Some New Insights on Regularization and Interpolation Motivated from Neural Networks

Tengyuan Liang

Econometrics and Statistics

CHICAGO BOOTH

The University of Chicago Booth School of Business

Outline

Generative Adversarial Networks

- statistical rates
- pair regularization
- optimization

Interpolation

- regularization?
- kernel ridgeless regression
- GD on two layers ReLU networks

Outline

Generative Adversarial Networks (unsupervised)

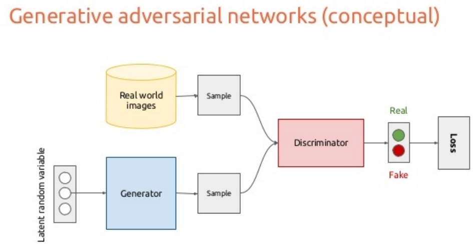
- statistical rates
- pair regularization
- optimization

Interpolation (supervised)

- regularization?
- kernel ridgeless regression
- GD on two layers ReLU networks

GANs

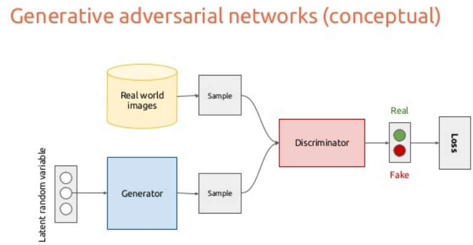
Generative adversarial networks

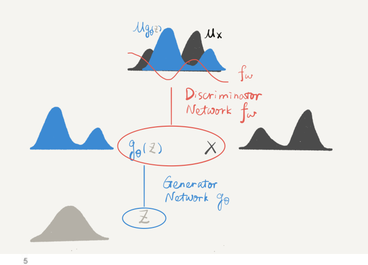


- GAN Goodfellow et al. (2014)
- WGAN Arjovsky et al. (2017); Arjovsky and Bottou (2017)
- MMD GAN Li, Swersky, and Zemel (2015); Dziugaite, Roy, and Ghahramani (2015); Arbel, Sutherland, Bińkowski, and Gretton (2018)
- f-GAN Nowozin, Cseke, and Tomioka (2016)
- Sobolev GAN Mroueh et al. (2017)
- many others... Liu, Bousquet, and Chaudhuri (2017); Tolstikhin, Gelly, Bousquet, Simon-Gabriel, and Schölkopf (2017)

5

Generative adversarial networks




$\mu_{g_\theta(z)}$
μ_x
f_ω
Discriminator Network f_ω
$g_\theta(Z)$
X
Generator Network g_θ
Z
5

Generator g_θ, Discriminator f_ω

$$U(\theta, \omega) = \mathop{\mathbb{E}}_{X \sim \mathcal{P}_{\text{real}}} h_1(f_\omega(X)) - \mathop{\mathbb{E}}_{Z \sim \mathcal{P}_{\text{input}}} h_2(f_\omega(g_\theta(Z)))$$

$$\min_{\theta} \max_{\omega} \; U(\theta, \omega)$$

GANs are widely used in practice, however

MUCH NEEDS TO BE UNDERSTOOD, IN THEORY

- Approximation:

  what dist. can be approximated by the generator $g_\theta(Z)$?

- **Statistical**:

  given n samples, what is the statistical/generalization error rate?

- Computational:

  local convergence for practical optimization, how to stablize?

- Landscape:

  are local saddle points good globally?

Formulation

$\mathcal{D}_G$ dist. class by generator, $\mathcal{F}_D$ func. class by discriminator, ν target dist.

population $$\mu_* := \operatorname*{arg\,min}_{\mu \in \mathcal{D}_G} \max_{f \in \mathcal{F}_D} \mathop{\mathbb{E}}_{Y \sim \mu} f(Y) - \mathop{\mathbb{E}}_{X \sim \nu} f(X)$$

FORMULATION

$\mathcal{D}_G$ dist. class by generator, $\mathcal{F}_D$ func. class by discriminator, ν target dist.

population $$\mu_* := \arg\min_{\mu \in \mathcal{D}_G} \max_{f \in \mathcal{F}_D} \mathop{\mathbb{E}}_{Y \sim \mu} f(Y) - \mathop{\mathbb{E}}_{X \sim \nu} f(X)$$

$\widehat{\nu}^n$ empirical dist.

empirical $$\widehat{\mu}_n := \arg\min_{\mu \in \mathcal{D}_G} \max_{f \in \mathcal{F}_D} \mathop{\mathbb{E}}_{Y \sim \mu} f(Y) - \mathop{\mathbb{E}}_{X \sim \widehat{\nu}^n} f(X)$$

FORMULATION

$\mathcal{D}_G$ dist. class by generator, $\mathcal{F}_D$ func. class by discriminator, ν target dist.

population $$\mu_* := \arg\min_{\mu\in\mathcal{D}_G}\max_{f\in\mathcal{F}_D} \mathop{\mathbb{E}}_{Y\sim\mu} f(Y) - \mathop{\mathbb{E}}_{X\sim\nu} f(X)$$

$\widehat{\nu}^n$ empirical dist.

empirical $$\widehat{\mu}_n := \arg\min_{\mu\in\mathcal{D}_G}\max_{f\in\mathcal{F}_D} \mathop{\mathbb{E}}_{Y\sim\mu} f(Y) - \mathop{\mathbb{E}}_{X\sim\widehat{\nu}^n} f(X)$$

- Density learning/estimation: long history nonparametric statistics target density $\nu \in W^\alpha$ - Sobolev space with smoothness $\alpha \geq 0$

  Stone (1982); Nemirovski (2000); Tsybakov (2009); Wassermann (2006)

- GAN statistical theory is needed

  Arora and Zhang (2017); Arora et al. (2017a,b); Liu et al. (2017)

DISCRIMINATOR METRIC

Define the critic metric (IPM)

$$d_{\mathcal{F}}(\mu, \nu) := \sup_{f \in \mathcal{F}} \mathop{\mathbb{E}}_{Y \sim \mu} f(Y) - \mathop{\mathbb{E}}_{X \sim \nu} f(X) \ .$$

Discriminator metric

Define the critic metric (IPM)

$$d_{\mathcal{F}}(\mu, \nu) := \sup_{f \in \mathcal{F}} \mathop{\mathbb{E}}_{Y \sim \mu} f(Y) - \mathop{\mathbb{E}}_{X \sim \nu} f(X) \; .$$

- $\mathcal{F}$ Lip-1: Wasserstein metric d_W
- $\mathcal{F}$ bounded by 1: total variation/Radon metric d_{TV}
- RKHS $\mathcal{H}$, $\mathcal{F} = \{f \in \mathcal{H}, \|f\|_{\mathcal{H}} \leq 1\}$: MMD GAN
- $\mathcal{F}$ Sobolev smoothness β: Sobolev GAN

Statistical question: statistical error rate with n-i.i.d samples, $\mathbb{E}\, d_{\mathcal{F}}(\nu, \widehat{\mu}_n)$?

