2

Generalization in Deep Learning

K. Kawaguchi, Y. Bengio, and L. Kaelbling

Abstract: This chapter provides theoretical insights into why and how deep learning can generalize well, despite its large capacity, complexity, possible algorithmic instability, non-robustness, and sharp minima. This chapter forms a response to an open question in the literature. We also discuss approaches to provide non-vacuous generalization guarantees for deep learning. On the basis of theoretical observations, we propose new open problems.

2.1 Introduction

Deep learning has seen significant practical success and has had a profound impact on the conceptual bases of machine learning and artificial intelligence. Along with its practical success, the theoretical properties of deep learning have been a subject of active investigation. For the *expressivity* of neural networks, there are classical results regarding their universality (Leshno et al., 1993) and their exponential advantages over hand-crafted features (Barron, 1993). Another series of theoretical studies has considered how *trainable* (or optimizable) deep hypothesis spaces are, revealing structural properties that may enable non-convex optimization (Choromanska et al., 2015; Kawaguchi, 2016b). However, merely having an *expressive* and *trainable* hypothesis space does not guarantee good performance in predicting the values of future inputs, because of possible over-fitting to training data. This leads to the study of *generalization*, which is the focus of this chapter.

Some classical theory work attributes generalization ability to the use of a low-capacity class of hypotheses (Vapnik, 1998; Mohri et al., 2012). From the viewpoint of compact representation, which is related to small capacity, it has been shown that deep hypothesis spaces have an exponential advantage over shallow hypothesis spaces for representing some classes of natural target functions (Pascanu et al., 2014; Montufar et al., 2014; Livni et al., 2014; Telgarsky, 2016; Poggio et al., 2017). In other words, when some assumptions implicit in the hypothesis space (e.g., the

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

deep composition of piecewise linear transformations) are approximately satisfied by the target function, one can achieve very good generalization, compared with methods that do not rely on that assumption. However, a recent paper (Zhang et al., 2017) showed empirically that successful deep hypothesis spaces have sufficient capacity to memorize random labels. This observation has been called an "apparent paradox" and has led to active discussion by many researchers (Arpit et al., 2017; Krueger et al., 2017; Hoffer et al., 2017; Wu et al., 2017; Dziugaite and Roy, 2017; Dinh et al., 2017). Zhang et al. (2017) concluded with an open problem stating that understanding such observations requires the rethinking of generalization, while Dinh et al. (2017) stated that explaining why deep learning models can generalize well, despite their overwhelming capacity, is an open area of research.

In §2.3 we illustrate that, even in the case of linear models, hypothesis spaces with overwhelming capacity can result in arbitrarily small test errors and expected risks. Here, the *test error* is the error of a learned hypothesis on data on which it was not trained, but which is often drawn from the same distribution. Test error is a measure of how well the hypothesis generalizes to new data. We will examine this phenomenon closely, extending the original open problem from previous papers (Zhang et al., 2017; Dinh et al., 2017) into a new open problem that strictly includes the original. We reconcile an apparent paradox by checking theoretical consistency and identifying a difference in the underlying assumptions. Considering the differences in focus of theory and practice, we outline possible practical roles that generalization theory can play.

Towards addressing these issues, §2.4 presents generalization bounds based on validation datasets, which can provide non-vacuous and numerically-tight generalization guarantees for deep learning in general. Section 2.5 analyzes generalization errors based on training datasets, focusing on a specific case of feed-forward neural networks with ReLU units and max pooling. Under these conditions, the developed theory provides quantitatively tight theoretical insights into the generalization behavior of neural networks.

2.2 Background

Let $\mathcal{R}[f]$ be the expected risk of a function f, $\mathcal{R}[f] = \mathbb{E}_{X,Y\sim\mathbb{P}_{(X,Y)}}[\mathcal{L}(f(X),Y)]$, where X and Y are a input and a target, $\mathcal{L}$ is a loss function, and $\mathbb{P}_{(X,Y)}$ is the true distribution. Let $\hat{f}_{\mathcal{A},S}\colon \mathcal{X} \to \mathcal{Y}$ be a model learned by a learning algorithm $\mathcal{A}$ (including random seeds for simplicity) using a training dataset $S = ((X^{(i)},Y^{(i)}))_{i=1}^{m}$ of size m. Let $\mathcal{R}_S[f]$ be the empirical risk of f as $\mathcal{R}_S[f] = \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}(f(X^{(i)}),Y^{(i)})$ with $\{(X^{(i)},Y^{(i)})\}_{i=1}^{m} = S$. Let $\mathcal{F}$ be a set of functions endowed with some structure, i.e., a *hypothesis space*. All vectors are *column* vectors in this chapter. For any given variable v, let d_v be its dimensionality.

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

A goal in machine learning is typically framed as the minimization of the expected risk $\mathcal{R}[\hat{f}_{\mathcal{A},S}]$. We typically aim to minimize the non-computable expected risk $\mathcal{R}[\hat{f}_{\mathcal{A},S}]$ by minimizing the computable empirical risk $\mathcal{R}_S[\hat{f}_{\mathcal{A},S}]$ (i.e., empirical risk minimization). One goal of generalization theory is to explain and justify when and how minimizing $\mathcal{R}_S[\hat{f}_{\mathcal{A},S}]$ is a sensible approach to minimizing $\mathcal{R}[\hat{f}_{\mathcal{A},S}]$ by analyzing

$$\text{the generalization gap} := \mathcal{R}[\hat{f}_{\mathcal{A},S}] - \mathcal{R}_S[\hat{f}_{\mathcal{A},S}].$$

In this section only, we use the typical assumption that S is generated by independent and identically distributed (i.i.d.) draws according to the true distribution $\mathbb{P}_{(X,Y)}$; the following sections of this chapter do not utilize this assumption. Under this assumption, a primary challenge of analyzing the generalization gap stems from the *dependence* of $\hat{f}_{\mathcal{A},S}$ on the same dataset S as that used in the definition of $\mathcal{R}_S$. Several approaches in *statistical learning theory* have been developed to handle this dependence.

The *hypothesis-space complexity* approach handles this dependence by decoupling $\hat{f}_{\mathcal{A},S}$ from the particular dataset S by considering the worst-case gap for functions in the hypothesis space as

$$\mathcal{R}[\hat{f}_{\mathcal{A},S}] - \mathcal{R}_S[\hat{f}_{\mathcal{A},S}] \leq \sup_{f \in \mathcal{F}} \mathcal{R}[f] - \mathcal{R}_S[f],$$

and by carefully analyzing the right-hand side. Because the cardinality of $\mathcal{F}$ is typically (uncountably) infinite, a direct use of the union bound over all elements in $\mathcal{F}$ yields a vacuous bound, leading to the need to consider different quantities for characterizing $\mathcal{F}$, e.g., Rademacher complexity and the Vapnik–Chervonenkis (VC) dimension. For example, if the codomain of $\mathcal{L}$ is in $[0,1]$, we have (Mohri et al., 2012, Theorem 3.1) that for any $\delta > 0$, with probability at least $1 - \delta$,

$$\sup_{f \in \mathcal{F}} \mathcal{R}[f] - \mathcal{R}_S[f] \leq 2\mathfrak{R}_m(\mathcal{F}) + \sqrt{\frac{\ln \frac{1}{\delta}}{2m}},$$

where $\mathfrak{R}_m(\mathcal{F})$ is the Rademacher complexity of the set $\{(X,Y) \mapsto \mathcal{L}(f(X),Y) \colon f \in \mathcal{F}\}$ and is defined by

$$\mathfrak{R}_m(\mathcal{F}) = \mathbb{E}_{S,\xi}\left[\sup_{f \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^{m} \xi_i \mathcal{L}(f(X^{(i)}),Y^{(i)})\right],$$

where $\xi = (\xi_1, \ldots, \xi_n)$ and $\xi_1, \ldots, \xi_n$ are independent uniform random variables taking values in $\{-1,+1\}$ (i.e., Rademacher variables). For the deep-learning hypothesis spaces $\mathcal{F}$, there are several well-known bounds on $\mathfrak{R}_m(\mathcal{F})$, including those with explicit exponential dependence on depth (Sun et al., 2016; Neyshabur et al.,

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

2015a; Xie et al., 2015) and explicit linear dependence on the number of trainable parameters (Shalev-Shwartz and Ben-David, 2014). There has been significant work on improving the bounds in this approach, but all existing solutions with this approach still depend on the complexity of a hypothesis space or a sequence of hypothesis spaces, resulting in vacuous, and numerically too loose, generalization bounds.

The *stability* approach deals with the dependence of $\hat{f}_{\mathcal{A},S}$ on the dataset S by considering the *stability* of algorithm $\mathcal{A}$ with respect to different datasets. The considered stability is a measure of how much changing a data point in S can change $\hat{f}_{\mathcal{A},S}$. For example, an algorithm $\mathcal{A}$ is said to have uniform stability β (w.r.t. $\mathcal{L}$) if we have that for all $S \in (\mathcal{X} \times \mathcal{Y})^m$, all $i \in \{1, \ldots, m\}$, and all $(X, Y) \in \mathcal{X} \times \mathcal{Y}$,

$$|\mathcal{L}(\hat{f}_{\mathcal{A},S}(X), Y) - \mathcal{L}(\hat{f}_{\mathcal{A},S^{\setminus i}}(X), Y)| \leq \beta,$$

where $S^{\setminus i} = ((X^{(1)}, Y^{(1)}), \ldots, (X^{(i-1)}, Y^{(i-1)}), (X^{(i+1)}, Y^{(i+1)}), \ldots, (X^{(m)}, Y^{(m)}))$ ($S^{\setminus i}$ is S with the ith sample being removed). If the algorithm $\mathcal{A}$ has uniform stability β (w.r.t. $\mathcal{L}$) and if the codomain of $\mathcal{L}$ is in $[0, M]$, we have (Bousquet and Elisseeff, 2002) that for any $\delta > 0$, with probability at least $1 - \delta$,

$$\mathcal{R}[\hat{f}_{\mathcal{A},S}] - \mathcal{R}_S[\hat{f}_{\mathcal{A},S}] \leq 2\beta + (4m\beta + M)\sqrt{\frac{\ln \frac{1}{\delta}}{2m}}.$$

On the basis of previous work on stability (e.g., Hardt et al., 2016; Kuzborskij and Lampert, 2017; Gonen and Shalev-Shwartz, 2017), one may conjecture some reason for generalization in deep learning.

The *robustness* approach avoids dealing with certain details of the dependence of $\hat{f}_{\mathcal{A},S}$ on S by considering the robustness of algorithm $\mathcal{A}$ for all possible datasets. In contrast with stability, robustness is the measure of how much the loss value can vary with respect to the *space* of the values of (X, Y). More precisely, an algorithm $\mathcal{A}$ is said to be $(|\Omega|, \zeta(\cdot))$-robust if $\mathcal{X} \times \mathcal{Y}$ can be partitioned into $|\Omega|$ disjoint sets $\Omega_1, \ldots, \Omega_{|\Omega|}$, such that, for all $S \in (\mathcal{X} \times \mathcal{Y})^m$, all $(X, Y) \in S$, all $(X', Y') \in \mathcal{X} \times \mathcal{Y}$, and all $i \in \{1, \ldots, |\Omega|\}$, if $(X, Y), (X', Y') \in \Omega_i$ then

$$|\mathcal{L}(\hat{f}_{\mathcal{A},S}(X), Y) - \mathcal{L}(\hat{f}_{\mathcal{A},S}(X'), Y')| \leq \zeta(S).$$

If algorithm $\mathcal{A}$ is $(\Omega, \zeta(\cdot))$-robust and the codomain of $\mathcal{L}$ is upper-bounded by M, given a dataset S we have (Xu and Mannor, 2012) that for any $\delta > 0$, with probability at least $1 - \delta$,

$$|\mathcal{R}[\hat{f}_{\mathcal{A},S}] - \mathcal{R}_S[\hat{f}_{\mathcal{A},S}]| \leq \zeta(S) + M\sqrt{\frac{2|\Omega| \ln 2 + 2 \ln \frac{1}{\delta}}{m}}.$$

The robustness approach requires an *a priori known and fixed* partition of the input

space such that the number of sets in the partition is $|\Omega|$ and the change of loss values in each set of the partition is bounded by $\zeta(S)$ *for all* S (Definition 2 and the proof of Theorem 1 in Xu and Mannor, 2012). In classification, if the *margin* is ensured to be large, we can fix the partition with balls of radius corresponding to this large margin, filling the input space. Recently, this idea was applied to deep learning (Sokolic et al., 2017a,b), producing insightful and effective generalization bounds while still suffering from the curse of the dimensionality of the a priori known and fixed input manifold.

With regard to the above approaches, *flat minima* can be viewed as the concept of low variation in the *parameter space*; i.e., a small perturbation in the parameter space around a solution results in a small change in the loss surface. Several studies have provided arguments for generalization in deep learning based on flat minima (Keskar et al., 2017). However, Dinh et al. (2017) showed that flat minima in practical deep-learning hypothesis spaces can be turned into sharp minima via re-parameterization without affecting the generalization gap, indicating that this requires further investigation.

There have been investigations into the connection between generalization and stochastic gradient descent (SGD) based on the Rademacher complexity, stability, and flat minima. For the Rademacher complexity, we can define the hypothesis space $\mathcal{F}$ explored by SGD, and could argue that the Rademacher complexity of it is somehow small; e.g., SGD with an appropriate initialization finds a minimal-norm solution (Poggio et al., 2018). The stability of SGD has been also analyzed (e.g., Hardt et al., 2016), but it is known that the existing bounds quickly become too loose and vacuous as the training time increases, even in the practical regime of the training time where a neural network can still generalize well. One can also argue that SGD prefers flat minima and degenerate minima, resulting in better generalization (Keskar et al., 2017; Banburski et al., 2019). Stochastic gradient descent with added noise has been also studied, but its convergence rate grows exponentially as the number of parameters increases (Raginsky et al., 2017). For all these approaches, there is yet no theoretical proof with a non-vacuous and numerically tight generalization bound on the practical regime of deep learning.

