UNRAVEL, December 17, 2024

Generalization of Stochastic Gradient Descent via Algorithm-dependent Rademacher Complexity

Tim van Erven

University of Amsterdam

Joint work with:

Sarah Sachs

Liam Hodgkinson

Rajiv Khanna

Umut Şimşekli

Outline

Context: Big Interest in New Complexity Measures Because of Deep Learning

New Algorithm-dependent Complexity Measure

Computation & Generalization in Deep Learning

Non-convex Optimization:

- Millions of data points: too many to process all at once
- Process one data point (or minibatch) at a time using stochastic gradient descent

Computation & Generalization in Deep Learning

Non-convex Optimization:

- Millions of data points: too many to process all at once
- Process one data point (or minibatch) at a time using stochastic gradient descent

High-dimensional Setting:

- Still many more parameters than data points (e.g. 25 times as many)
- Statistically obvious: we cannot estimate so many parameters unless we add constraints (e.g. restrict to L_p ball)

Computation & Generalization in Deep Learning

Non-convex Optimization:

- Millions of data points: too many to process all at once
- Process one data point (or minibatch) at a time using stochastic gradient descent

High-dimensional Setting:

- Still many more parameters than data points (e.g. 25 times as many)
- Statistically obvious: we cannot estimate so many parameters unless we add constraints (e.g. restrict to L_p ball)
- But even if you disable all standard regularization, it still works! (Zhang, Bengio, Hardt, Recht, and Vinyals, 2021)
- So how are the parameters restricted?

Computation & Generalization in Deep Learning

Non-convex Optimization:

- Millions of data points: too many to process all at once
- Process one data point (or minibatch) at a time using stochastic gradient descent

High-dimensional Setting:

- Still many more parameters than data points (e.g. 25 times as many)
- Statistically obvious: we cannot estimate so many parameters unless we add constraints (e.g. restrict to L_p ball)
- But even if you disable all standard regularization, it still works! (Zhang, Bengio, Hardt, Recht, and Vinyals, 2021)
- So how are the parameters restricted? **By the behavior of the optimization algorithm!**

Computation & Generalization in Deep Learning

Non-convex Optimization:

- Millions of data points: too many to process all at once
- Process one data point (or minibatch) at a time using stochastic gradient descent

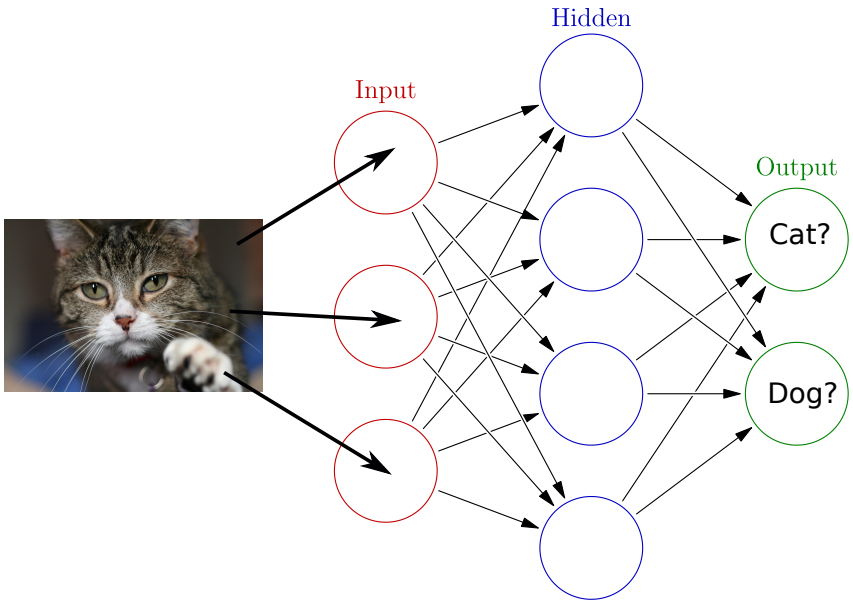
High-dimensional Setting:

- Still many more parameters than data points (e.g. 25 times as many)
- Statistically obvious: we cannot estimate so many parameters unless we add constraints (e.g. restrict to L_p ball)
- But even if you disable all standard regularization, it still works! (Zhang, Bengio, Hardt, Recht, and Vinyals, 2021)
- So how are the parameters restricted? **By the behavior of the optimization algorithm!**

Big open question: Can we **understand** why common optimization methods (on realistic data) **generalize** from the training data to unseen data?
See e.g. (Belkin, Hsu, Ma, and Mandal, 2019).