Minimax optimal rates: Sobolev GAN

Consider the target density $\nu \in \mathcal{G} = W^{\alpha}$ Sobolev space with smoothness $\alpha > 0$, and the evaluation metric $\mathcal{F} = W^{\beta}$ with smoothness $\beta > 0$.

Theorem (L. '17 & L. '18, Sobolev).

The **minimax optimal rate** is

$$\inf_{\widetilde{\nu}_n} \sup_{\nu \in \mathcal{G}} \mathbb{E}\, d_{\mathcal{F}}(\nu, \widetilde{\nu}_n) \asymp n^{-\frac{\alpha+\beta}{2\alpha+d}} \vee n^{-\frac{1}{2}} \; .$$

Here $\widetilde{\nu}_n$ any estimator based on n samples. d-dim.

Mair and Ruymgaart (1996); Liang (2017); Singh et al. (2018)

MINIMAX OPTIMAL RATES: MMD GAN

Consider a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$

- integral operator $\mathcal{T}$ with eigenvalue decay $t_i \asymp i^{-\kappa}$, $0 < \kappa < \infty$
- evaluation metric $\mathcal{F} = \{f \in \mathcal{H} \mid \|f\|_{\mathcal{H}} \leq 1\}$
- target density $\nu(x)$ in $\mathcal{G} = \{\nu \mid \|\mathcal{T}^{-(\alpha-1)/2}\nu\|_{\mathcal{H}} \leq 1\}$ with smoothness $\alpha > 0$

Theorem (L. '18, RKHS)**.**

The **minimax optimal rate** is

$$\inf_{\widetilde{\nu}_n} \sup_{\nu \in \mathcal{G}} \mathbb{E}\, d_{\mathcal{F}}(\nu, \widetilde{\nu}_n) \precsim n^{-\frac{(\alpha+1)\kappa}{2\alpha\kappa+2}} \vee n^{-\frac{1}{2}} \ .$$

MINIMAX OPTIMAL RATES: MMD GAN

Consider a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$

- integral operator $\mathcal{T}$ with eigenvalue decay $t_i \asymp i^{-\kappa}$, $0 < \kappa < \infty$
- evaluation metric $\mathcal{F} = \{f \in \mathcal{H} \mid \|f\|_{\mathcal{H}} \leq 1\}$
- target density $\nu(x)$ in $\mathcal{G} = \{\nu \mid \|\mathcal{T}^{-(\alpha-1)/2}\nu\|_{\mathcal{H}} \leq 1\}$ with smoothness $\alpha > 0$

Theorem (L. '18, RKHS)**.**

The **minimax optimal rate** is

$$\inf_{\widetilde{\nu}_n} \sup_{\nu \in \mathcal{G}} \mathbb{E}\, d_{\mathcal{F}}(\nu, \widetilde{\nu}_n) \precsim n^{-\frac{(\alpha+1)\kappa}{2\alpha\kappa+2}} \vee n^{-\frac{1}{2}} .$$

$\kappa > 1$: intrinsic dim. $\sum_{i \geq 1} t_i = \sum_{i \geq 1} i^{-\kappa} \leq C$, parametric rate $n^{-\frac{(\alpha+1)\kappa}{2\alpha\kappa+2}} \vee n^{-\frac{1}{2}} = n^{-1/2}$.

$\kappa < 1$: sample complexity scales $n = \epsilon^{2+\frac{2}{\alpha+1}\left(\frac{1}{\kappa}-1\right)}$, "effective dim." $1/\kappa$.

Oracle inequality

Generator class $\mathcal{D}_G$ may not contain the target density ν: oracle approach.

Let $\mathcal{D}_G$ be any generator class. The discriminator metric $\mathcal{F}_D = W^\beta$, target density $\nu \in W^\alpha$.

Corollary (L. '18)**.**

With empirical density $\widehat{\nu}^n(x)$ as plug-in, the GAN estimator

$$\widehat{\mu}_n \in \operatorname*{arg\,min}_{\mu \in \mathcal{D}_G} \max_{f \in \mathcal{F}_D} \left\{ \int f(x)\mu(x)dx - \int f(x)\widehat{\nu}^n(x)dx \right\},$$

attains a **sub-optimal rate**

$$\mathbb{E}\, d_{\mathcal{F}_D}(\widehat{\mu}_n, \nu) \leq \min_{\mu \in \mathcal{D}_G} d_{\mathcal{F}_D}(\mu, \nu) + \boxed{n^{-\frac{\beta}{d}} \vee \frac{\log n}{\sqrt{n}}}.$$

Oracle inequality

Generator class $\mathcal{D}_G$ may not contain the target density ν: oracle approach.

Let $\mathcal{D}_G$ be any generator class. The discriminator metric $\mathcal{F}_D = W^\beta$, target density $\nu \in W^\alpha$.

Corollary (L. '18).

With empirical density $\widehat{\nu}^n(x$
in, the GAN estimator

$$\widehat{\mu}_n \in \operatorname*{arg\,min}_{\mu \in \mathcal{D}_G} \max_{f \in \mathcal{F}_D} \left\{ \int f(x)\mu(\right.$$

attains a **sub-optimal rate**

$$\mathbb{E}\, d_{\mathcal{F}_D}(\widehat{\mu}_n, \nu) \leq \min_{\mu \in \mathcal{D}_G} d_{\mathcal{F}_D}(\mu, \nu) + \boxed{n^{-\frac{\beta}{d}} \vee \frac{\log n}{\sqrt{n}}}.$$

Corollary (L. '18).

In contrast, a smoothed/regularized empirical density $\widetilde{\nu}^n(x)$ as plug-in

$$\widetilde{\mu}_n \in \operatorname*{arg\,min}_{\mu \in \mathcal{D}_G} \max_{f \in \mathcal{F}_D} \left\{ \int f(x)\mu(x)dx - \int f(x)\widetilde{\nu}^n(x)dx \right\},$$

a **faster rate** is attainable

$$\mathbb{E}\, d_{\mathcal{F}_D}(\widetilde{\mu}_n, \nu) \leq \min_{\mu \in \mathcal{D}_G} d_{\mathcal{F}_D}(\mu, \nu) + \boxed{n^{-\frac{\alpha+\beta}{2\alpha+d}} \vee \frac{1}{\sqrt{n}}}.$$

Canas and Rosasco (2012)

Sub-optimality and Regularization

Regularization helps achieve faster rate!

however, notions of regularization/complexity is yet understood well for neural nets...

Use $\tilde{\nu}_n$ "smoothed" empirical estimate, that serves as regularization

For example, kernel smoothing - $\widetilde{\nu}^n(x) = \frac{1}{nh_n} K\left(\frac{x-x_i}{h_n}\right)$

practice: SGD still carries through, as sample from $\tilde{\nu}_n$ is easy as Gaussian mixtures

Turns out, this is used in practice, called "instance noise" or "data augmentation"

Sønderby et al. (2016); Liang et al. (2017); Arjovsky and Bottou (2017); Mescheder et al. (2018)

Parametric results and pair regularization

Consider the parametrized GAN estimator

$$\widehat{\theta}_{m,n} \in \mathop{\arg\min}_{\theta: g_\theta \in \mathcal{G}} \max_{\omega: f_\omega \in \mathcal{F}} \left\{ \widehat{\mathbb{E}}_m f_\omega(g_\theta(Z)) - \widehat{\mathbb{E}}_n f_\omega(X) \right\},$$

where m and n denote the number of the generator samples and real samples.