2.3 Rethinking Generalization

Zhang et al. (2017) demonstrated empirically that several deep hypothesis spaces can memorize random labels, while having the ability to produce zero training error and small test errors for particular natural datasets (e.g., CIFAR-10). They also observed empirically that regularization on the norm of weights seemed to be unnecessary to obtain small test errors, in contradiction to conventional wisdom. These observations suggest the following open problem.

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

Open Problem 1. How to tightly characterize the expected risk $\mathcal{R}[f]$ or the generalization gap $\mathcal{R}[f] - \mathcal{R}_S[f]$ with a sufficiently complex deep-learning hypothesis space $\mathcal{F} \ni f$, to produce theoretical insights and distinguish the case of "natural" problem instances $(\mathbb{P}_{(X,Y)}, S)$ (e.g., images with natural labels) from the case of other problem instances $(\mathbb{P}'_{(X,Y)}, S')$ (e.g., images with random labels).

In support of and extending the empirical observations by Zhang et al. (2017), we provide a theorem (Theorem 2.1) stating that the hypothesis space of over-parameterized linear models can memorize any training data *and* decrease the training and test errors arbitrarily close to zero (including zero itself) *with* arbitrarily large parameters norms, *even when* the parameters are arbitrarily far from the ground-truth parameters. Furthermore, Corollary 2.2 shows that conventional wisdom regarding the norms of the parameters w can fail to explain generalization, even in linear models that might seem not to be over-parameterized. All proofs for this chapter are presented in the appendix.

Theorem 2.1. *Consider a linear model with the training prediction* $\hat{\mathbf{Y}}(w) = \Phi w \in \mathbb{R}^{m\times s}$*, where* $\Phi \in \mathbb{R}^{m\times n}$ *is a fixed feature matrix of the training inputs. Let* $\hat{\mathbf{Y}}_{\text{test}}(w) = \Phi_{\text{test}} w \in \mathbb{R}^{m_{\text{test}}\times s}$ *be the test prediction, where* $\Phi_{\text{test}} \in \mathbb{R}^{m_{\text{test}}\times n}$ *is a fixed feature matrix of the test inputs. Let* $M = [\Phi^\top, \Phi_{\text{test}}^\top]^\top$*. Then, if* $n > m$ *and if* $\operatorname{rank}(\Phi) = m$ *and* $\operatorname{rank}(M) < n$*,*

(i) *for any* $\mathbf{Y} \in \mathbb{R}^{m\times s}$*, there exists a parameter* w' *such that* $\hat{\mathbf{Y}}(w') = \mathbf{Y}$*, and*
(ii) *if there exists a ground truth* w^* *satisfying* $\mathbf{Y} = \Phi w^*$ *and* $\mathbf{Y}_{\text{test}} = \Phi_{\text{test}} w^*$ *then, for any* $\epsilon, \delta \geq 0$*, there exists a parameter* w *such that*
  (a) $\hat{\mathbf{Y}}(w) = \mathbf{Y} + \epsilon A$ *for some matrix* A *with* $\|A\|_F \leq 1$*, and*
  (b) $\hat{\mathbf{Y}}_{\text{test}}(w) = \mathbf{Y}_{\text{test}} + \epsilon B$ *for some matrix* B *with* $\|B\|_F \leq 1$*, and*
  (c) $\|w\|_F \geq \delta$ *and* $\|w - w^*\|_F \geq \delta$*.*

Corollary 2.2. *If* $n \leq m$ *and if* $\operatorname{rank}(M) < n$*, then statement* (ii) *in Theorem* 2.1 *holds.*

Whereas Theorem 2.1 and Corollary 2.2 concern test errors rather than the expected risk (in order to be consistent with empirical studies), Proposition 2.3 below shows the same phenomena for the expected risk for general machine learning models not limited to deep learning and linear hypothesis spaces; i.e., Proposition 2.3 shows that, regarding small capacity, low complexity, stability, robustness, and flat minima, none of these is *necessary* for generalization in machine learning for any given problem instance $(\mathbb{P}_{(X,Y)}, S)$, although one of them can be *sufficient* for generalization. This statement does not contradict the necessary conditions and the no-free-lunch theorem from previous learning theory, as will be explained later in the chapter.

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

Proposition 2.3. *Given a pair* $(\mathbb{P}_{(X,Y)}, S)$ *and a desired* $\epsilon > \inf_{f \in \mathcal{Y}^{\mathcal{X}}} \mathcal{R}[f] - \mathcal{R}_S[f]$, *let* f^*_ϵ *be a function such that* $\epsilon \geq \mathcal{R}[f^*_\epsilon] - \mathcal{R}_S[f^*_\epsilon]$. *Then,*

(i) *for any hypothesis space* $\mathcal{F}$ *whose hypothesis-space complexity is large enough to memorize any dataset and which includes* f^*_ϵ *possibly at an arbitrarily sharp minimum, there exist learning algorithms* $\mathcal{A}$ *such that the generalization gap of* $\hat{f}_{\mathcal{A},S}$ *is at most* ϵ, *and*

(ii) *there exist arbitrarily unstable and arbitrarily non-robust algorithms* $\mathcal{A}$ *such that the generalization gap of* $\hat{f}_{\mathcal{A},S}$ *is at most* ϵ.

Proposition 2.3 is a direct consequence of the following remark which captures the essence of all the above observations (see Appendix A5 for the proof of Proposition 2.3).

Remark 2.4. The expected risk $\mathcal{R}[f]$ and the generalization gap $\mathcal{R}[f] - \mathcal{R}_S[f]$ of a hypothesis f with a true distribution $\mathbb{P}_{(X,Y)}$ and a dataset S are completely determined by the tuple $(\mathbb{P}_{(X,Y)}, S, f)$, independently of other factors, such as the hypothesis space $\mathcal{F}$ (and hence its properties such as capacity, Rademacher complexity, and flat-minima) or the properties of random datasets different from the given S (e.g., the stability and robustness of the learning algorithm $\mathcal{A}$). In contrast, conventional wisdom states that these other factors are what matter. This has created the "apparent paradox" in the literature.

From these observations, we propose the following open problem.

Open Problem 2. To tightly characterize the expected risk $\mathcal{R}[f]$ or the generalization gap $\mathcal{R}[f] - \mathcal{R}_S[f]$ of a hypothesis f with a pair $(\mathbb{P}_{(X,Y)}, S)$ of a true distribution and a dataset, *so as to produce theoretical insights* based only on properties of the hypothesis f and the pair $(\mathbb{P}_{(X,Y)}, S)$.

Solving Open Problem 2 for deep learning implies solving Open Problem 1, but not vice versa. Open Problem 2 encapsulates the essence of Open Problem 1 and all the issues from our Theorem 2.1, Corollary 2.2, and Proposition 2.3.

2.3.1 Consistency of Theory

The empirical observations in Zhang et al. (2017) and our results above may seem to contradict the results of statistical learning theory. However, there is no contradiction, and the apparent inconsistency arises from the misunderstanding and misuse of the precise meanings of the theoretical statements.

Statistical learning theory can be considered to provide two types of statements relevant to the scope of this chapter. The first type (which comes from upper

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

bounds) is logically in the form of "p implies q," where p := "the hypothesis-space complexity is small" (or another statement about stability, robustness, or flat minima), and q := "the generalization gap is small." Notice that "p implies q" does not imply "q implies p." Thus, based on statements of this type, it is entirely possible that the generalization gap is small even when the hypothesis-space complexity is large or the learning mechanism is unstable, non-robust, or subject to sharp minima.

The second type of statement (which comes from lower bounds) is, logically, in the following form. In a set U_{all} of all possible problem configurations, there exists a subset $U \subseteq U_{\text{all}}$ such that "q implies p" in U (with the same definitions of p and q as in the previous paragraph). For example, Mohri et al. (2012, Section 3.4) derived lower bounds on the generalization gap by showing the existence of a "bad" distribution that characterizes U. Similarly, the classical no-free-lunch theorems are the results that give a worst-case distribution for each algorithm. However, if the problem instance at hand (e.g., object classification with MNIST or CIFAR-10) is not in such a subset U in the proofs (e.g., if the data distribution is not among the "bad" ones considered in the proofs), q does not necessarily imply p. Thus, it is still naturally possible that the generalization gap is small with large hypothesis-space complexity, instability, non-robustness, and sharp minima. Therefore, there is no contradiction or paradox.

2.3.2 Differences in Assumptions and Problem Settings

Under certain assumptions, many results in statistical learning theory have been shown to be tight and insightful (e.g., Mukherjee et al., 2006; Mohri et al., 2012). Hence, the need to rethink generalization comes partly from differences in the assumptions and problem settings.

Figure 2.1 illustrates the differences between the assumptions in statistical learning theory and in some empirical studies. On the one hand, in statistical learning theory a distribution $\mathbb{P}_{(X,Y)}$ and a dataset S are usually unspecified except that $\mathbb{P}_{(X,Y)}$ is in some set $\mathcal{P}$ and the dataset $S \in D$ is drawn randomly according to $\mathbb{P}_{(X,Y)}$ (typically with the i.i.d. assumption). On the other hand, in most empirical studies and in our theoretical results (Theorem 2.1 and Proposition 2.3), the distribution $\mathbb{P}_{(X,Y)}$ is still unknown, yet specified (e.g., via a real-world process), and the dataset S is specified and usually known (e.g., CIFAR-10 or ImageNet). Intuitively, whereas statistical learning theory needs to consider a set $\mathcal{P} \times D$ because of weak assumptions, some empirical studies can focus on a specified point $(\mathbb{P}_{(X,Y)}, S)$ in a set $\mathcal{P} \times D$ because of stronger assumptions. Therefore, by using the same terminology such as "expected risk" and "generalization" in both cases, we are susceptible to confusion and apparent contradiction.

Lower bounds, necessary conditions, and tightness in statistical learning theory

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

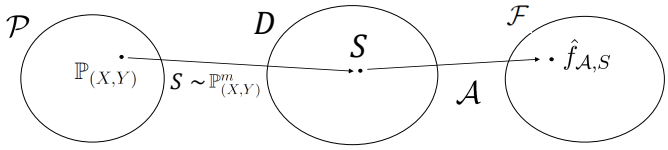

Figure 2.1 An illustration of differences in assumptions. Statistical learning theory analyzes the generalization behaviors of $\hat{f}_{\mathcal{A},S}$ over randomly drawn *unspecified* datasets $S \in D$ according to some *unspecified* distribution $\mathbb{P}_{(X,Y)} \in \mathcal{P}$. Intuitively, statistical learning theory is concerned more with questions regarding the set $\mathcal{P} \times D$ because of the *unspecified* nature of $(\mathbb{P}_{(X,Y)}, S)$, whereas certain empirical studies (e.g., Zhang et al., 2017) can focus on questions regarding each *specified* point $(\mathbb{P}_{(X,Y)}, S) \in \mathcal{P} \times D$.

are typically defined via a worst-case distribution $\mathbb{P}^{\text{worst}}_{(X,Y)} \in \mathcal{P}$. For instance, classical "no-free-lunch" theorems and certain lower bounds on the generalization gap (e.g., Mohri et al., 2012, Section 3.4) have actually been proven *for* the worst-case distribution $\mathbb{P}^{\text{worst}}_{(X,Y)} \in \mathcal{P}$. Therefore, "tight" and "necessary" typically mean "tight" and "necessary" for the set $\mathcal{P} \times D$ (e.g., through the worst or average case), but *not* for each particular point $(\mathbb{P}_{(X,Y)}, S) \in \mathcal{P} \times D$. From this viewpoint, we can understand that even if the quality of the set $\mathcal{P} \times D$ is "bad" overall, there may exist a "good" point $(\mathbb{P}_{(X,Y)}, S) \in \mathcal{P} \times D$.

Several approaches to statistical learning theory, such as the data-dependent and Bayesian ones (Herbrich and Williamson, 2002; Dziugaite and Roy, 2017), use further assumptions on the set $\mathcal{P} \times D$ to take advantage of more prior and posterior information; these have the ability to tackle Open Problem 1. However, these approaches do not apply to Open Problem 2 as they still depend on factors other than the given $(\mathbb{P}_{(X,Y)}, S, f)$. For example, data-dependent bounds with the *luckiness framework* (Shawe-Taylor et al., 1998; Herbrich and Williamson, 2002) and *empirical* Rademacher complexity (Koltchinskii and Panchenko, 2000; Bartlett et al., 2002) still depend on the concept of hypothesis spaces (or the sequence of hypothesis spaces), and the robustness approach (Xu and Mannor, 2012) depends on datasets different from a given S via the definition of robustness (i.e., in §2.2, $\zeta(S)$ is a data-dependent term, but the definition of ζ itself and Ω depend on datasets other than S).

We note that analyzing a set $\mathcal{P} \times D$ is of significant interest for its own merits and is a natural task in the field of computational complexity (e.g., categorizing a set of problem instances into subsets with or without polynomial solvability). Indeed, a situation where theory theory focuses on a set whereas many practical studies focus on each element in the set is prevalent in computer science (see the discussion in Appendix A2 for more detail).

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

2.3.3 Practical Role of Generalization Theory

From the discussions above, we can see that there is a *logically expected* difference between the scope of theory and the focus in practice; it is logically expected that there are problem instances where theoretical bounds are pessimistic. In order for generalization theory to have maximal impact in practice, we must be clear on the set of different roles it can play regarding practice, and then work to extend and strengthen it in each of these roles. We have identified the following practical roles for a theory:

Role 1 To provide guarantees on expected risk.
Role 2 To guarantee that the generalization gap:
  Role 2.1 is small for a given fixed S, and/or
  Role 2.2 approaches zero for *a fixed model class* as m increases.
Role 3 To provide theoretical insights to guide the search over model classes.