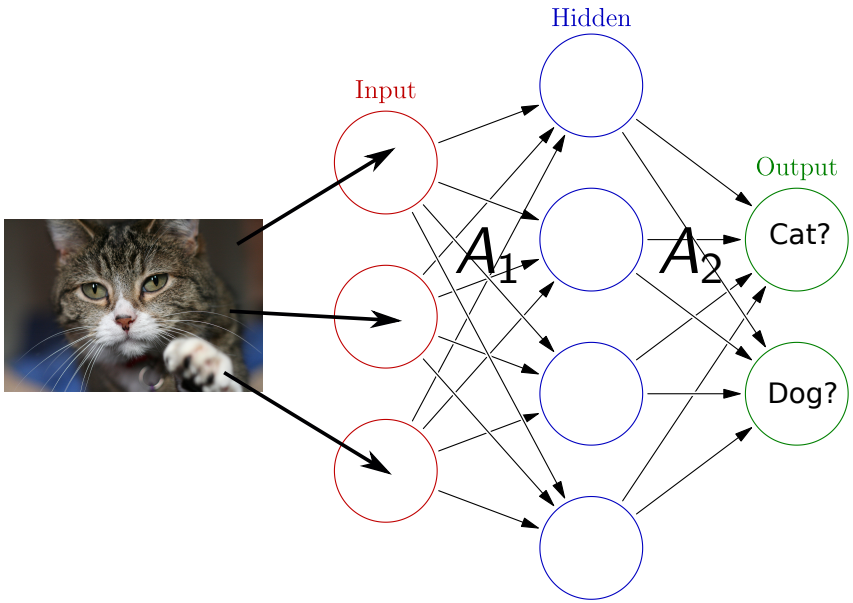
Fully Connected Neural Networks

Simplest case we do not understand



Fully Connected Neural Networks

Simplest case we do not understand



Class of **non-convex** functions parametrized by matrices $\theta = (A_1, \ldots, A_m)$:

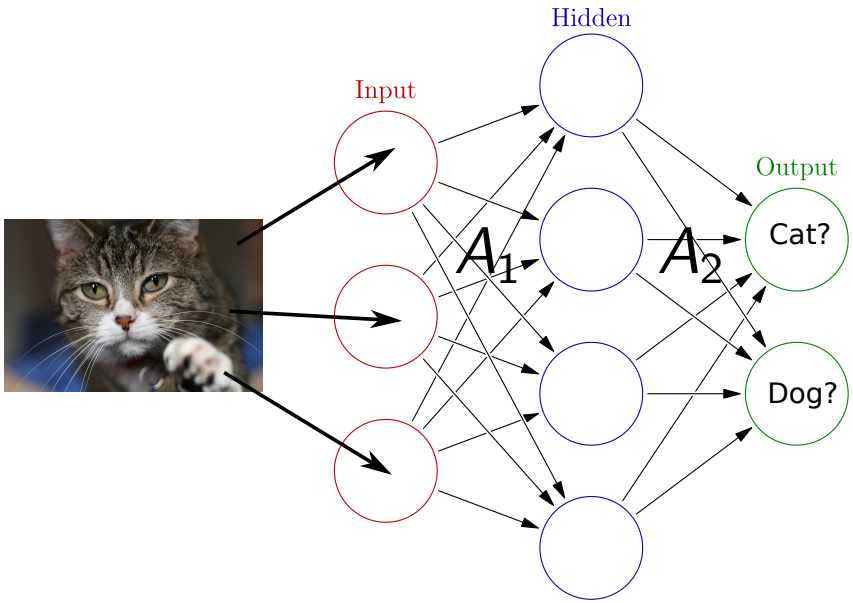
Fully connected network: $f_\theta(X) = A_m \sigma A_{m-1} \cdots \sigma A_1 X \qquad (\theta \in \mathbb{R}^d)$

with **activation function** $\sigma(z)$ applied component-wise to vectors. E.g.

