Yale University CPSC 683
Stability in Machine Learning: Generalization, Privacy & Replicability

Lecture 4: Failures of Uniform Convergence and Domain Adaptation

Instructor: Alkis Kalavasis, alkis.kalavasis@yale.edu

In Lecture 4, we will discuss the work of Nagarajan and Kolter (2019) on why uniform convergence fails to explain generalization in deep learning. In the second part of Lecture 4, we will talk about domain adaptation and extrapolation.

1 Uniform Convergence Fails to Explain Deep Learning

The presentation of this Section follows the work of Nagarajan and Kolter (2019) but is also inspired by https://locuslab.github.io/2019-07-09-uniform-convergence/.

Explaining why overparameterized deep networks generalize well has become an important open question in deep learning. To theoretically study the generalization behavior of a model, one must upper bound the gap between the population error (expected error under the underlying distribution) and the training error (under the assumption that the observations are i.i.d from the same populations). As we saw in Lectures 1 and 2, generalization bounds have the following form:

$$\text{Population Error} \leq \text{Training Error} + O\left(\frac{\text{Complexity Measure}}{\sqrt{m}}\right),$$

where m is the sample size. The way the complexity measure term is quantified in deep networks has evolved over time. We will now provide a quick overview.

Classics Standard generalization bounds (based e.g., on the VC dimension) consider a uniform convergence bound on the class of functions representable by a deep network to control the complexity of the concept class.

This approach is already known to fail in deep learning as we saw in the previous Lectures (e.g., (Zhang et al., 2021)). Recall that standard VC bounds fail because the obtained complexity measure for neural networks scales as width times depth (i.e., proportionally to the size of the network) and so the generalization bound is:

$$\text{Population Error} \leq \text{Training Error} + O\left(\frac{\text{width} \times \text{depth}}{\sqrt{m}}\right).$$

which is vacuous in the overparameterized setting (where the number of parameters is much larger than the sample size). As a remark, the above bound does not say much about the Rademacher complexity of the models (for nice data distributions, it could be much smaller than the above pessimistic bound).

Modern Approach. The issues based on the classical approach were reconsider in the work of Neyshabur et al. (2015), which sparked several new ideas and work in the deep learning theory community.

The key intuition is the one we already discussed in Lecture 1: *understand the inductive bias of SGD in deep learning*. The main question raised was to identify how does the underlying data distribution and the training algorithm as a pair restrict the "complexity" of the search (i.e., representation) space of the deep networks to a "simple" enough class of functions? Given this simple class, we could obtain generalization bounds based on its complexity, which will be significantly smaller and less pessimistic.

For instance, the generalization error will look like this:

$$\text{Population Error} \leq \text{Training Error} + O\left(\frac{\text{Complexity of Capacity Controlled NN}}{\sqrt{m}}\right).$$

Roughly speaking, these capacity controlled neural networks will correspond for instance to the *norms of the network weights* controlled by SGD:

$$\text{PopulationError} \leq \text{TrainingError} + O\left(\frac{\text{Weight Norms Controlled by SGD for Given Distribution}}{\sqrt{m}}\right).$$

This inspired several lines of work using various tools such as

1. Rademacher Complexity (Neyshabur et al., 2015; Golowich et al., 2018)
2. Covering Numbers (Bartlett et al., 2017)
3. PAC-Bayes (Neyshabur et al., 2017; Dziugaite and Roy, 2017)
4. Compression (Arora et al., 2018)

Recall that the empirical Rademacher complexity is defined as

$$\widehat{R}_m(\mathcal{H}, \{x_1, ..., x_m\}) = \mathop{\mathbf{E}}_{\sigma \sim \mathcal{U}(\{\pm 1\}^m)} \sup_{h \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(x_i).$$

The standard modern approach (Neyshabur et al., 2015; Golowich et al., 2018) is to give bounds on the Rademacher complexity (sometimes independent of $x_1, ..., x_m$ as long as they are assumed to have norm at most B), with respect to classes of neural networks with various norm constraints. Using standard arguments, an upper bound on the Rademacher complexity can be converted to bounds on the generalization error, assuming access to a sample of m i.i.d. training examples.