2.4 Generalization Bounds via Validation

In practical deep learning, we typically adopt the training–validation paradigm, usually with a held-out validation set. We then search over hypothesis spaces by changing architectures (and other hyper-parameters) to obtain a low validation error. In this view we can conjecture the reason why deep learning can sometimes generalize well to be as follows: it is partially because we can obtain a good model via search using a validation dataset. Indeed, Proposition 2.5 states that if the validation error of a hypothesis is small, it is guaranteed to generalize well regardless of its capacity, Rademacher complexity, stability, robustness, or flat minima. Let $S^{(\mathrm{val})}$ be a held-out validation dataset, of size m_{val}, which is independent of the training dataset S.

Proposition 2.5. (Generalization guarantee via validation error) *Assume that* $S^{(\mathrm{val})}$ *is generated by i.i.d. draws according to a true distribution* $\mathbb{P}_{(X,Y)}$. *Let* $\kappa_{f,i} = \mathcal{R}[f] - \mathcal{L}(f(X^{(i)}), Y^{(i)})$ *for* $(X^{(i)}, Y^{(i)}) \in S^{(\mathrm{val})}$. *Suppose that* $\mathbb{E}[\kappa_{f,i}^2] \leq \gamma^2$ *and* $|\kappa_{f,i}| \leq C$ *almost surely, for all* $(f,i) \in \mathcal{F}_{\mathrm{val}} \times \{1,\ldots,m_{\mathrm{val}}\}$. *Then, for any* $\delta > 0$, *with probability at least* $1-\delta$, *the following holds for all* $f \in \mathcal{F}_{\mathrm{val}}$:

$$\mathcal{R}[f] \leq \mathcal{R}_{S^{(\mathrm{val})}}[f] + \frac{2C\ln(|\mathcal{F}_{\mathrm{val}}|/\delta)}{3m_{\mathrm{val}}} + \sqrt{\frac{2\gamma^2\ln(|\mathcal{F}_{\mathrm{val}}|/\delta)}{m_{\mathrm{val}}}}.$$

Here, $\mathcal{F}_{\mathrm{val}}$ is defined as a set of models f that is independent of a held-out validation dataset $S^{(\mathrm{val})}$. Importantly, $\mathcal{F}_{\mathrm{val}}$ *can, however, depend on the training dataset* S, because a dependence on S does not imply a dependence on $S^{(\mathrm{val})}$. For example, $\mathcal{F}_{\mathrm{val}}$ can contain a set of models f such that each f is a model, obtained

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

at the end of training, with at least 99.5% *training* accuracy. In this example, $|\mathcal{F}_{\text{val}}|$ equals at most the number of end epochs times the cardinality of the set of possible hyper-parameter settings, and it is likely to be much smaller than that because of the 99.5% training accuracy criterion and the fact that a space of many hyper-parameters is narrowed down by using the training dataset as well as other datasets from different tasks. If a hyper-parameter search depends on the validation dataset, $\mathcal{F}_{\text{val}}$ will be the possible space of the search instead of the space actually visited by the search. We can also use a sequence $(\mathcal{F}_{\text{val}}^{(j)})_j$ to obtain validation-data-dependent bounds (see Appendix A6).

The bound in Proposition 2.5 is non-vacuous and tight enough to be practically meaningful. For example, consider a classification task with 0–1 loss. Set $m_{\text{val}} = 10,000$ (e.g., as in MNIST and CIFAR-10) and $\delta = 0.1$. Then, even in the worst case, with $C = 1$ and $\gamma^2 = 1$, and even with $|\mathcal{F}_{\text{val}}| = 1,000,000,000$, we have, with probability at least 0.9, that $\mathcal{R}[f] \leq \mathcal{R}_{S^{(\text{val})}}[f] + 6.94\%$ for all $f \in \mathcal{F}_{\text{val}}$. In a non-worst-case scenario, for example, with $C = 1$ and $\gamma^2 = (0.05)^2$, we can replace 6.94% by 0.49%. With a larger validation set (e.g., as in ImageNet) and/or more optimistic C and γ^2 values, we can obtain much better bounds.

Although Proposition 2.5 poses the concern of increasing that the generalization bound will be increased when one is using a single validation dataset with too large a value of $|\mathcal{F}_{\text{val}}|$, the rate of increase goes as only $\ln |\mathcal{F}_{\text{val}}|$ and $\sqrt{\ln |\mathcal{F}_{\text{val}}|}$. We can also avoid dependence on the cardinality of $\mathcal{F}_{\text{val}}$ by using Remark 2.6.

Remark 2.6. Assume that $S^{(\text{val})}$ is generated by i.i.d. draws according to $\mathbb{P}_{(X,Y)}$. By applying Mohri et al. (2012, Theorem 3.1) to $\mathcal{F}_{\text{val}}$, if the codomain of $\mathcal{L}$ is in $[0, 1]$, with probability at least $1 - \delta$, then, for all $f \in \mathcal{F}_{\text{val}}$,

$$\mathcal{R}[f] \leq \mathcal{R}_{S^{(\text{val})}}[f] + 2\Re_m(F_{\text{val}}) + \sqrt{(\ln 1/\delta)/m_{\text{val}}}.$$

Unlike the standard use of Rademacher complexity with a training dataset, the set $\mathcal{F}_{\text{val}}$ cannot depend on the validation set S but can depend on the training dataset S in any manner, and hence $\mathcal{F}_{\text{val}}$ differs significantly from the typical hypothesis space defined by the parameterization of models. We can thus end up with a very different effective capacity and hypothesis complexity (as selected by model search using the validation set) depending on whether the training data are random or have interesting structure which the neural network can capture.

2.5 Direct Analyses of Neural Networks

Unlike the previous section, this section analyzes the generalization gap with a training dataset S. In §2.3, we extended Open Problem 1 to Open Problem 2, and identified the different assumptions in theoretical and empirical studies. Accord-

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

Table 2.1 *Additional notation for DAG in Section 2.5*

Description	Notation
pre-activation output of the lth hidden layer given (X, w)	$z^{[l]}(X, w)$
component of an input $X^{(i)}$ used for the jth path	$\bar{X}_j^{(i)}$
path activation for the jth path given $X^{(i)}$ and $(X^{(i)}, w)$	$\bar{\sigma}_j(X^{(i)}, w)$
vector of trainable parameters	w
the jth path weight for the kth output unit	$\bar{w}_{k,j}$
vector of parameters trained with $(\mathcal{A}, S)$	w^S
vector of parameters frozen in two-phase training	w_σ
weight matrix connecting the $(l-1)$th layer to the lth layer	$W^{[l]}$
weight matrix connecting the l'th layer to the lth layer	$W^{(l,l')}$

ingly, this section aims to address these problems, to some extent, both in the case of particular specified datasets and the case of random unspecified datasets. To achieve this goal, this section presents a *direct analysis* of neural networks, rather than deriving results about neural networks from more generic theories based on capacity, Rademacher complexity, stability, or robustness.

Sections 2.5.2 and 2.5.3 deal with the squared loss, while §2.5.4 considers 0–1 loss with multi-labels. Table 2.1 summarizes the notation used in this section for a directed acyclic graph (DAG).

2.5.1 Model Description via Deep Paths

We consider general neural networks of any depth that have the structure of a DAG with ReLU nonlinearity and/or max pooling. This includes a feedforward network of any structure and with convolutional and/or fully connected layers, potentially with skip connections. For pedagogical purposes, we first discuss our model description for layered networks without skip connections, and then describe it for DAGs.

Layered nets without skip connections. Let $z^{[l]}(X, w) \in \mathbb{R}^{n_l}$ be the pre-activation vector of the lth hidden layer, where n_l is the width of the lth hidden layer and w represents the trainable parameters. Let $L-1$ be the number of hidden layers. For layered networks without skip connections, the pre-activation (or pre-nonlinearity) vector of the lth layer can be written as

$$z^{[l]}(X, w) = W^{[l]} \sigma^{(l-1)} \left(z^{[l-1]}(X, w) \right),$$

with a boundary definition $\sigma^{(0)} \left(z^{[0]}(X, w) \right) \equiv X$, where $\sigma^{(l-1)}$ represents nonlinearity via ReLU and/or max pooling at the $(l-1)$th hidden layer, and $W^{[l]} \in \mathbb{R}^{n_l \times n_{l-1}}$

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

is a matrix of weight parameters connecting the $(l-1)$th layer to the lth layer. Here, $W^{[l]}$ can have *any* structure (e.g., shared and sparse weights to represent a convolutional layer). Let $\dot{\sigma}^{[l]}(X,w)$ be a vector, each of whose elements is 0 or 1, such that $\sigma^{[l]}\left(z^{[l]}(X,w)\right) = \dot{\sigma}^{[l]}(X,w) \circ z^{[l]}(X,w)$, which is an elementwise product of the vectors $\dot{\sigma}^{[l]}(X,w)$ and $z^{[l]}(X,w)$. Then we can write the pre-activation of the kth output unit at the last layer $l = L$ as

$$z_k^{[L]}(X,w) = \sum_{j_{L-1}=1}^{n_{L-1}} W_{kj_{L-1}}^{[L]} \dot{\sigma}_{j_{L-1}}^{(L-1)}(X,w) z_{j_{L-1}}^{[L-1]}(X,w).$$

By expanding $z^{[l]}(X,w)$ repeatedly and exchanging the sum and product via the distributive law of multiplication, we obtain

$$z_k^{[L]}(X,w) = \sum_{j_{L-1}=1}^{n_{L-1}} \sum_{j_{L-2}=1}^{n_{L-2}} \cdots \sum_{j_0=1}^{n_0} \overline{W}_{kj_{L-1}j_{L-2}\cdots j_0} \dot{\sigma}_{j_{L-1}j_{L-2}\cdots j_1}(X,w) X_{j_0},$$

where

$$\overline{W}_{kj_{L-1}j_{L-2}\cdots j_0} = W_{kj_{L-1}}^{[L]} \prod_{l=1}^{L-1} W_{j_l j_{l-1}}^{[l]}$$

and

$$\dot{\sigma}_{j_{L-1}j_{L-2}\cdots j_1}(X,w) = \prod_{l=1}^{L-1} \dot{\sigma}_{j_l}^{[l]}(X,w).$$

By merging the indices $j_0, \ldots, j_{L-1}$ into j with some bijection between

$$\{1,\ldots,n_0\} \times \cdots \times \{1,\ldots,n_{L-1}\} \ni (j_0,\ldots,j_{L-1}) \quad \text{and} \quad \{1,\ldots,n_0 n_1 \cdots n_{L-1}\} \ni j,$$

we have

$$z_k^{[L]}(X,w) = \textstyle\sum_j \bar{w}_{k,j} \bar{\sigma}_j(X,w) \bar{X}_j,$$

where $\bar{w}_{k,j}, \bar{\sigma}_j(X,w)$ and $\bar{X}_j$ represent $\overline{W}_{kj_{L-1}j_{L-2}\cdots j_0}, \dot{\sigma}_{j_{L-1}j_{L-2}\cdots j_1}(X,w)$ and X_{j_0}, respectively with the change of indices (i.e., $\sigma_j(X,w)$ and $\bar{X}_j$, respectively, contain the n_0 numbers and $n_1 \cdots n_{L-1}$ numbers of the same copy of each $\dot{\sigma}_{j_{L-1}j_{L-2}\cdots j_1}(X,w)$ and X_{j_0}). Note that $\sum_j$ represents summation over all the paths from the input X to the kth output unit.

DAGs. Remember that every DAG has at least one topological ordering, which can be used to to create a layered structure with possible skip connections (e.g., see Healy and Nikolov, 2001; Neyshabur et al., 2015a). In other words, we consider

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

DAGs such that the pre-activation vector of the lth layer can be written as

$$z^{[l]}(X,w) = \sum_{l'=0}^{l-1} W^{(l,l')}\sigma^{[l']}\left(z^{[l']}(X,w)\right)$$

with a boundary definition $\sigma^{(0)}\left(z^{[0]}(X,w)\right) \equiv X$, where $W^{(l,l')} \in \mathbb{R}^{n_l \times n_{l'}}$ is a matrix of weight parameters connecting the l'th layer to the lth layer. Again, $W^{(l,l')}$ can have *any* structure. Thus, in the same way as with layered networks without skip connections, for all $k \in \{1,\ldots,d_y\}$,

$$z_k^{[L]}(X,w) = \textstyle\sum_j \bar{w}_{k,j}\bar{\sigma}_j(X,w)\bar{X}_j,$$

where $\sum_j$ represents summation over all paths from the input X to the kth output unit; i.e., $\bar{w}_{k,j}\bar{\sigma}_j(X,w)\bar{X}_j$ is the contribution from the jth path to the kth output unit. Each of $\bar{w}_{k,j}$, $\bar{\sigma}_j(X,w)$, and $\bar{X}_j$ is defined in the same manner as in the case of layered networks without skip connections. In other words, the jth path weight $\bar{w}_{k,j}$ is the product of the weight parameters in the jth path, and $\bar{\sigma}_j(X,w)$ is the product of the 0–1 activations in the jth path, corresponding to ReLU nonlinearity and max pooling; $\bar{\sigma}_j(X,w) = 1$ if all units in the jth path are active, and $\bar{\sigma}_j(X,w) = 0$ otherwise. Also, $\bar{X}_j$ is the input used in the jth path. Therefore, for DAGs, including layered networks without skip connections,

$$z_k^{[L]}(X,w) = [\bar{X} \circ \bar{\sigma}(X,w)]^\top \bar{w}_k, \tag{2.1}$$

where $[\bar{X} \circ \bar{\sigma}(X,w)]_j = \bar{X}_j\bar{\sigma}_j(X,w)$ and $(\bar{w}_k)_j = \bar{w}_{k,j}$ are vectors whose size is the number of paths.