- Rectified linear unit (ReLU): $\sigma(z) = \max\{0, z\}$
- Sigmoid: $\sigma(z) = 1/(1 + e^{-z})$

Fully Connected Neural Networks

Simplest case we do not understand



Class of **non-convex** functions parametrized by matrices $\theta = (A_1, \ldots, A_m) \in \mathbb{R}^d$:

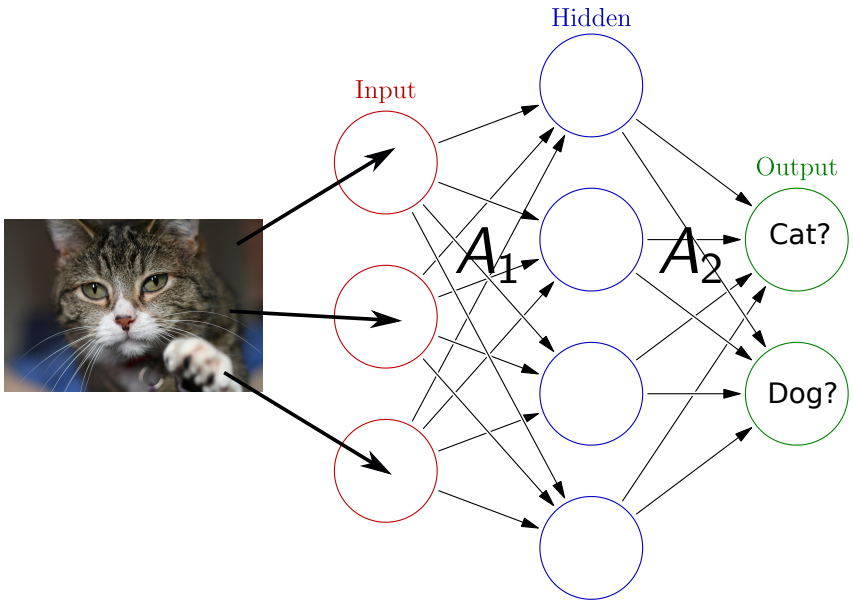
Fully connected network: $f_\theta(X) = A_m \sigma A_{m-1} \cdots \sigma A_1 X \qquad (\theta \in \mathbb{R}^d)$

with **activation function** $\sigma(z)$ applied component-wise to vectors. E.g.

- Rectified linear unit (ReLU): $\sigma(z) = \max\{0, z\}$
- Sigmoid: $\sigma(z) = 1/(1 + e^{-z})$

Fully Connected Neural Networks

Simplest case we do not understand



VC-dimension dependence on nr. of parameters d:

ReLU: $\tilde{\Theta}(d)$ (Bartlett et al., 2017)

Sigmoid: $\Theta(d^2)$ (Anthony and Bartlett, 2009)

Conclusion: standard VC-dimension bounds vacuous unless $n \gg d$

Class of **non-convex** functions parametrized by matrices $\theta = (A_1, \ldots, A_m) \in \mathbb{R}^d$:

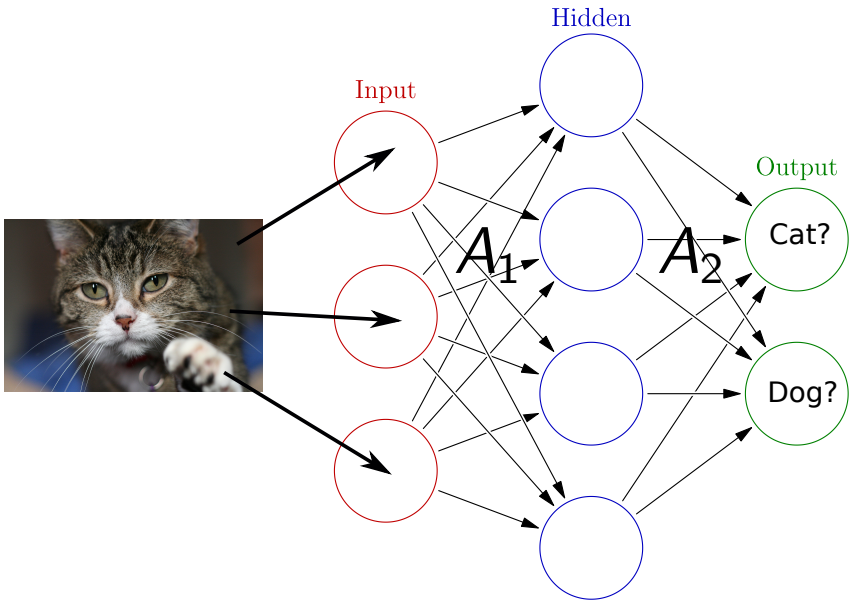
Fully connected network: $f_\theta(X) = A_m \sigma A_{m-1} \cdots \sigma A_1 X \qquad (\theta \in \mathbb{R}^d)$

with **activation function** $\sigma(z)$ applied component-wise to vectors. E.g.