Generalized oracle inequality

$$\text{approx. err.} \quad A_1(\mathcal{F}, \mathcal{G}, \nu) := \sup_{\theta} \inf_{\omega} \left\| \log \frac{\nu}{\mu_\theta} - f_\omega \right\|_\infty, \quad A_2(\mathcal{G}, \nu) := \inf_{\theta} \left\| \log \frac{\mu_\theta}{\nu} \right\|_\infty^{1/2},$$

$$\text{sto. err.} \quad S_{n,m}(\mathcal{F}, \mathcal{G}) := \sqrt{\mathrm{Pdim}(\mathcal{F}) \left(\frac{\log m}{m} \vee \frac{\log n}{n} \right)} \vee \sqrt{\mathrm{Pdim}(\mathcal{F} \circ \mathcal{G}) \frac{\log m}{m}},$$

$\mathrm{Pdim}(\cdot)$ the pseudo-dimension of the neural network function.

Theorem (L. '18).

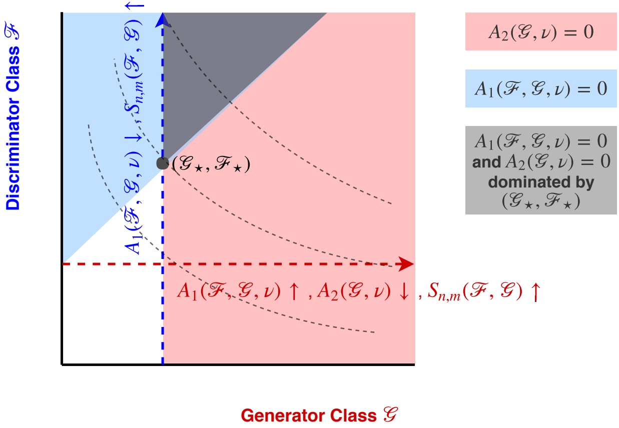
$$\mathbb{E}\, d_{TV}^2 \left(\nu, \mu_{\widehat{\theta}_{m,n}} \right), \mathbb{E}\, d_W^2 \left(\nu, \mu_{\widehat{\theta}_{m,n}} \right), \mathbb{E}\, d_{KL} \left(\nu \| \mu_{\widehat{\theta}_{m,n}} \right) + \mathbb{E}\, d_{KL} \left(\mu_{\widehat{\theta}_{m,n}} \| \nu \right)$$
$$\leq A_1(\mathcal{F}, \mathcal{G}, \nu) + A_2(\mathcal{G}, \nu) + S_{n,m}(\mathcal{F}, \mathcal{G}) \quad .$$

We emphasize on the interplay between $(\mathcal{G}, \mathcal{F})$ as a **pair** of tuning parameters for **regularization**.

Pair regularization

for instance, one simple form of the interplay is:

fix $\mathcal{G}$, as $\mathcal{F}$ increase : $A_1(\mathcal{F},\mathcal{G},\nu)$ decrease, $A_2(\mathcal{G},\nu)$ constant, $S_{n,m}(\mathcal{F},\mathcal{G})$ increase,
fix $\mathcal{F}$, as $\mathcal{G}$ increase : $A_1(\mathcal{F},\mathcal{G},\nu)$ increase, $A_2(\mathcal{G},\nu)$ decrease, $S_{n,m}(\mathcal{F},\mathcal{G})$ increase.



Discriminator Class $\mathscr{F}$

$A_1(\mathscr{F},\mathscr{G},\nu)\downarrow$, $S_{n,m}(\mathscr{F},\mathscr{G})\uparrow$

$(\mathscr{G}_\star,\mathscr{F}_\star)$

$A_1(\mathscr{F},\mathscr{G},\nu)\uparrow$, $A_2(\mathscr{G},\nu)\downarrow$, $S_{n,m}(\mathscr{F},\mathscr{G})\uparrow$

Generator Class $\mathscr{G}$

$A_2(\mathscr{G},\nu)=0$

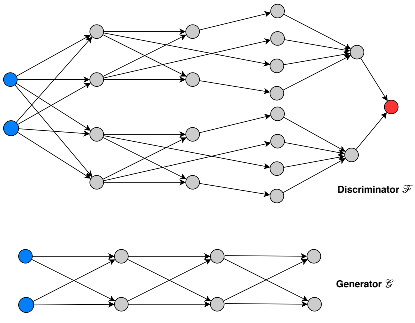
$A_1(\mathscr{F},\mathscr{G},\nu)=0$

$A_1(\mathscr{F},\mathscr{G},\nu)=0$ and $A_2(\mathscr{G},\nu)=0$ dominated by $(\mathscr{G}_\star,\mathscr{F}_\star)$

Applications of pair regularization

Application I: parametric rates for leaky ReLU networks

When the generator $\mathcal{G}$ and discriminator $\mathcal{F}$ are both leaky ReLU networks with depth L (width properly chosen depends on dimension).



When the target density is realizable by the generator.

$$\log \mu_\theta(x) = c_1 \sum_{l=1}^{L-1} \sum_{i=1}^{d} \mathbf{1}_{m_{li}(x) \geq 0} + c_0,$$

Bai et al. (2018)

Application I: parametric rates for leaky ReLU networks

When the generator $\mathcal{G}$ and discriminator $\mathcal{F}$ are both leaky ReLU networks with depth L (width properly chosen depends on dimension).

Theorem (L. ’18, leaky ReLU)**.**

$$\mathbb{E}\, d_{TV}^2\left(\nu, \mu_{\widehat{\theta}_{m,n}}\right) \precsim \sqrt{d^2 L^2 \log(dL) \left(\frac{\log m}{m} \vee \frac{\log n}{n}\right)}.$$

The results hold for very deep networks with depth $L = o(\sqrt{n/\log n})$.

