{D} .

We start by setting some notation. For random variables X, Y , we use the notation $d_{\text{TV}}(X, Y)$ to denote the total variation (i.e., the statistical distance) between the distributions of X and Y . For clarity, we also twist the notation we used earlier a little bit, and use $\text{KL}(X||Y)$ to denote the KL-divergence between the *distributions* of X and Y . Also, in the following, we use the notation (X, Y) to denote the joint distribution of X and Y , and $X \times Y$ to denote the product distribution resulting from the marginal distributions of X and Y .

Finally, as typical in stability arguments, for any sample S and any example $z \in \mathcal{X} \times \{0, 1\}$, we will use the notation $S^{(i, z)}$ to denote the set resulting from replacing the i -th example in S by z .

The proof relies on the following two lemmas.

Lemma 17 *If $I(\mathcal{A}(S); S) \leq d$, then*

$$\frac{1}{m} \sum_{i=1}^m \sqrt{I(\mathcal{A}(S); S_i)} \leq \sqrt{\frac{d}{m}}$$

where S_i denotes the i -th example in S .

Lemma 18 *For any $i \in [m]$, we have*

$$\sqrt{I(\mathcal{A}(S); S_i)} \geq \mathbb{E}_z \left[d_{\text{TV}} \left(\mathcal{A} \left(S^{(i, z)} \right), \mathcal{A}(S) \right) \right]$$

where $z = (x, y) \sim \mathcal{D}$ independently from S .

Proof [Proof of Lemma 17] By the independence of the samples S_1, \dots, S_m and the fact that conditioning reduces entropy,

$$I(\mathcal{A}(S); S) \geq \sum_{i=1}^m I(\mathcal{A}(S); S_i)$$

By the Cauchy-Schwartz inequality,

$$\sum_{i=1}^m \sqrt{I(\mathcal{A}(S); S_i)} \leq \sqrt{m \sum_{i=1}^m I(\mathcal{A}(S); S_i)}.$$

■

Proof [Proof of Lemma 18]

$$\begin{aligned}
 \sqrt{I(\mathcal{A}(S); S_i)} &= \sqrt{\text{KL}((\mathcal{A}(S), S_i) \parallel \mathcal{A}(S) \times S_i)} \\
 &\geq d_{\text{TV}}((\mathcal{A}(S), S_i), \mathcal{A}(S) \times S_i) && \text{(Pinsker's inequality)} \\
 &= d_{\text{TV}}((\mathcal{A}(S^{(i,z)}), z), \mathcal{A}(S) \times z) \\
 &= \mathbb{E}_z \left[d_{\text{TV}}(\mathcal{A}(S^{(i,z)}), \mathcal{A}(S)) \right].
 \end{aligned}$$

where the third step follows from the fact that $(\mathcal{A}(S), S_i)$ and $(\mathcal{A}(S^{(i,z)}), z)$ are identically distributed, and the fact that S_i and z are identically distributed. ■

We are now ready to prove (1). Recall that for any example $S_i = (x_i, y_i)$, we have $\widehat{\text{err}}(\mathcal{A}(S); S_i) \triangleq \mathbf{1}(\mathcal{A}(S)(x_i) \neq y_i)$, where $\mathcal{A}(S)(x_i)$ denotes the label of the output hypothesis $\mathcal{A}(S)$ on x_i . Let z denote a fresh example $(x, y) \sim \mathcal{D}$ independent of S . Let \mathcal{U} denote the uniform distribution over $[m]$.

The two lemmas above imply that

$$\mathbb{E}_{i \sim \mathcal{U}, z} \left[d_{\text{TV}}(\mathcal{A}(S^{(i,z)}), \mathcal{A}(S)) \right] \leq \sqrt{\frac{d}{m}}.$$

It follows that for any $\tilde{z} \in \mathcal{X} \times \{0, 1\}$, we must have

$$\mathbb{E}_{i \sim \mathcal{U}, z} \left[d_{\text{TV}}(\widehat{\text{err}}(\mathcal{A}(S^{(i,z)}); \tilde{z}), \widehat{\text{err}}(\mathcal{A}(S); \tilde{z})) \right] \leq \sqrt{\frac{d}{m}},$$

which is equivalent to

$$\mathbb{E}_{i \sim \mathcal{U}, z} \left[\left| \mathbb{E}_{S, \mathcal{A}} \left[\widehat{\text{err}}(\mathcal{A}(S^{(i,z)}); \tilde{z}) - \widehat{\text{err}}(\mathcal{A}(S); \tilde{z}) \right] \right| \right] \leq \sqrt{\frac{d}{m}},$$

which implies

$$\mathbb{E}_{i \sim \mathcal{U}, z} \left[\mathbb{E}_{S, \mathcal{A}} \left[\widehat{\text{err}}(\mathcal{A}(S^{(i,z)}); S_i) - \widehat{\text{err}}(\mathcal{A}(S); S_i) \right] \right] \leq \sqrt{\frac{d}{m}}.$$

Finally, we use the fact that $\widehat{\text{err}}(\mathcal{A}(S^{(i,z)}); S_i)$ and $\widehat{\text{err}}(\mathcal{A}(S); z)$ are identically distributed to get

$$\mathbb{E}_{i \sim \mathcal{U}} \left[\mathbb{E}_{S, z, \mathcal{A}} \left[\widehat{\text{err}}(\mathcal{A}(S); z) - \widehat{\text{err}}(\mathcal{A}(S); S_i) \right] \right] \leq \sqrt{\frac{d}{m}},$$

which leads directly to (1).