2.5.2 *Theoretical Insights via Tight Theory for Every Pair* $(\mathbb{P}, S)$

Theorem 2.7 below solves Open Problem 2 (and hence Open Problem 1) for neural networks with squared loss by stating that the generalization gap of a trainable parameter vector w with respect to a problem $(\mathbb{P}_{(X,Y)}, S)$ is tightly analyzable with theoretical insights, based only on the quality of w and the pair $(\mathbb{P}_{(X,Y)}, S)$. We do *not* assume that S is generated randomly on the basis of some relationship with $\mathbb{P}_{(X,Y)}$; the theorem holds for any dataset, regardless of how it was generated. Let w^S and $\bar{w}_k^S$ be the parameter vectors w and $\bar{w}_k$ learned with a dataset S and $\mathcal{A}$. Let $\mathcal{R}[w^S]$ and $\mathcal{R}_S[w^S]$ be the expected risk and empirical risk of the model with the learned parameter w^S. Let $z_i = [\bar{X}^{(i)} \circ \bar{\sigma}(X^{(i)}, w^S)]$. Let $G = \mathbb{E}_{X,Y\sim\mathbb{P}_{(X,Y)}}[zz^\top] - \frac{1}{m}\sum_{i=1}^m z_i z_i^\top$ and $v = \frac{1}{m}\sum_{i=1}^m Y_k^{(i)} z_i - \mathbb{E}_{X,Y\sim\mathbb{P}_{(X,Y)}}[Y_k z]$. Given a matrix M, let $\lambda_{\max}(M)$ be the largest eigenvalue of M.

Theorem 2.7. *Let* $\{\lambda_j\}_j$ *and be a set of eigenvalues and* $\{u_j\}_j$ *the corresponding*

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

orthonormal set of eigenvectors of G. Let $\theta^{(1)}_{\bar{w}_k,j}$ *be the angle between* u_j *and* $\bar{w}_k$*. Let* $\theta^{(2)}_{\bar{w}_k}$ *be the angle between* v *and* $\bar{w}_k$*. Then (deterministically),*

$$\begin{aligned}\mathcal{R}[w^S] - \mathcal{R}_S[w^S] - c_y &= \sum_{k=1}^{s}\left(2\|v\|_2\|\bar{w}_k^s\|_2\cos\theta^{(2)}_{\bar{w}_k^s} + \|\bar{w}_k^s\|_2^2\sum_j \lambda_j\cos^2\theta^{(1)}_{\bar{w}_k^s,j}\right)\\ &\le \sum_{k=1}^{s}\left(2\|v\|_2\|\bar{w}_k^s\|_2 + \lambda_{\max}(G)\|\bar{w}_k^s\|_2^2\right),\end{aligned}$$

where $c_y = \mathbb{E}_Y[\|Y\|_2^2] - \frac{1}{m}\sum_{i=1}^m \|Y^{(i)}\|_2^2$.

Proof Idea From Equation (2.1) with squared loss, we can decompose the generalization gap into three terms:

$$\begin{aligned}\mathcal{R}[w^S] - \mathcal{R}_S[w^S] = &\sum_{k=1}^{s}\left[(\bar{w}_k^s)^\top\left(\mathbb{E}[zz^\top] - \frac{1}{m}\sum_{i=1}^m z_i z_i^\top\right)\bar{w}_k^s\right]\\ &+ 2\sum_{k=1}^{s}\left[\left(\frac{1}{m}\sum_{i=1}^m Y_k^{(i)} z_i^\top - \mathbb{E}[Y_k z^\top]\right)\bar{w}_k^s\right]\\ &+ \mathbb{E}[Y^\top Y] - \frac{1}{m}\sum_{i=1}^m (Y^{(i)})^\top Y^{(i)}. \qquad (2.2)\end{aligned}$$

By manipulating each term, we obtain the desired statement. See Appendix C3 for a complete proof. □

In Theorem 2.7, there is no issue of a vacuous or too loose a bound. Instead, it indicates that if the norm of the weights $\|\bar{w}_k^s\|_2$ is small then the generalization gap is small, with the tightest bound (i.e., equality) having no dependence on the hypothesis space.

Importantly, in Theorem 2.7, there are two other significant factors in addition to the norm of the weights $\|\bar{w}_k^s\|_2$. First, the eigenvalues of G and v measure the concentration of the given dataset S with respect to the (unknown) $\mathbb{P}_{(X,Y)}$ in the space of the learned representation $z_i = [\bar{X}^{(i)} \circ \bar{\sigma}(X^{(i)}, w^S)]$. Here, we can see the benefit of deep learning from the viewpoint of "deep-path" feature learning: even if a given S is not concentrated in the original space, optimizing w can result in concentrating it in the space of z. Similarly, c_y measures the concentration of $\|Y\|_2^2$, but c_y is independent of w and remains unchanged after a pair $(\mathbb{P}_{(X,Y)}, S)$ is given. Second, the $\cos\theta$ terms measure the similarity between $\bar{w}_k^s$ and these concentration terms. Because the norm of the weights $\|\bar{w}_k^s\|_2$ is multiplied by these other factors, the generalization gap can remain small, even if $\|\bar{w}_k^s\|_2$ is large, as long as some of the other factors are small.

On the basis of a generic bound-based theory, Neyshabur et al. (2015b) and

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

Neyshabur et al. (2015a) proposed controlling the norm of the *path* weights $\|\bar{w}_k\|_2$, which is consistent with our direct bound-less result (and which is as computationally tractable as a standard forward–backward pass[1]). Unlike the previous results, we do *not* require a predefined bound on $\|\bar{w}_k\|_2$ over different datasets, but require only its final value for each S in question, in addition to tighter insights (besides the norm) via equality as discussed above. In addition to the predefined norm bound, these previous results have an explicit exponential dependence on the depth of the network, which does not appear in our Theorem 2.7. Similarly, some previous results specific to layered networks without skip connections (Sun et al., 2016; Xie et al., 2015) contain the 2^{L-1} factor *and* a bound on the product of the norms of weight matrices, $\prod_{l=1}^{L} \|W^{(l)}\|$, rather than on $\sum_k \|\bar{w}_k^s\|_2$. Here, $\sum_k \|\bar{w}_k\|_2^2 \leq \prod_{l=1}^{L} \|W^{(l)}\|_F^2$ because the latter contains all the same terms as the former as well as additional non-negative additive terms after expanding the sums in the definition of the norms.

Therefore, unlike previous bounds, Theorem 2.7 generates these new theoretical insights from a *the tight equality* (in the first line of the equation in Theorem 2.7). Notice that, without manipulating the generalization gap, we can always obtain equality. However, the question answered here is whether or not we can obtain competitive theoretical insights (the path norm bound) via equality instead of inequality. From a practical view point, if the insights obtained are the same (e.g., they regularize the norm), then equality-based theory has the obvious advantage of being more precise.

2.5.3 Probabilistic Bounds over Random Datasets

While the previous subsection tightly analyzed each given point $(\mathbb{P}_{(X,Y)}, S)$, this subsection considers the set $\mathcal{P} \times D \ni (\mathbb{P}_{(X,Y)}, S)$, where D is the set of possible datasets S endowed with an i.i.d. product measure $\mathbb{P}_{(X,Y)}^m$ where $\mathbb{P}_{(X,Y)} \in \mathcal{P}$ (see §2.3.2).

In Equation (2.2), the generalization gap is decomposed into three terms, each containing the difference between a sum of *dependent* random variables and its expectation. The dependence comes from the fact that the $z_i = [\bar{X}^{(i)} \circ \bar{\sigma}(X^{(i)}, w^S)]$ are dependent on the sample index i, because of the dependence of w^S on the entire dataset S. We then observe the following: in $z_k^{[L]}(X, w) = [\bar{X} \circ \bar{\sigma}(X, w)]^\top \bar{w}$, the derivative of $z = [\bar{X} \circ \bar{\sigma}(X, w)]$ with respect to w is zero everywhere (except for the measure-zero set, where the derivative does not exist). Therefore, each step of the (stochastic) gradient descent greedily chooses the best direction in terms of $\bar{w}$ (with the current $z = [\bar{X} \circ \bar{\sigma}(X, w)]$), but not in terms of the w in $z = [\bar{X} \circ \bar{\sigma}(X, w)]$

[1] From the derivation of Equation (2.1), one can compute $\|\bar{w}_k^s\|_2^2$ with a single forward pass using element-wise squared weights, an identity input, and no nonlinearity. One can also follow (Neyshabur et al., 2015b) for this computation.

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

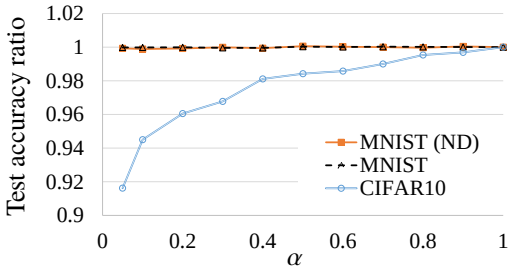

Figure 2.2 Test accuracy ratio (two-phase/base). Notice that the y-axis starts with high initial accuracy, even with a very small dataset size, αm, for learning w_σ.

(see Appendix A3 for more detail). This observation leads to a conjecture that the dependence of $z_i = [\bar{X}^{(i)} \circ \bar{\sigma}(X^{(i)}, w^S)]$, via the training process on the whole dataset S, is not entirely "bad" in terms of the concentration of the sum of the terms involving z_i.

Empirical Observations

As a first step in investigating the dependences of z_i, we evaluated the following novel *two-phase* training procedure which explicitly breaks the dependence of z_i on the sample index i. We first train a network in a standard way, but only using a *partial* training dataset $S_{\alpha m} = \{(X^{(1)}, Y^{(1)}), \ldots, (X^{(\alpha m)}, Y^{(\alpha m)})\}$ of size αm, where $\alpha \in (0, 1)$ (this is the standard phase of the procedure). We then assign the value of $w^{S_{\alpha m}}$ to a new placeholder $w_\sigma := w^{S_{\alpha m}}$ and freeze w_σ, meaning that, as w changes, w_σ does not change. At this point, we have that $z_k^{[L]}(X, w^{S_{\alpha m}}) = [\bar{X} \circ \bar{\sigma}(X, w_\sigma)]^\top \bar{w}_k^{S_{\alpha m}}$. We then keep training only the $\bar{w}_k^{S_{\alpha m}}$ part with the entire training dataset of size m (the freeze phase), yielding the final model via this two-phase training procedure as

$$\tilde{z}_k^{[L]}(X, w^S) = [\bar{X} \circ \bar{\sigma}(X, w_\sigma)]^\top \bar{w}_k^S. \tag{2.3}$$

Note that the vectors $w_\sigma = w^{S_{\alpha m}}$ and $\bar{w}_k^S$ contain the untied parameters in $\tilde{z}_k^{[L]}(X, w^S)$. See Appendix A4 for a simple implementation of this two-phase training procedure that requires at most (approximately) twice as much computational cost as the normal training procedure.

We implemented the two-phase training procedure with the MNIST and CIFAR-10 datasets. The test accuracies of the standard training procedure (the base case) were 99.47% for MNIST (ND), 99.72% for MNIST, and 92.89% for CIFAR-10. Here MNIST (ND) indicates MNIST with no data augmentation. The experimental details are given in Appendix B.

Our source code is available at:

http://lis.csail.mit.edu/code/gdl.html

Figure 2.2 presents the test accuracy ratios for varying α: the test accuracy

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

of the two-phase training procedure divided by the test accuracy of the standard (base) training procedure. The plot in Figure 2.2 begins with $\alpha = 0.05$, for which $\alpha m = 3000$ in MNIST and $\alpha m = 2500$ in CIFAR-10. Somewhat surprisingly, using a much smaller dataset for learning w_σ still resulted in a competitive performance. A dataset from which we could more easily obtain a better generalization (i.e., MNIST) allowed us to use a smaller value of αm to achieve a competitive performance, which is consistent with our discussion above.

Theoretical Results

We now prove a probabilistic bound for the hypotheses resulting from the two-phase training algorithm. Let $\tilde{z}_i = [\bar{X}^{(i)} \circ \bar{\sigma}(X^{(i)}, w_\sigma)]$ where $w_\sigma := w^{S_{\alpha m}}$, as defined in the two-phase training procedure above. Our two-phase training procedure forces $\tilde{z}_{\alpha m+1}, \ldots, \tilde{z}_m$ over samples to be independent random variables (each $\tilde{z}_i$ is dependent over coordinates, which is taken care of in our proof), while maintaining the competitive practical performance of the output model $\tilde{z}_k^{[L]}(\cdot, w^S)$. As a result, we obtain the following bound on the generalization gap for the practical deep models $\tilde{z}_k^{[L]}(\cdot, w^S)$. Let $m_\sigma = (1-\alpha)m$. Given a matrix M, let $\|M\|_2$ be the spectral norm of M.

Assumption 2.8. Let $G^{(i)} = \mathbb{E}_X[\tilde{z}\tilde{z}^\top] - \tilde{z}_i\tilde{z}_i^\top$, $V_{kk'}^{(i)} = Y_k^{(i)}\tilde{z}_{i,k'} - \mathbb{E}_{X,Y}[Y_k\tilde{z}_{k'}]$, and $c_y^{(i)} = \mathbb{E}_Y[\|Y\|_2^2] - \|Y^{(i)}\|_2^2$. Assume that, for all $i \in \{\alpha m + 1, \ldots, m\}$:

- $C_{zz} \geq \lambda_{\max}(G^{(i)})$ and $\gamma_{zz}^2 \geq \|\mathbb{E}_X[(G^{(i)})^2]\|_2$;
- $C_{yz} \geq \max_{k,k'} |V_{kk'}^{(i)}|$ and $\gamma_{yz}^2 \geq \max_{k,k'} \mathbb{E}_X[(V_{kk'}^{(i)})^2])$;
- $C_y \geq |c_y^{(i)}|$ and $\gamma_y^2 \geq \mathbb{E}_X[(c_y^{(i)})^2]$.

