JMLR: Workshop and Conference Proceedings vol 49:1–28, 2016

Adaptive Learning with Robust Generalization Guarantees

Rachel Cummings RACHELC@CALTECH.EDU
California Institute of Technology

Katrina Ligett KATRINA@CALTECH.EDU
California Institute of Technology and Hebrew University of Jerusalem

Kobbi Nissim KOBBI@SEAS.HARVARD.EDU
Ben-Gurion University and Harvard University

Aaron Roth AAROTH@CIS.UPENN.EDU
Zhiwei Steven Wu WUZHIWEI@CIS.UPENN.EDU
University of Pennsylvania

Abstract

The traditional notion of *generalization*—i.e., learning a hypothesis whose empirical error is close to its true error—is surprisingly brittle. As has recently been noted (Dwork et al., 2015c), even if several algorithms have this guarantee in isolation, the guarantee need not hold if the algorithms are composed adaptively. In this paper, we study three notions of generalization—increasing in strength—that are *robust* to postprocessing and amenable to adaptive composition, and examine the relationships between them.

We call the weakest such notion *Robust Generalization*. A second, intermediate, notion is the stability guarantee known as *differential privacy*. The strongest guarantee we consider we call *Perfect Generalization*. We prove that every hypothesis class that is PAC learnable is also PAC learnable in a robustly generalizing fashion, with almost the same sample complexity. It was previously known that differentially private algorithms satisfy robust generalization. In this paper, we show that robust generalization is a strictly weaker concept, and that there is a learning task that can be carried out subject to robust generalization guarantees, yet cannot be carried out subject to differential privacy. We also show that perfect generalization is a strictly stronger guarantee than differential privacy, but that, nevertheless, many learning tasks can be carried out subject to the guarantees of perfect generalization.

Keywords: Adaptive learning, generalizations, compression schemes, and composition.

1. Introduction

Generalization, informally, is the ability of a learner to reflect not just its training data, but properties of the underlying distribution from which the data are drawn. When paired with empirical risk minimization, it is one of the fundamental tools of learning. Typically, we say that a learning algorithm *generalizes* if, given access to some training set drawn i.i.d. from an underlying data distribution, it returns a hypothesis whose empirical error (on the training data) is close to its true error (on the underlying distribution).

This is, however, a surprisingly brittle notion—even if the output of a learning algorithm generalizes, one may be able to extract additional hypotheses by performing further computations on the output hypothesis—i.e., by postprocessing—that do not themselves generalize. As an example, notice that the standard notion of generalization does not prevent a learner from encoding the entire

training set in the hypothesis that it outputs, which in turn allows a data analyst to generate a hypothesis that over-fits to an arbitrary degree. In this sense, *traditional generalization is not robust to misinterpretation by subsequent analyses (postprocessing)* (either malicious or naive).

Misinterpretation of learning results is only one face of the threat—the problem is much more alarming. Suppose the output of a (generalizing) learning algorithm influences, directly or indirectly, the choice of future learning tasks. For example, suppose a scientist chooses a scientific hypothesis to explore on some data, on the basis of previously (generalizingly!) learned correlations in that data set. Or suppose a data scientist repeatedly iterates a model selection procedure while validating it on the same holdout set, attempting to optimize his empirical error. These approaches are very natural, but also can lead to false discovery in the first case, and disastrous overfitting to the holdout set in the second (Dwork et al., 2015b), because *traditional generalization is not robust to adaptive composition*.

In this paper, we study two refined notions of generalization—*robust generalization* and *perfect generalization*, each of which is preserved under post-processing (we discuss their adaptive composition guarantees more below). Viewed in relation to these two notions, *differential privacy* can also be cast as a third, intermediate generalization guarantee. It was previously known that differentially private algorithms were also robustly generalizing (Dwork et al., 2015c; Bassily et al., 2016). As we show in this paper, however, differential privacy is a strictly stronger guarantee—there are proper learning problems that can be solved subject to robust generalization that cannot be solved subject to differential privacy (or with any other method previously known to guarantee robust generalization). Moreover, we show that every PAC learnable class (even over infinite data domains) is learnable subject to robust generalization, with almost no asymptotic blowup in sample complexity (a comparable statement is not known for differentially private algorithms, and is known to be false for algorithms satisfying *pure* differential privacy). We also show that, in a sense, differential privacy is a strictly weaker guarantee than perfect generalization. We provide a number of generic techniques for learning under these notions of generalization and prove useful properties for each. As we will discuss, perfect generalization also can be interpreted as a *privacy guarantee*, and thus may also be of interest to the privacy community.

1.1. Our Results

Informally, we say that a learning algorithm has a guarantee of *robust generalization* if it is not only guaranteed to output a hypothesis whose empirical error is close to the true error (and near optimal), but if no adversary taking the output hypothesis as input can find another hypothesis whose empirical error differs substantially from its true error. (In particular, robustly generalizing algorithms are inherently robust to post-processing, and hence can be used to generate other test statistics in arbitrary ways without worry of overfitting). We say that a learning algorithm has the stronger guarantee of *perfect generalization* if its output reveals almost nothing about the training data that could not have been learned via only direct oracle access to the underlying data distribution.

It was previously known (Dwork et al., 2015c,a; Bassily et al., 2016) that both *differential privacy* and *bounded description length outputs* are sufficient conditions to guarantee that a learning algorithm satisfies robust generalization. However, prior to this work, it was possible that differential privacy was *equivalent* to robust generalization in the sense that any learning problem that could be solved subject to the guarantees of robust generalization could also be solved via a differentially

private algorithm.[1] Indeed, this was one of the open questions stated in Dwork et al. (2015a). We resolve this question (Section 3.3) by showing a simple proper learning task (learning threshold functions over the real line) that can be solved with guarantees of robust generalization (indeed, with the optimal sample complexity) but that cannot be non-trivially properly learned by any differentially private algorithm (or any algorithm with bounded description length outputs). We do so (Theorem 18) by showing that generalization guarantees that follow from *compression schemes* (Littlestone and Warmuth, 1986) carry over to give guarantees of robust generalization (thus giving a third technique, beyond differential privacy and description length arguments, for establishing robust generalization). In addition to threshold learning, important learning procedures like SVMs have optimal compression schemes, and so satisfy robust generalization without modification. We also show (Theorem 19) that compression schemes satisfy an adaptive composition theorem, and so can be used for adaptive data analysis while guaranteeing robust generalization. Note that, somewhat subtly, robustly generalizing algorithms derived by other means need not necessarily maintain their robust generalization guarantees under adaptive composition (a sequence of computations in which later computations have access not only to the training data, but also to the outputs of previous computations). Using the fact that boosting implies the existence of a near optimal variable-length compression scheme for every VC-class (see David et al. (2016)), we show (Theorem 26) that any PAC learnable hypothesis class (even over an infinite domain) is also learnable with robust generalization, with at most a logarithmic blowup in sample complexity. (In fact, merely *subsampling* gives a simple "approximate compression scheme" for any VC-class, but one that would imply a quadratically suboptimal sample complexity bound. In contrast, we show that almost no loss in sample complexity –on top of the sample complexity needed for outputting an accurate hypothesis– is necessary in order to get the guarantees of robust generalization.)

We then show (Theorem 33) that perfectly generalizing algorithms can be compiled into differentially private algorithms (in a black box way) with little loss in their parameters, and that (Theorem 39) differentially private algorithms are perfectly generalizing, but with a loss of a factor of $\sqrt{n}$ in the generalization parameter. Moreover, we show (Theorem 40) that this $\sqrt{n}$ loss is necessary. Because differentially private algorithms satisfy an adaptive composition theorem, this gives a method for designing perfectly generalizing algorithms that are robust to arbitrary adaptive composition. Despite this $\sqrt{n}$ blowup in the generalization parameter, we show (Section 4.1) that any *finite* hypothesis class can be PAC learned subject to perfect generalization.

1.2. Related work

Classically, machine learning has been concerned only with the basic generalization guarantee that the empirical error of the learned hypothesis be close to the true error. There are three main approaches to proving standard generalization guarantees of this sort. The first is by bounding various notions of complexity of the range of the algorithm—most notably, the VC-dimension (see, e.g., Kearns and Vazirani (1994) for a textbook introduction). These guarantees are *not* robust to post-processing or adaptive composition. The second follows from an important line of work (Bousquet and Elisseeff, 2002; Poggio et al., 2004; Shalev-Shwartz et al., 2010) that establishes connections

1. More precisely, it was known that algorithms with bounded description length could give robust generalization guarantees for the computation of high sensitivity statistics that could not be achieved via differential privacy (Dwork et al., 2015a). However, for low-sensitivity statistics (like the empirical error of a classifier, and hence for the problem of learning), there was no known separation.

between the *stability* of a learning algorithm and its ability to generalize. Most of these classic stability notions are defined over some metric on the output space (rather than on the distribution over outputs), and for these reasons are also brittle to post-processing and adaptive composition. The third is the compression-scheme method first introduced by Littlestone and Warmuth (1986) (see, e.g., Shalev-Shwartz and Ben-David (2014) for a textbook introduction). As we show in this paper, the generalization guarantees that follow from compression schemes *are* robust to post-processing and adaptive composition. A longstanding conjecture (Warmuth, 2003) states that VC-classes of dimension d have compression schemes of size d, but it is known that boosting Freund and Schapire (1997) implies the existence of a *variable-length* compression scheme that for any function from a VC-class of dimension d can compress n examples to an empirical risk minimizer defined by a subset of only $O(d \log n)$ many examples David et al. (2016).

A recent line of work (Dwork et al., 2015c,a; Bassily et al., 2016; Russo and Zou, 2016) has studied algorithmic conditions that guarantee the sort of *robust* generalization guarantees we study in this paper, suitable for adaptive data analysis. Dwork et al. (2015c) show that differential privacy (a stability guarantee on the output *distribution* of an algorithm) is sufficient to give robust generalization guarantees, and Dwork et al. (2015a) show that description length bounds on the algorithm's output (i.e., Occam style bounds Blumer et al. (1990), which have long been known to guarantee standard generalization) are also sufficient.

Differential privacy was introduced by Dwork et al. (2006b) (see Dwork and Roth (2014) for a textbook introduction), and private learning has been a central object of study since Kasiviswanathan et al. (2011). The key results we use here are the upper bounds for private learning proven by Kasiviswanathan et al. (2011) using the exponential mechanism of McSherry and Talwar (2007), and the lower bounds for private proper threshold learning due to Bun et al. (2015). A measure similar to, but distinct from, the notion of *perfect generalization* that we introduce here was briefly studied as a privacy solution concept in Blum et al. (2008) under the name "distributional privacy."

2. Preliminaries

2.1. Learning Theory Background

Let $\mathcal{X}$ denote a *domain*, which contains all possible *examples*. A *hypothesis* $h \colon \mathcal{X} \to \{0,1\}$ is a boolean mapping that labels examples by $\{0,1\}$, with $h(x) = 1$ indicating that x is a positive instance and $h(x) = 0$ indicating that x is a negative instance. A *hypothesis class* is a set of hypotheses. Throughout the paper, we elide dependencies on the dimension of the domain.

We will sometimes write $\mathcal{X}_L$ for $\mathcal{X} \times \{0,1\}$, i.e., labelled examples. Let $\mathcal{D}_L \in \Delta\mathcal{X}_L$ be a distribution over labelled examples; we will refer to it as the *underlying distribution*. We write $S_L \sim_{i.i.d.} \mathcal{D}_L^n$ to denote a *sample* of n labelled examples drawn i.i.d. from $\mathcal{D}_L$. A learning algorithm takes such a sample S_L (also known as a *training set*) as input, and outputs a hypothesis. Note that we use subscript-L to denote labeling of examples in the domain, in samples, and in distributions. When $\mathcal{D}_L$ is well-defined, we also sometimes write $\mathcal{D}$ for the marginal distribution of $\mathcal{D}_L$ over $\mathcal{X}$; similarly for S and S_L.

Typically, the goal when selecting a hypothesis is to minimize the *true error* (also known as the expected error) of the selected hypothesis on the underlying distribution:

$$err(h) = \Pr_{(x,y)\sim\mathcal{D}_L}[h(x) \neq y].$$

This is in contrast to the *empirical error* (also known as the training error), which is the error of the selected hypothesis h on the sample S_L:

$$err(S_L, h) \equiv \frac{1}{|S_L|} \sum_{(x_i, y_i) \in S_L} \mathbf{1}[h(x_i) \neq y_i].$$

In order to minimize true error, learning algorithms typically seek to (approximately) minimize their empirical error, and to combine this with a generalization guarantee, which serves to translate low empirical error into a guarantee of low true error.

For any set $S \in \mathcal{X}^n$, let $\mathcal{E}_S$ denote the empirical distribution that assigns weight $1/n$ on every observation in S. For any hypothesis $h\colon \mathcal{X} \to \{0,1\}$, we will write $h(\mathcal{D})$ to denote $\mathbb{E}_{x\sim\mathcal{D}}[h(x)]$ and $h(S)$ to denote $h(\mathcal{E}_S) = \mathbb{E}_{x\sim\mathcal{E}_S}[h(x)] = 1/n \sum_{x_i \in S} h(x_i)$. We say that a hypothesis $h : \mathcal{X} \to \{0,1\}$ α-*overfits* to the sample S taken from $\mathcal{D}$ if $|err(h) - err(S_L, h)| \geq \alpha$. Traditional generalization requires that a mechanism output a hypothesis that does not overfit to the sample.