- Rectified linear unit (ReLU): $\sigma(z) = \max\{0, z\}$
- Sigmoid: $\sigma(z) = 1/(1 + e^{-z})$

Fully Connected Neural Networks

Simplest case we do not understand



VC-dimension dependence on nr. of parameters d:

ReLU: $\tilde{\Theta}(d)$ (Bartlett et al., 2017)

Sigmoid: $\Theta(d^2)$ (Anthony and Bartlett, 2009)

Conclusion: standard VC-dimension bounds vacuous unless $n \gg d$

Class of **non-convex** functions parametrized by matrices $\theta = (A_1, \ldots, A_m) \in \mathbb{R}^d$:

F

w

- Sigmoid: $\sigma(z) = 1/(1 + e^{-z})$

Why do neural networks generalize?

- Classical VC bounds: need sample size $n \gg$ nr. parameters d
- In practice: sample size $n \ll$ nr. parameters d

Learning Setting

Given:

- Data: $S^n = \begin{pmatrix} Y_1 \\ X_1 \end{pmatrix}, \ldots, \begin{pmatrix} Y_n \\ X_n \end{pmatrix} \overset{i.i.d.}{\sim} P$
- Model: $\mathcal{F} = \{f_\theta : \theta \in \Theta \subset \mathbb{R}^d\}$
- Bounded loss: $\ell(\hat{Y}, Y) \in [0, b]$

We want an

- Algorithm: $\hat{\theta} \equiv \mathrm{Alg}(S^n) \in \Theta \subset \mathbb{R}^d$

That achieves small **risk** $R(\hat{\theta})$, which measures its performance on unseen data:

$$R(\theta) = \mathop{\mathbb{E}}_{(X,Y)\sim P}[\ell(f_\theta(X), Y)]$$

Learning Setting

Given:

- Data: $S^n = \begin{pmatrix} Y_1 \\ X_1 \end{pmatrix}, \ldots, \begin{pmatrix} Y_n \\ X_n \end{pmatrix} \overset{i.i.d.}{\sim} P$
- Model: $\mathcal{F} = \{f_\theta : \theta \in \Theta \subset \mathbb{R}^d\}$
- Bounded loss: $\ell(\hat{Y}, Y) \in [0, b]$

We want an

- Algorithm: $\hat{\theta} \equiv \mathrm{Alg}(S^n) \in \Theta \subset \mathbb{R}^d$

That achieves small **risk** $R(\hat{\theta})$, which measures its performance on unseen data:

$$R(\theta) = \mathop{\mathbb{E}}_{(X,Y)\sim P}[\ell(f_\theta(X), Y)]$$

Example: binary classification

$f(X), Y \in \{-1, +1\}$ both binary

$$\ell(f(X), Y) = \begin{cases} 0 & \text{if } f(X) = Y \\ 1 & \text{otherwise} \end{cases}$$

Learning Setting

Given:

- Data: $S^n = \begin{pmatrix} Y_1 \\ X_1 \end{pmatrix}, \ldots, \begin{pmatrix} Y_n \\ X_n \end{pmatrix} \overset{i.i.d.}{\sim} P$
- Model: $\mathcal{F} = \{f_\theta : \theta \in \Theta \subset \mathbb{R}^d\}$
- Bounded loss: $\ell(\hat{Y}, Y) \in [0, b]$

We want an

- Algorithm: $\hat{\theta} \equiv \mathrm{Alg}(S^n) \in \Theta \subset \mathbb{R}^d$

That achieves small **risk** $R(\hat{\theta})$, which measures its performance on unseen data:

$$R(\theta) = \mathop{\mathbb{E}}_{(X,Y)\sim P}[\ell(f_\theta(X), Y)]$$

Example: binary classification

$f(X), Y \in \{-1, +1\}$ both binary

$$\ell(f(X), Y) = \begin{cases} 0 & \text{if } f(X) = Y \\ 1 & \text{otherwise} \end{cases}$$