A.4 Proof IV: PAC-Bayes

The fourth proof is straightforward via a connection between information learners and the PAC-Bayes framework. The PAC-Bayes framework considers distributions over hypotheses. It is convenient to think of a distribution \mathcal{P} over hypotheses as a randomized hypothesis. We extend the notions of error to randomized hypotheses as follows

$$\text{err}(\mathcal{Q}; \mathcal{D}) = \mathbb{E}_{h \sim \mathcal{Q}} [\text{err}(h; \mathcal{D})]$$

and

$$\widehat{\text{err}}(\mathcal{Q}; S) = \mathbb{E}_{h \sim \mathcal{Q}} [\widehat{\text{err}}(h; S)].$$

Fix some randomized hypothesis \mathcal{P} . The following theorem known as the PAC-Bayes bound gives a bound on the generalization error simultaneously for all randomized hypotheses \mathcal{Q} in terms of their KL-divergence with \mathcal{P} .

Theorem 19 (McAllester 2003; Shalev-Shwartz and Ben-David 2014) *Let \mathcal{D} be a distribution over examples, and let \mathcal{P} be a fixed but otherwise arbitrary distribution over hypotheses. Let S denote a set of m i.i.d. examples generated by \mathcal{D} . Then, the following event occurs with probability at least $1 - \delta$: for every distribution \mathcal{Q} over hypotheses,*

$$\text{err}(\mathcal{Q}; \mathcal{D}) - \widehat{\text{err}}(\mathcal{Q}; S) \leq \sqrt{\frac{\text{KL}(\mathcal{Q}||\mathcal{P}) + \ln(m/\delta)}{m}}.$$

Following the Bayesian reasoning approach, the distribution \mathcal{P} can be thought of as the a priori output of the algorithm for the target concept, and after seeing the input sample S the learning process outputs the distribution \mathcal{Q} (which may depend on S), which is its a posteriori output for the target concept. The PAC-Bayes theorem bounds the generalization error of the algorithm in terms of the KL-divergence between the a priori and the a posteriori outputs.

Technically, the proof of this section follows from expressing the mutual information between random variables X, Y in terms of the KL-divergence between $X|Y$ and X : Let \mathcal{A} be a d -bit information learner for \mathcal{H} , and let \mathcal{D} be a realizable distribution. Then

$$d \geq I(S; \mathcal{A}(S)) = \mathbb{E}_{S \sim \mathcal{D}^m} [\text{KL}(P_{h|S}||P_h)]$$

(recall that P_h denotes the marginal distribution of $\mathcal{A}(S)$ for $S \sim \mathcal{D}^m$). Therefore, by Markov's inequality, with probability $1 - \delta$ it holds that

$$\text{KL}(P_{h|S}||P_h) \leq d/\delta.$$

Now we can apply the standard PAC-Bayes generalization bound (Theorem 19) and deduce that with probability at least $1 - \delta$ over the choice of the sample, the *expected* generalization error is

$$O\left(\sqrt{\frac{d/\delta + \ln(m/\delta)}{m}}\right),$$

where the expectation is taken over $h \sim P_{h|S}$. Indeed, this follows by choosing the prior \mathcal{P} in the PAC-Bayes bound to be the marginal distribution P_h of the output of \mathcal{A} , and choosing the posterior \mathcal{Q} to be the conditional distribution $P_{h|S}$. Hence, by rephrasing the statement of this bound we can obtain a form similar to Theorem 8.

Theorem 20 *Assuming $m \geq 5 \frac{(d+1)}{\varepsilon^2} \ln \left(\frac{d+1}{\varepsilon} \right)$, for every $\varepsilon > 0$,*

$$\mathbb{P}_S \left[\mathbb{E}_{\mathcal{A}} [\text{err}(\mathcal{A}(S); \mathcal{D}) - \widehat{\text{err}}(\mathcal{A}(S); S)] > \varepsilon \right] < \frac{d+1}{m\varepsilon^2}$$

where the probability is taken over the randomness in the sample S and the expectation inside the probability is taken over the random coins of \mathcal{A} .