Definition 1 ((Traditional) Generalization) *Let $\mathcal{X}$ be an arbitrary domain. A mechanism $\mathcal{M}\colon \mathcal{X}_L^n \to (\mathcal{X} \to \{0,1\})$ is* (α, β)-generalizing *if for all distributions $\mathcal{D}_L$ over $\mathcal{X}_L$, given a sample $S_L \sim_{i.i.d.} \mathcal{D}_L^n$,*

$$\Pr\left[\mathcal{M}(S_L) \text{ outputs } h\colon \mathcal{X} \to \{0,1\} \text{ such that } |err(h) - err(S_L, h)| \leq \alpha\right] \geq 1 - \beta,$$

where the probability is over the choice of the sample S_L and the randomness of $\mathcal{M}$.

Note that (traditional) generalization does not prevent $\mathcal{M}$ from encoding its input sample S_L in the hypothesis h that it outputs.

Note that throughout the paper, we focus only on *proper learning*, wherein the learner is required to return a hypothesis from the class it is learning, rather than from, e.g., some superset of that class. For simplicity, we frequently omit the word "proper." Within the setting of proper learning, we consider two different models of learning. In the setting of *PAC learning*, we assume that the examples in the support of the underlying distribution are labelled consistently with some *target* hypothesis h^* from a known hypothesis class $\mathcal{H}$. In this case, we could write $err(h) = \Pr_{x\sim\mathcal{D}}[h(x) \neq h^*(x)]$.

Definition 2 (PAC Learning) *A hypothesis class $\mathcal{H}$ over domain $\mathcal{X}$ is* PAC learnable *if there exists a polynomial $n_{\mathcal{H}} : \mathbb{R}^2 \to \mathbb{R}$ and a learning algorithm $\mathcal{A}$ such that for all hypotheses $h^* \in \mathcal{H}_d$, all $\alpha, \beta \in (0, 1/2)$, and all distributions $\mathcal{D}$ over $\mathcal{X}$, given inputs α, β and a sample $S_L = (z_1, \ldots, z_n)$, where $n \geq n_{\mathcal{H}}(1/\alpha, \log(1/\beta))$, $z_i = (x_i, h^*(x_i))$ and the x_i's are drawn $i.i.d.$ from $\mathcal{D}$, the algorithm $\mathcal{A}$ outputs a hypothesis $h \in \mathcal{H}$ with the following guarantee:*

$$\Pr[err(h) \leq \alpha] \geq 1 - \beta.$$

The probability is taken over both the randomness of the examples and the internal randomness of $\mathcal{A}$. We will say that $\mathcal{H}$ is PAC learnable with a learning rate $n_{\mathcal{H}}$, and call a learning algorithm with the above guarantee (α, β)-accurate.

In the setting of *agnostic learning*, we do not assume that the labels of the underlying data distribution are consistent with some hypothesis in $\mathcal{H}$. The goal then becomes finding a hypothesis whose true error is almost optimal within the hypothesis class $\mathcal{H}$.

Definition 3 (Agnostic Learning) Agnostically learnable *is defined identically to PAC learnable with two exceptions:*

1. *the data are drawn and labelled from an arbitrary distribution $\mathcal{D}_L$ over $\mathcal{X} \times \{0,1\}$*
2. *the output hypothesis h satisfies the following*

$$\Pr[err(h) \leq \mathrm{OPT} + \alpha] \geq 1 - \beta,$$

where $\mathrm{OPT} = \min_{f \in \mathcal{H}}\{err(f)\}$ and the probability is taken over both the randomness of the data and the internal randomness of the algorithm.

It is known that (in the binary classification setting we study), a hypothesis class is learnable if and only if its *VC-dimension* is polynomially bounded:

Definition 4 (VC Dimension Vapnik and Chervonenkis (1971)) *A set $S \subseteq \mathcal{X}$ is* shattered by *a hypothesis class $\mathcal{H}$ if $\mathcal{H}$ restricted to S contains all $2^{|S|}$ possible functions from S to $\{0,1\}$. The* VC dimension *of $\mathcal{H}$ denoted $\mathsf{VCDIM}(\mathcal{H})$, is the cardinality of a largest set S shattered by $\mathcal{H}$.*

2.2. Notions of Generalization

In this section, we introduce the three notions of generalization that are studied throughout this paper. We say that a mechanism $\mathcal{M}$ *robustly generalizes* if the mechanism does not provide information that helps overfit the sample it is given as input. Formally:

Definition 5 (Robust Generalization) *Let $\mathcal{R}$ be an arbitrary range and $\mathcal{X}$ an arbitrary domain. A mechanism $\mathcal{M} \colon \mathcal{X}_L^n \to \mathcal{R}$ is* (ε, δ)-robustly generalizing *if for all distributions $\mathcal{D}_L$ over $\mathcal{X}_L$ and any adversary $\mathcal{A}$, with probability $1 - \zeta$ over the choice of sample $S_L \sim_{i.i.d.} \mathcal{D}_L^n$,*

$$\Pr\left[\mathcal{A}(\mathcal{M}(S_L)) \text{ outputs } h \colon \mathcal{X} \to \{0,1\} \text{ such that } |h(S) - h(\mathcal{D})| \leq \varepsilon\right] \geq 1 - \gamma,$$

for some ζ, γ such that $\delta = \zeta + \gamma$, where the probability is over the randomness of $\mathcal{M}$ and $\mathcal{A}$.[2]

For our other notions of generalization we require the following definition of distributional closeness.

Definition 6 ((ε, δ)-Closeness) *Let $\mathcal{R}$ be an arbitrary range, and let $\Delta\mathcal{R}$ denote the set of all probability distributions over $\mathcal{R}$. We say that distributions $\mathcal{J}_1, \mathcal{J}_2 \in \Delta\mathcal{R}$ are* (ε, δ)-close *and write $\mathcal{J}_1 \approx_{\varepsilon,\delta} \mathcal{J}_2$ if for all $\mathcal{O} \subseteq \mathcal{R}$,*

$$\Pr_{y \sim \mathcal{J}_1}[y \in \mathcal{O}] \leq \exp(\varepsilon) \Pr_{y \sim \mathcal{J}_2}[y \in \mathcal{O}] + \delta \quad \text{and} \quad \Pr_{y \sim \mathcal{J}_2}[y \in \mathcal{O}] \leq \exp(\varepsilon) \Pr_{y \sim \mathcal{J}_1}[y \in \mathcal{O}] + \delta.$$

Given an arbitrary domain $\mathcal{Y}$, we say samples $T, T' \in \mathcal{Y}^n$ are *neighboring* if they differ on exactly one element. A mechanism $\mathcal{M}$ is *differentially private* if the distributions of its outputs are close on neighboring samples.

2. Note that we do not state the robust generalization guarantee in terms of the difference $|err(h') - err(S_L, h')|$ between true error and empirical error for for some hypothesis h' (as in Theorem 1), and our definition is in fact more general — in particular, we can let $h((x,y)) = \mathbf{1}[h'(x) \neq y]$ to capture the generalization notion in terms of error.

Definition 7 (Differential Privacy, Dwork et al. (2006b)) *A mechanism $\mathcal{M}\colon \mathcal{Y}^n \to \mathcal{R}$ is* (ε, δ)-differentially private *if for every pair of neighboring samples $T, T' \in \mathcal{Y}^n$, $\mathcal{M}(T) \approx_{\varepsilon,\delta} \mathcal{M}(T')$.*

Let $\mathcal{Y}$ be an arbitrary domain and $\mathcal{R}$ be an arbitrary range, and let $\Delta\mathcal{Y}$ denote the set of all probability distributions over $\mathcal{Y}$. A *simulator* $\textsc{Sim}\colon \Delta\mathcal{Y} \to \mathcal{R}$ is a (randomized) mechanism that takes a probability distribution over $\mathcal{Y}$ as input, and outputs an outcome in the range $\mathcal{R}$. For any fixed distribution $\mathcal{C} \in \Delta\mathcal{Y}$, we sometimes write $\textsc{Sim}_{\mathcal{C}}$ to denote the output distribution $\textsc{Sim}(\mathcal{C})$.

We say that a mechanism $\mathcal{M}$ *perfectly generalizes* if the distribution of its output when run on a sample is close to that of a simulator that did not have access to the sample.

Definition 8 (Perfect Generalization) *Let $\mathcal{R}$ be an arbitrary range and $\mathcal{Y}$ an arbitrary domain. Let $0 \leq \beta < 1$, $\varepsilon \geq 0$, and $0 \leq \delta < 1$. A mechanism $\mathcal{M}\colon \mathcal{Y}^n \to \mathcal{R}$ is* $(\beta, \varepsilon, \delta)$-perfectly generalizing *if for every distribution $\mathcal{C}$ over $\mathcal{Y}$ there exists a* simulator $\textsc{Sim}_{\mathcal{C}}$ *such that with probability at least $1 - \beta$ over the choice of sample $T \sim_{i.i.d.} \mathcal{C}^n$, $\mathcal{M}(T) \approx_{\varepsilon,\delta} \textsc{Sim}_{\mathcal{C}}$.*

Discussion of the generalization notions We will see that all three of the above generalization notions are robust to postprocessing and compatible with adaptive composition,[3] making each of them much more appealing than traditional generalization for learning contexts. Perfect generalization also has an intuitive interpretation as a privacy solution concept that guarantees privacy not just to the individuals in a data sample, but to the sample as a whole (one can think of this as providing privacy to a data provider such as a school or a hospital, when each provider's data comes from the same underlying distribution). Despite the very strong guarantee it gives, we will see that many tasks are achievable under perfect generalization.

2.3. Basic Properties of the Generalization Notions

Here we state several basic properties of the generalization notions defined above. Proofs are deferred to Appendix A.

The following lemma is a useful tool for bounding the closeness parameters between two distributions via an intermediate distribution, such as that of the simulator. It allows us to say (Corollary 10) that for any perfectly generalizing mechanism, any two "typical" samples will induce similar output distributions.

Lemma 9 *Let $\mathcal{J}_1, \mathcal{J}_2, \mathcal{J}_3$ be distributions over an abstract domain $\mathcal{R}$. That is, $\mathcal{J}_1, \mathcal{J}_2, \mathcal{J}_3 \in \Delta\mathcal{R}$. If $\mathcal{J}_1 \approx_{\varepsilon,\delta} \mathcal{J}_2$ and $\mathcal{J}_2 \approx_{\varepsilon',\delta'} \mathcal{J}_3$ where $\varepsilon, \varepsilon' < \ln 2$ then $\mathcal{J}_1 \approx_{\varepsilon+\varepsilon', 2(\delta+\delta')} \mathcal{J}_3$. If $\delta = \delta'$, then $\mathcal{J}_1 \approx_{\varepsilon+\varepsilon', 3\delta} \mathcal{J}_3$.*

Corollary 10 *Suppose that a mechanism $\mathcal{M}\colon \mathcal{Y}^n \to \mathcal{R}$ is $(\beta, \varepsilon, \delta)$-perfectly generalizing, where $\varepsilon < \ln 2$. Let $T_1, T_2 \sim_{i.i.d.} \mathcal{C}^n$ be two independent samples. Then with probability at least $1 - 2\beta$ over the random draws of T_1 and T_2, the following holds*

$$\mathcal{M}(T_1) \approx_{2\varepsilon, 3\delta} \mathcal{M}(T_2).$$

3. Specifically, differentially private algorithms can be adaptively composed in a black box manner, and can be compiled into perfectly generalizing mechanisms (with some loss in their parameters). This gives a recipe for designing perfectly generalizing mechanisms that compose adaptively. Similarly, many methods for guaranteeing robust generalization (including differential privacy, description length bounds, and compression schemes) compose adaptively, giving a recipe for designing robustly generalizing mechanisms that compose adaptively.

We can show that both robust generalization and perfect generalization are robust to postprocessing, i.e., arbitrary interpretation. It is known that differential privacy is also robust to postprocessing (Dwork and Roth, 2014).

Lemma 11 (Robustness to Postprocessing) *Given any (α, β)-robustly generalizing (resp. $(\beta, \varepsilon, \delta)$-perfectly generalizing) mechanism $\mathcal{M}\colon \mathcal{Y}^n \to \mathcal{R}$ and any post-processing procedure $\mathcal{A}\colon \mathcal{R} \to \mathcal{R}'$, the composition $\mathcal{A} \circ \mathcal{M}\colon \mathcal{Y}^n \to \mathcal{R}$ is also (α, β)-robustly generalizing (resp. $(\beta, \varepsilon, \delta)$-perfectly generalizing).*

Theorem 12 says that the composition of multiple $(\beta, \varepsilon, 0)$-perfectly generalizing mechanisms is also perfectly generalizing, where the β and ε parameters "add up".

Theorem 12 (Basic Composition) *Let $\mathcal{M}_i\colon \mathcal{Y}^n \to \mathcal{R}_i$ be $(\beta_i, \varepsilon_i, 0)$-perfectly generalizing for $i = 1, \ldots, k$. The composition $\mathcal{M}_{[k]}\colon \mathcal{Y}^n \to \mathcal{R}_1 \times \cdots \times \mathcal{R}_k$, defined as $\mathcal{M}_{[k]}(T) = (\mathcal{M}_1(T), \ldots \mathcal{M}_k(T))$ is $(\sum_{i=1}^k \beta_i, \sum_{i=1}^k \varepsilon_i, 0)$-perfectly generalizing.*

A very recent work by Bassily and Freund (2016) studies the notion of *typical stability*, which generalizes perfect generalization. In particular, a mechanism is perfectly generalizing if it is typically stable with respect to product distributions $\mathcal{D}^n$. They show the class of typically stable mechanisms is closed under adaptive composition, implying an adaptive composition theorem for perfectly generalizing mechanisms.[4]

3. Robust Generalization via Compression Schemes

In this section, we present a new technique, based on the idea of *compression bounds*, for designing learning algorithms with the robust generalization guarantee. Recent work (Dwork et al., 2015c,a; Bassily et al., 2016) gives two other techniques for obtaining robust generalizing mechanisms. As we will see, our new technique allows one to learn hypothesis classes under robust generalization for which the two previous techniques do not apply. More surprisingly, we show that any PAC/agnostically learnable hypothesis class can also be learned under robust generalization with nearly optimal sample complexity.