Example: logistic/cross-entropy loss

$f(X)$ bounded, $Y \in \{-1, +1\}$

$\ell(f(X), Y) = \log(1 + e^{-Yf(X)})$

The Generalization Error

$$R(\theta) = \mathop{\mathbb{E}}_{(X,Y)\sim P}[\ell(f_\theta(X), Y)]$$

Most algorithms (approximately) minimize the **empirical risk**

$$\hat{R}(\theta, S^n) = \frac{1}{n}\sum_{i=1}^{n} \ell(f_\theta(X_i), Y_i)]$$

So want to control the **generalization error**:

$$\boldsymbol{R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)}$$

The Generalization Error

$$R(\theta) = \mathop{\mathbb{E}}_{(X,Y)\sim P}[\ell(f_\theta(X), Y)]$$

Most algorithms (approximately) minimize the **empirical risk**

$$\hat{R}(\theta, S^n) = \frac{1}{n}\sum_{i=1}^{n} \ell(f_\theta(X_i), Y_i)]$$

So want to control the **generalization error**:

$$\boldsymbol{R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)}$$

Example: binary classification

$$\sup_\theta \left|R(\theta) - \hat{R}(\theta, S^n)\right| = O\left(\sqrt{\frac{\mathrm{VCdim}(\mathcal{F}) + \log(1/\delta)}{n}}\right) \qquad \text{w.p. } \geq 1-\delta$$

The Generalization Error

$$R(\theta) = \mathop{\mathbb{E}}_{(X,Y)\sim P}[\ell(f_\theta(X), Y)]$$

Most algorithms (approximately) minimize the **empirical risk**

$$\hat{R}(\theta, S^n) = \frac{1}{n}\sum_{i=1}^{n} \ell(f_\theta(X_i), Y_i)]$$

So want to control the **generalization error**:

$$\boldsymbol{R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)}$$

Example: linear model with Lipschitz loss
$\hat{Y} \mapsto \ell(\hat{Y}, Y))$ L-Lipschitz, $\ell(\hat{Y}, Y) \in [0, b]$
$f_\theta(X) = X^\intercal\theta$, $\|X\| \le 1$, $\|\theta\| \le B$

$$\sup_\theta \left|R(\theta) - \hat{R}(\theta, S^n)\right| = O\left(\frac{LB}{\sqrt{n}} + b\sqrt{\frac{\log(1/\delta)}{n}}\right) \qquad \text{w.p.} \ge 1-\delta$$

The Generalization Error

$$R(\theta) = \mathop{\mathbb{E}}_{(X,Y)\sim P}[\ell(f_\theta(X), Y)]$$

Most algorithms (approximately) minimize the **empirical risk**

$$\hat{R}(\theta, S^n) = \frac{1}{n}\sum_{i=1}^{n} \ell(f_\theta(X_i), Y_i)]$$

So want to control the **generalization error**:

$$\boldsymbol{R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)}$$

Example: linear model with Lipschitz loss
$\hat{Y} \mapsto \ell(\hat{Y}, Y))$ L-Lipschitz, $\ell(\hat{Y}, Y) \in [0, b]$
$f_\theta(X) = X^\intercal\theta$, $\|X\| \le 1$, $\|\theta\| \le B$

$$\sup_\theta \left|R(\theta) - \hat{R}(\theta, S^n)\right| = O\Big(\frac{LB}{\sqrt{n}} + b\sqrt{\frac{\log(1/\delta)}{n}}\Big) \qquad \text{w.p. } \ge 1-\delta$$

Both bounds can be proved using Rademacher complexity, and give minimax optimal rates when algorithm is empirical risk minimizer $\hat{\theta}_{\mathrm{ERM}} \in \arg\min_{\theta\in\Theta} \hat{R}(\theta, S^n)$.

Need for Algorithm-dependent Analysis Tools

Convex Lipschitz losses:

$\theta \mapsto \ell(f_\theta(X), Y))$ convex and L-Lipschitz

$\|\theta\| \leq B$

Algorithm-independent Bounds Not Good Enough:

- Uniform bounds on generalization error not good enough to analyse all algorithms even in this simple setting:
- Gradient Descent with right tuning of step size η and large enough number of iterations $T = \Omega(n^2)$ is known to achieve optimal rate.
- But need algorithm-dependent bound on generalization error, which depends on η, to prove it: (Bassily et al., 2020; Haghifam et al., 2023)

$$\mathbb{E}\left[R(\hat{\theta}_{\mathrm{GD}}) - \hat{R}(\hat{\theta}_{\mathrm{GD}}, S^n)\right] \leq 4L^2\sqrt{T}\eta + \frac{4L^2 T\eta}{n}$$

Better for smaller η! (Depends on algorithm.)