Appendix B. Sharpness of Theorem 8

Proposition 21 *Let $n \geq m \geq 4$ be integers such that n is sufficiently large. Let $\mathcal{X} = [n]$ and let \mathcal{D} be the uniform distribution on examples of the form $\{(x, 1) : x \in [n]\}$. There is a deterministic learning algorithm $\mathcal{A} : (\mathcal{X} \times \{0, 1\})^m \rightarrow \{0, 1\}^{\mathcal{X}}$ with sample size m and mutual information $O(1)$ so that*

$$\mathbb{P}_S [|\widehat{\text{err}}(\mathcal{A}(S); S) - \text{err}(\mathcal{A}(S); \mathcal{D})| \geq 1/2] \geq \frac{1}{m}$$

where S is generated i.i.d. from \mathcal{D} and f .

The construction is based on the following claim.

Claim 22 *For a sufficiently large n , there are $M = \Theta\left(\frac{2^m}{m}\right)$ subsets T_1, \dots, T_M of $[n]$ each of size at most $n/2$ so that the \mathcal{D} -measure of $T = \bigcup_i (T_i \times \{1\})^m$ is between $1/m$ and $4/m$.*

Proof Let T_1, \dots, T_M be i.i.d. uniformly random subsets of \mathcal{X} , each of size $k = \lfloor n/2 \rfloor$. The size of each T_i^m is k^m . For $i \neq j$ we have

$$\begin{aligned} \mathbb{E} [|T_i^m \cap T_j^m|] &= \mathbb{E} [|T_i \cap T_j|^m] \\ &= \sum_{x_1, \dots, x_m} \mathbb{E} [1_{x_1, \dots, x_m \in T_i}] \mathbb{E} [1_{x_1, \dots, x_m \in T_j}] \\ &\leq n^m \cdot \left(\frac{\binom{n-m}{k-m}}{\binom{n}{k}} \right)^2 + m^2 n^{m-1} \cdot 1, \end{aligned}$$

where in the last inequality, the first term corresponds to sequences x_1, \dots, x_m where all elements are distinct, and the second term corresponds to sequences x_1, \dots, x_m for which there are $i \neq j$ such that $x_i = x_j$. Now, since $\frac{\binom{n-m}{k-m}}{\binom{n}{k}} = \frac{k(k-1)\dots(k-m+1)}{n(n-1)\dots(n-m+1)} \leq \left(\frac{k}{n}\right)^m$,

$$\mathbb{E} [|T_i^m \cap T_j^m|] \leq \frac{k^{2m}}{n^m} + m^2 n^{m-1}.$$

Therefore, since $\sum |T_i^m| \geq |\cup_i T_i^m| \geq \sum |T_i^m| - \sum_{i \neq j} |T_i^m \cap T_j^m|$,

$$Mk^m \geq \mathbb{E}[|T|] \geq Mk^m - \frac{M^2 k^{2m}}{2 n^m} - \frac{M^2 m^2 n^{m-1}}{2} \geq \frac{Mk^m}{2} - \frac{M^2 m^2 n^{m-1}}{2},$$

as long as $M \leq n^m/k^m$. Hence, plugging $M = \frac{4n^m}{mk^m} = \Theta(\frac{2^m}{m})$ yields that

$$\frac{4}{m} \geq \mathbb{E}[|T|/n^m] \geq \frac{2}{m} - o(1),$$

where the $o(1)$ term approaches 0 as n approaches ∞ . So, for a sufficiently large n there is a choice of T_1, \dots, T_M as claimed. \blacksquare

Proof [Proof of Proposition 21] The algorithm \mathcal{A} is defined as follows. Let T_1, \dots, T_M be the sets given in Claim 22. For each $i \in [M]$, let h_i be the hypothesis that is 1 on T_i and 0 elsewhere. Given a sample $((x_1, 1), \dots, (x_m, 1))$, the algorithm outputs h_i , where i is the minimum index so that $\{x_1, \dots, x_m\} \subset T_i$; if no such index exists the algorithm outputs the all-ones function.

The empirical error of the algorithm is 0, and with probability $p \in [1/m, 4/m]$ it outputs a hypothesis with true error at least $1/2$. The amount of information it provides on its inputs can be bounded as follows: letting p_i be the probability that the algorithm outputs h_i , we have

$$\begin{aligned} I(S; \mathcal{A}(S)) &= H(\mathcal{A}(S)) \\ &= (1-p) \log \frac{1}{1-p} + p \sum_i \frac{p_i}{p} \log \frac{1}{p_i} \\ &\leq (1-p) \log \frac{1}{1-p} + p \log \frac{M}{p} && \text{(convexity)} \\ &\leq O(1). \end{aligned}$$

\blacksquare

Appendix C. Proof of Theorem 9

C.1 Warm Up

We first prove Theorem 9 for the special case of deterministic learning algorithms. Let \mathcal{A} be a consistent deterministic learning algorithm for \mathcal{T} . Define the $2^n \times 2^n$ upper triangular matrix M as follows. For all $i < j$,

$$M_{ij} = k$$

where f_k is the output of \mathcal{A} on a sample of the form

$$S_{ij} = \left(\underbrace{(1, 0), \dots, (1, 0)}_{m-2}, (i, 0), (j, 1) \right)$$

and $M_{ij} = 0$ for all $i \geq j$.