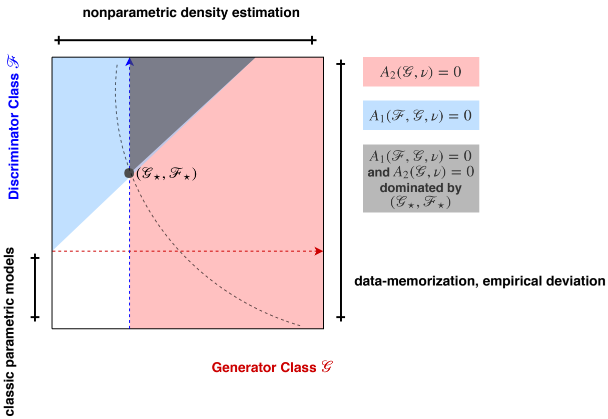
Application II: learning multivariate Gaussian

Corollary (L. '18, Gaussian)**.**

GANs enjoy near optimal sampling complexity (w.r.t. dim. d), with proper choices of the architecture and activation,

$$\mathbb{E}\, d_{TV}^2 \left(\nu, \mu_{\widehat{\theta}_{m,n}}\right) \precsim \sqrt{\frac{d^2 \log d}{n \wedge m}}.$$

Pair regularization: why GANs might be better



nonparametric density estimation

Discriminator Class $\mathscr{F}$

classic parametric models

Generator Class $\mathscr{G}$

$(\mathscr{G}_\star, \mathscr{F}_\star)$

data-memorization, empirical deviation

🟥 $A_2(\mathscr{G}, \nu) = 0$

🟦 $A_1(\mathscr{F}, \mathscr{G}, \nu) = 0$

▨ $A_1(\mathscr{F}, \mathscr{G}, \nu) = 0$ and $A_2(\mathscr{G}, \nu) = 0$ dominated by $(\mathscr{G}_\star, \mathscr{F}_\star)$

Optimization: local convergence

Formulation

Generator g_θ, Discriminator f_ω

$$U(\theta,\omega) = \mathop{\mathbb{E}}_{X\sim\mathcal{P}_{\text{real}}} h_1(f_\omega(X)) - \mathop{\mathbb{E}}_{Z\sim\mathcal{P}_{\text{input}}} h_2(f_\omega(g_\theta(Z)))$$

$$\min_{\theta}\max_{\omega}\ U(\theta,\omega)$$

- global optimization for general $U(\theta,\omega)$ is hard Singh et al. (2000); Pfau and Vinyals (2016); Salimans et al. (2016)

Local saddle point (θ_*,ω_*) such that no incentive to deviate locally

$$U(\theta_*,\omega) \le U(\theta_*,\omega_*) \le U(\theta,\omega_*)\ ,$$

for (θ,ω) in an open neighborhood of (θ_*,ω_*).

- also called local Nash Equilibrium (NE)
- modest goal: initialized properly, algorithm converges to a local NE

MAIN MESSAGE: INTERACTION MATTERS

Exponential local convergence to **stable equilibrium**

However, “interaction term” matters, slows down the convergence ⇐ curse

What if unstable? turns out “interaction term” matters, utilize it renders exponential convergence ⇐ blessing

Main message: interaction matters

Exponential local convergence to **stable equilibrium**
analog to GD in single-player optimization, strongly convex case
intuitive picture: discrete-time SGA cycles inward to a stable equilibrium fast

However, "interaction term" matters, slows down the convergence $\Leftarrow$ curse
compared to conventional GD, strongly convex case, due to presence of $\nabla_{\theta\omega} U \nabla_{\theta\omega} U^T$
also we show a lower bound on T_{SGA} to show the **curse** is necessary

What if unstable? turns out "interaction term" matters, utilize it renders exponential convergence $\Leftarrow$ blessing

- SGA fails, modify the dynamics to utilize interaction
- analog to single-player optimization, non-strongly convex case, is surprising
  - single-player: first order methods **cannot obtain error better than** $1/T^2$ in smooth, but non-strongly convex case, classic result Nemirovski and Yudin (1983); Nesterov (2013)
  - two-player: we will show first order method can obtain **exponential convergence** to **unstable equilibrium** $\exp(-cT)$

> *"However, no guarantees are known beyond the convex-concave setting and, more importantly for the paper, even in convex-concave games, no guarantees are known for the last-iterate pair."*
>
> — Daskalakis, Ilyas, Syrgkanis, and Zeng (2017)

Exponential convergence to unstable equilibrium

OMD proposed in Daskalakis et al. (2017)

$$\theta_{t+1} = \theta_t - 2\eta\nabla_\theta U(\theta_t, \omega_t) + \boxed{\eta\nabla_\theta U(\theta_{t-1}, \omega_{t-1})}$$
$$\omega_{t+1} = \omega_t + 2\eta\nabla_\omega U(\theta_t, \omega_t) - \boxed{\eta\nabla_\omega U(\theta_{t-1}, \omega_{t-1})}$$

For bi-linear game $U(\theta, \omega) = \theta^T C\omega$, to obtain ϵ-close solution

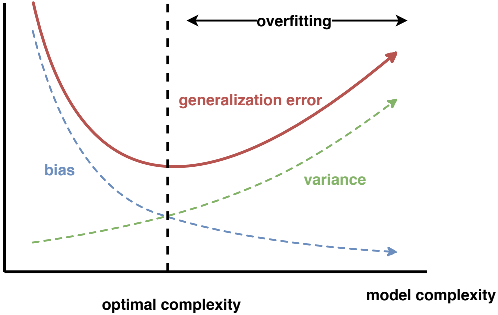
shown in Daskalakis et al. (2017) : $T \gtrsim \boxed{\epsilon^{-4}\log\frac{1}{\epsilon}} \cdot \text{Poly}\left(\frac{\lambda_{\max}(CC^T)}{\lambda_{\min}(CC^T)}\right)$

Corollary (L. & Stokes, '18)**.**

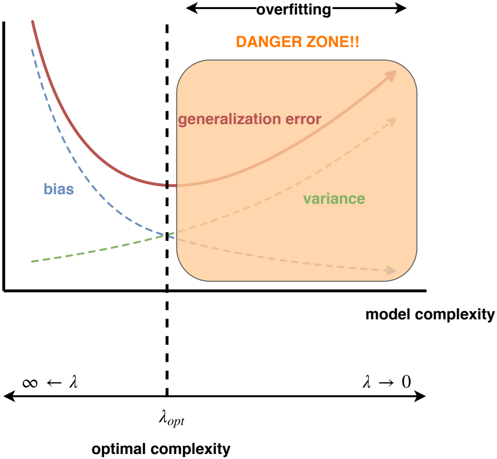
we show : $T \gtrsim \boxed{\log\frac{1}{\epsilon}} \cdot \frac{\lambda_{\max}(CC^T)}{\lambda_{\min}(CC^T)}$

Interpolation

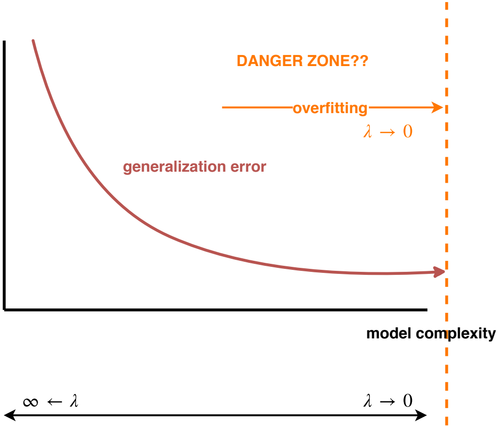
How do we teach stat/ml?



HOW DO WE TEACH STAT/ML?



Is this really what's happening in practice?