We first give a definition for what it means to learn a hypothesis under robust generalization.

Definition 13 (RG PAC/Agnostic Learning) *A hypothesis class $\mathcal{H}$ over domain $\mathcal{X}$ is* PAC/agnostically learnable under robust generalization (RG-PAC/agnostically learnable) *if there exists a polynomial $n_{\mathcal{H}}\colon \mathbb{R}^4 \to \mathbb{R}$ and a learning algorithm $\mathcal{A}$ such that for all $\alpha, \beta, \varepsilon, \delta \in (0, 1/2)$, given inputs $\alpha, \beta, \varepsilon, \delta$ and a sample $S_L \in \mathcal{X}_L^n$ where $n \geq n_{\mathcal{H}}(1/\alpha, 1/\varepsilon, \log(1/\beta), \log(1/\delta))$, the algorithm $\mathcal{A}$ is an (α, β)-accurate PAC/agnostic learner, and is (ε, δ)-robustly generalizing.*

3.1. Compression Learners

For any function $k\colon \mathbb{N} \to \mathbb{N}$, we say that a hypothesis class has a compression scheme of size k if any arbitrary set S_L of n labelled examples can be mapped to a *sequence* of $k(n)$ input examples, from which it is possible to compute an empirical risk minimizer for S_L.

4. A previous version of our paper contained an error in the proof of the adaptive composition theorem for perfect generalization. We are grateful to Raef Bassily and Adam Smith for bringing this to our attention.

Definition 14 (Compression Scheme Littlestone and Warmuth (1986)) *Let $\mathcal{H}$ be a hypothesis class and let $k\colon \mathbb{N} \to \mathbb{N}$. We say that $\mathcal{H}$ has a* compression scheme *of size k if for all $n \in \mathbb{N}$, there exists an integer $k' \leq k(n)$, a* compression algorithm *$A\colon \mathcal{X}_L^n \to \mathcal{X}_L^{k'}$ and an* encoding algorithm *$B\colon \mathcal{X}_L^{k'} \to \mathcal{H}$ such that for any arbitrary set S_L of n labelled examples, A will select a sequence of examples $A(S_L) = (z_{i_1}, z_{i_2}, \ldots, z_{i_{k'}}) \in S_L^{k'}$, and B will output a hypothesis $h' = B(A(S_L))$ that is an* empirical risk minimizer*; i.e. $err(S_L, h') \leq err(S_L, h)$ for all $h \in \mathcal{H}$. We will call the algorithm $\mathcal{L} = B \circ A$ a compression learner of size k for the hypothesis class $\mathcal{H}$.*[5]

Remark 15 *A natural extension to the compression scheme defined above is* approximate *compression schemes David et al. (2016), which produce approximate empirical risk minimizers rather than exact empirical risk minimizers. A particularly simple and naive approximate compression scheme results from subsampling: since it is possible to produce an ε-approximate empirical risk minimizer for any function drawn from a VC-class of dimension d using $O(d/\varepsilon^2)$ samples, it immediately follows that every VC-class of dimension d admits an ε-approximate compression scheme of size $k = O(d/\varepsilon^2)$. As we will see, such a compression scheme is in general quite inefficient in terms of sample complexity, and by using a more sophisticated boosting-based compression scheme David et al. (2016), it is possible to obtain robust generalization with nearly optimal sample complexity for every VC-class.*

Next, we want to show that any compression learner of small size satisfies robust generalization. As an intermediate step, we recall the following result, which follows from a standard application of a concentration bound.

Lemma 16 (see, e.g., Shalev-Shwartz and Ben-David (2014) Theorem 30.2) *Let $n, k' \in \mathbb{N}$ such that $n \geq 2k'$. Let $A\colon \mathcal{X}_L^n \to \mathcal{X}_L^{k'}$ be an algorithm that takes a sample S_L of n labelled examples as input, and selects a sequence of labelled examples $A(S_L) = (z_{i_1}, z_{i_2}, \ldots, z_{i_{k'}}) \in S_L^{k'}$ of length k'. Let algorithm $B\colon \mathcal{X}_L^{k'} \to (\mathcal{X} \to \{0,1\})$ take a sequence of k' labelled examples and return a hypothesis.*

For any random sample $S_L \sim_{i.i.d.} \mathcal{D}_L^n$, let $V_L = \{z \mid z \notin A(S_L)\}$ be the set of examples not selected by A, and write V for the unlabelled version of V_L. Let $h = B(A(S_L))$ be the hypothesis output by B. Then, with probability of at least $1 - \delta$ over the random draws of S_L and the randomness of A and B, we have

$$|h(V) - h(\mathcal{D})| \leq \sqrt{h(V)\frac{4k' \log(2n/\delta)}{n}} + \frac{8k' \log(2n/\delta)}{n}$$

Recall that $h(\mathcal{D}) = \mathbb{E}_{x\sim\mathcal{D}}[h(x)]$ denotes the expected value of h, and $h(V) = \frac{1}{n-k'}\sum_{x\in V} h(x)$ is the average value of h over the examples in V.

This theorem is useful in analyzing the guarantees of a compression learner. If we interpret A as a compression algorithm, and B as an encoding algorithm that outputs a hypothesis h, Lemma 16 says that the empirical error of h over the remaining subset V is close to its true error.

However, we can also interpret algorithm B as an adversary who is trying to overfit a hypothesis to the input sample S_L. Since the hypothesis output by a compression algorithm is uniquely

5. Note that this definition of variable-length compression scheme (where the number of examples output by the compression algorithm depends on the input sample size) is more general than the one defined in Littlestone and Warmuth (1986).

determined by the sequence of examples output by the compression algorithm A, we could think of the adversary post-processing the size-k' sequence of examples that defines the output hypothesis. Therefore, it suffices to show that the compression algorithm A is robustly generalizing. We will establish this by showing that any algorithm that outputs a small sequence of the input sample is robustly generalizing:

Lemma 17 *Let n, k' be integers, $\varepsilon, \delta > 0$, and let $A\colon \mathcal{X}_L^n \to \mathcal{X}_L^{k'}$ be an algorithm that takes any set $S_L \in \mathcal{X}_L^n$ as input and outputs a sequence $T \in S_L^{k'}$ of size k'. Then A is (ε, δ)-robustly generalizing for*

$$\varepsilon = \sqrt{\frac{4k' \log(n/\delta)}{n}} + \frac{8k' \log(2n/\delta)}{n} + \frac{k'}{n}.$$

Proof We will appeal to Theorem 16. Let $F\colon \mathcal{X}_L^{k'} \to \{\mathcal{X} \to \{0,1\}\}$ be a deterministic mapping from samples of size k' to hypotheses. Let $S_L \sim_{i.i.d.} \mathcal{D}^n$ be a random sample of size n, $T = A(S_L)$ be the sequence output by the compression algorithm, V be the examples (without labels) not selected by A, and $f = F(T)$ be the function output by the adversary. By the result of Lemma 16, we know that with probability at least $1 - \delta$ over the random draws of S_L, the following holds,

$$|f(V) - f(\mathcal{D})| \leq \sqrt{\frac{4k' \log(2n/\delta)}{n}} + \frac{8k' \log(2n/\delta)}{n} \equiv C$$

Let S be the examples in S_L but without labels. By the triangle inequality we have

$$\begin{aligned}
|f(S_L) - f(\mathcal{D})| &\leq \frac{1}{n}\left|\sum_{z \in S_L} (f(z) - f(\mathcal{D}))\right| \\
&= \frac{1}{n}\left|\sum_{z \in V} (f(z) - f(\mathcal{D})) + \sum_{z \notin V} (f(z) - f(\mathcal{D}))\right| \\
&\leq \frac{1}{n}\left|\sum_{z \in V} (f(z) - f(\mathcal{D}))\right| + \frac{1}{n}\left|\sum_{z \notin V} (f(z) - f(\mathcal{D}))\right| \\
&\leq \frac{C\,n}{n} + \frac{k'}{n} = C + \frac{k'}{n} \qquad (1)
\end{aligned}$$

which recovers our stated bound. ■

Now we are ready to show that any hypothesis class that admits a compression scheme of small size is learnable under robust generalization.

Theorem 18 (Compression implies RG Learnability) *Let $\mathcal{H}$ be a hypothesis class with a compression scheme of size $k\colon \mathbb{N} \to \mathbb{N}$, and let $\mathcal{A}\colon \mathcal{H} \to \mathcal{H}$ be any adversary. Then given any input sample $S_L \sim_{i.i.d.} \mathcal{D}_L^n$ of size n, the compression learner $\mathcal{L}$ for $\mathcal{H}$ outputs an hypothesis h such that with probability at least $1 - \delta$, the error satisfies $err(h) \leq \min_{h'} err(h') + \varepsilon$, and the adversary outputs a hypothesis $f = \mathcal{A}(h)$ that satisfies $|f(S_L) - f(\mathcal{D})| \leq \varepsilon$ with*

$$\varepsilon = O\left(\sqrt{\frac{k(n)\log(n/\delta)}{n}}\right),$$

as long as $n \geq 8k(n)\,\log(2n/\delta)$.

Proof Note that when $n \geq 8k(n)\ \log(2n/\delta)$, the bound on ε in Lemma 17 becomes $O\left(\sqrt{\frac{k(n)\log(n/\delta)}{n}}\right)$. Then by applying Lemma 17, we can guarantee that $|f(S_L) - f(\mathcal{D})| \leq \varepsilon$ with probability at least $1-\delta$. Then the accuracy guarantee of the learner's output hypothesis directly follows by setting $\mathcal{A}$ to be the identity map. ■

We can also show that compression learners continue to give robust generalization under adaptive composition.

Theorem 19 (Adaptive Composition for Compression Learners) *Let $\mathcal{M}_{[m]} \colon \mathcal{X}^n \to \mathcal{H}^m$ be an adaptive composition of compression schemes such that for any $S \in \mathcal{X}^n$, $\mathcal{M}_{[m]}(S) = (h_1, \ldots, h_m)$, where $h_1 = \mathcal{M}_1(S)$, $h_2 = \mathcal{M}_2(S; h_1), \ldots, h_m = \mathcal{M}(S; h_1, \ldots, h_{m-1})$, where $\mathcal{M}_i(\cdot; h_1, \ldots, h_{i-1})$ is a compression learner of size k_i for all choices of $h_1, \ldots, h_{i-1}$. Let $k = \sum_{i=1}^m k_i$. Then $\mathcal{M}_{[m]}$ is (ε, δ)-robustly generalizing, where*

$$\varepsilon = O\left(\sqrt{\frac{k\log(n/\beta)}{n}}\right),$$

as long as $n \geq 8k\log(2n/\beta)$.

Proof For each $\mathcal{M}_i$, we can write it as $\mathcal{M}_i(\cdot; h_1, \ldots, h_{i-1}) = (B_i \circ A_i)$, where A_i is the compression algorithm and B_i is the encoding algorithm. Note that the sequence of output hypotheses is just a postprocessing of the sequence of examples output by the compression algorithms—that is, given the sequence of examples output by the compression algorithms, we can uniquely determine the set of output hypotheses. So it suffices to prove that the adaptive composition of the compression algorithms satisfies robust generalization. Note that the composed compression algorithms can be viewed as a single compression algorithm that releases a sequence of examples of length k. By directly applying Lemma 17, we recover the stated bound. ■

3.2. Robust Generalization via Differential Privacy and Description Length

We briefly review two existing techniques for obtaining algorithms with robust generalization guarantees, from the recent line of work starting with Dwork et al. (2015c), and followed by Dwork et al. (2015a); Blum and Hardt (2015); Bassily et al. (2016). Here we will rephrase their results in terms of robust generalization (this terminology is new to the present paper).

First, it is known that differential privacy implies robust generalization.

Theorem 20 (Bassily et al. (2016)) *Let $\mathcal{M} \colon \mathcal{X}_L^n \to \mathcal{R}$ be a (ε, δ)-differentially private mechanism for $n \geq O(\ln(1/\delta)/\varepsilon^2)$. Then $\mathcal{M}$ also satisfies $(O(\varepsilon), O(\delta/\varepsilon))$-robust generalization.*

Algorithms with a small output range (i.e., each output can be described using a small number of bits) also enjoy robust generalization.

Theorem 21 (Dwork et al. (2015a)) *Let $\mathcal{M} \colon \mathcal{X}_L^n \to \mathcal{R}$ be a mechanism such that $|\mathcal{R}|$ is bounded. Then $\mathcal{M}$ satisfies (α, β)-robust generalization, with $\alpha = \sqrt{\frac{\ln(|\mathcal{R}|/\beta)}{2n}}$.*

3.3. Case Study: Proper Threshold Learning

Next, we consider the problem of properly learning *thresholds* in the PAC setting. We will first note that when the domain size is infinite, there is no proper PAC learner that is differentially private or has finite output range. In contrast to these impossibility results, we show that the class of threshold functions admits a simple compression scheme, and hence a PAC learning algorithm that satisfies robust generalization. This result, in particular, gives a separation between the power of learning under robust generalization and that of learning under differential privacy.