The matrix M summarizes the behavior of \mathcal{A} on some of its inputs. Our goal is to identify a sub-structure in M , and then use it to define the distribution \mathcal{D} .

We start with the following lemma.

Lemma 23 *Let $Q \in \text{Mat}_{2^n \times 2^n}(\mathbb{N})$ be a symmetric matrix that has the property that for all i, j :*

$$\min\{i, j\} \leq Q_{ij} \leq \max\{i, j\} \tag{i}$$

Then Q contains a row with at least $n + 1$ different values (and hence also a column with $n + 1$ different values).

Proof The proof is by induction on n . In the base case $n = 1$ we have

$$Q = \begin{bmatrix} 1 & x \\ x & 2 \end{bmatrix}$$

and the lemma indeed holds.

For the induction step, let

$$Q = \begin{bmatrix} Q_1 & Q_2 \\ Q_3 & Q_4 \end{bmatrix} \in \text{Mat}_{2^{n+1} \times 2^{n+1}}(\mathbb{N})$$

where $Q_1, Q_2, Q_3, Q_4 \in \text{Mat}_{2^n \times 2^n}(\mathbb{N})$.

All the values in Q_1 are in the interval $[1, 2^n]$ and Q_1 is also a symmetric matrix and satisfies property (i). So Q_1 contains some row r with at least $n + 1$ distinct values in the interval $[1, 2^n]$.

Similarly, all the values in Q_4 are in the interval $[2^n + 1, 2^{n+1}]$, and we can write Q_4 as

$$Q_4 = Q'_4 + 2^n \cdot J$$

where J is the all-1 matrix and $Q'_4 = Q_4 - (2^n \cdot J)$ is a symmetric matrix satisfying property (i). From the induction hypothesis it follows that Q_4 contains a column k with at least $n + 1$ different values in the interval $[2^n + 1, 2^{n+1}]$.

Now consider the value Q_{rk} . If $(Q_2)_{rk} \in [1, 2^n]$ then the column in Q corresponding to k contains $n + 2$ different values. Otherwise, the row corresponding to r contains $n + 2$ different values. ■

Next, consider the matrix

$$Q = M + M^t + \text{diag}(1, 2, 3, \dots, 2^n).$$

It is a symmetric matrix satisfying property (i), and so it contains a row r with at least $n + 1$ distinct values. If row r contains at least $\frac{n}{2}$ distinct values above the diagonal then row r in M also contains $\frac{n}{2}$ distinct values above the diagonal. Otherwise, row r contains at least $\frac{n}{2}$ distinct values below the diagonal, and then column r in M contains $\frac{n}{2}$ distinct values above the diagonal.

The proof proceeds by separately considering each of these two cases (row or column):

Case 1: M contains a row r with $\frac{n}{2}$ distinct values. Let $k_1, \dots, k_{n/2}$ be columns such that all the values M_{rk_i} are distinct. Let \mathcal{D} be the distribution that gives the point 1 a mass of $1 - \frac{1}{m-2}$, gives r a mass of $\frac{1}{2(m-2)}$ and evenly distributes the remaining mass on the values $k_1, \dots, k_{n/2}$. The function labeling the examples is chosen to be $f_{r+1} \in \mathcal{T}$.

Let E be the indicator random variable of the event that the sample S is of the form

$$S = \left(\underbrace{(1, 0), \dots, (1, 0)}_{m-2}, (r, 0), (k_i, 1) \right)$$

for some k_i . The probability of $E = 1$ is at least

$$\left(1 - \frac{1}{m-2}\right)^{m-2} \cdot \left(\frac{1}{2(m-2)}\right)^2 = \Omega\left(\frac{1}{m^2}\right).$$

From the definition of M , when $E = 1$ the algorithm outputs $h = f_{M_{rk_i}}$ where k_i is uniformly distributed uniformly over $\frac{n}{2}$ values, and so $H(\mathcal{A}(S)|E = 1) = \log(n/2)$. This yields

$$\begin{aligned} I(S; \mathcal{A}(S)) &= I(S, E; \mathcal{A}(S)) \\ &\geq I(S; \mathcal{A}(S)|E) \\ &= H(\mathcal{A}(S)|E) \\ &\geq \mathbb{P}[E = 1] H(\mathcal{A}(S)|E = 1) \\ &\geq \Omega\left(\frac{\log n}{m^2}\right). \end{aligned}$$

Case 2: M contains a column k with $\frac{n}{2}$ distinct values. Let $r_1, \dots, r_{n/2}$ be rows such that all the values $M_{r_i k}$ are distinct. Now, consider the event that

$$S = \left(\underbrace{(1, 0), \dots, (1, 0)}_{m-2}, (r_i, 0), (k, 1) \right)$$

for some r_i . The rest of the argument is the same as for the previous case.

C.2 Framework for Lower Bounding Mutual Information

Here we describe a simple framework that allows to lower bound the mutual information between two random variables.