For instance, let us consider standard depth-d neural networks, of the form

$$x \mapsto W_d \sigma_{d-1}(W_{d-1}\sigma_{d-2}(...W_2\sigma_1(W_1 x))).$$

Theorem 1: Neyshabur et al. (2015)

If the parameter matrices $W_1, ..., W_d$ for each one of the d layers have Frobenious norm bounds $M_F(d), ..., M_F(d)$, $\|x\| \leq B$ and under suitable assumptions on the activation functions, the generalization error scales (with high probability) as

$$\frac{B \cdot 2^d \cdot \prod_i M_F(i)}{\sqrt{m}}.$$

Theorem 2: Golowich et al. (2018)

If the parameter matrices $W_1, ..., W_d$ for each one of the d layers have Frobenious norm bounds $M_F(d), ..., M_F(d)$, $\|x\| \leq B$ and under suitable assumptions on the activation functions, the generalization error scales (with high probability) as

$$\frac{B \cdot \sqrt{d} \cdot \prod_i M_F(i)}{\sqrt{m}}.$$

However, behind all these results, the main tool needed is still uniform convergence. A lot of ongoing efforts in this space has been focused on developing more novel or refined variations of such uniform convergence-based generalization bounds to better capture generalization in deep learning. However, the goal of providing a complete explanation of generalization through these bounds appears to be elusive.

Observation The first observation of Nagarajan and Kolter (2019) is that the norm-based bounds are usually vacuous because as the training size increases, the weight norms also increase.

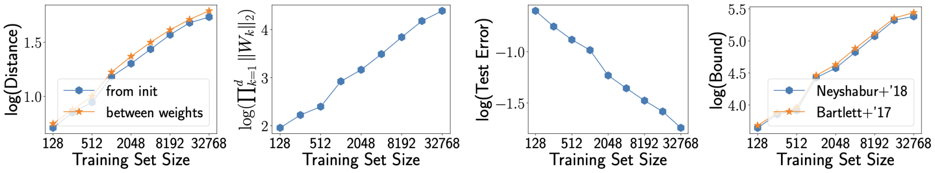

Figure 1: **Experiments in Section 2:** In the **first** figure, we plot (i) ℓ_2 the distance of the network from the initialization and (ii) the ℓ_2 distance between the weights learned on two random draws of training data starting from the same initialization. In the **second** figure we plot the product of spectral norms of the weights matrices. In the **third** figure, we plot the test error. In the **fourth** figure, we plot the bounds from [32, 3]. Note that we have presented log-log plots and the exponent of m can be recovered from the slope of these plots.

Figure 1: Weight Norms grow with the sample size. **Taken from Nagarajan and Kolter (2019).**

Uniform Convergence For a given $\delta \in (0,1)$, the generalization error of the algorithm is essentially a bound on the difference between the error of the hypothesis h_S learned on a training set S and the expected error over D, that holds with high probability of at least $1-\delta$ over the draws of S. More formally:

> **Definition 1: Generalization Error**
>
> The generalization error of A with respect to L is the smallest $\epsilon_{gen}(n,\delta)$ such that
>
> $$\Pr_{S\sim\mathcal{D}^n}[L_{\mathcal{D}}(A(S)) - L_S(A(S)) \leq \epsilon_{gen}(n,\delta)] \geq 1-\delta$$

As we saw in Lecture 1, the standard way to control the generalization error is uniform convergence:

> **Definition 2: Uniform Convergence**
>
> The uniform convergence bound with respect to loss L is the smallest value $\epsilon_{unif}(n,\delta)$ such that
>
> $$\Pr_{S\sim\mathcal{D}^n}[\sup_{h\in\mathcal{H}} L_{\mathcal{D}}(h) - L_S(h) \leq \epsilon_{unif}(n,\delta)] \geq 1-\delta$$
>
> Equivalently, we can say that $\epsilon_{unif}(n,\delta)$ is the smallest value for which there exists a set of sample sets $S_\delta \subseteq (\mathcal{X}\times\{0,1\})^n$ with $\mathbf{Pr}_S(S\in S_\delta) \geq 1-\delta$ and $\sup_{S\in S_\delta}\sup_{h\in\mathcal{H}}|L_{\mathcal{D}}(h) - L_S(h)| \leq \epsilon_{unif}(n,\delta)$.