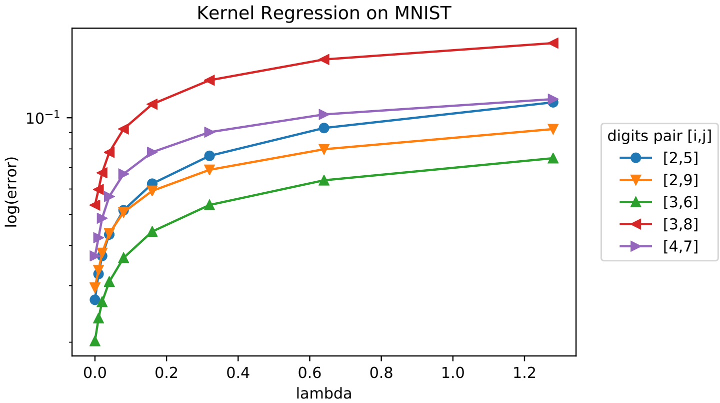


Is explicit regularization λ_{opt} really needed?

Is explicit regularization λ_{opt} really needed?

Is interpolation really bad for statistics and machine learning?

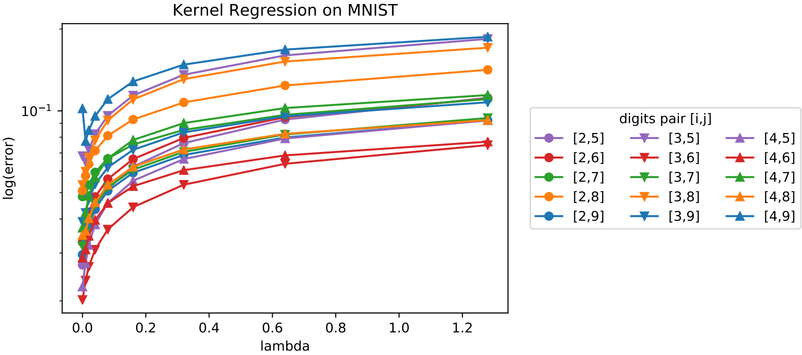
An empirical example



$\lambda = 0$: the interpolated solution, perfect fit on training data.

MNIST data from LeCun et al. (2010)

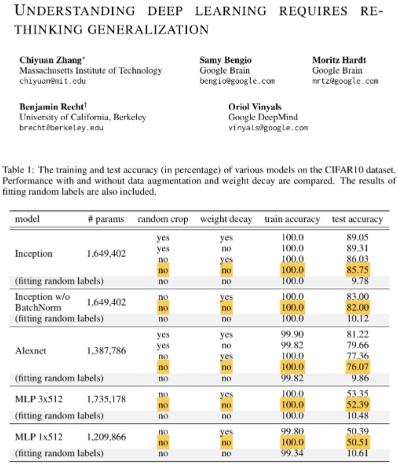
An empirical example



λ = 0: the interpolated solution, perfect fit on training data.

MNIST data from LeCun et al. (2010)

Isolated phenomenon? No



Understanding deep learning requires rethinking generalization

Chiyuan Zhang*
Massachusetts Institute of Technology
chiyuan@mit.edu

Samy Bengio
Google Brain
bengio@google.com

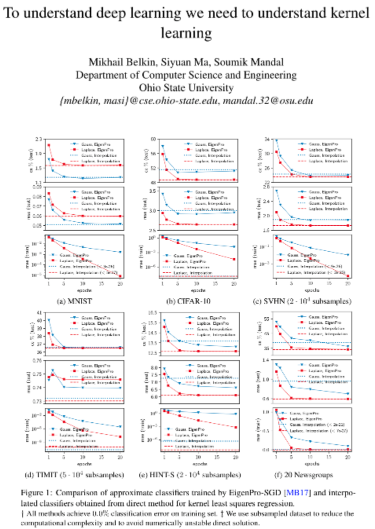
Moritz Hardt
Google Brain
mrtz@google.com

Benjamin Recht†
University of California, Berkeley
brecht@berkeley.edu

Oriol Vinyals
Google DeepMind
vinyals@google.com

Table 1: The training and test accuracy (in percentage) of various models on the CIFAR10 dataset. Performance with and without data augmentation and weight decay are compared. The results of fitting random labels are also included.

model	# params	random crop	weight decay	train accuracy	test accuracy
Inception	1,649,402	yes	yes	100.0	89.05
		yes	no	100.0	89.31
		no	yes	100.0	86.03
		no	no	100.0	85.75
(fitting random labels)		no	no	100.0	9.78
Inception w/o BatchNorm	1,649,402	no	yes	100.0	83.00
		no	no	100.0	82.00
(fitting random labels)		no	no	100.0	10.12
Alexnet	1,387,786	yes	yes	99.90	81.22
		yes	no	99.82	79.66
		no	yes	100.0	77.36
		no	no	100.0	76.07
(fitting random labels)		no	no	99.82	9.86
MLP 3x512	1,735,178	no	yes	100.0	53.35
		no	no	100.0	52.39
(fitting random labels)		no	no	100.0	10.48
MLP 1x512	1,209,866	no	yes	99.80	50.39
		no	no	100.0	50.51
(fitting random labels)		no	no	99.34	10.61



To understand deep learning we need to understand kernel learning

Mikhail Belkin, Siyuan Ma, Soumik Mandal
Department of Computer Science and Engineering
Ohio State University
{mbelkin, masi}@cse.ohio-state.edu, mandal.32@osu.edu

(a) MNIST (b) CIFAR-10 (c) SVHN ($2 \cdot 10^4$ subsamples)

(d) TIMIT ($5 \cdot 10^4$ subsamples) (e) HINT-S ($2 \cdot 10^4$ subsamples) (f) 20 Newsgroups

Figure 1: Comparison of approximate classifiers trained by EigenPro-SGD [MB17] and interpolated classifiers obtained from direct method for kernel least squares regression.
| All methods achieve 0.0% classification error on training set. ‡ We use subsampled dataset to reduce the computational complexity and to avoid numerically unstable direct solution.

- Methodology: deep learning, kernel learning, boosting, random forests . . . Zhang, Bengio, Hardt, Recht, and Vinyals (2016); Wyner, Olson, Bleich, and Mease (2017); Maennel, Bousquet, and Gelly (2018)
- Datasets: MNIST, CIFAR-10, others Belkin, Ma, and Mandal (2018b)

PUZZLES

Interpolated solutions performs very well in practice
for many (modern) methodology and datasets!

What is happening? "Overfitting" is not that bad . . .

OUR MESSAGE

Geometric properties of the data design X, **high dimensionality**, and **curvature of the kernel** ⇒ interpolated solution generalizes.