No dependence on B!

Algorithm-dependent Control via Mutual Information

Bound with **mutual information** (Catoni, 2007; Russo and Zou, 2016):

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \lesssim \sqrt{\frac{I(\hat{\theta}; S^n)}{n}}$$

Algorithm-dependent Control via Mutual Information

Bound with **mutual information** (Catoni, 2007; Russo and Zou, 2016):

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \lesssim \sqrt{\frac{I(\hat{\theta}; S^n)}{n}}$$

Refined to **conditional mutual information** via symmetrization with a ghost sample (Steinke and Zakynthinou, 2020):

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \lesssim \sqrt{\frac{\mathrm{CMI}(\mathrm{Alg})}{n}}$$

Algorithm-dependent Control via Mutual Information

Bound with **mutual information** (Catoni, 2007; Russo and Zou, 2016):

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \lesssim \sqrt{\frac{I(\hat{\theta}; S^n)}{n}}$$

Refined to **conditional mutual information** via symmetrization with a ghost sample (Steinke and Zakynthinou, 2020):

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \lesssim \sqrt{\frac{\mathrm{CMI}(\mathrm{Alg})}{n}}$$

- Promising: information-theoretic techniques achieved **numerically non-vacuous estimates** for stochastic gradient Langevin dynamics in non-trivial deep learning benchmarks (Negrea et al., 2019)

Algorithm-dependent Control via Mutual Information

Bound with **mutual information** (Catoni, 2007; Russo and Zou, 2016):

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \lesssim \sqrt{\frac{I(\hat{\theta}; S^n)}{n}}$$

Refined to **conditional mutual information** via symmetrization with a ghost sample (Steinke and Zakynthinou, 2020):

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \lesssim \sqrt{\frac{\mathrm{CMI}(\mathrm{Alg})}{n}}$$

- Promising: information-theoretic techniques achieved **numerically non-vacuous estimates** for stochastic gradient Langevin dynamics in non-trivial deep learning benchmarks (Negrea et al., 2019)
- But CMI and other information-theoretic complexity measures **insufficient to prove non-vacuous generalization bound** for GD case from previous slide (Haghifam et al., 2023)

Control via Rademacher Complexity

$$R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n) \leq \sup_{\theta \in \Theta} \Big(R(\theta) - \hat{R}(\theta, S^n) \Big) \qquad (*)$$

Lemma (Algorithm-independent upper bound)

$$\mathbb{E}\left[\sup_{\theta \in \Theta} \left(R(\theta) - \hat{R}(\theta, S^n) \right) \right] \leq 2 \, \mathbb{E}_{S^n}[\mathbf{Rad}(\Theta, S^n)]$$

and, with probability at least $1 - \delta$,

$$\sup_{\theta \in \Theta} \left(R(\theta) - \hat{R}(\theta, S^n) \right) \leq 2 \, \mathbb{E}_{S^n}[\mathbf{Rad}(\Theta, S^n)] + b \sqrt{\frac{\log(2/\delta)}{2n}}$$

Empirical Rademacher complexity:

$$\mathrm{Rad}(\Theta, S^n) = \frac{1}{n} \, \mathbb{E}_{\sigma}[\sup_{\theta \in \Theta} \sum_{i=1}^{n} \sigma_i \ell(f_\theta(X_i), Y_i)],$$

where $\sigma = (\sigma_1, \ldots, \sigma_n)$ with $\Pr(\sigma_i = -1) = \Pr(\sigma_i = +1) = 1/2$.

Outline

Context: Big Interest in New Complexity Measures Because of Deep Learning

New Algorithm-dependent Complexity Measure

Algorithm-dependent Rademacher Complexity

Lemma (Algorithm-dependent upper bound)

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \leq 2 \mathop{\mathbb{E}}_{S_-^n, S_+^n} [\mathrm{Rad}(\hat{\Theta}^{\boldsymbol{n}}, S_+^n)]$$

Algorithm-dependent Rademacher Complexity

$$\hat{\Theta}^n := \left\{ \mathrm{Alg}(S^n_\sigma) : \sigma \in \{-1,+1\}^n \right\} \subset \Theta.$$

$$\left.\begin{aligned} S^n_- &= (Z_1^{-1}, \ldots, Z_n^{-1}) \\ S^n_+ &= (Z_1^{+1}, \ldots, Z_n^{+1}) \end{aligned}\right\} \quad S^n_\sigma = (Z_1^{\sigma_1}, \ldots, Z_n^{\sigma_n})$$