Theorem 2.9. *Suppose that Assumption* 2.8 *holds. Assume that $S \setminus S_{\alpha m}$ is generated by i.i.d. draws according to the true distribution $\mathbb{P}_{(X,Y)}$. Assume further that $S \setminus S_{\alpha m}$ is independent of $S_{\alpha m}$. Let $\hat{f}_{\mathcal{A},S}$ be the model learned by the two-phase training procedure with S. Then, for each $w_\sigma := w^{S_{\alpha m}}$, for any $\delta > 0$, with probability at least $1 - \delta$,*

$$\mathcal{R}[\hat{f}_{\mathcal{A},S}] - \mathcal{R}_{S\setminus S_{\alpha m}}[\hat{f}_{\mathcal{A},S}] \leq \beta_1 \sum_{k=1}^{s} \left\|\bar{w}_k^S\right\|_1 + 2\beta_2 \sum_{k=1}^{s} \left\|\bar{w}_k^S\right\|_2^2 + \beta_3,$$

where

$$\beta_1 = \frac{2C_{zz}}{3m_\sigma} \ln \frac{3d_z}{\delta} + \sqrt{\frac{2\gamma_{zz}^2}{m_\sigma} \ln \frac{3d_z}{\delta}},$$

$$\beta_2 = \frac{2C_{yz}}{3m_\sigma} \ln \frac{6d_y d_z}{\delta} + \sqrt{\frac{\gamma_{yz}^2}{m_\sigma} \ln \frac{6d_y d_z}{\delta}},$$

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

and

$$\beta_2 = \frac{2C_y}{3m_\sigma} \ln \frac{3}{\delta} + \sqrt{\frac{2\gamma_y^2}{m_\sigma} \ln \frac{3}{\delta}}.$$

Our proof does *not* require independence of the coordinates of $\tilde{z}_i$ from the entries of the random matrices $\tilde{z}_i \tilde{z}_i^\top$ (see the proof of Theorem 2.9).

The bound in Theorem 2.9 is data dependent because the norms of the weights $\bar{w}_k^S$ depend on each particular S. As with Theorem 2.7, the bound in Theorem 2.9 does not contain a predetermined bound on the norms of weights and can be independent of the choice of hypothesis space, if desired; i.e., Assumption 2.8 can be also satisfied without referencing a hypothesis space of w, because $\tilde{z} = [\bar{X}^{(i)} \circ \bar{\sigma}(X^{(i)}, w_\sigma)]$ with $\bar{\sigma}_j(X^{(i)}, w_\sigma) \in \{0, 1\}$. However, unlike Theorem 2.7, Theorem 2.9 *implicitly* contains the properties of datasets different from a given S, via the predefined bounds in Assumption 2.8. This is expected since Theorem 2.9 makes claims about the set of random datasets S rather than about each instantiated S. Therefore, while Theorem 2.9 presents a strongly data-dependent bound (over random datasets), Theorem 2.7 is tighter for each given S; indeed, the main equality of Theorem 2.7 is as tight as possible.

Theorems 2.7 and 2.9 provide generalization bounds for practical deep learning models that do not necessarily have an explicit dependence on the number of weights or an exponential dependence on depth or the effective input dimensionality. Although the size of the vector $\bar{w}_k^S$ can be exponentially large in the depth of the network, the norms of the vector need not be. Because $\tilde{z}_k^{[L]}(X, w^S) = \|\bar{X} \circ \bar{\sigma}(X, w_\sigma)\|_2 \|\bar{w}_k^S\|_2 \cos\theta$, we have that $\|\bar{w}_k^S\|_2 = z_k^{[L]}(X, w)/(\|\bar{X} \circ \bar{\sigma}(X, w_\sigma)\|_2 \cos\theta)$ (unless the denominator is zero), where θ is the angle between $\bar{X} \circ \bar{\sigma}(X, w_\sigma)$ and $\bar{w}_k^S$. Additionally, as discussed in §2.5.2, $\sum_k \|\bar{w}_k\|_2^2 \leq \prod_{l=1}^L \|W^{(l)}\|_F^2$.

2.5.4 Probabilistic Bound for 0–1 Loss with Multi-Labels

For a 0–1 loss with multi-labels, the two-phase training procedure in §2.5.3 yields the generalization bound in Theorem 2.10. Similarly to the bounds in Theorems 2.7 and 2.9, the generalization bound in Theorem 2.10 does not necessarily have a dependence on the number of weights, or an exponential dependence on depth, or effective input dimensionality.

Theorem 2.10. *Assume that $S \setminus S_{\alpha m}$ is generated by i.i.d. draws according to a true distribution $\mathbb{P}_{(X,Y)}$. Assume also that $S \setminus S_{\alpha m}$ is independent of $S_{\alpha m}$. Fix $\rho > 0$ and w_σ. Let $\mathcal{F}$ be the set of models with the two-phase training procedure. Suppose that $\mathbb{E}_X[\|\bar{X} \circ \bar{\sigma}(X, w_\sigma)\|_2^2] \leq C_\sigma^2$ and $\max_k \|\bar{w}_k\|_2 \leq C_w$ for all $f \in \mathcal{F}$. Then, for*

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

any $\delta > 0$, with probability at least $1 - \delta$, the following holds for all $f \in \mathcal{F}$:

$$\mathcal{R}[f] \leq \mathcal{R}^{(\rho)}_{S \setminus S_{\alpha m}}[f] + \frac{2d_y^2(1-\alpha)^{-1/2}C_\sigma C_w}{\rho\sqrt{m_\sigma}} + \sqrt{\frac{\ln\frac{1}{\delta}}{2m_\sigma}}.$$

Here, the empirical margin loss $\mathcal{R}^{(\rho)}_S[f]$ is defined as

$$\mathcal{R}^{(\rho)}_S[f] = \frac{1}{m}\sum_{i=1}^{m} \mathcal{L}_{\text{margin},\rho}(f(X^{(i)}), Y^{(i)}),$$

where $\mathcal{L}_{\text{margin},\rho}$ is defined as follows:

$$\mathcal{L}_{\text{margin},\rho}(f(X), Y) = \mathcal{L}^{(2)}_{\text{margin},\rho}(\mathcal{L}^{(1)}_{\text{margin},\rho}(f(X), Y)),$$

with

$$\mathcal{L}^{(1)}_{\text{margin},\rho}(f(X), Y) = z_y^{[L]}(X) - \max_{y \neq y'} z_{y'}^{[L]}(X) \in \mathbb{R}$$

and

$$\mathcal{L}^{(2)}_{\text{margin},\rho}(t) = \begin{cases} 0 & \text{if } \rho \leq t, \\ 1 - t/\rho & \text{if } 0 \leq t \leq \rho, \\ 1 & \text{if } t \leq 0. \end{cases}$$

2.6 Discussions and Open Problems

It is very difficult to make a detailed characterization of how well a specific hypothesis generated by a certain learning algorithm will generalize in the absence of detailed information about the given problem instance. Traditional learning theory addresses this very difficult question and has developed bounds that are as tight as possible given the generic information available. In this chapter, we have worked toward drawing stronger conclusions by developing theoretical analyses tailored for the situations with more detailed information, including actual neural network structures, and actual performance on a validation set.

Optimization and generalization in deep learning are closely related via the following observation: if we make optimization easier by changing the model architecture, the generalization performance can be degraded, and vice versa. Hence, the non-pessimistic generalization theory discussed in this chapter might allow more architectural choices and assumptions in optimization theory.

We now note an additional important open problem designed to address the gap between learning theory and practice: for example, theoretically motivated algorithms can degrade actual performances when compared with heuristics. Define

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

the partial order of problem instances $(\mathbb{P}, S, f)$ as

$$(\mathbb{P}, S, f) \leq (\mathbb{P}', S', f') \quad \Leftrightarrow \quad \mathcal{R}_{\mathbb{P}}[f] - \mathcal{R}_S[f] \leq \mathcal{R}_{\mathbb{P}'}[f'] - \mathcal{R}_{S'}[f'],$$

where $\mathcal{R}_{\mathbb{P}}[f]$ is the expected risk with probability measure $\mathbb{P}$. Then, any theoretical insights without partial order preservation can be misleading as they can change the preference ranking of $(\mathbb{P}, S, f)$. For example, theoretically motivated algorithms can be worse than heuristics, if the theory does not preserve the partial order of $(\mathbb{P}, S, f)$. This observation suggests the following open problem.

Open Problem 3. Tightly characterize the expected risk $\mathcal{R}[f]$ or the generalization gap $\mathcal{R}[f] - \mathcal{R}_S[f]$ of a hypothesis f together with a pair $(\mathbb{P}, S)$, producing theoretical insights while partially yet provably preserving the partial order of $(\mathbb{P}, S, f)$.

Theorem 2.7 addresses Open Problem 3 by preserving the exact ordering via equality without bounds, *while producing the same and more tight practical insights* (e.g., regularizing the norm) when compared with existing bound-based theory. However, it would be beneficial also to consider a weaker notion of order preservation in order to gain analyzability with more useful insights, as stated in Open Problem 3.

Our discussion of Proposition 2.5 and Remark 2.6 suggests another open problem: analyzing the role and influence of *human intelligence* on generalization. For example, human intelligence often seems to be able to find good architectures (and other hyper-parameters) that get low validation errors (without non-exponentially large $|\mathcal{F}_{\text{val}}|$ as in Proposition 2.5, or a low complexity of $\mathcal{F}_{\text{val}}$ as in Remark 2.6). A close look at the deep learning literature seems to suggest that this question is fundamentally related to the progress of science and engineering, because many successful architectures have been designed based on the physical properties and engineering priors of the problems at hand (e.g., their hierarchical nature, convolution, and architecture for motion, such as that considered by Finn et al., 2016, memory networks, and so on). While this further open problem is a hard question, understanding it would be beneficial for further automating the role of human intelligence towards the goal of artificial intelligence.

Acknowledgements

We gratefully acknowledge support from NSF grants 1420316, 1523767, and 1723381, from AFOSR FA9550-17-1-0165, from ONR grant N00014-14-1-0486, and from ARO grant W911 NF1410433, as well as support from the NSERC, CIFAR, and Canada Research Chairs. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of our sponsors.

Appendix A Additional Discussions

This appendix contains additional results and discussions.

A1 Simple Regularization Algorithm

In general, theoretical bounds from statistical learning theory can be too loose to be directly used in practice. In addition, many theoretical results in statistical learning theory end up simply suggesting the regularization of some notion of smoothness of a hypothesis class. Indeed, by upper-bounding a distance between two functions (e.g., a hypothesis and the ground-truth function corresponding to the expected true labels), one can immediately see *without statistical learning theory* that regularizing the smoothness of the hypothesis class helps guarantees on generalization. Then, by an Occam's razor argument, one might prefer a simpler (yet still rigorous) theory and a corresponding simpler algorithm.

Accordingly, this subsection examines another simple regularization algorithm that directly regularizes the smoothness of the learned hypothesis. In this subsection we focus on multi-class classification problems with d_y classes, such as object classification with images. Accordingly, we analyze the expected risk with 0–1 loss as $\mathcal{R}[f] = \mathbb{E}_X[1\{f(X) = Y\}]$, where $f(X) = \text{argmax}_{k\in\{1,\ldots,d_y\}}(z_k^{[L]}(X))$ is the model prediction and $Y \in \{1,\ldots,d_y\}$ is the true label of X.

This subsection proposes the following family of simple regularization algorithms: given any architecture and method, add a new regularization term for each training mini-batch as

$$\text{loss} = \text{original loss} + \frac{\lambda}{\bar{m}}\left|\max_k \sum_{i=1}^{\bar{m}} \xi_i z_k^{[L]}(X^{(i)})\right|,$$

where $X^{(i)}$ is drawn from some distribution approximating the true distribution of X, $\xi_1,\ldots,\xi_{\bar{m}}$ are independently and uniformly drawn from $\{-1,1\}$, $\bar{m}$ is the mini-batch size, and λ is a hyper-parameter. Importantly, the distribution approximating the true distribution of X is used only for regularization purposes and hence need not be precisely accurate (as long as it plays its role in regularization). For example, the true distribution can be approximated by populations generated by a generative neural network and/or an extra data augmentation process. For simplicity, we refer to this family of methods as directly approximately regularizing complexity (DARC).

In this chapter, as a first step, we evaluated only a very simple version of the proposed family of methods. That is, our experiments employed the following

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

Table A.2 *Test error (%)*

Method	MNIST	CIFAR-10
Baseline	0.26	3.52
DARC1	0.20	3.43

Table A.3 *Test error ratio (DARC1/Base)*

	MNIST (ND)		MNIST		CIFAR-10	
	mean	stdv	mean	stdv	mean	stdv
Ratio	0.89	0.61	0.95	0.67	0.97	0.79

Table A.4 *Values of* $\frac{1}{m}\left(\max_k \sum_{i=1}^{m} |z_k^{[L]}(X^{(i)})|\right)$

Method	MNIST (ND)		MNIST		CIFAR-10	
	mean	stdv	mean	stdv	mean	stdv
Base	17.2	2.40	8.85	0.60	12.2	0.32
DARC1	1.30	0.07	1.35	0.02	0.96	0.01

simple and easy-to-implement method, called DARC1:

$$\text{loss} = \text{original loss} + \frac{\lambda}{\bar{m}}\left(\max_k \sum_{i=1}^{\bar{m}} |z_k^{[L]}(X^{(i)})|\right), \tag{A.1}$$

where $X^{(i)}$ is the ith sample in the training mini-batch. The additional computational cost and programming effort due to this new regularization is almost negligible because $z_k^{[L]}(X^{(i)})$ is already used in computing the original loss. This simplest version was derived by approximating the true distribution of X by the empirical distribution of the training data.

We evaluated the proposed method (DARC1) by simply adding the new regularization term in (A.1) to the existing standard codes for MNIST and CIFAR-10. Standard variants of LeNet (LeCun et al., 1998) and ResNeXt-29(16×64d) (Xie et al., 2017) were used for MNIST and CIFAR-10, and compared with the addition of the studied regularizer. For all the experiments, we fixed $(\lambda/\bar{m}) = 0.001$ with $\bar{m} = 64$. We used a single model without ensemble methods. The experimental details are given in Appendix B. The source code is available at

http://lis.csail.mit.edu/code/gdl.html

Table A.2 shows the error rates in comparison with previous results. To the best of our knowledge, the previous state-of-the-art classification error is 0.23% for MNIST with a single model (Sato et al., 2015) (and 0.21% with an ensemble,

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

by Wan et al., 2013). To further investigate the improvement, we ran 10 random trials with computationally less expensive settings, to gather the mean and standard deviation (stdv). For MNIST, we used fewer epochs with the same model. For CIFAR-10, we used a smaller model class (pre-activation ResNet with only 18 layers). Table A.3 summarizes the improvement ratio, i.e., the new model's error divided by the base model's error. We observed improvements in all cases. The test errors (standard deviations) of the base models were 0.53 (0.029) for MNIST (ND), 0.28 (0.024) for MNIST, and 7.11 (0.17) for CIFAR-10 (all in %).