Consider the hypothesis class of *threshold functions* $\{h_x\}_{x\in\mathcal{X}}$ over a totally ordered domain $\mathcal{X}$, where $h_x(y) = 1$ if $y \leq x$ and $h_x(y) = 0$ if $y > x$. We will first recall an impossibility result for privately learning thresholds.

Theorem 22 (Bun et al. (2015) Theorem 6.2) *Let $\alpha > 0$ be the accuracy parameter (as in Theorem 2). For every $n \in \mathbb{N}$, and $\delta \leq 1/(1500n^2)$, any $(1/2, \delta)$-differentially private and $(\alpha, 1/8)$-accurate (proper) PAC learner for threshold functions requires sample complexity $n = \Omega\left(\log^* |\mathcal{X}|/\alpha\right)$.*

In particular, the result of Theorem 22 implies that there is no private proper PAC learner for threshold functions over an infinite domain. Similarly, we can show that there is no proper PAC learner for thresholds that has a finite outcome range.

Lemma 23 *Let $\mathcal{H}$ be the hypothesis class of threshold functions. For any $n \in \mathbb{N}$ and any learner $\mathcal{M} \colon \mathcal{X}_L^n \to \mathcal{H}'$ such that the output hypothesis class $\mathcal{H}'$ is a subset of $\mathcal{H}$ and has bounded cardinality, there exits a distribution $\mathcal{D} \in \Delta\mathcal{X}$ such that the output hypothesis has true error $err(h) \geq 1/2$.*

Proof Let $|\mathcal{H}'| = m$. We can write $\mathcal{H}' = \{h_{x_1}, h_{x_2}, \ldots, h_{x_m}\}$ such that $x_1 < x_2 < \ldots < x_m$. Let y, z be points such that $x_1 < y < z < x_2$. Let $\mathcal{D}$ be a distribution over $\mathcal{X}$ that puts half of the probability mass on y and the other half on z. Suppose our target hypothesis is $c = h_y$. Then $c(y) = 1$ and $c(z) = 0$. Note that for each $h \in \mathcal{H}'$, it must be case that $h(y) = h(z)$, and thus its true error must be at least $1/2$. ■

Now we will show that the class of threshold functions can be properly PAC learned under the constraint of robust generalization *even when* the domain size is infinite.

Theorem 24 *Let $\mathcal{H}$ be the hypothesis class of threshold functions. There exists a compression learner for $\mathcal{H}$ such that when given a input sample $S_L \sim_{i.i.d.} \mathcal{D}_L^n$ of size n, it is both (ε, δ)-accurate and (ε, δ)-robustly generalizing for any $\delta \in (0, 1)$ and*

$$\varepsilon = O\left(\sqrt{\frac{\log(n/\delta)}{n}}\right)$$

as long as $n \geq 8\log(2n/\delta)$.

Proof Consider the compression function A, that, given a sample, outputs the largest positive example s_+ in the sample. Then consider the encoding function B that, given any example s_+, returns the threshold function h_{s_+}. Such a threshold function will correctly label all the examples in the sample. This gives us a compression scheme of size 1 for the class of threshold functions. Then the result follows directly from Theorem 18. ■

3.4. Every Learnable Class is Learnable under Robust Generalization

Finally, we will show that any PAC-learnable hypothesis class can be learned under robust generalization with a logarithmic blowup in the sample complexity. We will rely on a result due to David et al. (2016), which shows that any learnable class admits a compression scheme of size scaling logarithmically in the input sample size n.

Theorem 25 (David et al. (2016) (see Theorem 3.1)) *Let $\mathcal{H}$ be a hypothesis class that is PAC/agnostically learnable with VC-dimension d; then it has a compression scheme of size*

$$k(n) = O(d\log(n)\log\log(n) + d\log(n)\log(d)).$$

Our result then follows directly from Theorem 18 and Theorem 25.

Theorem 26 *Let $\mathcal{H}$ be a hypothesis class. Suppose that $\mathcal{H}$ is PAC/agnostically learnable with $\mathsf{VCDIM}(\mathcal{H}) = d$. Then there exists a compression learner for $\mathcal{H}$ such that when given input sample $S_L \sim_{i.i.d.} \mathcal{D}_L^n$, $\mathcal{L}$ is both (ε, δ)-accurate and (ε, δ)-robustly generalizing for any $\delta \in (0, 1)$ and $\varepsilon = \widetilde{O}\left(\sqrt{d/n}\right)$ as long as $n \geq 16d\log(d)\log^3(n/\delta)$.*

Remark 27 *Note that we can obtain a similar result with the approximate compression scheme of subsampling. In particular, for every VC-class of dimension d, the compression learner that uses subsampling as its compression algorithm is both (ε, δ)-accurate and (ε, δ)-robustly generalizing with:*

$$\varepsilon = O\left(\left(\frac{d\log(n/\delta)}{n}\right)^{1/4}\right)$$

which is polynomial, but is quadratically suboptimal.

4. Learning under Perfect Generalization

In this section, we will focus on the problem of agnostic learning under the constraint of perfect generalization. Our main result gives a perfectly generalizing generic learner in the settings where the domain $\mathcal{X}$ or the hypothesis class $\mathcal{H}$ has bounded size. The sample complexity will depend logarithmically on these two quantities. Furthermore, we give a reduction from any perfectly generalizing learner to a differentially private learner that preserves the sample complexity bounds (up to constant factors). This allows us to carry over lower bounds for differentially private learning to learning under perfect generalization. In particular, we will show that proper threshold learning with unbounded domain size is impossible under perfect generalization.

We will first define what it means to learn a hypothesis under perfect generalization.

Definition 28 (PG PAC/Agnostic Learning) *A hypothesis class $\mathcal{H}$ over domain $\mathcal{X}$ is PAC/agnostically learnable under perfect generalization (PG-PAC/agnostically learnable) if there exists a polynomial $n_{\mathcal{H}} \colon \mathbb{R}^5 \to \mathbb{R}$ and a learning algorithm $\mathcal{A}$ such that for all $\alpha, \gamma, \beta, \varepsilon, \delta \in (0, 1/2)$, given inputs $\alpha, \gamma, \beta, \varepsilon, \delta$ and a sample $S_L \in \mathcal{X}_L^n$ where $n \geq n_{\mathcal{H}}(1/\alpha, 1/\varepsilon, \log(1/\gamma), \log(1/\beta), \log(1/\delta))$, the algorithm $\mathcal{A}$ is an (α, γ)-accurate PAC/agnostic learner, and is $(\beta, \varepsilon, \delta)$-perfectly generalizing.*

4.1. Generic PG Agnostic Learner

Now we present a generic perfectly generalizing agnostic learner, which is based on the *exponential mechanism* of McSherry and Talwar (2007) and analogous to the generic learner of Kasiviswanathan et al. (2011).

Our learner, formally presented in Algorithm 1, takes generalization parameters ε, β, a sample of n labelled examples $S_L \sim_{i.i.d.} \mathcal{D}_L^n$, and a hypothesis class $\mathcal{H}$ as input, and samples a random hypothesis with probability that is exponentially biased towards hypotheses with small empirical error. We show that this algorithm is perfectly generalizing.

Algorithm 1 Generic Agnostic Learner $\mathcal{A}(\beta, \varepsilon, S_L, \mathcal{H})$

Output $h \in \mathcal{H}$ with probability proportional to $\exp\left(\frac{-\sqrt{|S_L|}\cdot\varepsilon\cdot err(S_L,h)}{\sqrt{2\ln(2|\mathcal{H}|/\beta)}}\right)$

Lemma 29 *Given any $\varepsilon, \beta > 0$ and finite hypothesis class $\mathcal{H}$, the learning algorithm $\mathcal{A}(\beta, \varepsilon, \cdot, \cdot)$ is $(\beta, \varepsilon, 0)$-perfectly generalizing.*

Proof Let $S_L \sim_{i.i.d.} \mathcal{D}_L^n$ be a labelled random sample of size n. Note that since each (x_i, y_i) in S_L is drawn from the underlying distribution $\mathcal{D}_L$, we know that for each $h \in \mathcal{H}$,

$$\mathop{\mathbb{E}}_{S_L \sim_{i.i.d.} \mathcal{D}_L^n}[err(S_L, h)] = err(h).$$

Fix any $h \in \mathcal{H}$. Then by a Chernoff-Hoeffding bound, we know that with probability at least $1 - \beta/|\mathcal{H}|$, the following holds:

$$|err(S_L, h) - err(h)| \leq \sqrt{\frac{1}{2n}\ln\left(\frac{2|\mathcal{H}|}{\beta}\right)}. \tag{2}$$

Applying a union bound, we know that the above holds for all $h \in \mathcal{H}$ with probability at least $1 - \beta$. We will condition on this event for the remainder of the proof. Now consider the following randomized simulator:

$$\textsc{Sim}(\mathcal{D}_L): \text{ output } h \in \mathcal{H} \text{ with probability proportional to } \exp\left(\frac{-\varepsilon\cdot\sqrt{n}\cdot err(h)}{\sqrt{2\ln(2|\mathcal{H}|/\beta)}}\right).$$

We want to show that the output distributions satisfy $\mathcal{A}(\beta, \varepsilon, S_L) \approx_{\varepsilon,0} \textsc{Sim}(\mathcal{D}_L)$, where $S_L \sim_{i.i.d.} \mathcal{D}_L^n$ is a labelled random sample of size n. Let $Z = \sum_{h\in\mathcal{H}} \exp\left(\frac{-\varepsilon\sqrt{n}\cdot err(S_L,h)}{\sqrt{2\ln(2|\mathcal{H}|/\beta)}}\right)$ and $Z' =$

$\sum_{h\in\mathcal{H}} \exp\left(\frac{-\varepsilon\cdot\sqrt{n}\cdot err(h)}{\sqrt{2\ln(2|\mathcal{H}|/\beta)}}\right)$. For each $h \in \mathcal{H}$,

$$\begin{aligned}
\frac{\Pr[\mathcal{A}(\beta,\varepsilon,S_L,\mathcal{H})=h]}{\Pr[\textsc{Sim}(\mathcal{D}_L)=h]} &= \frac{\exp\left(\frac{-\varepsilon\cdot\sqrt{n}\cdot err(S_L,h)}{\sqrt{2\ln(2|\mathcal{H}|/\beta)}}\right)/Z}{\exp\left(\frac{-\varepsilon\cdot\sqrt{n}\cdot err(h)}{\sqrt{2\ln(2|\mathcal{H}|/\beta)}}\right)/Z'} \\
&= \exp\left(\frac{\varepsilon\cdot\sqrt{n}\left(err(h)-err(S_L,h)\right)}{\sqrt{2\ln(2|\mathcal{H}|/\beta)}}\right)\cdot\frac{Z'}{Z} \\
&\leq \exp\left(\frac{\varepsilon}{2}\right)\exp\left(\frac{\varepsilon}{2}\right)\cdot\frac{Z}{Z} \\
&= \exp(\varepsilon).
\end{aligned}$$

A symmetric argument would also show that $\frac{\Pr[\textsc{Sim}(\mathcal{D}_L)=h]}{\Pr[\mathcal{A}(\beta,\varepsilon,S_L,\mathcal{H})=h]} \leq \exp(\varepsilon)$. Therefore, $\mathcal{A}(\beta,\varepsilon,\cdot,\cdot)$ is $(\beta,\varepsilon,0)$-perfectly generalizing. ■

Theorem 30 *Let $\mathcal{H}$ be a finite hypothesis class and $\alpha,\gamma>0$. Then the generic learner Algorithm 1 instantiated as $\mathcal{A}(\gamma,\varepsilon,\cdot,\mathcal{H})$ is (α,γ)-accurate as long as the sample size*

$$n \geq \frac{6}{\varepsilon^2\alpha^2}\left(\ln(2|\mathcal{H}|)+\ln(1/\gamma)\right)^3.$$

Proof Let $S_L \sim_{i.i.d.} \mathcal{D}_L^n$, and let the algorithm $\mathcal{A}(\gamma,\varepsilon,S_L,\mathcal{H})$ be the Generic Agnostic Learner of Algorithm 1. Consider the event $E = \{\mathcal{A}(\gamma,\varepsilon,S_L,\mathcal{H}) = h \mid err(h) > \alpha + \mathrm{OPT}\}$, where α is our target accuracy parameter. We want to show that $\Pr[E] \leq \gamma$ as long as the sample size n satisfies the stated bound.