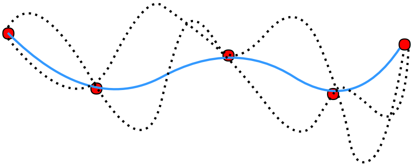
Potential theory in statistics/learning for interpolated solution:

- Analysis through explicit regularization (✗)
- Capacity control (✗)
- Early stopping (algorithmic) (✗)
- Stability analysis (algorithmic) (✗)
- Nonparametric smoothing analysis (✗)
- Inductive bias (✓?) at least promising

Belkin, Hsu, and Mitra (2018a)

Inductive bias

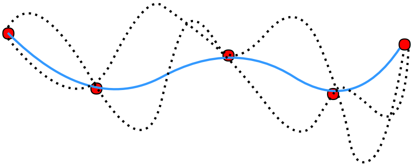
There are many functions that behave exactly the same on training data, but the method/algo. prefers certain functions



Zhang et al. (2016); Neyshabur et al. (2017); Liang et al. (2017); Belkin et al. (2018b)

INDUCTIVE BIAS

There are many functions that behave exactly the same on training data, but the method/algo. prefers certain functions



- kernels/RKHS: Representer Thm., min norm interpolation
- over-parametrized linear regression: 0-initialization, min norm interpolation
- matrix factorization, etc. Gunasekar, Woodworth, Bhojanapalli, Neyshabur, and Srebro (2017); Li, Ma, and Zhang (2017)
- two layers ReLU network Maennel, Bousquet, and Gelly (2018)

- Inductive bias (✓?), at least promising

Zhang et al. (2016); Neyshabur et al. (2017); Liang et al. (2017); Belkin et al. (2018b)

History: interpolation rules

Understudied in the literature: especially when there is label noise

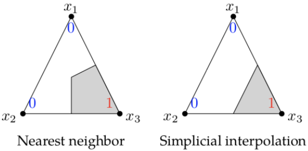
Recent progress on local/direct interpolation schemes:

- Geometric simplicial interpolation and weighted kNN Belkin, Hsu, and Mitra (2018a)
- Nonparametric Nadaraya-Watson estimator with singular kernels Shepard (1968); Devroye, Györfi, and Krzyżak (1998); Belkin, Rakhlin, and Tsybakov (2018c)

Simplicial interpolation

Belkin, Hsu, and Mitra (2018a) showed under regularity conditions, simplicial interpolation $\widehat{f}_n$

$$\limsup_{n\to\infty} \mathbb{E}(\widehat{f}_n(\mathbf{x}) - f_*(\mathbf{x}))^2 \leq \frac{2}{d+2}\,\mathbb{E}(f_*(\mathbf{x}) - \mathbf{y})^2$$



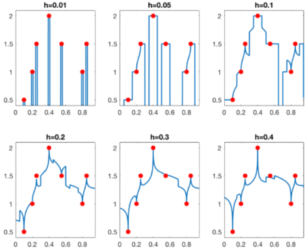
SINGULAR KERNEL

Shepard (1968); Devroye et al. (1998); Belkin et al. (2018c) showed for singular kernels

$$K(u) := \|u\|^{-a}\mathbf{I}\{\|u\| \le 1\} \Rightarrow \lim_{u\to 0} K(u) = \infty$$

the Nadaraya-Watson estimator $\widehat{f_n} = \frac{\sum_{i=1}^n \mathbf{y}_i K(\frac{x-\mathbf{x}_i}{h})}{\sum_{i=1}^n K(\frac{x-\mathbf{x}_i}{h})}$ achieves the optimal error when f_* lies in Hölder space with smoothness β

$$\mathbb{E}(\widehat{f_n}(\mathbf{x}) - f_*(\mathbf{x}))^2 \sim n^{-\frac{2\beta}{2\beta+d}}$$



Global/inverse interpolation methods (kernel machines/neural networks/boosting) performs better than the local interpolation schemes empirically.

Several conjectures have been made about global/inverse interpolation methods, such as kernel machines in Belkin, Hsu, and Mitra (2018a), two layers ReLU nets Maennel, Bousquet, and Gelly (2018).

Interpolated min-norm solution for kernel ridge regression

PROBLEM FORMULATION

Given n i.i.d. pairs (x_i, y_i) drawn from unknown μ: x_i are d-dim covariates in $\Omega \subset \mathbb{R}^d$, $y_i \in \mathbb{R}$ are the response/labels.

To estimate

$$f_*(x) = \mathbb{E}(\mathbf{y}|\mathbf{x} = x),$$

which is assumed to lie in a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$ with kernel $K(\cdot, \cdot)$.

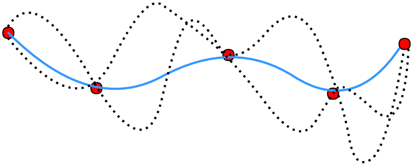
Smola and Schölkopf (1998); Wahba (1990); Shawe-Taylor and Cristianini (2004)

Conventional wisdom: Kernel Ridge Regression, explicit regularization $\lambda \neq 0$ added when $\mathcal{H}$ is high- or infinite-dimensional

$$\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|_{\mathcal{H}}^2 \quad .$$

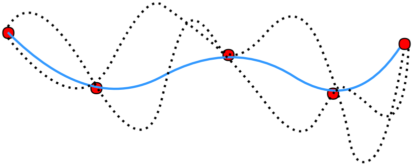
We study min-norm interpolation estimator $\widehat{f}$

$$\widehat{f} := \arg\min_{f \in \mathcal{H}} \|f\|_{\mathcal{H}}, \quad \text{s.t. } f(x_i) = y_i, \ \forall i \leq n \quad .$$



We study min-norm interpolation estimator $\widehat{f}$

$$\widehat{f} := \operatorname*{arg\,min}_{f \in \mathcal{H}} \|f\|_{\mathcal{H}}, \quad \text{s.t. } f(x_i) = y_i, \ \forall i \leq n \quad .$$



Equivalently

$$\widehat{f}(x) = K(x, X) K(X, X)^{-1} Y$$

when $K(X, X) \in \mathbb{R}^{n \times n}$ is invertible.

Look for **adaptive** "data-dependent" bounds $\phi_{n,d}(X, f_*)$ to understand when and why interpolated estimator $\widehat{f}$ generalizes.

We provide high-probability bounds on

$$\begin{aligned} \text{integrated squared risk} \quad & \mathbb{E}(\widehat{f}(\mathbf{x}) - f_*(\mathbf{x})) \leq \phi_{n,d}(X, f_*) \\ \text{generalization error} \quad & \mathbb{E}(\widehat{f}(\mathbf{x}) - \mathbf{y})^2 - \mathbb{E}(f_*(\mathbf{x}) - \mathbf{y})^2 \leq \phi_{n,d}(X, f_*) \end{aligned}$$

Assumptions

(A.1) High-dim: $c \leq d/n \leq C$
$\Sigma_d = \mathbb{E}_\mu[\mathbf{x}\mathbf{x}^*]$ satisfies $\|\Sigma_d\| \leq C$ and $\mathrm{Tr}(\Sigma_d)/d \geq c$.

(A.2) $(8+m)$-moments: $z_i := \Sigma_d^{-1/2} x_i$
each entries of z_i are i.i.d. mean zero, with bounded $(8+m)$-moments.

(A.3) Noise: $\mathbb{E}[(f_*(\mathbf{x}) - \mathbf{y})^2 | \mathbf{x} = x] \leq \sigma^2$ for all $x \in \Omega$.