STANDARD LEMMAS⁶

Lemma 24 *For any two distribution p and q , the contribution of the terms with $p(x) < q(x)$ to the divergence is at least -1 :*

$$\sum_{x:p(x)<q(x)} p(x) \log \frac{p(x)}{q(x)} > -1.$$

⁶We include the proofs for completeness.

Proof Let E denote the subset of x 's for which $p(x) < q(x)$. Then we have

$$\begin{aligned} \sum_{x \in E} p(x) \log \frac{p(x)}{q(x)} &\geq -p(E) \cdot \sum_{x \in E} p(x|E) \log \frac{q(x)}{p(x)} \\ &\geq -p(E) \cdot \log \sum_{x \in E} p(x|E) \frac{q(x)}{p(x)} \\ &= -p(E) \cdot \log \frac{q(E)}{p(E)} \\ &\geq p(E) \cdot \log p(E). \end{aligned}$$

For $0 \leq z \leq 1$, $z \log z$ is maximized when its derivative is 0: $\log e + \log x = 0$. So the maximum is attained at $z = 1/e$, proving that $p(E) \log p(E) \geq \frac{-\log e}{e} > -1$. ■

Lemma 25 (Data processing) *Let X, Y, Z be random variables such that $X - Y - Z$ form a Markov chain; that is, X and Z are independent conditioned on Y . Then*

$$I(X; Y) \geq I(X; Z)$$

Proof The chain rule for mutual information yields that

$$\begin{aligned} I(X; Y) &= I(X; Z) + I(X; Y|Z) - I(X; Z|Y) && \text{(chain rule)} \\ &= I(X; Z) + I(X; Y|Z) && (X - Y - Z) \\ &\geq I(X; Z). && \text{(information is non-negative)} \end{aligned}$$

■

THE FRAMEWORK

The following lemma is the key tool in proving a lower bound on the mutual information.

Lemma 26 *Let $n \in \mathbb{N}$, and let p_1, \dots, p_n be probability distributions over the set $[n]$ such that for all $i \in [n]$,*

$$p_i(i) \geq \frac{1}{2} \tag{iv}$$

Let U be a random variable distributed uniformly over $[n]$. Let T be a random variable over $[n]$ that results from sampling an index i according to U and then sampling an element of $[n]$ according to p_i . Then

$$I(U; T) = \Omega(\log n).$$

Proof

$$\begin{aligned} I(U; T) &= \sum_{i=1}^n \sum_{t=1}^n p_U(i) p_i(t) \log \frac{p_i(t)}{p_T(t)} \\ &= \sum_i p_U(i) p_i(i) \log \frac{p_i(i)}{p_T(i)} + \sum_i \sum_{t \neq i} p_U(i) p_i(t) \log \frac{p_i(t)}{p_T(t)}. \end{aligned}$$

Consider the first sum (the “diagonal”):

$$\begin{aligned} \sum_i p_U(i) p_i(i) \log \frac{p_i(i)}{p_T(i)} &= \frac{1}{n} \sum_i p_i(i) \log \frac{p_i(i)}{p_T(i)} \\ &\geq \frac{1}{n} \sum_i \frac{1}{2} \log \frac{\frac{1}{2}}{p_T(i)} && \text{(iv)} \\ &\geq \frac{1}{n} \cdot \frac{n}{2} \log \frac{\frac{n}{2}}{1} && \text{(log-sum inequality)} \\ &= \frac{1}{2} \log \frac{n}{2}. \end{aligned}$$

Finally, Lemma 24 implies that the second sum (the “off-diagonal”) is at least -1 . ■

We generalize the previous lemma as follows.

Lemma 27 *Let p_1, \dots, p_n be probability distributions over \mathcal{X} . Let $S_1, \dots, S_n \subset \mathcal{X}$ be pairwise disjoint events such that for all $i \in [n]$,*

$$p_i(S_i) \geq \frac{1}{2} \tag{v}$$

Let U be a random variable distributed uniformly over $[n]$. Let W be a random variable taking values in \mathcal{X} that results from sampling an index i according to U and then sampling an element of \mathcal{X} according to p_i . Then,

$$I(U; W) = \Omega(\log n).$$

Proof Let $S_0 = \mathcal{X} \setminus (S_1 \cup \dots \cup S_n)$, and let T be the random variable taking values in \mathbb{N} defined by $T(W) = i$ iff $W \in A_i$. Hence, T satisfies the conditions of Lemma 26. Furthermore, we have that the random variables U, W, T form the following Markov chain: $U - W - T$. We thus conclude that

$$I(U; W) \geq I(U; T) = \Omega(\log n)$$

where the inequality is according to the data processing inequality (Lemma 25) and the equality follows from Lemma 26. ■

C.3 Proof for General Case

We start with the analog of Lemma 23 from the warm up. Let $\Delta([2^n])$ be the set of all probability distribution over $[2^n]$. Let $Q \in \text{Mat}_{2^n \times 2^n}(\Delta([2^n]))$, i.e., Q is a $2^n \times 2^n$ matrix where each cell contains a probability distribution.