By a Chernoff-Hoeffding bound (similar to Equation (2)), we have that with probability at least $1-\gamma/2$, the following condition holds for each $h \in \mathcal{H}$:

$$|err(S_L,h) - err(h)| \leq \sqrt{\frac{1}{2n}\ln\left(\frac{4|\mathcal{H}|}{\gamma}\right)} \equiv B(n).$$

We will condition on the event above. Let $h^* = \arg\min_{h'\in\mathcal{H}} err(h')$ and let $\mathrm{OPT} = err(h^*)$, then

$$\min_{h'\in\mathcal{H}} err(S_L,h') \leq err(S_L,h^*) \leq err(h^*) + B(n) = \mathrm{OPT} + B(n)$$

Recall that for each $h \in \mathcal{H}$, the probability that the hypothesis output by $\mathcal{A}(\gamma, \varepsilon, S_L, \mathcal{H})$ is h is,

$$\begin{aligned}
&\frac{\exp\left(-\varepsilon\sqrt{n} \cdot err(S_L, h)/\sqrt{2\ln(2|\mathcal{H}|/\gamma)}\right)}{\sum_{h' \in \mathcal{H}} \exp\left(-\varepsilon\sqrt{n} \cdot err(S_L, h')/\sqrt{2\ln(2|\mathcal{H}|/\gamma)}\right)} \\
&\leq \frac{\exp\left(-\varepsilon\sqrt{n} \cdot err(S_L, h)/\sqrt{2\ln(2|\mathcal{H}|/\gamma)}\right)}{\max_{h' \in \mathcal{H}} \exp\left(-\varepsilon\sqrt{n} \cdot err(S_L, h')/\sqrt{2\ln(2|\mathcal{H}|/\gamma)}\right)} \\
&= \exp\left(-\varepsilon\sqrt{n} \cdot (err(S_L, h) - \min_{h' \in \mathcal{H}} err(S_L, h'))/\sqrt{2\ln(2|\mathcal{H}|/\gamma)}\right) \\
&\leq \exp\left(-\varepsilon\sqrt{n} \cdot (err(S_L, h) - \mathrm{OPT} - B(n))/\sqrt{2\ln(2|\mathcal{H}|/\gamma)}\right).
\end{aligned}$$

Taking a union bound, we know that the probability that $\mathcal{A}(\gamma, \varepsilon, S_L, \mathcal{H})$ outputs a hypothesis h with empirical error $err(S_L, h) \geq \mathrm{OPT} + 2B(n)$ is at most $|\mathcal{H}| \exp\left(-\varepsilon\sqrt{n}B(n)/\sqrt{2\ln(2|\mathcal{H}|/\gamma)}\right)$.

Set $B(n) = \alpha/3$, and the event $E = \{\mathcal{A}(\gamma, \varepsilon, S_L, \mathcal{H}_d) = h \mid err(h) > \alpha + \mathrm{OPT}\}$ implies

$$err(S_L, h) \geq \mathrm{OPT} + 2\alpha/3 = \mathrm{OPT} + 2B(n) \quad \text{or} \quad |err(S_L, h) - err(h)| \geq \alpha/3 = B(n).$$

It is sufficient to set n large enough to bound the probabilities of these two events. Further if we a sample size $n \geq \frac{6}{\varepsilon^2\alpha^2}\left(\ln(2|\mathcal{H}|/\gamma)\right)^3$, both probabilities are bounded by $\gamma/2$, which means we must have $\Pr[E] \leq \gamma$. ■

4.2. PG Learning with VC Dimension Sample Bounds

We can also extend the sample complexity bound in Theorem 30 to one that is dependent on the VC-dimension of the hypothesis class $\mathcal{H}$, but resulting bound will have a logarithmic dependence on the size of the domain $|\mathcal{X}|$.

Corollary 31 *Every hypothesis class $\mathcal{H}$ with finite VC dimension is PG agnostically learnable with a sample size of $n = O\left((\mathsf{VCDIM}(\mathcal{H}) \cdot \ln|\mathcal{X}| + \ln\frac{1}{\beta})^3 \cdot \frac{1}{\varepsilon^2\alpha^2}\right)$.*

Proof By Sauer's lemma (see e.g., Kearns and Vazirani (1994)), we know that there are at most $O(|\mathcal{X}|^{\mathsf{VCDIM}(\mathcal{H})})$ different labelings of the domain $\mathcal{X}$ by the hypotheses in $\mathcal{H}$. We can run the exponential mechanism over such a hypothesis class $\mathcal{H}'$ with cardinality $|\mathcal{H}'| = O\left(|\mathcal{X}|^{\mathsf{VCDIM}(\mathcal{H})}\right)$. The complexity bound follows from Theorem 30 directly. ■

4.3. Limitations of PG learning

We have so far given a generic agnostic learner with perfect generalization in the cases where either $|\mathcal{X}|$ or $|\mathcal{H}|$ is finite. We now show that the finiteness condition is necessary, by revisiting the threshold learning problem in Section 3.3. In particular, we will show that when both of the domain size and hypothesis class are infinite, properly learning thresholds under perfect generalization is

impossible. Our result crucially relies on a reduction from a perfectly generalizing learner to a differentially private learner, which allows us to apply lower bound results of differentially private learning(such as Theorem 22) to PG agnostic learning.

First, let's consider the reduction in Algorithm 2, which is a black-box mechanism that takes as input a perfectly generalizing mechanism $\mathcal{M}\colon \mathcal{X}_L^n \to \mathcal{R}$ and a labelled sample $S_L \in \mathcal{X}_L^n$, and outputs an element of $\mathcal{R}$. We show that this new mechanism $\mathcal{M}'(\mathcal{M}, \cdot)$ is differentially private.

Algorithm 2 $\mathcal{M}'(\mathcal{M}\colon \mathcal{X}_L^n \to \mathcal{R}, S_L \in \mathcal{X}_L^n)$

Let $\mathcal{E}_{S_L}$ be the empirical distribution that assigns weight $1/n$ to each of the data points in S_L
Sample $T_L \sim_{i.i.d.} (\mathcal{E}_{S_L})^n$
Output $\mathcal{M}(T_L) \in \mathcal{R}$

Theorem 32 *Let $\beta < 1/2e$ and $\varepsilon \leq \ln(2)$, and $\mathcal{M}$ be a $(\beta, \varepsilon, \delta)$-perfectly generalizing mechanism, then the mechanism $\mathcal{M}'(\mathcal{M}, \cdot)$ of Algorithm 2 is $(4\varepsilon, 16\delta + 2\beta)$-differentially private.*

Proof Let $S_L, S_L' \in \mathcal{X}^n$ be neighboring databases that differ on the ith entry, and let $\mathcal{E}_{S_L}$ and $\mathcal{E}_{S_L'}$ denote their corresponding empirical distributions. Since $\mathcal{M}$ is $(\beta, \varepsilon, \delta)$-perfectly generalizing, there exists a simulator SIM such that with probability at least $1-\beta$ over choosing $T_L \sim_{i.i.d.} (\mathcal{E}_{S_L})^n$,

$$\mathcal{M}(T_L) \approx_{\varepsilon,\delta} \text{SIM}. \tag{3}$$

Similarly, there exists a simulator SIM$'$ such that with probability at least $1 - \beta$ over choosing $T_L' \sim_{i.i.d.} (\mathcal{E}_{S_L'})^n$,

$$\mathcal{M}(T_L') \approx_{\varepsilon,\delta} \text{SIM}'. \tag{4}$$

Let $R_1 = \{T_L \in \mathcal{X}_L^n \mid \mathcal{M}(T_L) \approx_{\varepsilon,\delta} \text{SIM}\}$ and $R_2 = \{T_L' \in \mathcal{X}_L^n \mid \mathcal{M}(T_L') \approx_{\varepsilon,\delta} \text{SIM}'\}$. We want to first show that there exists a dataset T_L^* such that $T_L^* \in R_1$ and $T_L^* \in R_2$.

Let $\{(x_i, y_i)\} = S_L \setminus S_L'$ and let $R_3 = \{T_L \in \mathcal{X}_L^n \mid (x_i, y_i) \notin T_L\}$.

$$\Pr_{T_L \sim_{i.i.d.} (\mathcal{E}_{S_L})^n}[T_L \in R_3] = \Pr_{T_L' \sim_{i.i.d.} (\mathcal{E}_{S_L'})^n}[T_L' \in R_3] = (1 - 1/n)^n \approx 1/e.$$

Moreover, for any $T \in R_3$,

$$\Pr_{T_L \sim_{i.i.d.} (\mathcal{E}_{S_L})^n}[T_L = T] = \Pr_{T_L' \sim_{i.i.d.} (\mathcal{E}_{S_L'})^n}[T_L' = T]$$

Note that any dataset T_L in R_3 also lies in the supports of both $(\mathcal{E}_{S_L})^n$ and $(\mathcal{E}_{S_L'})^n$. It follows that

$$\begin{aligned}
&\Pr_{T_L \sim_{i.i.d.} (\mathcal{E}_{S_L}^n)}[T_L \in (R_1 \cap R_2)] \\
\geq &\Pr_{T_L \sim_{i.i.d.} (\mathcal{E}_{S_L}^n)}[T_L \in (R_1 \cap R_2 \cap R_3)] \\
\geq &\Pr_{T_L \sim_{i.i.d.} (\mathcal{E}_{S_L}^n)}[T_L \in R_3] - \Pr_{T_L \sim_{i.i.d.} (\mathcal{E}_{S_L}^n)}[T_L \in R_3 \text{ and } T_L \notin R_1] - \Pr_{T_L \sim_{i.i.d.} (\mathcal{E}_{S_L}^n)}[T_L \in R_3 \text{ and } T_L \notin R_2] \\
\geq &1/e - \beta - \beta > 0
\end{aligned}$$

Therefore, there exists a $T_L^* \in R_1 \in R_2$. Since $\mathcal{M}$ is perfectly generalizing, we have that,

$$\mathcal{M}(T_L^*) \approx_{\varepsilon,\delta} \textsc{Sim} \quad \text{and} \quad \mathcal{M}(T_L^*) \approx_{\varepsilon,\delta} \textsc{Sim}'. \tag{5}$$

This means with probability at least $1 - 2\beta$, we also have

$$\mathcal{M}(T_L) \approx_{\varepsilon,\delta} \textsc{Sim} \approx_{\varepsilon,\delta} \mathcal{M}(T_L^*) \approx_{\varepsilon,\delta} \textsc{Sim}' \approx_{\varepsilon,\delta} \mathcal{M}(T_L').$$

By Lemma 9, with probability at least $1 - 2\beta$,

$$\mathcal{M}'(S_L) = \mathcal{M}(T_L) \approx_{4\varepsilon,16\delta} \mathcal{M}(T_L') = \mathcal{M}'(S_L').$$

Therefore, $\mathcal{M}'$ is $(4\varepsilon, 16\delta + 2\beta)$-differentially private. ■

Theorem 33 *Let $\mathcal{H}$ be a hypothesis class with finite VC dimension d. Suppose that $\mathcal{H}$ admits an agnostic learner $\mathcal{M}\colon \mathcal{X}_L^n \to \mathcal{H}$ that is (α,γ)-accurate and $(\beta,\varepsilon,\delta)$-perfectly generalizing. Then algorithm $\mathcal{M}'(\mathcal{M},\cdot)$ defined as in Algorithm 2 is $(4\varepsilon, 16\delta + 2\beta)$-differentially private, and is also an $(O(\alpha), O(\gamma))$-accurate agnostic learner for $\mathcal{H}$.*

We will rely on the following result on the uniform convergence properties of any hypothesis class with finite VC dimension.

Theorem 34 (see, e.g., Shalev-Shwartz and Ben-David (2014) Theorem 6.8) *Let $\mathcal{H}$ be a hypothesis class of VC dimension $d < \infty$. Then there are constants C_1 and C_2 such that the following holds:*

1. *Fix any $\alpha, \gamma > 0$. Let $S_L \sim_{i.i.d.} \mathcal{D}_L^n$, then with probability at least $1 - \gamma$, $|err(S_L, h) - err(h)| \leq \alpha$ for all $h \in \mathcal{H}$, as long as*

$$n \geq C_1 \frac{d + \log(1/\gamma)}{\alpha^2}$$

2. *Any agnostic learner that is (α,γ)-accurate requires a sample of size*

$$n \geq C_2 \frac{d + \log(1/\gamma)}{\alpha^2}$$

Proof [Proof of Theorem 33] Let $S_L \sim_{i.i.d.} \mathcal{D}_L^n$ be a random sample of size n. By Part 2 of Theorem 34 and our assumption that $\mathcal{M}$ is an (α,γ)-accurate agnostic learner, we know that $n \geq C_2 \frac{(d+\log(1/\gamma))}{\alpha^2}$. By Part 1 of Theorem 34, we have with probability at least $1 - \gamma$ over the random draws of S_L, for each $h \in \mathcal{H}$,

$$|err(S_L, h) - err(h)| \leq O(\alpha). \tag{6}$$

Let $\hat{h} = \mathcal{M}'(\mathcal{M}, S_L)$. First, we can view $\mathcal{E}_{S_L}$ as some distribution over the labelled examples. Since $\mathcal{M}$ is an (α,γ)-accurate learner, we have with probability at least $1 - \gamma$,

$$err(S_L, \hat{h}) \leq \min_{h \in \mathcal{H}} err(S_L, h) + \alpha. \tag{7}$$

Let's condition on guarantee of both Equations (6) and (7). Let $h^* = \arg\min_{h\in\mathcal{H}} err(h)$. Then by combining Equations (6) and (7), we get

$$err(\hat{h}) \leq err(S_L, \hat{h}) + O(\alpha) \leq err(S_L, h^*) + O(\alpha) \leq err(h^*) + O(\alpha)$$

which recovers the stated utility guarantee. By Theorem 32, know that the mechanism $\mathcal{M}'(\mathcal{M}, \cdot)$ is also $(4\varepsilon, 16\delta + 2\beta)$-differentially private. ■

The result of Theorem 33 implies that the existence of a perfectly generalizing agnostic learner would imply the existence of a differentially private one. Moreover, the lower bound results for private learning would apply to a perfectly generalizing learner as well. In particular, based on the result of Bun et al. (2015), we can show that there is no proper threshold learner that satisfies perfect generalization when the domain size is infinite.