(A.4) Non-linear kernel: for a non-linear smooth function $h(\cdot)$

$$K(x, x') = h\left(\frac{1}{d}\langle x, x' \rangle\right)$$

Define the following quantities related to curvature of $h(\cdot)$

$$\alpha := h(0) + h''(0)\frac{\mathrm{Tr}(\Sigma_d^2)}{d^2}, \quad \beta := h'(0),$$
$$\gamma := h\left(\frac{\mathrm{Tr}(\Sigma_d)}{d}\right) - h(0) - h'(0)\frac{\mathrm{Tr}(\Sigma_d)}{d}.$$

Main results

Theorem (L. & Rakhlin, '18).

Define

$\phi_{n,d}(X, f_*) :=$

$$\frac{8\sigma^2 \|\Sigma_d\|}{d} \sum_j \frac{\lambda_j\left(\frac{XX^*}{d} + \frac{\alpha}{\beta} 11^*\right)}{\left[\frac{\gamma}{\beta} + \lambda_j\left(\frac{XX^*}{d} + \frac{\alpha}{\beta} 11^*\right)\right]^2} + \|f_*\|_{\mathcal{H}}^2 \inf_{0 \le k \le n} \left\{ \frac{1}{n} \sum_{j>k} \lambda_j(K_X K_X^*) + 2M\sqrt{\frac{k}{n}} \right\}$$

Main results

Theorem (L. & Rakhlin, '18).

Define

$\phi_{n,d}(X, f_*) :=$

$$\frac{8\sigma^2 \|\Sigma_d\|}{d} \sum_j \frac{\lambda_j\left(\frac{XX^*}{d} + \frac{\alpha}{\beta} 11^*\right)}{\left[\frac{\gamma}{\beta} + \lambda_j\left(\frac{XX^*}{d} + \frac{\alpha}{\beta} 11^*\right)\right]^2} + \|f_*\|_{\mathcal{H}}^2 \inf_{0 \le k \le n} \left\{ \frac{1}{n} \sum_{j > k} \lambda_j(K_X K_X^*) + 2M\sqrt{\frac{k}{n}} \right\}$$

Under (A.1)-(A.4), with prob. $1 - 2\delta - d^{-2}$, interpolation estimator $\widehat{f}$

$$\begin{array}{c} \mathbb{E}_{Y|X} \|\widehat{f} - f_*\|_{L^2_\mu}^2 \\ \mathbb{E}_{Y|X} \left\{ \mathbb{E}(\widehat{f}(\mathbf{x}) - \mathbf{y})^2 - \mathbb{E}(f_*(\mathbf{x}) - \mathbf{y})^2 \right\} \end{array} \le \phi_{n,d}(X, f_*) + \epsilon(n, d).$$

The remainder term $\epsilon(n, d) = O(d^{-\frac{m}{8+m}} \log^{4.1} d) + O(n^{-\frac{1}{2}} \log^{0.5}(n/\delta))$.

Main Message

Geometric properties of the data design X**, high dimensionality**, and **curvature of the kernel** ⇒ interpolated solution generalizes.

$$\phi_{n,d}(X, f_*) :=$$

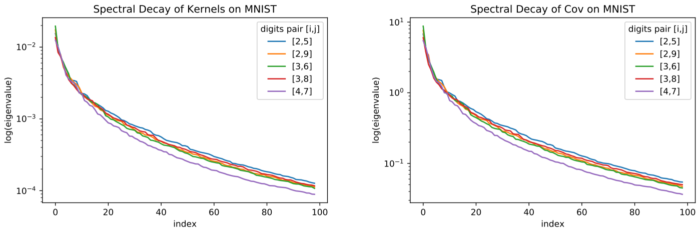
$$\frac{8\sigma^2\|\Sigma_d\|}{d}\sum_j \frac{\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)}{\left[\frac{\gamma}{\beta}+\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)\right]^2} + \|f_*\|_{\mathcal{H}}^2 \inf_{0\le k\le n}\left\{\frac{1}{n}\sum_{j>k}\lambda_j(K_X K_X^*) + 2M\sqrt{\frac{k}{n}}\right\}$$

Proof is **different** from classic RKHS analysis with explicit regularization.

Geometric properties of design

Geometric properties of the data design X, high dimensionality, and curvature of the kernel ⇒ interpolated solution generalizes.

$$\frac{8\sigma^2\|\Sigma_d\|}{d}\sum_j \frac{\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)}{\left[\frac{\gamma}{\beta}+\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)\right]^2} + \|f_*\|_{\mathcal{H}}^2 \inf_{0\le k\le n}\left\{\frac{1}{n}\sum_{j>k}\lambda_j(K_X K_X^*) + 2M\sqrt{\frac{k}{n}}\right\}$$



Spectral decay of $\lambda_j(K_X K_X^*)$ and $\lambda_j\left(\frac{XX^*}{d}\right)$

Preferable geometric properties of the design. Not all design works!

HIGH DIMENSIONALITY

Geometric properties of the data design X, high dimensionality, and curvature of the kernel ⇒ interpolated solution generalizes.

$$\frac{8\sigma^2 \|\Sigma_d\|}{d} \sum_j \frac{\lambda_j \left(\frac{XX^*}{d} + \frac{\alpha}{\beta} 11^* \right)}{\left[\frac{\gamma}{\beta} + \lambda_j \left(\frac{XX^*}{d} + \frac{\alpha}{\beta} 11^* \right) \right]^2} + \|f_*\|_{\mathcal{H}}^2 \inf_{0 \le k \le n} \left\{ \frac{1}{n} \sum_{j > k} \lambda_j (K_X K_X^*) + 2M \sqrt{\frac{k}{n}} \right\}$$

High dimensionality

Geometric properties of the data design X, high dimensionality, and curvature of the kernel ⇒ interpolated solution generalizes.

$$\frac{8\sigma^2\|\Sigma_d\|}{d}\sum_j \frac{\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)}{\left[\frac{\gamma}{\beta}+\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)\right]^2} + \|f_*\|_{\mathcal{H}}^2 \inf_{0\le k\le n}\left\{\frac{1}{n}\sum_{j>k}\lambda_j(K_XK_X^*)+2M\sqrt{\frac{k}{n}}\right\}$$

Scalings:

- $c < d/n < C$, typical high-dim scaling in RMT, El Karoui (2010); Johnstone (2001)
- scaling: $K(x,x') = h(\langle x,x'\rangle/d)$, default choice for high dim. data in computing packages, e.g. Scikit-learn Pedregosa et al. (2011)
- bounds work for large (d,n) regime,
  $\epsilon(n,d) = O(d^{-m/(8+m)}\log^{4.1} d) + O(n^{-1/2}\log^{0.5}(n/\delta))$

Blessings of high dimensionality:

- similar effect observed in local/direct interpolating schemes in Belkin et al. (2018a) for simplicial interpolation and weighted kNN
- Kernel “ridgeless” regression is a global/inverse interpolation scheme

Curvature and implicit regularization

Geometric properties of the data design X, high dimensionality, and curvature of the kernel ⇒ interpolated solution generalizes.