Table A.4 contains the values of the regularization term $\frac{1}{m}(\max_k \sum_{i=1}^m |z_k^{[L]}(X^{(i)})|)$ for each obtained model. The models learned with the proposed method were significantly different from the base models in terms of this value. Interestingly, a comparison of the base cases for MNIST (ND) and MNIST shows that data augmentation by itself *implicitly* regularized what we explicitly regularized in the proposed method.

A2 Relationship to Other Fields

The situation where theoretical studies focus on a set of problems and practical applications are concerned with each element in a set is prevalent in the machine learning and computer science literature, and is not limited to the field of learning theory. For example, for each practical problem instance $q \in Q$, the size of the set Q that had been analyzed in theory for optimal exploration in Markov decision processes (MDPs) was demonstrated to be frequently too pessimistic, and a methodology to partially mitigate the issue was proposed (Kawaguchi, 2016a). Bayesian optimization would suffer from a pessimistic set Q regarding each problem instance $q \in Q$, the issue of which had been partially mitigated (Kawaguchi et al., 2015).

Moreover, characterizing a set of problems Q only via a worst-case instance $q' \in Q$ (i.e., worst-case analysis) is known to have several issues in theoretical computer science, and so-called *beyond worst-case analysis* (e.g., smoothed analysis) is an active area of research to mitigate these issues.

A3 SGD Chooses Direction in Terms of $\bar{w}$

Recall that

$$z_k^{[L]}(X, w) = z^\top \bar{w} = [\bar{X} \circ \bar{\sigma}(X, w)]^\top \bar{w}.$$

Note that $\sigma(X, w)$ is 0 or 1 for max pooling and/or ReLU nonlinearity. Thus, the derivative of $z = [\bar{X} \circ \bar{\sigma}(X, w)]$ with respect to w is zero everywhere (except at the measure-zero set, where the derivative does not exists). Thus, by the chain rule (and power rule), the gradient of the loss with respect to w contains only a contribution

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

from the derivative of $z_k^{[L]}$ with respect to $\bar{w}$, but not from that with respect to w in z.

A4 Simple Implementation of Two-Phase Training Procedure

Directly implementing Equation (2.3) requires a summation over all paths, which can be computationally expensive. To avoid thi, we implemented it by creating two deep neural networks, one of which defines $\bar{w}$ paths hierarchically, the other defining w_σ paths hierarchically, resulting in a computational cost at most (approximately) twice as much as the original cost of training standard deep learning models. We tied w_σ and $\bar{w}$ in the two networks during the standard phase, and untied them during the freeze phase.

Our source code is available at http://lis.csail.mit.edu/code/gdl.html

The computation of the standard network without skip connection can be re-written as

$$\begin{aligned} z^{[l]}(X,w) &= \sigma^{[l]}(W^{[l]}z^{[l-1]}(X,w)) \\ &= \dot{\sigma}^{[l]}(W^{[l]}z^{[l-1]}(X,w)) \circ W^{[l]}z^{[l-1]}(X,w) \\ &= \dot{\sigma}^{[l]}(W_\sigma^{[l]}z_\sigma^{[l-1]}(X,w)) \circ W^{[l]}z^{[l-1]}(X,w), \end{aligned}$$

where

$$W_\sigma^{[l]} := W^{[l]}, \qquad z_\sigma^{[l-1]} := \sigma(W_\sigma^{[l]}z_\sigma^{[l-1]}(X,w))$$

and

$$\dot{\sigma}_j^{[l]}(W^{[l]}z^{[l-1]}(X,w)) = \begin{cases} 1 & \text{if the } j\text{th unit at the } l\text{th layer is active,} \\ 0 & \text{otherwise.} \end{cases}$$

Note that because $W_\sigma^{[l]} = W^{[l]}$, we have that $z_\sigma^{[l-1]} = z^{[l]}$ in the standard phase and standard models.

In the two-phase training procedure, we created two networks for $W_\sigma^{[l]}z_\sigma^{[l-1]}(X,w)$ and $W^{[l]}z^{[l-1]}(X,w)$ separately. We then set $W_\sigma^{[l]} = W^{[l]}$ during the standard phase, and froze $W_\sigma^{[l]}$ and just trained $W^{[l]}$ during the freeze phase. By following the same derivation of Equation (2.1), we can see that this defines the desired computation without explicitly computing the summation over all paths. By the same token, this applies to DAGs.

A5 On Proposition 2.3

Proposition 2.3 is a direct consequence of Remark 2.4. Consider statement (i). Given such an $\mathcal{F}$, consider any tuples $(\mathcal{A}, \mathcal{F}, S)$ such that $\mathcal{A}$ takes $\mathcal{F}$ and S as input

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

and outputs f^*_ϵ. Clearly, there are many such tuples $(\mathcal{A}, \mathcal{F}, S)$ because of Remark 2.4. This establishes statement (i).

Consider statement (ii). Given any dataset S', consider any algorithm $\mathcal{A}'$ that happens to output f^*_ϵ if $S = S'$ and outputs any f' otherwise, such that f' is arbitrarily non-robust and $|\mathcal{L}(f^*_\epsilon(X), Y) - \mathcal{L}(f'(X), Y)|$ is arbitrarily large (i.e., arbitrarily non-stable). This proves statement (ii). Note that although this particular A' suffices to prove statement (ii), there are clearly many other tuples $(\mathcal{A}, \mathcal{F}, \mathbb{P}_{(X,Y)}, S)$ that could be used to prove statement (ii) because of Remark 2.4.

A6 On Extensions

Theorem 2.7 addresses Open Problem 2 with limited applicability, i.e., to certain neural networks with squared loss. In contrast, a parallel study (Kawaguchi et al., 2018) presented a novel generic learning theory to address Open Problem 2 for general cases in machine learning. It would be beneficial to explore both a generic analysis (Kawaguchi et al., 2018) and a concrete analysis in deep learning to get theoretical insights that are tailored for each particular case.

For previous bounds with a hypothesis space $\mathcal{F}$, if we try different such spaces then $\mathcal{F}$, depending on S, the basic proof breaks down. An easy recovery at the cost of an extra quantity in a bound is achieved by taking a union bound over all possible $\mathcal{F}_j$ for $j = 1, 2, \ldots$, where we pre-decide $(\mathcal{F}_j)_j$ without dependence on S (because simply considering the "largest" $\mathcal{F} \supseteq \mathcal{F}_j$ can result in a very loose bound for each $\mathcal{F}_j$). Then, after training with a dataset S, if we have $\hat{f}_{\mathcal{A},S} \in \mathcal{F}_j$, we can use the complexity of $\mathcal{F}_j$ without using the complexity of other $\mathcal{F}_i$ with $i \neq j$. Because the choice of $\mathcal{F}_j$ (out of $(\mathcal{F}_j)_j$) depends on S, it is called a data-dependent bound and indeed this is the idea behind data-dependent bounds in statistical learning theory.

Similarly, if we need to try many $w_\sigma := w^{S_{\alpha m}}$ depending on the whole of S in Theorem 2.9, we can take a union bound over $w^{(j)}_\sigma$ for $j = 1, 2, \ldots$, where we pre-determine $\{w^{(j)}_\sigma\}_j$ without dependence on $S \setminus S_{\alpha m}$ but with dependence on $S_{\alpha m}$. We can do the same with Proposition 2.5 and Remark 2.6 to use many different $\mathcal{F}_{\text{val}}$ depending on the validation dataset $S^{(\text{val})}$ with a predefined sequence.

Appendix B Experimental Details

For MNIST: We used the following fixed architecture:

(i) Convolutional layer with 32 filters with filter size 5 by 5, followed by max pooling of size 2 by 2 and ReLU.
(ii) Convolution layer with 32 filters with filter size 5 by 5, followed by max pooling of size 2 by 2 and ReLU.

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

(iii) Fully connected layer with output 1024 units, followed by ReLU and Dropout with its probability set to 0.5.

(iv) Fully connected layer with output 10 units.

Layer (iv) outputs $z^{[L]}$ in our notation. For training purpose, we used the softmax of $z^{[L]}$. Also, $f(X) = \text{argmax}(z^{[L]}(X))$ was taken as the label prediction.

We fixed the learning rate to be 0.01, the momentum coefficient to be 0.5, and the optimization algorithm to be the standard stochastic gradient descent (SGD). We fixed the data augmentation process as: a random crop with size 24, a random rotation up to ±15 degrees, and scaling of 15%. We used 3000 epochs for Table A.2, and 1000 epochs for Tables A.3 and A.4.

For CIFAR-10: For data augmentation, we used a random horizontal flip with probability 0.5 and a random crop of size 32 with padding of size 4.

For Table A.2, we used ResNeXt-29(16×64d) (Xie et al., 2017). We set the initial learning rate to be 0.05, decreasing it to 0.005 at 150 epochs and to 0.0005 at 250 epochs. We fixed the momentum coefficient to be 0.9, the weight decay coefficient to be 5×10^{-4}, and the optimization algorithm to be stochastic gradient descent (SGD) with Nesterov momentum. We stopped the training at 300 epochs.

For Tables A.3 and A.4, we used pre-activation ResNet with only 18 layers (pre-activation ResNet-18) (He et al., 2016). We fixed learning rate to be 0.001 and momentum coefficient to be 0.9, and optimization algorithm to be (standard) stochastic gradient descent (SGD). We used 1000 epochs.

Appendix C Proofs

We will use the following lemma in the proof of Theorem 2.7.

Lemma 2.11. (Matrix Bernstein inequality: corollary to Theorem 1.4 in Tropp, 2012) *Consider a finite sequence $\{M_i\}$ of independent, random, self-adjoint matrices with dimension d. Assume that each random matrix satisfies that $\mathbb{E}[M_i] = 0$ and $\lambda_{\max}(M_i) \leq R$ almost surely. Let $\gamma^2 = \| \sum_i \mathbb{E}[M_i^2] \|_2$. Then, for any $\delta > 0$, with probability at least $1 - \delta$,*

$$\lambda_{\max}\left(\sum_i M_i\right) \leq \frac{2R}{3} \ln \frac{d}{\delta} + \sqrt{2\gamma^2 \ln \frac{d}{\delta}}.$$

Proof Theorem 1.4 of Tropp (2012) states that, for all $t \geq 0$,

$$\mathbb{P}\left[\lambda_{\max}\left(\sum_i M_i\right) \geq t\right] \leq d \cdot \exp\left(\frac{-t^2/2}{\gamma^2 + Rt/3}\right).$$

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

Setting $\delta = d \exp\left(-\frac{t^2/2}{\gamma^2 + Rt/3}\right)$ implies

$$-t^2 + \frac{2}{3}R(\ln d/\delta)t + 2\gamma^2 \ln d/\delta = 0.$$

Solving for t with the quadratic formula and bounding the solution using the sub-additivity of square roots of non-negative terms (i.e., $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for all $a, b \geq 0$), gives

$$t \leq \frac{2}{3}R(\ln d/\delta) + 2\gamma^2 \ln d/\delta. \qquad \square$$

C1 Proof of Theorem 2.1

Proof For any matrix M, let $\mathrm{Col}(M)$ and $\mathrm{Null}(M)$ be the column space and null space of M. Since $\mathrm{rank}(\Phi) \geq m$ and $\Phi \in \mathbb{R}^{m \times n}$, the set of its columns spans $\mathbb{R}^m$, which proves statement (i). Let $w^* = w_1^* + w_2^*$ where $\mathrm{Col}(w_1^*) \subseteq \mathrm{Col}(M^T)$ and $\mathrm{Col}(w_2^*) \subseteq \mathrm{Null}(M)$. For statement (ii), set the parameter as $w := w_1^* + \epsilon C_1 + \alpha C_2$ where $\mathrm{Col}(C_1) \subseteq \mathrm{Col}(M^T)$, $\mathrm{Col}(C_2) \subseteq \mathrm{Null}(M)$, $\alpha \geq 0$, and $C_2 = \frac{1}{\alpha} w_2^* + \bar{C}_2$. Since $\mathrm{rank}(M) < n$, $\mathrm{Null}(M) \neq \{0\}$ and there exist non-zero $\bar{C}_2$. Then

$$\hat{\mathbf{Y}}(w) = \mathbf{Y} + \epsilon \Phi C_1$$

and

$$\hat{\mathbf{Y}}_{\text{test}}(w) = \mathbf{Y}_{\text{test}} + \epsilon \Phi_{\text{test}} C_1.$$

Setting $A = \Phi C_1$ and $B = \Phi_{\text{test}} C_1$ with a proper normalization of C_1 yields (ii)(a) and (ii)(b) in statement (ii) (note that C_1 has an arbitrary freedom in the bound on its scale because the only condition on it is $\mathrm{Col}(C_1) \subseteq \mathrm{Col}(M^T)$). At the same time, with the same parameter, since $\mathrm{Col}(w_1^* + \epsilon C_1) \perp \mathrm{Col}(C_2)$ we have

$$\|w\|_F^2 = \|w_1^* + \epsilon C_1\|_F^2 + \alpha^2 \|C_2\|_F^2$$

and

$$\|w - w^*\|_F^2 = \|\epsilon C_1\|_F^2 + \alpha^2 \|\bar{C}_2\|_F^2,$$

which grows unboundedly as $\alpha \to \infty$ without changing A and B, proving (ii)(c) in statement (ii). $\square$

C2 Proof of Corollary 2.2

Proof This follows the fact that the proof in Theorem 2.1 uses the assumption of $n > m$ and $\mathrm{rank}(\Phi) \geq m$ only for statement (i). $\square$