Given the discussions in Lectures 2 and 3, one might suspect that pruning the search space and keeping only the concepts explored by the training algorithm, would imply generalization bounds (via uniform convergence in the pruned space). The main goal of Nagarajan and Kolter (2019) is to show that this intuition is also false.

Tightest algorithm-dependent uniform convergence. The bound given by ϵ_{unif} can be tightened by ignoring many extraneous hypotheses in $\mathcal{H}$ never picked/explored by the algorithm A for a given distribution $\mathcal{D}$. This is typically done by focusing on a norm-bounded class of hypotheses that the algorithm A implicitly restricts itself to (recall the bounds we saw before).

The main idea of Nagarajan and Kolter (2019) is to take this intuition to the extreme and apply uniform convergence on the *smallest possible class* of concepts, namely, only those hypotheses that are picked by A under D, excluding everything else. The reason is that pruning the hypothesis class any further would not imply a bound on the generalization error, and hence applying uniform convergence on this extremely pruned hypothesis class would yield the tightest possible uniform convergence bound.

Definition 3: Algorithm-Dependent Uniform Convergence

The tightest algorithm-dependent uniform convergence bound with respect to loss L is the smallest value $\epsilon_{unif,alg}(n,\delta)$ for which there exists a set of sample sets S_δ such that $\mathbf{Pr}_{S\sim\mathcal{D}^n}[S \in S_\delta] \geq 1-\delta$ and and if we define the space of hypotheses explored by A as $\mathcal{H}_\delta = \cup_{S\in S_\delta} A(S) \subseteq \mathcal{H}$ then $\sup_{S\in S_\delta} \sup_{h\in\mathcal{H}_\delta} |L_\mathcal{D}(h) - L_S(h)| \leq \epsilon_{unif,alg}(n,\delta)$.

Through examples of overparameterized models trained by GD (or SGD), Nagarajan and Kolter (2019) argue how even the above tightest algorithm-dependent uniform convergence can fail to explain generalization. i.e., in these settings, even though ϵ_{gen} is smaller than a negligible value ϵ, they show that $\epsilon_{unif,alg}$ is large (close to $1-\epsilon$).

Sketch of the Idea. Consider a scenario where the algorithm generalizes well i.e., for every training set S, $h_S = A(S)$ has zero error on S and has small test error. This means that the generalization error is small.

While this means that h_S has small error on random draws of a test set, it may still be possible that for every such h_S, there exists a corresponding "bad" dataset S' – that is not random, but rather dependent on S – on which h_S has a large empirical error (say 1).

Unfortunately, uniform convergence runs into trouble while dealing with such bad datasets. Specifically, as we can see from the above definition, uniform convergence demands that $L_\mathcal{D}(h_S) - L_S(h_S)$ be small on all datasets in S_δ , which excludes a δ fraction of the datasets.

While it may be tempting to think that we can somehow exclude the bad dataset as part of the δ-fraction, there is a significant catch here: we *can not* carve out a δ fraction specific to each hypothesis; we *can ignore only a single chunk* of δ-mass common to all hypotheses in $\mathcal{H}_\delta$.

This restriction turns out to be a tremendous bottleneck: despite ignoring this δ fraction, for most $h \in \mathcal{H}_\delta$, the corresponding bad set S' would still be left in S_δ.

Then, for all such h_S in $\mathcal{H}_\delta$, $L_\mathcal{D}(h_S)$ will be small but in the definition of uniform convergence we can put $S' \in S_\delta$: this will make $L_{S'}(h_S)$ large when we take the supremum over the sets S_δ: this means that $\epsilon_{unif,alg}$ is indeed vacuous.