Lemma (Algorithm-dependent upper bound)

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \leq 2 \mathop{\mathbb{E}}_{S^n_-, S^n_+} [\mathrm{Rad}(\hat{\Theta}^n, S^n_+)]$$

- Like normal Rademacher bound, but with $\hat{\Theta}^n$ instead of Θ
- Symmetrization with ghost sample S^n_- like CMI
- Proof: similar to standard proof, but upper bound $\hat{\theta}$ by supremum over θ later, after symmetrization

Algorithm-dependent Rademacher Complexity

$$\hat{\Theta}^n := \left\{ \mathrm{Alg}(S^n_\sigma) : \sigma \in \{-1,+1\}^n \right\} \subset \Theta.$$

$$\left.\begin{aligned} S^n_- &= (Z_1^{-1}, \ldots, Z_n^{-1}) \\ S^n_+ &= (Z_1^{+1}, \ldots, Z_n^{+1}) \end{aligned}\right\} \quad S^n_\sigma = (Z_1^{\sigma_1}, \ldots, Z_n^{\sigma_n})$$

Lemma (Algorithm-dependent upper bound)

$$\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)] \leq 2 \mathop{\mathbb{E}}_{S^n_-, S^n_+} [\mathrm{Rad}(\hat{\Theta}^n, S^n_+)]$$

and, with probability at least $1-\delta$,

$$R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n) \leq 4 \operatorname*{ess\,sup}_{S^n_-, S^n_+} \mathrm{Rad}(\hat{\Theta}^n, S^n_+) + b\sqrt{\frac{8\log(2/\delta)}{n}}$$

- Refines special case of a result by Foster et al. (2019)

Consequences 1: Generalization for SGD

Greatly **simplified proof** of result by Park, Şimşekli, and Erdogdu (2022):

Stochastic Gradient Descent (SGD) with constant step size η:

$$\hat{\theta}_{t+1} = \hat{\theta}_t - \eta \nabla \ell(f_{\hat{\theta}_t}(X_{i_t}), Y_{i_t}) \qquad (i_t \sim \mathrm{Unif}[1, \ldots, n])$$

Consequences 1: Generalization for SGD

Greatly **simplified proof** of result by Park, Şimşekli, and Erdogdu (2022):

Stochastic Gradient Descent (SGD) with constant step size η:

$$\hat{\theta}_{t+1} = \hat{\theta}_t - \eta \nabla \ell(f_{\hat{\theta}_t}(X_{i_t}), Y_{i_t}) \qquad (i_t \sim \mathrm{Unif}[1, \ldots, n])$$

Suppose $\theta \mapsto \ell(f_\theta(X), Y)$:

- α-**strongly convex**
- β-**smooth**
- L-Lipschitz
- $\|\theta\| \leq B$

Theorem

For $\eta \in (0, \beta)$ with **any random or non-random choice of the i_t**, *w.p. $\geq 1 - \delta$*

$$R(\hat{\theta}_{T+1}) - \hat{R}(\hat{\theta}_{T+1}, S^n) = O\Big(\sqrt{\frac{\log(Bn)}{n \log(\frac{1}{\gamma})}} + \sqrt{\frac{\log(1/\delta)}{n}} + \frac{L}{n}\Big),$$

where $\gamma = \sqrt{1 - 2\alpha\eta + \alpha\beta\eta^2}$.

Consequences 2: Properties Like CMI

Generalization for VC Classes:

For binary classification:

$$\mathrm{Rad}(\hat{\Theta}^n, S_+^n) \leq \mathrm{Rad}(\Theta, S_+^n) = O\Big(\sqrt{\frac{\mathrm{VCdim}(\mathcal{F})}{n}}\Big)$$

Generalization for compression schemes:

If Alg is a k-compression scheme: $\mathrm{Alg}(S^n) = \mathrm{Alg}_2(\mathrm{Alg}_1(S^n))$, where Alg_1 maps any sample S^n to a subsample S^k of size k. Then

$$\mathrm{Rad}(\hat{\Theta}^n, S_+^n) = O\Big(\sqrt{\frac{k \log n}{n}}\Big)$$

Consequences 3: Topological Bounds

Define the (random) set $\hat{\Theta} := \bigcup_{n=1}^{\infty} \hat{\Theta}^n$

Minkowski dimension: $\overline{\dim}_{\mathcal{M}}(\hat{\Theta}) = \limsup_{\delta \to 0^+} \dfrac{\log \mathrm{Cover}(\hat{\Theta}, \|\cdot\|, \delta)}{\log(1/\delta)}$