Lemma 28 *Assume that Q is symmetric and that it has the property that for all i, j ,*

$$\text{supp}(Q_{ij}) \subseteq [\min\{i, j\}, \max\{i, j\}]. \quad (\text{ii})$$

Then, Q contains a row with $n + 1$ distributions p_1, \dots, p_{n+1} such that there exist pairwise disjoint sets $S_1, \dots, S_{n+1} \subset [2^n]$ so that for all $i \in [n + 1]$,

$$p_i(S_i) \geq \frac{1}{2} \quad (\text{iii})$$

(and hence it also contains such a column).

Proof Again, the proof is by induction on n . The base case is easily verified. For the step, let

$$Q = \begin{bmatrix} Q_1 & Q_2 \\ Q_3 & Q_4 \end{bmatrix}$$

where $Q_1, Q_2, Q_3, Q_4 \in \text{Mat}_{2^n \times 2^n}(\Delta([2^n]))$.

The matrix Q_1 is symmetric, it satisfies property (ii), and that the supports of all the entries in Q_1 are contained in $[2^n]$. So by induction Q_1 contains a row r with probability functions p_1, \dots, p_{n+1} and pairwise disjoint sets $A_1, \dots, A_{n+1} \subset [2^n]$ that satisfy property (iii).

Similarly, one sees that Q_4 satisfies property (iii) as well. Namely, Q_4 contains a column k with probabilities q_1, \dots, q_{n+1} and pairwise disjoint sets $B_1, \dots, B_{n+1} \subset [2^n + 1, 2^{n+1}]$ with $q_i(B_i) \geq \frac{1}{2}$ for all i .

We now consider the probability distribution Q_{rk} . If $Q_{rk}([2^n]) \geq \frac{1}{2}$ then we define $q_{n+2} = Q_{rk}$ and $B_{n+2} = [2^n]$. Thus, column k of Q satisfies property (iii) with probabilities q_1, \dots, q_{n+2} and sets B_1, \dots, B_{n+2} . Otherwise, we choose $p_{n+2} = Q_{rk}$ and $A_{n+2} = [2^n + 1, 2^{n+1}]$. In this case, the row of Q corresponding to r satisfies property (iii) with probabilities p_1, \dots, p_{n+2} and sets A_1, \dots, A_{n+2} . ■

In the previous section we proved that property (iii) yields a lower bound on mutual information. We are prepared to prove the desired lower bound for probabilistic algorithms.

Proof [Proof of Theorem 9] Let \mathcal{A} be a consistent and proper learning algorithm for \mathcal{T} . Let $M \in \text{Mat}_{2^n \times 2^n}(\Delta([2^n]))$ be the matrix whose (i, j) entry for $i \neq j$ is

the probability distribution \mathcal{A} induces on \mathcal{T} with input

$$\left(\underbrace{(1, 0), \dots, (1, 0)}_{m-2}, (\min\{i, j\}, 0), (\max\{i, j\}, 1) \right)$$

and whose (i, i) entry for all i is the degenerate distribution that assigns probability 1 to i . The matrix M is symmetric, and because \mathcal{A} is consistent and proper it follows that M satisfies property (ii). By Lemma 28 therefore M contains a row r with probabilities p_1, \dots, p_{n+1} for which there are pairwise disjoint sets $S_1, \dots, S_{n+1} \subseteq [2^n]$ such that $p_i(S_i) \geq \frac{1}{2}$ for all i .

We assume without loss of generality that the probabilities $p_1, \dots, p_{n/2}$ are located on row r above the diagonal in cells $(r, k_1), \dots, (r, k_{n/2})$; the symmetric case can be handled similarly.

We now construct the probability \mathcal{D} over \mathcal{X} . The probability of 1 is $1 - \frac{1}{m-2}$, the probability of r is $\frac{1}{2(m-2)}$ and rest of the mass is uniformly distributed on $k_1, \dots, k_{n/2}$.⁷ Consider the indicator random variable E of the event that S is of the form

$$S = \left(\underbrace{(1, 0), \dots, (1, 0)}_{m-2}, (r, 0), (k_i, 1) \right)$$

for some k_i . For every S in this event, denote by i_S the index so that k_{i_S} is in the last example of S . Again, $\Pr[E = 1] \geq \Omega(1/m^2)$. Finally, as in the warm up,

$$\begin{aligned} I(S; \mathcal{A}(S)) &\geq I(S; \mathcal{A}(S)|E) \\ &\geq \Pr[E = 1] \cdot I(S; \mathcal{A}(S)|E = 1) \\ &\geq \Pr[E = 1] \cdot I(i_S; \mathcal{A}(S)|E = 1) \\ &\stackrel{(*)}{\geq} \Omega\left(\frac{\log n}{m^2}\right), \end{aligned}$$

where $(*)$ is justified as follows: Given that $E = 1$, we know that i_S is uniformly distributed on $[n/2]$. Furthermore, $\mathcal{A}(S)$ is the result of sampling a hypothesis according to the distribution p_{i_S} . The lower bound hence follows from Lemma 27. \blacksquare

Appendix D. Classes for the generic algorithm

We demonstrate the behavior of the generic algorithm in a couple of simple cases.