The above argument is formalized as follows.

Theorem 3: Nagarajan and Kolter (2019)

Consider an algorithm that has zero training error and a test error of ϵ (on almost all draws of the training set). Assume the algorithm is such that, for nearly all draws of S, one can construct another dataset S' of size n (which can be depend on S) such that:

- h_S misclassifies S' completely.
- The law of S' satisfies $S' \sim \mathcal{D}^n$ (note that S' is a random variable that is a function of S, and here we want the distribution of S', with S marginalized out, to be equal to $\mathcal{D}^n$).

If the algorithm satisfies these requirements, then: $\epsilon_{unif,alg} \geq 1-\epsilon$.

Exercise 1

Understand the proofs in the paper.

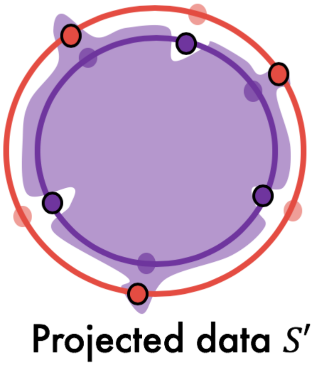

Figure 2: We design the bad dataset S' merely by projecting every point in S to its opposite hypersphere and flipping its label. That is, every point on the inner hypersphere is projected onto the outer hypersphere and vice versa, and then the labels are fixed accordingly. **Taken from**: `https://locuslab.github.io/2019-07-09-uniform-convergence/`

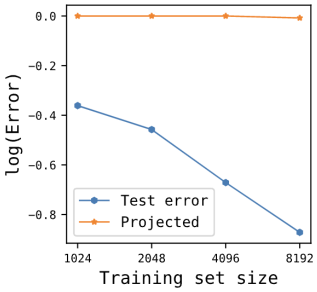

Figure 3: While the test loss goes to 0 with the number of samples, the loss on S' is not improving indicating that UC fails. **Taken from** `https://locuslab.github.io/2019-07-09-uniform-convergence/`.

Insight 1

The decision boundary learned by SGD on overparameterized deep networks can have certain complexities that hurt uniform convergence without hurting generalization.
The conjecture of Nagarajan and Kolter (2019) was the following:

1. the decision boundary is macroscopically simple, leading to good generalization (the trained classifier will not encounter the ”hard” designed dataset).

2. The complex skews in the decision boundary are only microscopic (i.e., they cover only low probability regions in the input space). Hence, they will not affect generalization error. However, uniform convergence, which is by its nature extremely worst-case (even in the algorithm-dependent version) cannot ignore these skews since it takes a supremum over pairs in $(S_\delta, \mathcal{H}_\delta)$.

2 Domain Adaptation

Having extensively discussed about generalization, we have ignored an important aspect of it: Perhaps the most concerning issue regarding generalization in Machine Learning is the assumption that the training distribution is exactly the same as the testing distribution.

The ultimate of Machine Learning is to shed light on understanding how algorithms can be trained on data from the source domain and then adeptly applied to the target domain. The most interesting examples correspond to cases where the target distribution of data, denoted as Q, differs from that of the source domain, characterized by distribution P.

2.1 Domain Adaptation in Regression

In the second part of Lecture 4, we will study domain adaptation through the lens of regression. Here, our goal is to identify a *regressor* f that aims to minimize the mean squared error $\mathbf{E}_{x\sim P}[(f(x)-f^\star(x))^2]$, using samples in the form (x,y), where x is drawn from distribution P and $\mathbf{E}[y|x]=f^\star(x)$. The twist in transfer learning emerges when the objective shifts towards minimizing the expected error $\mathbf{E}_{x\sim Q}[(f(x)-f^\star(x))^2]$, despite the fact that our available samples are drawn from P. This raises the pivotal question: is it possible to establish a relationship between the expected errors $\mathbf{E}_{x\sim P}[(f(x)-f^\star(x))^2]$ and $\mathbf{E}_{x\sim Q}[(f(x)-f^\star(x))^2]$? In essence, this inquiry delves into the capability of the regressor f to *extrapolate* effectively to samples that are out-of-distribution, originating from the target distribution Q, based on its training with the source distribution P.[1]