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

C3 Proof of Theorem 2.7

Proof From Equation (2.1), the squared loss of the deep models for each point (X,Y) can be rewritten as

$$\sum_{k=1}^{s}(z^\top \bar{w}_k - Y_k)^2 = \sum_{k=1}^{s} \bar{w}_k^\top (zz^\top)\bar{w}_k - 2Y_k z^\top \bar{w}_k + Y_k^2.$$

Thus, from Equation (2.1) for a squared loss, we can decompose the generalization gap into three terms as

$$\begin{aligned}\mathcal{R}[w^S] - \mathcal{R}_S[w^S] = &\sum_{k=1}^{s}\left[(\bar{w}_k^s)^\top \left(\mathbb{E}[zz^\top] - \frac{1}{m}\sum_{i=1}^{m} z_i z_i^\top\right)\bar{w}_k^s\right] \\ &+ 2\sum_{k=1}^{s}\left[\left(\frac{1}{m}\sum_{i=1}^{m} Y_k^{(i)} z_i^\top - \mathbb{E}[Y_k z^\top]\right)\bar{w}_k^s\right] \\ &+ \left(\mathbb{E}[Y^\top Y] - \frac{1}{m}\sum_{i=1}^{m}(Y^{(i)})^\top Y^{(i)}\right).\end{aligned}$$

As G, defined before Theorem 2.7, is a real symmetric matrix, we can write an eigendecomposition of G as $G = U\Lambda U^\top$ where the diagonal matrix Λ contains eigenvalues $\Lambda_{jj} = \lambda_j$ with corresponding orthogonal eigenvector matrix U; u_j is the jth column of U. Then

$$(\bar{w}_k^s)^\top G \bar{w}_k^s = \sum_j \lambda_j (u_j^\top \bar{w}_k^s)^2 = \|\bar{w}_k^s\|_2^2 \sum_j \lambda_j \cos^2 \theta^{(1)}_{\bar{w}_k^s, j}$$

and

$$\sum_j \lambda_j (u_j^\top \bar{w}_k^s)^2 \le \lambda_{\max}(G)\|U^\top \bar{w}_k^s\|_2^2 = \lambda_{\max}(G)\|\bar{w}_k^s\|_2^2.$$

Also,

$$v^\top \bar{w}_k^s = \|v\|_2 \|\bar{w}_k^s\|_2 \cos\theta^{(2)}_{\bar{w}_k^s} \le \|v\|_2\|\bar{w}_k^s\|_2.$$

Using these expressions we obtain

$$\begin{aligned}&\mathcal{R}[w^S] - \mathcal{R}_S[w^S] - c_y \\ &= \sum_{k=1}^{s}\left(2\|v\|_2\|\bar{w}_k^s\|_2 \cos\theta^{(2)}_{\bar{w}_k^s} + \|\bar{w}_k^s\|_2^2 \sum_j \lambda_j \cos^2\theta^{(1)}_{\bar{w}_k^s, j}\right) \\ &\le \sum_{k=1}^{s}\left(2\|v\|_2\|\bar{w}_k^s\|_2 + \lambda_{\max}(G)\|\bar{w}_k^s\|_2^2\right)\end{aligned}$$

as required. □

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

C4 Proof of Theorem 2.9

Proof We do *not* require independence of the coordinates of $\tilde{z}_i$ and the entries of the random matrices $\tilde{z}_i\tilde{z}_i^\top$ because of the definition of independence required for the matrix Bernstein inequality (for $\frac{1}{m_\sigma}\sum_{i=1}^{m_\sigma}\tilde{z}_i\tilde{z}_i^\top$); see, e.g., Section 2.2.3 of Tropp *et al.*, 2015) and because of the union bound over the coordinates (for $\frac{1}{m_\sigma}\sum_{i=1}^{m_\sigma}Y_k^{(i)}\tilde{z}_i$). We use the fact that $\tilde{z}_{\alpha m+1},\dots,\tilde{z}_m$ are independent random variables *over the sample index* (although dependent on the coordinates), because each $w_\sigma := w^{S_{\alpha m}}$ is fixed and independent of $X^{(\alpha m+1)},\dots,X^{(m)}$.

From Equation (2.2), with the definition of induced matrix norm and the Cauchy–Schwarz inequality,

$$\begin{aligned}
&\mathcal{R}[\hat{f}_{\mathcal{A},S}] - \mathcal{R}_{S\setminus S_{\alpha m}}[\hat{f}_{\mathcal{A},S}] \\
&\quad\le \sum_{k=1}^{s}\left\|\bar{w}_k^S\right\|_2^2\,\lambda_{\max}\left(\mathbb{E}[\tilde{z}\tilde{z}^\top] - \frac{1}{m_\sigma}\sum_{i=\alpha m+1}^{m}\tilde{z}_i\tilde{z}_i^\top\right) \\
&\qquad + 2\sum_{k=1}^{s}\|\bar{w}_k^S\|_1\left\|\frac{1}{m_\sigma}\sum_{i=\alpha m+1}^{m}Y_k^{(i)}\tilde{z}_i - \mathbb{E}[Y_k\tilde{z}]\right\|_\infty \\
&\qquad + \left(\mathbb{E}[Y^\top Y] - \frac{1}{m_\sigma}\sum_{i=\alpha m+1}^{m}(Y^{(i)})^\top Y^{(i)}\right). \qquad \text{(C.1)}
\end{aligned}$$

In what follows, we bound each term on the right-hand side with concentration inequalities.

For the first term: The matrix Bernstein inequality (Lemma 2.11) states that for any $\delta > 0$, with probability at least $1 - \delta/3$,

$$\lambda_{\max}\left(\mathbb{E}[\tilde{z}\tilde{z}^\top] - \frac{1}{m_\sigma}\sum_{i=\alpha m+1}^{m}\tilde{z}_i\tilde{z}_i^\top\right) \le \frac{2C_{zz}}{3m_\sigma}\ln\frac{3d_z}{\delta} + \sqrt{\frac{2\gamma_{zz}^2}{m_\sigma}\ln\frac{3d_z}{\delta}}.$$

Here, the matrix Bernstein inequality was applied as follows. Let $M_i = (\frac{1}{m_\sigma}G^{(i)})$. Then $\sum_{i=\alpha m+1}^{m}M_i = \mathbb{E}[\tilde{z}\tilde{z}^\top] - \frac{1}{m_\sigma}\sum_{i=\alpha m+1}^{m}\tilde{z}_i\tilde{z}_i^\top$. We have that $\mathbb{E}[M_i] = 0$ for all i. Also, $\lambda_{\max}(M_i) \le \frac{1}{m_\sigma}C_{zz}$ and $\|\sum_i \mathbb{E}[M_i^2]\|_2 \le \frac{1}{m_\sigma}\gamma_{zz}^2$.

For the second term: To each $(k,k') \in \{1,\dots,s\}\times\{1,\dots,d_z\}$ we apply the matrix Bernstein inequality and take the union bound over $d_y d_z$ events, obtaining that for any $\delta > 0$, with probability at least $1 - \delta/3$, for all $k \in \{1,2,\dots,s\}$,

$$\left\|\frac{1}{m_\sigma}\sum_{i=\alpha m+1}^{m}Y_k^{(i)}\tilde{z}_i - \mathbb{E}[Y_k\tilde{z}]\right\|_\infty \le \frac{2C_{yz}}{3m_\sigma}\ln\frac{6d_yd_z}{\delta} + \sqrt{\frac{\gamma_{yz}^2}{m_\sigma}\ln\frac{6d_yd_z}{\delta}}.$$

For the third term: From the matrix Bernstein inequality, with probability at least

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

$1 - \delta/3$,

$$\mathbb{E}[Y^\top Y] - \frac{1}{m_\sigma} \sum_{i=\alpha m+1}^{m} (Y^{(i)})^\top Y^{(i)} \leq \frac{2C_y}{3m} \ln \frac{3}{\delta} + \sqrt{\frac{2\gamma_y^2}{m} \ln \frac{3}{\delta}}.$$

Putting it all together: Pulling all this together, for a fixed (or frozen) w_σ, with probability (over $S \setminus S_{\alpha m} = \{(X^{(\alpha m+1)}, Y^{(\alpha m+1)}), \ldots, (X^{(m)}, Y^{(m)})\}$) at least $1 - \delta$, we have that

$$\lambda_{\max} \left(\mathbb{E}[\tilde{z}\tilde{z}^\top] - \frac{1}{m_\sigma} \sum_{i=\alpha m+1}^{m} \tilde{z}_i \tilde{z}_i^\top \right) \leq \beta_1,$$

$$\left\| \frac{1}{m_\sigma} \sum_{i=\alpha m+1}^{m} Y_k^{(i)} \tilde{z}_i - \mathbb{E}[Y_k \tilde{z}] \right\|_\infty \leq \beta_2 \quad \text{(for all } k\text{)},$$

and

$$\mathbb{E}[Y^\top Y] - \frac{1}{m_\sigma} \sum_{i=\alpha m+1}^{m} (Y^{(i)})^\top Y^{(i)} \leq \beta_3.$$

Since Equation (C.1) always hold deterministically (with or without such a dataset), the desired statement of this theorem follows. □

C5 Proof of Theorem 2.10

Proof Define S_{m_σ} as

$$S_{m_\sigma} = S \setminus S_{\alpha m} = \{(X^{(\alpha m+1)}, Y^{(\alpha m+1)}), \ldots, (X^{(m)}, Y^{(m)})\}.$$

Recall the following fact: using the result of Koltchinskii and Panchenko (2002), we have that for any $\delta > 0$, with probability at least $1 - \delta$, the following holds for all $f \in \mathcal{F}$:

$$\mathcal{R}[f] \leq \mathcal{R}_{S_{m_\sigma}, \rho}[f] + \frac{2d_y^2}{\rho m_\sigma} \mathfrak{R}'_{m_\sigma}(\mathcal{F}) + \sqrt{\frac{\ln \frac{1}{\delta}}{2m_\sigma}},$$

where $\mathfrak{R}'_{m_\sigma}(\mathcal{F})$ is the Rademacher complexity, defined as

$$\mathfrak{R}'_{m_\sigma}(\mathcal{F}) = \mathbb{E}_{S_{m_\sigma}, \xi} \left[\sup_{k,w} \sum_{i=1}^{m_\sigma} \xi_i z_k^{[L]}(X^{(i)}, w) \right].$$

Here, ξ_i is the Rademacher variable, and the supremum is taken over all $k \in \{1, \ldots, s\}$ and all w allowed in $\mathcal{F}$. Then, for our parameterized hypothesis spaces,

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

with any frozen w_σ,

$$\begin{aligned}\mathfrak{R}'_{m_\sigma}(\mathcal{F}) &= \mathbb{E}_{S_{m_\sigma},\xi}\left[\sup_{k,\bar{w}_k}\sum_{i=1}^{m_\sigma}\xi_i[\bar{X}^{(i)}\circ\bar{\sigma}(X^{(i)},w_\sigma)]^\top\bar{w}_k\right]\\ &\le \mathbb{E}_{S_{m_\sigma},\xi}\left[\sup_{k,\bar{w}_k}\left\|\sum_{i=1}^{m_\sigma}\xi_i[\bar{X}^{(i)}\circ\bar{\sigma}(X^{(i)},w_\sigma)]\right\|_2\|\bar{w}_k\|_2\right]\\ &\le C_w\mathbb{E}_{S_{m_\sigma},\xi}\left[\left\|\sum_{i=1}^{m_\sigma}\xi_i[\bar{X}^{(i)}\circ\bar{\sigma}(X^{(i)},w_\sigma)]\right\|_2\right].\end{aligned}$$

Because the square root is concave in its domain, by using Jensen's inequality and linearity of expectation, we obtain

$$\begin{aligned}&\mathbb{E}_{S_{m_\sigma},\xi}\left[\left\|\sum_{i=1}^{m_\sigma}\xi_i[\bar{X}^{(i)}\circ\bar{\sigma}(X^{(i)},w_\sigma)]\right\|_2\right]\\ &\le\left(\mathbb{E}_{S_{m_\sigma}}\sum_{i=1}^{m_\sigma}\sum_{j=1}^{m_\sigma}\mathbb{E}_\xi[\xi_i\xi_j][\bar{X}^{(i)}\circ\bar{\sigma}(X^{(i)},w_\sigma)]^\top[\bar{X}_j\circ\bar{\sigma}(X_j,w_\sigma)]\right)^{1/2}\\ &=\left(\sum_{i=1}^{m_\sigma}\mathbb{E}_{S_{m_\sigma}}\left[\left\|[\bar{X}^{(i)}\circ\bar{\sigma}(X^{(i)},w_\sigma)]\right\|_2^2\right]\right)^{1/2}\\ &\le C_\sigma\sqrt{m_\sigma}.\end{aligned}$$

Putting it all together, we find $\mathfrak{R}'_m(\mathcal{F}) \le C_\sigma C_w\sqrt{m_\sigma}$. □

C6 Proof of Proposition 2.5

Proof Consider a fixed $f \in \mathcal{F}_{\text{val}}$. Because $\mathcal{F}_{\text{val}}$ is independent of the validation dataset $S^{(\text{val})}$, it follows that $\kappa_{f,1},\ldots,\kappa_{f,m_{\text{val}}}$ are independent zero-mean random variables, given, as above, a fixed $f \in \mathcal{F}_{\text{val}}$ (note that we can make $\mathcal{F}_{\text{val}}$ dependent on S because we are considering $S^{(\text{val})}$ here, resulting in the requirement that $\mathcal{F}_{\text{val}}$ is independent of $S^{(\text{val})}$, instead of S). Thus, we can apply the matrix Bernstein inequality, yielding

$$\mathbb{P}\left(\frac{1}{m_{\text{val}}}\sum_{i=1}^{m_{\text{val}}}\kappa_{f,i} > \epsilon\right) \le \exp\left(-\frac{\epsilon^2 m_{\text{val}}/2}{\gamma^2+\epsilon C/3}\right).$$