Theorem

Suppose $\ell(f_\theta(X), Y)$ is Lipschitz continuous in θ. Then

$$\limsup_{n \to \infty} \frac{\mathbb{E}[R(\hat{\theta}) - \hat{R}(\hat{\theta}, S^n)]}{\sqrt{\log(n)/n}} \leq b\sqrt{2\,\mathbb{E}[\overline{\dim}_{\mathcal{M}}(\hat{\Theta})]}.$$

- Avoids bad I_∞ term (much larger than regular mutual information) from previous topological bounds (Simsekli et al., 2020)

Summary

Algorithm-dependent Rademacher complexity:

- Rademacher complexity of algorithm- and data-dependent set $\hat{\Theta}^n$ controls generalization error

Consequences:

1. Greatly simplified proof of an algorithm-dependent generalization bound for SGD
2. Generalization for VC classes and compression schemes (like CMI)
3. New topological generalization bounds

References I

S. Sachs, T. van Erven, L. Hodgkinson, R. Khanna, U. Şimşekli **Generalization Guarantees via Algorithm-dependent Rademacher Complexity**, Conference on Learning Theory (COLT), 2023.

Other references:

Anthony, Martin and Peter L. Bartlett (2009). *Neural Network Learning: Theoretical Foundations*. 1st. USA: Cambridge University Press. ISBN: 052111862X.

Bassily, Raef, Vitaly Feldman, Cristóbal Guzmán, and Kunal Talwar (2020). "Stability of Stochastic Gradient Descent on Nonsmooth Convex Losses". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., pp. 4381–4391.

Belkin, Mikhail, Daniel Hsu, Siyuan Ma, and Soumik Mandal (2019). "Reconciling modern machine-learning practice and the classical bias-variance trade-off". In: *Proceedings of the National Academy of Sciences* 116.32, pp. 15849–15854. DOI: 10.1073/pnas.1903070116. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1903070116.

Catoni, Olivier (2007). *Pac-Bayesian Supervised Classification: The Thermodynamics of Statistical Learning*. Lecture Notes — Monograph Series, Volume 56.

References II

Foster, Dylan J., Spencer Greenberg, Satyen Kale, Haipeng Luo, Mehryar Mohri, and Karthik Sridharan (2019). "Hypothesis Set Stability and Generalization". In: *Advances in Neural Information Processing Systems 32*.

Haghifam, Mahdi, Borja Rodríguez-Gálvez, Ragnar Thobaben, Mikael Skoglund, Daniel M. Roy, and Gintare Karolina Dziugaite (20 Feb–23 Feb 2023). "Limitations of Information-Theoretic Generalization Bounds for Gradient Descent Methods in Stochastic Convex Optimization". In: *Proceedings of The 34th International Conference on Algorithmic Learning Theory*. Ed. by Shipra Agrawal and Francesco Orabona. Vol. 201. Proceedings of Machine Learning Research. PMLR, pp. 663–706.

Negrea, Jeffrey, Mahdi Haghifam, Gintare Karolina Dziugaite, Ashish Khisti, and Daniel M Roy (2019). "Information-Theoretic Generalization Bounds for SGLD via Data-Dependent Estimates". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc.

Park, Sejun, Umut Şimşekli, and Murat A. Erdogdu (2022). "Generalization Bounds for Stochastic Gradient Descent via Localized ε-Covers". In: *Advances in Neural Information Processing Systems*.

Russo, Daniel and James Zou (2016). "Controlling bias in adaptive data analysis using information theory". In: *Artificial Intelligence and Statistics*.

References III

Simsekli, Umut, Ozan Sener, George Deligiannidis, and Murat A Erdogdu (2020). "Hausdorff dimension, heavy tails, and generalization in neural networks". In: *Advances in Neural Information Processing Systems 33*.

Steinke, Thomas and Lydia Zakynthinou (2020). "Reasoning About Generalization via Conditional Mutual Information". In: *COLT*.

Zhang, Chiyuan, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals (Feb. 2021). "Understanding deep learning (still) requires rethinking generalization". In: *Commun. ACM* 64.3, pp. 107–115. ISSN: 0001-0782. DOI: 10.1145/3446776.