THRESHOLDS

Consider the class of thresholds \mathcal{T} over $\mathcal{X} = [N]$ and the generic algorithm $\mathcal{A}_{\mathcal{T}}$.

Theorem 29 *For every sample size, the generic algorithm $\mathcal{A} = \mathcal{A}_{\mathcal{T}}$ is a $(\log \log(N) + O(1))$ -bit information learner.*

Proof Suppose the labelling function is $f_k(x) = 1_{x \geq k}$. Consider the following distribution \mathcal{Q} over \mathcal{H} :

$$\mathcal{Q}(t) = \frac{c}{(1 + |k - t|) \log(N)},$$

⁷We assume for simplicity that 1, r and the k_i are all distinct.

where $c \geq 1/2$ is the normalizing constant. One can verify this lower bound on c by noting that

$$\sum_{t \in [N]} \frac{1}{1 + |k - t|} \leq \int_{t=1}^{N+1} \frac{1}{1 + |k - t|} dt \leq 2 \log(N).$$

Now, by plugging this choice of \mathcal{Q} into Lemma 11 and noting that for any realizable sample S , the distribution $P(h|S)$ is uniform over f_t for $t \in \{x_1, x_1 + 1, \dots, x_2\}$ for some $x_1 \leq k \leq x_2$ in \mathcal{X} , we can reach the desired bound:

$$\begin{aligned} \text{KL}(P_{h|S} \parallel \mathcal{Q}) &= \log \log(N) + \sum_{x_1 \leq t \leq x_2} \frac{1}{x_2 - x_1 + 1} \log \frac{1 + |k - t|}{c(x_2 - x_1 + 1)} \\ &\leq \log \log(N) + 1, \end{aligned}$$

since $\frac{1 + |k - t|}{c(x_2 - x_1 + 1)} \leq 2$. ■

Given our results in Section 5, we note that the above bound is indeed tight.

POINT FUNCTIONS

We can learn the class of point functions \mathcal{PF} on $\mathcal{X} = [N]$ with at most 2 bits of information. The learning algorithm is again the generic one.

Theorem 30 *For every realizable distribution \mathcal{D} , and for every sample size $m \leq N/2$, the generic algorithm $\mathcal{A} = \mathcal{A}_{\mathcal{PF}}$ has at most 2 bits of mutual information with respect to \mathcal{D} .*

Proof Suppose, without loss of generality, that the target concept is $f_1(x) = 1_{\{x=1\}}$. Pick \mathcal{Q} in the bound of Lemma 11 as follows:

$$\mathcal{Q}(x) = \begin{cases} 1/2 & x = 1, \\ \frac{1}{2(N-1)} & x \neq 1. \end{cases}$$

Let S be a sample. If 1 appears in S then

$$\text{KL}(P_{h|S} \parallel \mathcal{Q}) = 1 \cdot \log \frac{1}{1/2} = 1.$$

If 1 does not appear in S then $P_{h|S}$ is uniform on a subset of the form $\{1_{x=i} : i \in \mathcal{X}'\}$ for some $\mathcal{X}' \subset \mathcal{X}$ that contains 1 of size $k \geq N - m \geq N/2$, so

$$\text{KL}(P_{h|S} \parallel \mathcal{Q}) = \frac{1}{k} \log \frac{2}{k} + \frac{k-1}{k} \log \frac{2(N-1)}{k} \leq 2. ■$$

Appendix E. Mutual Information Bound

Given a sample S of size m labelled by the function $1_{x>1}$, we calculate the mutual information of $\mathcal{A} = \mathcal{A}_{\mathcal{H}}$ in this setting (we think of m as a large constant). We start with some preliminary calculations.

Conditional entropy: Let $s_1 = ((1, 0), (1, 0), \dots, (1, 0))$. Then

$$\begin{aligned} H(\mathcal{A}(S)|S) &= P(S = s_1)H(\mathcal{A}(S)|S = s_1) + \sum_{s \neq s_1} P(S = s)H(\mathcal{A}(S)|S = s) \\ &< \left(1 - \frac{1}{m}\right)^m \log N + 1 < \frac{\log N}{e} + 1. \end{aligned}$$

Marginal probabilities:

$$P(\mathcal{A}(S) = 1_{x>0}) = \left(1 - \frac{1}{m}\right)^m \frac{1}{N} + m \left(1 - \frac{1}{m}\right)^{m-1} \cdot \frac{1}{m} \cdot \frac{1}{2} + \dots > \frac{0.99}{2e}$$

and

$$P(\mathcal{A}(S) = 1_{x=i}) = \left(1 - \frac{1}{m}\right)^m \frac{1}{N} + m \left(1 - \frac{1}{m}\right)^{m-1} \cdot \frac{1}{m(N-1)} \cdot \frac{1}{2} + \dots < \frac{3.01}{2eN}.$$