By taking the union bound over all elements in $\mathcal{F}_{\text{val}}$, we find

$$\mathbb{P}\left(\bigcup_{f\in\mathcal{F}_{\text{val}}}\left\{\frac{1}{m_{\text{val}}}\sum_{i=1}^{m_{\text{val}}}\kappa_{f,i} > \epsilon\right\}\right) \le |\mathcal{F}_{\text{val}}|\exp\left(-\frac{\epsilon^2 m_{\text{val}}/2}{\gamma^2+\epsilon C/3}\right).$$

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

Setting $\delta = |\mathcal{F}_{\text{val}}| \exp\left(-\frac{\epsilon^2 m_{\text{val}}/2}{\gamma^2 + \epsilon C/3}\right)$ and solving for ϵ gives (via the quadratic formula),

$$\epsilon = \frac{2C \ln(\frac{|\mathcal{F}_{\text{val}}|}{\delta})}{6m_{\text{val}}} \pm \frac{1}{2}\sqrt{\left(\frac{2C \ln(\frac{|\mathcal{F}_{\text{val}}|}{\delta})}{3m_{\text{val}}}\right)^2 + \frac{8\gamma^2 \ln(\frac{|\mathcal{F}_{\text{val}}|}{\delta})}{m_{\text{val}}}}.$$

Noticing that the solution of ϵ with the minus sign results in $\epsilon < 0$, which is invalid for the matrix Bernstein inequality, we obtain the valid solution, the one with with the plus sign. Then we have

$$\epsilon \leq \frac{2C \ln(\frac{|\mathcal{F}_{\text{val}}|}{\delta})}{3m_{\text{val}}} + \sqrt{\frac{2\gamma^2 \ln(\frac{|\mathcal{F}_{\text{val}}|}{\delta})}{m_{\text{val}}}},$$

where we have used $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$. By taking the negation of the statement, we obtain that, for any $\delta > 0$, with probability at least $1 - \delta$, for all $f \in \mathcal{F}_{\text{val}}$,

$$\frac{1}{m_{\text{val}}} \sum_{i=1}^{m_{\text{val}}} \kappa_{f,i} \leq \frac{2C \ln(\frac{|\mathcal{F}_{\text{val}}|}{\delta})}{3m_{\text{val}}} + \sqrt{\frac{2\gamma^2 \ln(\frac{|\mathcal{F}_{\text{val}}|}{\delta})}{m}},$$

where

$$\frac{1}{m_{\text{val}}} \sum_{i=1}^{m_{\text{val}}} \kappa_{f,i} = \mathcal{R}[f] - \mathcal{R}_{S^{(\text{val})}}[f]. \qquad \square$$

References

Arpit, Devansh, Jastrzebski, Stanislaw, Ballas, Nicolas, Krueger, David, Bengio, Emmanuel, Kanwal, Maxinder S., Maharaj, Tegan, Fischer, Asja, Courville, Aaron, Bengio, Yoshua, et al. 2017. A Closer Look at Memorization in Deep Networks. In: *Proc. International Conference on Machine Learning*.

Banburski, Andrzej, Liao, Qianli, Miranda, Brando, Rosasco, Lorenzo, Liang, Bob, Hidary, Jack, and Poggio, Tomaso. 2019. Theory III: Dynamics and Generalization in Deep Networks. *Massachusetts Institute of Technology CBMM Memo No. 90*.

Barron, Andrew R. 1993. Universal Approximation Bounds for Superpositions of a Sigmoidal Function. *IEEE Transactions on Information theory*, **39**(3), 930–945.

Bartlett, Peter L., Boucheron, Stéphane, and Lugosi, Gábor. 2002. Model Selection and Error Estimation. *Machine Learning*, **48**(1), 85–113.

Bousquet, Olivier, and Elisseeff, André. 2002. Stability and Generalization. *Journal of Machine Learning Research*, **2**(Mar), 499–526.

https://doi.org/10.1017/9781009025096.003 Published online by Cambridge University Press

Choromanska, Anna, Henaff, Mikael, Mathieu, Michael, Ben Arous, Gerard, and LeCun, Yann. 2015. The Loss Surfaces of Multilayer Networks. Pages 192–204 of: *Proc. 18th International Conference on Artificial Intelligence and Statistics*.

Dinh, Laurent, Pascanu, Razvan, Bengio, Samy, and Bengio, Yoshua. 2017. Sharp Minima Can Generalize for Deep Nets. In: *International Conference on Machine Learning*.

Dziugaite, Gintare Karolina, and Roy, Daniel M. 2017. Computing Nonvacuous Generalization Bounds for Deep (Stochastic) Neural Networks with Many More Parameters than Training Data. In: *Proc. 33rd Conference on Uncertainty in Artificial Intelligence*.

Finn, Chelsea, Goodfellow, Ian, and Levine, Sergey. 2016. Unsupervised Learning for Physical Interaction through Video Prediction. Pages 64–72 of: *Advances in Neural Information Processing Systems*.

Gonen, Alon, and Shalev-Shwartz, Shai. 2017. Fast Rates for Empirical Risk Minimization of Strict Saddle Problems. Pages 1043–1063 of: *Proc. Conference on Learning Theory*.

Hardt, Moritz, Recht, Ben, and Singer, Yoram. 2016. Train Faster, Generalize Better: Stability of Stochastic Gradient Descent. Pages 1225–1234 of: *proc. International Conference on Machine Learning*.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. 2016. Identity Mappings in Deep Residual Networks. Pages 630–645 of: *Proc. European Conference on Computer Vision*. Springer.

Healy, Patrick, and Nikolov, Nikola S. 2001. How to Layer a Directed Acyclic Graph. Pages 16–30 of: *proc. International Symposium on Graph Drawing*. Springer.

Herbrich, Ralf, and Williamson, Robert C. 2002. Algorithmic Luckiness. *Journal of Machine Learning Research*, **3**, 175–212.

Hoffer, Elad, Hubara, Itay, and Soudry, Daniel. 2017. Train Longer, Generalize Better: Closing the Generalization Gap in Large Batch Training of Neural Networks. Pages 1731–1741 of: *Advances in Neural Information Processing Systems*.

Kawaguchi, Kenji. 2016a. Bounded Optimal Exploration in MDP. In: *Proc. 30th AAAI Conference on Artificial Intelligence*.

Kawaguchi, Kenji. 2016b. Deep Learning without Poor Local Minima. In: *Advances in Neural Information Processing Systems*.

Kawaguchi, Kenji, Bengio, Yoshua, Verma, Vikas, and Kaelbling, Leslie Pack. 2018. Generalization in Machine Learning via Analytical Learning Theory. Massachusetts Institute of Technology, Report MIT-CSAIL-TR-2018-019.

Kawaguchi, Kenji, Kaelbling, Leslie Pack, and Lozano-Pérez, Tomás. 2015. Bayesian Optimization with Exponential Convergence. In: *Advances in Neural Information Processing*.

Keskar, Nitish Shirish, Mudigere, Dheevatsa, Nocedal, Jorge, Smelyanskiy, Mikhail, and Tang, Ping Tak Peter. 2017. On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima. In: *Proc. International Conference on Learning Representations*.

Koltchinskii, Vladimir, and Panchenko, Dmitriy. 2000. Rademacher Processes and Bounding the Risk of Function Learning. Pages 443–457 of: *High Dimensional Probability II*. Springer.

Koltchinskii, Vladimir, and Panchenko, Dmitry. 2002. Empirical Margin Distributions and Bounding the Generalization Error of Combined Classifiers. *Annals of Statistics*, 1–50.

Krueger, David, Ballas, Nicolas, Jastrzebski, Stanislaw, Arpit, Devansh, Kanwal, Maxinder S., Maharaj, Tegan, Bengio, Emmanuel, Fischer, Asja, and Courville, Aaron. 2017. Deep Nets Don't Learn via Memorization. In: *proc. Workshop Track of International Conference on Learning Representations*.

Kuzborskij, Ilja, and Lampert, Christoph. 2017. Data-Dependent Stability of Stochastic Gradient Descent. ArXiv preprint arXiv:1703.01678.

LeCun, Yann, Bottou, Léon, Bengio, Yoshua, and Haffner, Patrick. 1998. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE*, **86**(11), 2278–2324.

Leshno, Moshe, Lin, Vladimir Ya., Pinkus, Allan, and Schocken, Shimon. 1993. Multilayer Feedforward Networks with a Nonpolynomial Activation Function can Approximate Any Function. *Neural Networks*, **6**(6), 861–867.

Livni, Roi, Shalev-Shwartz, Shai, and Shamir, Ohad. 2014. On the Computational Efficiency of Training Neural Networks. Pages 855–863 of: *Advances in Neural Information Processing Systems*.

Mohri, Mehryar, Rostamizadeh, Afshin, and Talwalkar, Ameet. 2012. *Foundations of Machine Learning*. MIT Press.

Montufar, Guido F., Pascanu, Razvan, Cho, Kyunghyun, and Bengio, Yoshua. 2014. On the Number of Linear Regions of Deep Neural Networks. Pages 2924–2932 of: *Advances in Neural Information Processing Systems*.

Mukherjee, Sayan, Niyogi, Partha, Poggio, Tomaso, and Rifkin, Ryan. 2006. Learning Theory: Stability is Sufficient for Generalization and Necessary and Sufficient for Consistency of Empirical Risk Minimization. *Advances in Computational Mathematics*, **25**(1), 161–193.

Neyshabur, Behnam, Tomioka, Ryota, and Srebro, Nathan. 2015a. Norm-Based Capacity Control in Neural Networks. Pages 1376–1401 of: *Proc. 28th Conference on Learning Theory*.

Neyshabur, Behnam, Salakhutdinov, Ruslan R., and Srebro, Nati. 2015b. Path-SGD: Path-Normalized Optimization in Deep Neural Networks. Pages 2422–2430 of: *Advances in Neural Information Processing Systems*.

Pascanu, Razvan, Montufar, Guido, and Bengio, Yoshua. 2014. On the Number of Response Regions of Deep Feed Forward Networks with Piece-Wise Linear Activations. In: *Proc. International Conference on Learning Representations*.

Poggio, Tomaso, Kawaguchi, Kenji, Liao, Qianli, Miranda, Brando, Rosasco, Lorenzo, Boix, Xavier, Hidary, Jack, and Mhaskar, Hrushikesh. 2018. Theory of Deep Learning III: Explaining the Non-overfitting Puzzle. Massachusetts Institute of Technology CBMM Memo No. 73.

Poggio, Tomaso, Mhaskar, Hrushikesh, Rosasco, Lorenzo, Miranda, Brando, and Liao, Qianli. 2017. Why and When Can Deep – But not Shallow – Networks Avoid the Curse of Dimensionality: A Review. *International Journal of Automation and Computing*, **14**, 1–17.

Raginsky, Maxim, Rakhlin, Alexander, and Telgarsky, Matus. 2017. Non-Convex Learning via Stochastic Gradient Langevin Dynamics: A Nonasymptotic Analysis. Pages 1674–1703 of: *Proc. Conference on Learning Theory*.

Sato, Ikuro, Nishimura, Hiroki, and Yokoi, Kensuke. 2015. Apac: Augmented Pattern Classification with Neural Networks. ArXiv preprint arXiv:1505.03229.

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

Shawe-Taylor, John, Bartlett, Peter L., Williamson, Robert C., and Anthony, Martin. 1998. Structural Risk Minimization over Data-Dependent Hierarchies. *IEEE Transactions on Information Theory*, **44**(5), 1926–1940.

Sokolic, Jure, Giryes, Raja, Sapiro, Guillermo, and Rodrigues, Miguel. 2017a. Generalization Error of Invariant Classifiers. Pages 1094–1103 of: *Artificial Intelligence and Statistics*.

Sokolic, Jure, Giryes, Raja, Sapiro, Guillermo, and Rodrigues, Miguel R. D. 2017b. Robust Large Margin Deep Neural Networks. *IEEE Transactions on Signal Processing*, **65**(16), 4265–4280.

Sun, Shizhao, Chen, Wei, Wang, Liwei, Liu, Xiaoguang, and Liu, Tie-Yan. 2016. On the Depth of Deep Neural Networks: A Theoretical View. Pages 2066–2072 of: *Proc. 30th AAAI Conference on Artificial Intelligence*. AAAI Press.

Telgarsky, Matus. 2016. Benefits of Depth in Neural Networks. Pages 1517–1539 of: *Proc. 29th Annual Conference on Learning Theory*.

Tropp, Joel A. 2012. User-Friendly Tail Bounds for Sums of Random Matrices. *Foundations of Computational Mathematics*, **12**(4), 389–434.

Tropp, Joel A., et al. 2015. An Introduction to Matrix Concentration Inequalities. *Foundations and Trends® in Machine Learning*, **8**(1-2), 1–230.

Vapnik, Vladimir. 1998. *Statistical Learning Theory*. Vol. 1. Wiley.

Wan, Li, Zeiler, Matthew, Zhang, Sixin, Cun, Yann L., and Fergus, Rob. 2013. Regularization of Neural Networks using Dropconnect. Pages 1058–1066 of: *Proc. 30th International Conference on Machine Learning*.

Wu, Lei, Zhu, Zhanxing, et al. 2017. Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes. ArXiv preprint arXiv:1706.10239.

Xie, Pengtao, Deng, Yuntian, and Xing, Eric. 2015. On the Generalization Error Bounds of Neural Networks under Diversity-Inducing Mutual Angular Regularization. ArXiv preprint arXiv:1511.07110.

Xie, Saining, Girshick, Ross, Dollár, Piotr, Tu, Zhuowen, and He, Kaiming. 2017. Aggregated Residual Transformations for Deep Neural Networks. Pages 1492–1500 of: *Proc. IEEE Conference on Computer Vision and Pattern Recognition*.

Xu, Huan, and Mannor, Shie. 2012. Robustness and Generalization. *Machine Learning*, **86**(3), 391–423.

Zhang, Chiyuan, Bengio, Samy, Hardt, Moritz, Recht, Benjamin, and Vinyals, Oriol. 2017. Understanding Deep Learning Requires Rethinking Generalization. In: *Proc. International Conference on Learning Representations*.