A natural approach to compare the mean squared error with respect to P with the mean squared error with respect to Q is to apply a change of measure:

$$\begin{aligned}\mathop{\mathbf{E}}_{x\sim Q}[(f(x)-f^\star(x))^2] &= \mathop{\mathbf{E}}_{x\sim P}\left[\frac{Q(x)}{P(x)}(f(x)-f^\star(x))^2\right]\\ &\le \left\|\frac{dQ}{dP}\right\|_\infty \mathop{\mathbf{E}}_{x\sim P}[(f(x)-f^\star(x))^2]\,, \end{aligned} \tag{1}$$

where $\|dQ/dP\|_\infty = \sup_{x\in\mathrm{supp}(Q)} Q(x)/P(x)$.

Due to the many instances where the ratio dQ/dP is unbounded (Kalavasis et al., 2024), a pertinent question arises:

Is transfer learning possible when $\|dQ/dP\|_\infty \to \infty$?

We consider the expressive and well-studied class of *low-degree polynomials*. We will show general transfer inequalities, for functions in this class, under mild assumptions on the distribution pair P and Q, going well beyond the boundedness of the ratio dQ/dP. A distribution with density Q is log-concave if $-\log Q$ is convex.

Theorem 4: Kalavasis et al. (2024)

For any pair of degree-d polynomials $f, f^\star : \mathbb{R}^n \to \mathbb{R}$, any log-concave distribution Q and any continuous distribution P, it holds that

$$\mathop{\mathbf{E}}_{Q}[(f(x)-f^\star(x))^2] \le C_d \cdot \left\|\frac{dP}{dQ}\right\|_\infty^{2d} \cdot \mathop{\mathbf{E}}_{P}[(f(x)-f^\star(x))^2]\,, \tag{2}$$

where $C_d > 0$ is a constant depending only on the degree d. Furthermore, even when Q is not log-concave, it holds that

$$\mathop{\mathbf{E}}_{Q}[(f(x)-f^\star(x))^2] \le C_d \cdot \inf_{\mu\in\mathcal{L}} \left\|\frac{dQ}{d\mu}\right\|_\infty \left\|\frac{dP}{d\mu}\right\|_\infty^{2d} \cdot \mathop{\mathbf{E}}_{P}[(f(x)-f^\star(x))^2]\,, \tag{3}$$

where the infimum is over the set $\mathcal{L}$ of log-concave distributions over $\mathbb{R}^n$ and P, Q are arbitrary[a]

[a] Observe that it is necessary for the inequality not being void to take the infimum over measures $\mu \in \mathcal{L}$ whose support contains supp(P) $\cup$ supp(Q).

[1] When talking about the Euclidean domain, we consider a probability space $(\Omega, \mathcal{F}, (P, Q))$ where $\Omega = \mathbb{R}^n$ and for clarity we will assume that both distributions P and Q are *continuous* distributions.

2.2 The Proof: Anti-Concentration Implies Extrapolation

For $\alpha \geq 1$, we define the divergence $D_\alpha(P \parallel Q) \triangleq \left(\mathbf{E}_{x\sim Q}\left[\left(\frac{P(x)}{Q(x)}\right)^{\alpha}\right]\right)^{1/\alpha}$; we denote both distributions and associated density functions by P, Q overloading the notation. For $\alpha = \infty : D_\alpha(P \parallel Q) = \|dP/dQ\|_\infty$.