Theorem 35 *Let $\alpha > 0$ be the accuracy parameter. For every $n \in \mathbb{N}$, and $\delta, \beta \leq 1/(10000n^2)$, any $(\beta, 1/8, \delta)$-perfectly generalizing and $(\alpha, 1/32)$-accurate proper agnostic learner for threshold function requires sample complexity $n = \Omega\left(\log^* |\mathcal{X}|/\alpha\right)$.*

5. Relationship between Perfect Generalization and Other Generalization Notions

In the previous sections we have studied the three generalization notions as learnability constraints, and we know that any class that learnable under perfect generalization is also learnable under differential privacy, and any class learnable under differential privacy is also learnable under robust generalization. In this section, we study these three notions from the algorithmic point of view, and explore the relationships among algorithms that satisfy perfect generalization, robust generalization and differential privacy. Section 5.1 shows that any perfectly generalizing algorithms is also robustly generalizing, but there exist robustly generalizing algorithms that are neither differentially private nor perfectly generalizing for any reasonable parameters. Section 5.2 shows that all differentially private algorithms are perfectly generalizing with some necessary loss in generalization parameters, but there exist perfectly generalizing algorithms which are not differentially private for any reasonable parameters.

5.1. Separation between Perfect and Robust Generalization

In this section we show that perfect generalization is a stronger requirement than robust generalization. Lemma 36 shows one direction of this, by showing that every perfectly generalizing mechanism also satisfies robust generalization with only a constant degradation in the generalization parameters.

Lemma 36 *For any $\beta, \varepsilon, \delta \in (0, 1)$, suppose that a mechanism $\mathcal{M} \colon \mathcal{X}_L^n \to \mathcal{R}$ with arbitrary range $\mathcal{R}$ is $(\beta, \varepsilon, \delta)$-perfectly generalizing. Then $\mathcal{M}$ is also $(\alpha, 2(\beta + \delta))$-robustly generalizing, where*

$$\alpha = \sqrt{\frac{2}{n} \ln\left(\frac{2(2\varepsilon + 1)}{\beta + \delta}\right)}.$$

Proof Let $\mathcal{A}\colon \mathcal{R} \to (\mathcal{X} \to \{0,1\})$ be function that takes in the output of $\mathcal{M}(S_L)$ and produces a hypothesis $h\colon \mathcal{X} \to \{0,1\}$. Our goal is to show that h will not overfit to the original sample S_L.

By Theorem 11, the composition of $\mathcal{A} \circ \mathcal{M}\colon \mathcal{X}^n \to (\mathcal{X}_L \to \{0,1\})$ is also $(\beta, \varepsilon, \delta)$-perfectly generalizing. This means there exists a simulator $\textsc{Sim}\colon \Delta\mathcal{X} \to \mathcal{R}$ such that with high probability over a random sample S_L, $\textsc{Sim}(\mathcal{D}) \approx_{\varepsilon,\delta} (\mathcal{A} \circ \mathcal{M})(S_L)$. Define the event $E = \{S_L \in \mathcal{X}^n \mid [\textsc{Sim}(\mathcal{D}) \approx_{\varepsilon,\delta} (\mathcal{A} \circ \mathcal{M})(S_L)]\}$. By perfect generalization, $\Pr_{S_L \sim_{i.i.d.} \mathcal{D}_L^n}[E] \geq 1 - \beta$.

Also by a Chernoff-Hoeffding bound, for any fixed $h \in \mathcal{H}$ and any $\alpha > 0$,

$$\Pr_{S \sim_{i.i.d.} \mathcal{D}^n}[|h(S) - h(\mathcal{D})| \geq \alpha] \leq 2\exp\left(-2\alpha^2 n\right).$$

The following bounds the probability that the hypothesis h output by $(\mathcal{A} \circ \mathcal{M})(S_L)$ overfits on the sample S_L, where $\wedge$ denotes the logical AND.

$$\begin{aligned}
&\Pr_{S_L \sim_{i.i.d.} \mathcal{D}_L^n}[h \leftarrow (\mathcal{A} \circ \mathcal{M}(S_L)) \wedge |h(S) - h(\mathcal{D})| \geq \alpha] \\
&\quad = \sum_{S_L \in \mathcal{X}_L^n} \Pr[S] \Pr[h \leftarrow (\mathcal{A} \circ \mathcal{M}(S_L)) \wedge |h(S) - h(\mathcal{D})| \geq \alpha \mid S] \\
&\quad \leq (1 - \Pr[E]) + \sum_{S \in E} \Pr[S] \Pr[h \leftarrow (\mathcal{A} \circ \mathcal{M}(S)) \wedge |h(S) - h(\mathcal{D})| \geq \alpha \mid S] \\
&\quad \leq (1 - \Pr[E]) + \sum_{S \in E} \Pr[S] \left(\Pr[h \leftarrow \textsc{Sim}(\mathcal{D}) \wedge |h(S) - h(\mathcal{D})| \geq \alpha \mid S] \cdot \exp(\varepsilon) + \delta\right) \\
&\quad \leq (1 - \Pr[E]) + \sum_{S \in \mathcal{X}^n} \Pr[S] \left(\Pr[h \leftarrow \textsc{Sim}(\mathcal{D}) \wedge |h(S) - h(\mathcal{D})| \geq \alpha \mid S] \cdot \exp(\varepsilon) + \delta\right) \\
&\quad = (1 - \Pr[E]) + \delta + \exp(\varepsilon) \Pr_{S \sim_{i.i.d.} \mathcal{D}^n}[h \leftarrow \textsc{Sim}(\mathcal{D}) \wedge |h(S) - h(\mathcal{D})| \geq \alpha] \\
&\quad \leq (1 - \Pr[E]) + \delta + 2\exp(\varepsilon) \cdot \exp(-2\alpha^2 n) \\
&\quad \leq \beta + \delta + 2\exp(\varepsilon) \cdot \exp(-2\alpha^2 n)
\end{aligned}$$

Setting $\alpha = \sqrt{\frac{2}{n} \ln\left(\frac{2(2\varepsilon+1)}{\beta+\delta}\right)}$ also gives $\exp(-2\alpha^2 n) = \frac{\beta+\delta}{2(2\varepsilon+1)}$. Plugging this into the above equations, we see that the probability that $(\mathcal{A} \circ \mathcal{M})(S_L)$ overfits to S_L by more than our choice of α is at most

$$\Pr_{S_L \sim_{i.i.d.} \mathcal{D}^n}[h \leftarrow (\mathcal{A} \circ \mathcal{M}(S_L)) \wedge |h(S) - h(\mathcal{D})| \geq \alpha] \leq \beta + \delta + 2\exp(\varepsilon)\frac{\beta + \delta}{2(1 + 2\varepsilon)} = 2(\beta + \delta).$$

Thus $\mathcal{M}$ is $(\alpha, 2(\beta + \delta))$-robustly generalizing for our specified value of α. ■

Our next result, Lemma 37, shows that there exist robustly generalizing mechanisms that are neither differentially private nor perfectly generalizing, for any reasonable parameters.

Lemma 37 *For any $\gamma > 0$ and $n \in \mathbb{N}$, there exists a mechanism $\mathcal{M}\colon \mathcal{X}_L^n \to \{0,1\}$ that is $(\sqrt{\ln(2/\gamma)/2n}, \gamma)$-robustly generalizing, but is not (ε, δ)-differentially private for any bounded ε and $\delta < 1$, and is not $(\beta, \varepsilon', \delta')$-perfectly generalizing for any $\beta < 1/2 - 1/\sqrt{n}$, bounded ε', and $\delta' < 1/2$.*

Proof Consider the domain $\mathcal{X} = \{0,1\}$, and the following deterministic mechanism $\mathcal{M}\colon \mathcal{X}^n \to \{0,1\}$: given a sample S, output 1 if more than $\lfloor n/2 \rfloor$ of the elements in S is 1, and output 0 otherwise. Note $\mathcal{M}$ has a small output space, so by Theorem 21, $\mathcal{M}$ is $(\sqrt{\ln(2/\gamma)/2n}, \gamma)$-robustly generalizing for any $\gamma > 0$.

Consider two neighboring samples S_1 and S_2 such that S_1 has $\lfloor n/2 \rfloor + 1$ number of 1's, and S_2 has $\lfloor n/2 \rfloor$ number of 1's. Then $\Pr[\mathcal{M}(S_1) = 1] = 1$ and $\Pr[\mathcal{M}(S_2) = 1] = 0$. Therefore, the mechanism is not (ε, δ)-differentially private for any bounded ε and $\delta < 1$.

To show that $\mathcal{M}$ is not perfectly generalizing, consider the distribution $\mathcal{D}$ that is uniform over $\mathcal{X} = \{0,1\}$. That is, $\Pr_{x\sim\mathcal{D}}[x = 1] = \Pr_{x\sim\mathcal{D}}[x = 0] = 1/2$. Suppose that $\mathcal{M}$ is $(\beta, \varepsilon', \delta')$-perfectly generalizing with $\beta < 1/2 - 1/\sqrt{n}$. In particular, this implies that $\beta < 1/2 - \frac{\binom{n}{\lfloor n/2 \rfloor}}{2^n}$. Let SIM be the associated simulator, and let $p = \Pr[\text{SIM}(\mathcal{D}) = 1]$.

Since each the events of $(\mathcal{M}(S) = 0)$ and $(\mathcal{M}(S) = 1)$ will occur with probability (over the random draws of S) greater than β, then there exist samples S_1 and S_2 such that both $\mathcal{M}(S_1), \mathcal{M}(S_2) \approx_{\varepsilon',\delta'} \text{SIM}(\mathcal{D})$, and furthermore $\mathcal{M}(S_1) = 1$ and $\mathcal{M}(S_2) = 0$ deterministically. This means,

$$p \leq \exp(\varepsilon') \cdot \Pr[\mathcal{M}(S_2) = 1] + \delta' = \delta' \quad \text{and,} \quad (1-p) \leq \exp(\varepsilon') \cdot \Pr[\mathcal{M}(S_1) = 0] + \delta' = \delta'.$$

It follows from above that $\delta' \geq 1/2$. Thus, $\mathcal{M}$ is not $(\beta, \varepsilon', \delta')$ for any $\beta < 1/2 - 1/\sqrt{n}$, bounded ε', and $\delta' < 1/2$. ■

5.2. Perfect Generalization and Differential Privacy

We now focus on the relationship between differential privacy and perfect generalization to show that perfect generalization is a strictly stronger definition in the sense that *problems* that can be solved subject to perfect generalization can also be solved subject to differential privacy with little loss in the parameters, whereas in the reverse direction, parameters necessarily degrade. Recall that we have already shown that any perfectly generalizing algorithm can be "compiled" into a differentially private algorithm with only a constant factor loss in parameters (Theorem 32). We here note however that this compilation is necessary – that perfectly generalizing algorithms are not necessarily themselves differentially private. In the reverse direction, we show that every differentially private algorithm is strongly generalizing, with some necessary degradation in the generalization parameters.

We first give an example of a perfectly generalizing algorithm that does not satisfy differential privacy for any reasonable parameters. The intuition behind this result is that perfect generalization requires an algorithm to behave similarly only on a $(1-\beta)$-fraction of samples, while differential privacy requires an algorithm to behave similarly on all neighboring samples. The algorithm of Theorem 38 exploits this difference to find a pair of unlikely neighboring samples which are treated very differently.

Theorem 38 *For any $\beta > 0$ and any $n \geq \log(1/\beta)$, there exists a algorithm $\mathcal{M}\colon \mathcal{X}^n \to \mathcal{R}$ which is $(\beta, 0, 0)$-perfectly generalizing but is not (ε, δ)-differentially private for any $\varepsilon < \infty$ and $\delta < 1$.*

Proof Consider the domain $\mathcal{X} = \{0,1\}$ and the following simple algorithm $\mathcal{M}$: given a sample $S = \{s_1, \ldots, s_n\}$ of size n, it will output "Strange" if the sample S satisfies:

$$s_1 = s_2 = \ldots = s_{\lfloor n/2 \rfloor} = 1 \qquad \text{and,} \qquad s_{\lfloor n/2 \rfloor + 1} = s_{\lfloor n/2 \rfloor + 2} = \ldots = s_n = 0,$$

and output "Normal" otherwise. We first show that $\mathcal{M}$ is $((1/2)^n, 0, 0)$-perfectly generalizing. Consider the following deterministic simulator SIM that simply outputs "Normal" no matter what the input distribution over the domain is.

Suppose that the distribution $\mathcal{D}$ over the domain satisfies $\Pr_{x\sim\mathcal{D}}[x = 1] = p$ for some $p \in [0, 1]$. Note that the probability (over the random draws of S) of outputting "Strange" is

$$\Pr[\mathcal{M}(S) = \text{"Strange"}] = p^{\lfloor n/2 \rfloor}(1-p)^{\lceil n/2 \rceil} =\leq (1/2)^n.$$

This means, with probability at least $1 - (1/2)^n$ over the random draws of S, $\mathcal{M}$ will output "Normal," and also

$$\frac{\Pr[M(S) = \text{"Normal"}]}{\Pr[\text{SIM}(\mathcal{D}) = \text{"Normal"}]} = 1 \leq \exp(0).$$

Therefore, $\mathcal{M}$ is $((1/2)^n, 0, 0)$-perfectly generalizing.