Entropy:

$$\begin{aligned} &H(\mathcal{A}(S)) \\ &= -P(\mathcal{A}(S) = 1_{x>0}) \log P(\mathcal{A}(S) = 1_{x>0}) - \sum_i P(\mathcal{A}(S) = 1_{x=i}) \log P(\mathcal{A}(S) = 1_{x=i}) \\ &> \frac{3}{2e} \log N. \end{aligned}$$

Finally, we can bound the information:

$$I(S; \mathcal{A}(S)) = H(\mathcal{A}(S)) - H(\mathcal{A}(S)|S) > \frac{\log N}{2e} - 1.$$

This calculation makes sense, since the only way to learn something substantial about the sample is when you get exactly one $x \neq 1$. This happens with probability $\approx 1/e$ and then the generic algorithm chooses a function that reveals x with probability $1/2$. This sample has entropy $\log(N-1)$, so we have the desired result.

Appendix F. Proof for the Distribution-Dependent Setting

Proof Let $\varepsilon_k = (1/(m+1))^k$ for $k > 0$. For each k , pick an ε_k -net N_k for \mathcal{H} with respect to \mathcal{D} of minimum size; that is, for every $h \in \mathcal{H}$ there is $f \in N_k$ so that $\Pr_{x \sim \mathcal{D}_X}(h(x) \neq f(x)) \leq \varepsilon_k$. A result of [Haussler \(1995\)](#) states that the size of N_k is at most $(4e^2/\varepsilon_k)^d \leq (4e^2m)^{kd}$.

The algorithm works as follows: given an input sample S , the algorithm checks if N_1 contains a consistent hypothesis. If it does, the algorithm outputs it. Otherwise, it checks in N_2 , and so forth.

As we explain below, the probability that the algorithm stops is one (even when \mathcal{X} is infinite). Denote by K the value of k in which the algorithm stops.

Bound the entropy of the output as follows:

$$\begin{aligned} H(\mathcal{A}(S)) &\leq H(\mathcal{A}(S), K) = H(\mathcal{A}(S)|K) + H(K) \leq \\ &\leq \sum_k \Pr[K = k] \cdot kd \log(4e^2(m+1)) + H(K) \end{aligned}$$

The labelling function $f \in \mathcal{H}$ is at distance of at most ε_k from N_k , so

$$\Pr[K \leq k] \geq (1 - \varepsilon_k)^m \geq 1 - \frac{1}{(m+1)^{k-1}},$$

which implies $\Pr[K = k+1] \leq \frac{1}{(m+1)^{k-1}} \leq \frac{1}{2^{k-1}}$ (this in particular implies that the algorithm terminates with probability one). Hence,

$$\sum_k \Pr[K = k] \cdot kd \log(4e^2(m+1)) \leq O(d \log(m+1))$$

and $H(K) \leq O(1)$. ■

Appendix G. Confidence Amplification

We show that the same standard procedure used for boosting the confidence of learning algorithms can be used with information learners at a modest cost.

Theorem B.1 (Confidence amplification.) Let \mathcal{A} be a d -bit information learner, and let \mathcal{D} be a distribution on examples. Assume that when \mathcal{A} receives m_0 examples from \mathcal{D} , then with probability $1/2$ it outputs a hypothesis with error at most ε . Then there is an algorithm \mathcal{B} such that for every $\delta > 0$,

- When \mathcal{B} receives

$$m = m_0 \lceil \log(2/\delta) \rceil + \frac{2 \ln(4 \log(2/\delta)/\delta)}{4\varepsilon^2}$$

examples from \mathcal{D} , then with probability at least $1 - \delta$ it outputs a hypothesis with error at most ε .

- \mathcal{B} is a $(\log \log(2/\delta) + d \log(2/\delta))$ -bit information learner.

Proof We use the natural confidence amplification procedure. Set $k = \lceil \log(2/\delta) \rceil$, and draw k subsamples $(S^{(1)}, \dots, S^{(k)}) \triangleq S$, each of size m_0 , and another “validation” set T of size $2 \ln(4 \log(2/\delta)/\delta)/\varepsilon^2$. Run \mathcal{A} independently on each of the k subsamples to output hypotheses $\vec{h} = (h_1, \dots, h_k)$. Next, we validate h_1, \dots, h_k on T and output a hypothesis $h^* \in \{h_1, \dots, h_k\}$ with minimal empirical error on T .

Item 1 in the above theorem follows directly from standard analysis (Chernoff plus union bound). To prove item 2, first we note that (S, h_1, \dots, h_k) is independent of T . So,

$$\begin{aligned}
I(h^*; S, T) &\leq I(h^*, \vec{h}; S, T) \\
&= I(\vec{h}; S, T) + I(h^*; S, T | \vec{h}) \\
&= I(\vec{h}; S) + [I(h^*; T | \vec{h}) + I(h^*; S | T, \vec{h})] \\
&\leq \sum_{i=1}^k I(h_i; S^{(i)}) + H(h^* | h_1, \dots, h_k) + 0 \\
&\leq dk + \log(k)
\end{aligned}$$

■