Let us fix some $\mu \in \mathcal{L}$ whose support contains both that of Q and that of P. We will overload the notation and denote by P (resp. Q, μ) the density of the associated distribution. The first step of the proof is to upper bound the expectation of f under Q by that under μ. This is done by a simple change of measure and an application of Hölder's inequality with parameters α, β:

$$\mathop{\mathbf{E}}_{Q}|f| = \mathop{\mathbf{E}}_{x\sim\mu}\left[\frac{Q(x)}{\mu(x)}|f(x)|\right] \leq D_\alpha(Q \parallel \mu)\left(\mathop{\mathbf{E}}_{\mu}|f|^{\beta}\right)^{1/\beta} . \tag{4}$$

As a next step we need to relate expectations under the log-concave measure μ and under the general measure P, which is the non-trivial part of the argument. We will first lower bound the value of $\mathbf{E}_P |f|^\beta$. For some $\gamma > 0$ to be decided, we have that

$$\mathop{\mathbf{E}}_{P}|f|^{\beta} \geq \mathop{\mathbf{E}}_{x\sim P}\left[|f|^{\beta}\,\Big|\,|f(x)|^{\beta} \geq \gamma\right] \mathop{\mathbf{Pr}}_{x\sim P}\left[|f(x)|^{\beta} \geq \gamma\right] \geq \gamma \mathop{\mathbf{Pr}}_{x\sim P}\left[|f(x)|^{\beta} \geq \gamma\right].$$

Hence we get that

$$\mathop{\mathbf{E}}_{P}|f|^{\beta} \geq \gamma\left(1 - \mathop{\mathbf{Pr}}_{x\sim P}\left[|f(x)|^{\beta} \leq \gamma\right]\right) . \tag{5}$$

In order to further lower bound the right-hand side of the above inequality, it suffices to obtain an upper bound on the probability mass of the event $\{x \in \mathbb{R}^n : |f(x)|^\beta \leq \gamma\}$ under P. We have that

$$\mathop{\mathbf{Pr}}_{x\sim P}\left[|f(x)| \leq \gamma^{1/\beta}\right] = \mathop{\mathbf{E}}_{x\sim\mu}\left[\frac{P(x)}{\mu(x)}\, 1\{|f(x)| \leq \gamma^{1/\beta}\}\right] .$$

We can upper bound this quantity using Hölder's inequality for $\alpha, \beta \in [1, \infty]$ with $1/\alpha + 1/\beta = 1$:

$$\mathop{\mathbf{Pr}}_{x\sim P}\left[|f(x)|^{\beta} \leq \gamma\right] \leq D_\alpha(P \parallel \mu)\left(\mathop{\mathbf{Pr}}_{x\sim\mu}\left[|f(x)| \leq \gamma^{1/\beta}\right]\right)^{1/\beta} \tag{6}$$

We now use the following result.

Theorem 5: Carbery and Wright (2001)

There exists an absolute constant C such that the following holds: let $f : \mathbb{R}^n \to \mathbb{R}$ be a non-zero polynomial of degree at most d, μ be a log-concave probability distribution over $\mathbb{R}^n$ and $q, \gamma \in (0, \infty)$. Then it holds that

$$\mathop{\mathbf{Pr}}_{x\sim\mu}\left[|f(x)| \leq \gamma\right] \leq \frac{Cq\gamma^{1/d}}{\left(\mathbf{E}_\mu |f|^{q/d}\right)^{1/q}} .$$

In particular, for $q = d$, it holds that $\mathbf{Pr}_{x\sim\mu}\left[|f(x)| \leq \gamma\right] \leq \frac{Cd\gamma^{1/d}}{\left(\mathbf{E}_\mu |f|\right)^{1/d}}$.