Now consider the sample $T = \{t_1, \ldots, t_n\}$ such that

$$t_1 = t_2 = \ldots = t_{\lfloor n/2 \rfloor} = 1 \qquad \text{and,} \qquad t_{\lfloor n/2 \rfloor + 1} = t_{\lfloor n/2 \rfloor + 2} = \ldots = t_n = 0.$$

Let T' be any neighboring sample of T such that $|T \Delta T'| = 1$. We know that $\mathcal{M}(T') =$ "Normal", so,

$$\frac{\Pr[\mathcal{M}(T') = \text{"Normal"}]}{\Pr[\mathcal{M}(T) = \text{"Normal"}]} = \frac{1}{0} = \infty.$$

Therefore, the algorithm $\mathcal{M}$ is not (ε, δ)-differentially private for any $\varepsilon < \infty$ and $\delta < 1$. ■

Now we show the other direction of the relationship between these two definitions: any differentially private algorithm is also perfectly generalizing. We begin with Theorem 39, which proves that every $(\varepsilon, 0)$-differentially private algorithm is also $(\beta, O(\sqrt{n \ln(1/\beta)}\varepsilon), 0)$-perfectly generalizing. We will later show that this dependence on n and β is tight.

Theorem 39 *Let $\mathcal{M} \colon \mathcal{X}^n \to \mathcal{R}$ be an $(\varepsilon, 0)$-differentially private algorithm, where $\mathcal{R}$ is an arbitrary finite range. Then $\mathcal{M}$ is also $(\beta, \sqrt{2n \ln(2|\mathcal{R}|/\beta)}\varepsilon, 0)$-perfectly generalizing.*

Proof Given an $(\varepsilon, 0)$-differentially private algorithm $\mathcal{M}$, consider the following log-likelihood function $q \colon \mathcal{X}^n \times \mathcal{R} \to \mathbb{R}$, such that for any sample $S \in \mathcal{X}^n$ and outcome $r \in \mathcal{R}$, we have

$$q(S, r) \stackrel{\text{def}}{=} \log\left(\Pr[\mathcal{M}(S) = r]\right).$$

Since $\mathcal{M}$ is $(\varepsilon, 0)$-differentially private, we know that for all neighboring $S, S' \in \mathcal{X}^n$, the function q satisfies,

$$\max_{r\in\mathcal{R}} \left|q(S, r) - q(S', r)\right| = \max_{r\in\mathcal{R}} \left|\ln\left(\frac{\Pr[\mathcal{M}(S) = r]}{\Pr[\mathcal{M}(S') = r]}\right)\right| \leq \varepsilon.$$

For any distribution $\mathcal{D} \in \Delta\mathcal{X}$, the sample $S = (s_1, \ldots, s_n) \sim_{i.i.d.} \mathcal{D}^n$ is now a random variable, rather than a fixed input. By an application of McDiarmid's inequality to the variables $s_1, \ldots s_n$, we have that for any $r \in \mathcal{R}$,

$$\Pr_{S\sim_{i.i.d.}\mathcal{D}^n}\left[\left|q(S, r) - \mathop{\mathbb{E}}_{S'\sim_{i.i.d.}\mathcal{D}^n}\left[q(S', r)\right]\right| \geq t\right] \leq 2\exp\left(\frac{-2t^2}{n\varepsilon^2}\right). \tag{8}$$

Instantiating Equation (8) with $t = \varepsilon\sqrt{(n/2)\ln(2|\mathcal{R}|/\beta)}$ and taking a union bound, we have that with probability at least $1-\beta$, it holds for all $r \in \mathcal{R}$ that,

$$\left| q(S,r) - \mathop{\mathbb{E}}_{S' \sim_{i.i.d.} \mathcal{D}^n} \left[q(S',r) \right] \right| \leq \varepsilon\sqrt{(n/2)\ln(2|\mathcal{R}|/\beta)}. \tag{9}$$

Define the simulator $\textsc{Sim}(\mathcal{D})$ for algorithm $\mathcal{M}$ on distribution $\mathcal{D}$ as follow for all $r \in \mathcal{R}$, output the r with probability proportional to $\exp\left(\mathbb{E}_{S \sim_{i.i.d.} \mathcal{D}^n}\left[q(S,r)\right]\right)$. Let

$$Z = \frac{\sum_{r \in \mathcal{R}} \exp\left(\mathbb{E}_{S' \sim_{i.i.d.} \mathcal{D}^n}\left[q(S',r)\right]\right)}{\sum_{r \in \mathcal{R}} \exp\left(q(S,r)\right)}$$

be the ratio between the normalization factors, and by Equation (9),

$$\exp\left(-\varepsilon\sqrt{(n/2)\ln(2|\mathcal{R}|/\beta)}\right) \leq Z \leq \exp\left(\varepsilon\sqrt{(n/2)\ln(2|\mathcal{R}|/\beta)}\right)$$

We condition on the bound in Equation (9) for the remainder of the proof, which holds except with probability β. For any $r \in \mathcal{R}$,

$$\begin{aligned}
\frac{\Pr[\mathcal{M}(S) = r]}{\Pr[\textsc{Sim}(\mathcal{D}) = r]} &= \frac{\exp\left(q(S,r)\right)}{\exp\left(\mathbb{E}_{S' \sim_{i.i.d.} \mathcal{D}^n}[q(S',r)]\right)/Z} \\
&= \exp\left(q(S,r) - \mathop{\mathbb{E}}_{S' \sim_{i.i.d.} \mathcal{D}^n}[q(S',r)]\right) \cdot Z \\
&\leq \exp\left(\varepsilon\sqrt{2n\ln(2|\mathcal{R}|/\beta)}\right),
\end{aligned}$$

where the last inequality is due to Equation (9).

For any $\mathcal{O} \subseteq \mathcal{R}$ and for $\varepsilon' = \varepsilon\sqrt{2n\ln(2|\mathcal{R}|/\beta)}$,

$$\begin{aligned}
\Pr[\mathcal{M}(S) \in \mathcal{O}] &= \sum_{r \in \mathcal{O}} \Pr[\mathcal{M}(S) = r] \\
&\leq \sum_{r \in \mathcal{O}} e^{\varepsilon'} \Pr[\textsc{Sim}(\mathcal{D}) = r] \\
&= e^{\varepsilon'} \Pr[\textsc{Sim}(\mathcal{D}) \in \mathcal{O}].
\end{aligned}$$

Similarly, we could also show

$$\frac{\Pr[\textsc{Sim}(\mathcal{D}) \in \mathcal{O}]}{\Pr[\mathcal{M}(S) \in \mathcal{O}]} \leq \exp\left(\varepsilon\sqrt{2n\ln(2|\mathcal{R}|/\beta)}\right).$$

Thus for any distribution $\mathcal{D} \in \Delta\mathcal{X}$, with probability at least $1-\beta$ over the choice of $S \sim_{i.i.d.} \mathcal{D}^n$, we have that $\mathcal{M}(S) \approx_{\varepsilon',0} \textsc{Sim}(\mathcal{D})$, for $\varepsilon' = \varepsilon\sqrt{2n\ln(2|\mathcal{R}|/\beta)}$, so $\mathcal{M}$ is $(\beta, \varepsilon\sqrt{2n\ln(2|\mathcal{R}|/\beta)}, 0)$-perfectly generalizing. ■

The following result proves that the degradation of ε in Theorem 39 is necessary, and the dependence on n and β is asymptotically tight.

Theorem 40 *For any $\varepsilon > 0$, $\beta \in (0,1)$ and $n \in \mathbb{N}$, there exists a algorithm $\mathcal{M}\colon \mathcal{X}^n \to \mathcal{R}$ that is $(\varepsilon, 0)$-differentially private, but not $(\beta, \varepsilon', 0)$-perfectly generalizing for any $\varepsilon' = o(\varepsilon\sqrt{n\ln(1/\beta)})$.*

Proof Consider the domain $\mathcal{X} = \{0, 1\}$ and the distribution $\mathcal{D}$ over $\mathcal{X}$ such that $\Pr_{x\sim\mathcal{D}}[x = 1] = \Pr_{x\sim\mathcal{D}}[x = 0] = 1/2$. Consider following algorithm $\mathcal{M}\colon \mathcal{X}^n \to \{0, 1\}$. Given a sample $S = \{s_1, \ldots, s_n\} \in \mathcal{X}^n$, $\mathcal{M}$ will do the following:

1. first compute the sample average $\overline{s} = \frac{1}{n}\sum_{i=1}^{n} s_i$;
2. then compute a noisy estimate $\hat{s} = \overline{s} + \mathrm{Lap}(\frac{1}{n\varepsilon})$ by adding Laplace noise with parameter $1/n\varepsilon$;
3. if $\hat{s} \leq 1/2$, output 0; otherwise, output 1.

In words, the algorithm tries to identify the majority in the sample based on the noisy estimate $\hat{s}$. Note that the average value $\overline{s}$ is a $(1/n)$-sensitive statistic — that is, changing a single sample point s_i in S will change the value of $\overline{s}$ by at most $1/n$. Also observe that $\mathcal{M}$ is the Laplace mechanism of Dwork et al. (2006b) composed with a (data independent) post-processing step, so we know $\mathcal{M}$ is $(\varepsilon, 0)$-differentially private.

Now suppose that $\mathcal{M}$ is $(\beta, \varepsilon', 0, n)$-strongly generalizing for some $\beta \in (0, 1)$. Using a standard tail bound for the Binomial distribution, we know that for any $S \sim_{i.i.d.} \mathcal{D}^n$ and $k \leq 1/8$, the sample average $\overline{s}$ satisfies

$$\Pr[\overline{s} \leq n/2 - k] = \Pr[\overline{s} \geq n/2 + k] \geq \frac{1}{15}\exp\left(-16nk^2\right).$$

In other words, for any $\gamma \in (0, 1)$, we have both $\Pr[\overline{s} \leq 1/2 - K] \geq \gamma$ and $\Pr[\overline{s} \geq 1/2 + K] \geq \gamma$, where $K = \frac{\sqrt{\ln(1/(15\gamma))}}{4\sqrt{n}}$. For the remainder of the proof, we will set $\gamma = 2\sqrt{\beta}$.

Let $S_1, S_2 \sim_{i.i.d.} \mathcal{D}^n$ be two random samples with sample averages $\overline{s}_1$ and $\overline{s}_2$. By Corollary 10, we know that $\Pr[\mathcal{M}(S_1) \not\approx_{2\varepsilon',0} \mathcal{M}(S_2)] \leq 2\beta$. Since $\gamma^2 > 2\beta$, it follows that with strictly positive probability over the random draws over S_1 and S_2, all of the events that $\overline{s}_1 \leq n/2 - K$, $\overline{s}_2 \geq n/2 + K$, and $\mathcal{M}(S_1) \approx_{2\varepsilon',0} \mathcal{M}(S_2)$ occur simultaneously. For the remainder of the proof, we condition on samples S_1 and S_2 satisfying these conditions, which will happen with probability greater than 2β.

If we apply our algorithm $\mathcal{M}$ to both samples, we will first obtain noisy estimates $\hat{s}_1$ and $\hat{s}_2$, and by the property of the Laplace distribution, we know for any $t > 0$

$$\Pr\left[|\hat{s}_1 - \overline{s}_1| < K\right] = 1 - \exp(-Kn\varepsilon) \qquad \text{and} \qquad \Pr\left[|\hat{s}_2 - \overline{s}_2| < K\right] = 1 - \exp(-Kn\varepsilon)$$

Note that the event $|\hat{s}_1 - \overline{s}_1| < K$ implies that $M(S_1) = 0$, and the event $|\hat{s}_2 - \overline{s}_2| < K$ implies that $M(S_2) = 1$. The condition of $M(S_1) \approx_{2\varepsilon',0} M(S_2)$ implies that

$$\exp(2\varepsilon') \geq \frac{\Pr[M(S_1) = 0]}{\Pr[M(S_2) = 0]} = \frac{\Pr[M(S_1) = 0]}{1 - \Pr[M(S_2) = 1]} \geq \frac{1 - \exp(-Kn\varepsilon)}{\exp(-Kn\varepsilon)} = \exp(Kn\varepsilon) - 1$$

It follows that we must have

$$\varepsilon' \geq \frac{1}{2}(Kn\varepsilon - 1) = \Omega\left(\varepsilon\sqrt{n\ln(1/\beta)}\right),$$

which recovers the stated bound. ■

Theorems 39 and Theorem 40 only show a relationship between $(\varepsilon, 0)$-differential privacy and strong generalization. To show such a relationship when $\delta > 0$, we appeal to *group privacy*, first studied by Dwork et al. (2006a), which says that if $\mathcal{M}$ is (ε, δ)-differentially private and two samples S, S' differ on k entries, then $\mathcal{M}(S) \approx_{k\varepsilon, ke^{(k-1)\varepsilon}\delta} \mathcal{M}(S')$. Using simulator $\textsc{Sim}_{\mathcal{D}} = \mathcal{M}(S^*)$ for any fixed sample $S^* \sim_{i.i.d.} \mathcal{D}^n$ and by the fact that any sample S can differ from S^* in an most n samples, we see that $\mathcal{M}$ is $(0, n\varepsilon, ne^{(n-1)\varepsilon}\delta)$-perfectly generalizing.

Unfortunately, this blowup in parameters is generally unacceptable for most tasks. We suspect that the necessary blowup in the ε parameter is closer to $\Theta\left(\sqrt{n \ln(1/\beta)}\right)$ as with $(\varepsilon, 0)$-differential privacy, but leave a formal proof as an open question for future work.

On the positive side, most known (ε, δ)-differentially private algorithms are designed by composing several $(\varepsilon', 0)$-differentially private algorithms, where the $\delta > 0$ is an artifact of the composition (see, e.g., Theorem 3.20 of Dwork and Roth (2014) for more details). Since perfect generalization enjoys adaptive composition (as shown in Bassily and Freund (2016)), we could also obtain $(\beta, \varepsilon, \delta)$-perfectly generalizing algorithms by composing a collection of $(\beta, \varepsilon, 0)$-perfectly generalizing algorithms together. This will give better generalization parameters than a direct reduction via group privacy.

Acknowledgments

We thank Adam Smith and Raef Bassily for helpful comments about adaptive composition of perfectly generalizing mechanisms and pointing out an error in an earlier version of this paper. We thank Shay Moran for telling us about variable length compression schemes and sharing with us his manuscript David et al. (2016). We thank our anonymous reviewers for numerous helpful comments.

The first author is supported in part by NSF grant 1254169, US-Israel Binational Science Foundation grant 2012348, and a Simons Graduate Fellowship. The second author is supported in part by NSF grants 1254169 and 1518941, US-Israel Binational Science Foundation Grant 2012348, the Charles Lee Powell Foundation, a Google Faculty Research Award, an Okawa Foundation Research Grant, a subcontract through the DARPA Brandeis project, a grant from the HUJI Cyber Security Research Center, and a startup grant from Hebrew University's School of Computer Science. Part of this work was completed when the second author was visiting the Simons Institute for the Theory of Computing at Berkeley. The third author is supported by grants from the Sloan Foundation, a Simons Investigator grant to Salil Vadhan, and NSF grant CNS-1237235. The fourth author is supported in part by an NSF CAREER award, NSF grant CNS-1513694, a subcontract through the DARPA Brandeis project, and a grant from the Sloan Foundation.

References

Raef Bassily and Yoav Freund. Typicality-based stability and privacy. *CoRR*, abs/1604.03336, 2016.

Raef Bassily, Kobbi Nissim, Adam D. Smith, Thomas Steinke, Uri Stemmer, and Jonathan Ullman. Algorithmic stability for adaptive data analysis. In *Proceedings of the 48th Annual ACM on Symposium on Theory of Computing, STOC*, 2016.

Avrim Blum and Moritz Hardt. The ladder: A reliable leaderboard for machine learning competitions. In *Proceedings of the 32nd International Conference on Machine Learning, ICML*, pages 1006–1014, 2015.

Avrim Blum, Katrina Ligett, and Aaron Roth. A learning theory approach to noninteractive database privacy. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, STOC*, pages 609–618, 2008.

Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K Warmuth. Occams razor. *Readings in machine learning*, pages 201–204, 1990.

Olivier Bousquet and André Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2:499–526, 2002.

Mark Bun, Kobbi Nissim, Uri Stemmer, and Salil P. Vadhan. Differentially private release and learning of threshold functions. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS*, pages 634–649, 2015.

Ofir David, Shay Moran, and Amir Yehudayof. Supervised learning through the lens of compression. *Preprint*, 2016.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. *Advances in Cryptology - EUROCRYPT 2006: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques. Proceedings*, chapter Our Data, Ourselves: Privacy Via Distributed Noise Generation, pages 486–503. Springer Berlin Heidelberg, 2006a.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography, TCC*, pages 265–284, 2006b.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toni Pitassi, Omer Reingold, and Aaron Roth. Generalization in adaptive data analysis and holdout reuse. In *Advances in Neural Information Processing Systems, NIPS*, pages 2341–2349, 2015a.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349(6248):636–638, 2015b.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. Preserving statistical validity in adaptive data analysis. In *Proceedings of the 47th Annual ACM on Symposium on Theory of Computing, STOC*, pages 117–126, 2015c.

Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

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

Michael J Kearns and Umesh Virkumar Vazirani. *An introduction to computational learning theory*. MIT press, 1994.

Nick Littlestone and Manfred Warmuth. Relating data compression and learnability. Technical report, 1986.

Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science, FOCS*, pages 94–103, 2007.

Tomaso Poggio, Ryan Rifkin, Sayan Mukherjee, and Partha Niyogi. General conditions for predictivity in learning theory. *Nature*, 428(6981):419–422, 2004.

Daniel Russo and James Zou. Controlling bias in adaptive data analysis using information theory. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, AISTATS*, 2016.

S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

Shai Shalev-Shwartz, Ohad Shamir, Nathan Srebro, and Karthik Sridharan. Learnability, stability and uniform convergence. *The Journal of Machine Learning Research*, 11:2635–2670, 2010.

Vladimir N Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of Complexity*, pages 11–30. Springer International Publishing, 1971.

Manfred K. Warmuth. *Compressing to VC Dimension Many Points*, volume 2777, pages 743–744. Springer Berlin Heidelberg, 2003.

Appendix A. Missing Proofs in Section 2

Proof [Proof of Lemma 9]
In the following, we will use $(a \wedge b)$ to denote $\min\{a, b\}$. For all $\mathcal{O} \subseteq \mathcal{R}$,

$$\begin{aligned}
\Pr_{y\sim\mathcal{J}_1}[y \in \mathcal{O}] &\leq (\exp(\varepsilon) \Pr_{y\sim\mathcal{J}_2}[y \in \mathcal{O}] + \delta) \wedge 1 \\
&\leq (\exp(\varepsilon) \Pr_{y\sim\mathcal{J}_2}[y \in \mathcal{O}]) \wedge 1 + \delta \\
&\leq \exp(\varepsilon)\left(\exp(\varepsilon') \Pr_{y\sim\mathcal{J}_3}[y \in \mathcal{O}] + \delta'\right) + \delta \\
&= \exp(\varepsilon+\varepsilon') \Pr_{y\sim\mathcal{J}_3}[y \in \mathcal{O}] + 2\delta' + \delta.
\end{aligned}$$

A similar argument gives $\Pr_{y\sim\mathcal{J}_3}[y \in \mathcal{O}] \leq \exp(\varepsilon+\varepsilon') \Pr_{y\sim\mathcal{J}_1}[y \in \mathcal{O}] + 2\delta + \delta'$. ■

Proof [Proof of Corollary 10] By a union bound, with probability $1 - 2\beta$ over the draws of $T_1, T_2 \sim_{i.i.d.} \mathcal{C}^n$,

$$\mathcal{M}(T_1) \approx_{\varepsilon,\delta} \textsc{Sim}_{\mathcal{C}} \approx_{\varepsilon,\delta} \mathcal{M}(T_2).$$

The result then follows from Lemma 9. ■

Proof [Proof of Lemma 11] The result for robustly generalizing mechanisms follows immediately from the definition: Assume by way of contradiction that there exists an (α, β)-robustly generalizing mechanism $\mathcal{M}\colon \mathcal{Y}^n \to \mathcal{R}$ and a post-processing procedure $\mathcal{A}\colon \mathcal{R} \to \mathcal{R}'$ such that $\mathcal{A} \circ \mathcal{M}$ is not (α, β)-robustly generalizing. Then there exists an adversary $\mathcal{A}'$ such that $\mathcal{A}'(\mathcal{A}(M(T)))$ outputs a hypothesis h that violates the robust generalization condition. However, this would imply that $\mathcal{A}' \circ \mathcal{A}$ is an adversary that violates the robust generalization condition, contradicting the assumption that $\mathcal{M}$ is (α, β)-robustly generalizing.

Let $\mathcal{M}\colon \mathcal{Y}^n \to \mathcal{R}$ be $(\beta, \varepsilon, \delta)$-perfectly generalizing, and let $\mathcal{A}\colon \mathcal{R} \to \mathcal{R}'$ be a post-processing procedure. Fix any distribution $\mathcal{C}$, and let $\textsc{Sim}_{\mathcal{C}}$ denote the simulator such that $\mathcal{M}(T) \approx_{\varepsilon,\delta} \textsc{Sim}_{\mathcal{C}}$ with probability $1 - \beta$ when $T \sim_{i.i.d.} \mathcal{C}^n$. We will show that with probability at least $1 - \beta$ over the sample $T \sim_{i.i.d.} \mathcal{C}^n$,

$$\mathcal{A}(\mathcal{M}(T)) \approx_{\varepsilon,\delta} \mathcal{A}(\textsc{Sim}_{\mathcal{C}}).$$

First note that any randomized mapping can be decomposed into a convex combination of deterministic mappings. Let

$$\mathcal{A} = \sum_{i=1} \gamma_i \mathcal{A}_i \quad \text{s.t.} \quad \sum_{i=1} \gamma_i = 1 \text{ and } 0 < \gamma_i \leq 1 \ \forall i,$$

where each $\mathcal{A}_i\colon \mathcal{R} \to \mathcal{R}'$ is deterministic. For the remainder of the proof, we will assume that $\mathcal{M}(T) \approx_{\varepsilon,\delta} \textsc{Sim}_{\mathcal{C}}$, which will be the case with probability $1 - \beta$.

Fix an arbitrary $\mathcal{O}' \subseteq \mathcal{R}'$ and define $\mathcal{O}_i = \{r \in \mathcal{R} \mid \mathcal{A}_i(r) \in \mathcal{O}'\}$ for $i \in [k]$.

$$\begin{aligned}
\Pr[\mathcal{A}(\mathcal{M}(T)) \in \mathcal{O}'] &= \sum_{i=1} \gamma_i \Pr[\mathcal{A}_i(\mathcal{M}(T)) \in \mathcal{O}'] \\
&= \sum_{i=1} \gamma_i \Pr[\mathcal{M}(T) \in \mathcal{O}_i] \\
&\leq \sum_{i=1} \gamma_i \left(e^{\varepsilon} \Pr[\textsc{Sim}_{\mathcal{C}} \in \mathcal{O}_i] + \delta\right) \\
&= \sum_{i=1} \gamma_i \left(e^{\varepsilon} \Pr[\mathcal{A}_i(\textsc{Sim}_{\mathcal{C}}) \in \mathcal{O}'] + \delta\right) \\
&= e^{\varepsilon} \Pr[\mathcal{A}(\textsc{Sim}_{\mathcal{C}}) \in \mathcal{O}'] + \delta.
\end{aligned}$$

A symmetric argument shows that

$$\Pr[\mathcal{A}(\textsc{Sim}_{\mathcal{C}}) \in \mathcal{O}'] \leq e^{\varepsilon} \Pr[\mathcal{A}(\mathcal{M}(T)) \in \mathcal{O}'] + \delta.$$

Thus with probability at least $1 - \beta$, $\mathcal{A}(\mathcal{M}(T)) \approx_{\varepsilon,\delta} \mathcal{A}(\textsc{Sim}_{\mathcal{C}})$. The mapping $\mathcal{A}(\textsc{Sim}_{\mathcal{C}})\colon \mathcal{Y}^n \to \mathcal{R}'$ is simply a new simulator, so $\mathcal{A} \circ \mathcal{M}$ is $(\beta, \varepsilon, \delta)$-perfectly generalizing. ■

Proof [Proof of Theorem 12] Fix any distribution $\mathcal{C}$, and for all $i \in [k]$ let $\textsc{Sim}_i(\mathcal{C})$ denote the simulator such that $\mathcal{M}_i(T) \approx_{\varepsilon,\delta} \textsc{Sim}_i(\mathcal{C})$ with probability $1 - \beta_i$ when $T \sim_{i.i.d.} \mathcal{C}^n$. Define $\textsc{Sim}_{[k]}(\mathcal{C}) = (\textsc{Sim}_1(\mathcal{C}), \ldots, \textsc{Sim}_k(\mathcal{C}))$. For the remainder of the proof, we will assume that $\mathcal{M}_i(T) \approx_{\varepsilon,\delta} \textsc{Sim}_i(\mathcal{C})$

for all $i \in [k]$, which will be the case with probability at least $1 - \sum_{i=1}^{k} \beta_i$ over the choice of the sample.

Fix any $(r_1, \ldots, r_k) \in \mathcal{R}_1 \times \cdots \times \mathcal{R}_k$:

$$\begin{aligned}\Pr[\mathcal{M}_{[k]}(T) = (r_1, \ldots, r_k)] &= \prod_{i=1}^{k} \Pr[\mathcal{M}_i(T) = r_i] \\ &\leq \prod_{i=1}^{k} e^{\varepsilon_i} \Pr[\textsc{Sim}_i(\mathcal{C}) = r_i] \\ &= e^{\sum_{i=1}^{k} \varepsilon_i} \Pr[\textsc{Sim}_{[k]}(\mathcal{C}) = (r_1, \ldots, r_k)]\end{aligned}$$

For any $\mathcal{O} \subseteq \mathcal{R}_1 \times \cdots \times \mathcal{R}_k$,

$$\begin{aligned}\Pr[\mathcal{M}_{[k]}(T) \in \mathcal{O}] &= \int_{o \in \mathcal{O}} \Pr[\mathcal{M}_{[k]}(T) = o] do \\ &\leq \int_{o \in \mathcal{O}} e^{\sum_{i=1}^{k} \varepsilon_i} \Pr[\textsc{Sim}_{[k]}(\mathcal{C}) = o] do = e^{\sum_{i=1}^{k} \varepsilon_i} \Pr[\textsc{Sim}_{[k]}(\mathcal{C}) \in \mathcal{O}].\end{aligned}$$

A symmetric argument would show that $\Pr[\textsc{Sim}_{[k]}(\mathcal{C}) \in \mathcal{O}] \leq e^{\sum_{i=1}^{k} \varepsilon_i} \Pr[\mathcal{M}_{[k]}(T) \in \mathcal{O}]$.

The mapping $\textsc{Sim}_{[k]}(\mathcal{C})$ serves as a simulator for $\mathcal{M}_{[k]}(T)$, so $\mathcal{M}_{[k]}$ is $(\sum_{i=1}^{k} \beta_i, \sum_{i=1}^{k} \varepsilon_i, 0)$-perfectly generalizing. ■