Applying the Carbery-Wright inequality in (6), by picking $q = \beta d$, we get that

$$\mathop{\mathbf{Pr}}_{x\sim P}\left[|f(x)| \leq \gamma^{1/\beta}\right] \leq D_\alpha(P \parallel \mu)\left(\frac{C\beta d\gamma^{1/(\beta d)}}{\left(\mathbf{E}_\mu |f|^{\beta}\right)^{1/\beta d}}\right)^{1/\beta} \tag{7}$$

We are now ready to pick γ. We choose γ so that $D_\alpha(P \parallel \mu)\left(\frac{C\beta d\gamma^{1/(\beta d)}}{\left(\mathbf{E}_\mu |f|^\beta\right)^{1/(\beta d)}}\right)^{1/\beta} = 1/2$, which implies that $\gamma = \frac{\mathbf{E}_\mu |f|^\beta}{\left(C\beta d 2^\beta D_\alpha(P\|\mu)^\beta\right)^{\beta d}}$. Using this choice we can return to (5) and get that

$$\mathop{\mathbf{E}}_P |f|^\beta \geq \frac{\mathbf{E}_\mu |f|^\beta}{2\left(Cd2^\beta D_\alpha(P \parallel \mu)^\beta\right)^{\beta d}}$$

We can now complete the proof using (4), which gives that

$$\mathop{\mathbf{E}}_Q |f| \leq D_\alpha(Q \parallel \mu)\left(\mathop{\mathbf{E}}_\mu |f|^\beta\right)^{1/\beta} \leq 2C^d d^d 2^{d\beta} D_\alpha(Q \parallel \mu) D_\alpha(P \parallel \mu)^{\beta d}\left(\mathop{\mathbf{E}}_P |f|^\beta\right)^{1/\beta}.$$

By minimizing over $\mu \in \mathcal{L}$, the proof is complete. The statement of Theorem 4 follows by setting $\alpha = \infty$.

2.3 Extrapolation in Classification

This is a research direction which is still wide open. Note that the above result about regression can be directly used for classification (why?).

Regarding binary classification, even identification is highly non-obvious, which we will shortly discuss.

Definition 4: Extrapolation

Fix distribution family $\mathcal{P}$ and survival set S. We will say that a concept $h^\star \in \mathcal{H} \subseteq \{0,1\}^{\mathbb{R}^d}$ can be extrapolated with respect to $(\mathcal{P}, S)$ if for any $Q \in \mathcal{P}$ and for any $h \in \mathcal{H}$, it holds that

$$h^\star \neq h \Rightarrow Q_{h^\star,S} \neq Q_{h,S}\,.$$

Assume that

1. It holds that $\partial h^\star \cap S \neq \emptyset$ for each connected component of $h^\star$.
2. It holds that $Q(S) > 0$.

We note that without capturing every connected decision boundary by the survival set, it is information theoretically impossible to extrapolate. We denote by $\mathrm{PTF}_{k,d}$ the class of polynomial-threshold functions of degree k in dimension d.

Theorem 6: Extrapolation of 2D PTFs

Any $h^\star$ in $\mathrm{PTF}_{k,2}$ can be extrapolated from any marginal distribution Q and survival set S that satisfy Conditions 1 and 2.

Exercise 2

Prove this result.

References

Arora, S., Ge, R., Neyshabur, B., and Zhang, Y. (2018). Stronger generalization bounds for deep nets via a compression approach. In *International conference on machine learning*, pages 254–263. PMLR.

Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. (2017). Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30.

Carbery, A. and Wright, J. (2001). Distributional and lq norm inequalities for polynomials over convex bodies in rn. *Mathematical research letters*, 8(3):233–248.

Dziugaite, G. K. and Roy, D. M. (2017). Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv preprint arXiv:1703.11008*.

Golowich, N., Rakhlin, A., and Shamir, O. (2018). Size-independent sample complexity of neural networks. In *Conference On Learning Theory*, pages 297–299. PMLR.

Kalavasis, A., Zadik, I., and Zampetakis, M. (2024). Transfer learning beyond bounded density ratios. *arXiv preprint arXiv:2403.11963*.

Nagarajan, V. and Kolter, J. Z. (2019). Uniform convergence may be unable to explain generalization in deep learning. *Advances in Neural Information Processing Systems*, 32.

Neyshabur, B., Bhojanapalli, S., and Srebro, N. (2017). A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*.

Neyshabur, B., Tomioka, R., and Srebro, N. (2015). Norm-based capacity control in neural networks. In *Conference on learning theory*, pages 1376–1401. PMLR.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2021). Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115.