$$\frac{8\sigma^2\|\Sigma_d\|}{d}\sum_j \frac{\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)}{\left[\frac{\gamma}{\beta}+\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)\right]^2} + \|f_*\|_{\mathcal{H}}^2 \inf_{0\le k\le n}\left\{\frac{1}{n}\sum_{j>k}\lambda_j(K_XK_X^*)+2M\sqrt{\frac{k}{n}}\right\}$$

Role of implicit regularization $\frac{\gamma}{\beta}\neq 0$: due to curvature/non-linearity of kernel

- the analysis is very different from that in explicit regularization in RKHS Caponnetto and De Vito (2007)
- borrow tools from recent development in RMT for kernel matrices in El Karoui (2010)

$$\text{effective dim}\begin{cases}\text{classic analysis:} & \sum_j \frac{\lambda_j}{\lambda_*+\lambda_j}\\ \text{our analysis:} & \sum_j \frac{\lambda_j}{(\lambda_*+\lambda_j)^2}\end{cases}$$

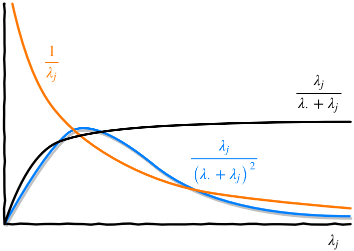
Curvature and implicit regularization

Geometric properties of the data design X, high dimensionality, and curvature of the kernel ⇒ interpolated solution generalizes.

$$\frac{8\sigma^2\|\Sigma_d\|}{d}\sum_j \frac{\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)}{\left[\frac{\gamma}{\beta}+\lambda_j\left(\frac{XX^*}{d}+\frac{\alpha}{\beta}11^*\right)\right]^2} + \|f_*\|_{\mathcal{H}}^2 \inf_{0\le k\le n}\left\{\frac{1}{n}\sum_{j>k}\lambda_j(K_X K_X^*)+2M\sqrt{\frac{k}{n}}\right\}$$

Role of implicit regularization $\frac{\gamma}{\beta}\neq 0$: due to curvature/non-linearity of kernel

effective dim $\begin{cases} \text{classic:} & \sum_j \frac{\lambda_j}{\lambda_*+\lambda_j} \\ \text{ours:} & \sum_j \frac{\lambda_j}{(\lambda_*+\lambda_j)^2} \\ \text{naive:} & \sum_j \frac{1}{\lambda_j} \end{cases}$



MAIN MESSAGE

Geometric properties of the data design X, **high dimensionality**, and **curvature of the kernel** ⇒ interpolated solution generalizes.

“implicit regularization” + “inductive bias”

"Explicit regularization may improve generalization performance, but is neither necessary nor by itself sufficient for controlling generalization error."

— Zhang, Bengio, Hardt, Recht, and Vinyals (2016)

Gradient descent on two layers ReLU Networks

Formulation

Two layers ReLU networks

$$f_t(x) = \sum_{i=1}^{m} w_i(t)\sigma(x^T u_i(t)).$$

with gradient descent (GD) on parameters $w_i(t), u_i(t)$

$$\frac{dw_i(t)}{dt} = -\mathbb{E}_{\mathbf{z}}\left[\frac{\partial l(\mathbf{y}, f(\mathbf{x}))}{\partial f}\sigma(\mathbf{x}^T u_i)\right]$$

$$\frac{du_i(t)}{dt} = -\mathbb{E}_{\mathbf{z}}\left[\frac{\partial l(\mathbf{y}, f(\mathbf{x}))}{\partial f} w_i \mathbf{1}_{\mathbf{x}^T u_i \geq 0}\mathbf{x}\right]$$

Initialization: m large, u_i random from uniform spherical dist. with

$$|w_i| = \|u_i\| = \frac{1}{\sqrt{m}}.$$

Algorithmic approximation: given $(\mathbf{x}, \mathbf{y})$, run GD with two layers ReLU networks, how $f_t(x)$ approximates $f_*(x) = \mathbb{E}(\mathbf{y}|\mathbf{x} = x)$? interpolates $\mathbf{y}$?

No further assumption on f_* besides it lies in L^2.

Maennel, Bousquet, and Gelly (2018)

View GD on ReLU network as dynamic kernels

Lemma (Dou & L. '18+).

$$d\mathbf{E}_{\mathbf{x}}\left[(f_*(\mathbf{x}) - f_t(\mathbf{x}))^2\right] = -2\mathbf{E}_{\mathbf{x},\tilde{\mathbf{x}}}\left[(f_*(\mathbf{x}) - f_t(\mathbf{x}))\boxed{\mathbf{K}_t(\mathbf{x},\tilde{\mathbf{x}})}(f_*(\tilde{\mathbf{x}}) - f_t(\tilde{\mathbf{x}}))\right]dt.$$

View NN as **fixed** kernel:

$$\lim_{m\to\infty} K_0(x,\tilde{x}) = 2\left[\frac{\pi - \arccos(t)}{\pi}t + \frac{\sqrt{1-t^2}}{2\pi}\right], \quad \text{where } t = \langle x, \tilde{x}\rangle$$

Rahimi and Recht (2008); Cho and Saul (2009); Daniely et al. (2016); Bach (2017)

We view NN as **dynamic** kernel! We provide mean-field approx. (as $m \to \infty$), PDE characterization (Distribution Dynamics) for ρ_t thus $\mathbf{K}_t$

Mei, Montanari, and Nguyen (2018); Rotskoff and Vanden-Eijnden (2018)

Representation Benefits

NN: data-dependent basis, an adaptive representation learned from data
Classic nonparametric: fixed basis from analysis, not adaptive to data

Heuristic justification, but what does it really mean mathematically?

Representation Benefits

NN: data-dependent basis, an adaptive representation learned from data
Classic nonparametric: fixed basis from analysis, not adaptive to data

Heuristic justification, but what does it really mean mathematically?

Follow GD dynamics to **any stationarity**, denote **corresp. RKHS** as $\mathcal{K}_\star$ (kernel $\mathbf{K}_\star$):

Theorem (Dou & L., '18+).

For any $f_*\in L^2_\mu$

- Function computed by GD on NN is proj. to RKHS $\mathcal{K}_\star$

$$\boxed{\lim_{t\to\infty} f_t^{GD} = \mathbf{H}_\star f_*} \in \mathcal{K}_\star$$

- Residual lies in a smaller space

$$residual := f_* - \lim_{t\to\infty} f_t^{GD}$$

$$residual \in \boxed{\mathcal{K}_{GD}^\perp \subset \mathcal{K}_\star^\perp}$$

$\mathbf{K}_\star$ is **adaptive** to f_*! **Gap in space: non-trivial decomposition.**

Interpolation Benefits

Running GD on NN is learning the **data-dependent kernel** and **performing least-squares (RKHS)** simultaneously.

The kernel is adaptive to task f_*, so the least squares proj. f_∞^{GD} lies in $\mathcal{K}_\star$, and the residual $f_* - f_\infty^{GD}$ is smaller.

Having an (trainable) additional layer serves as **"implicit regularization"** on $\widehat{\mathbf{K}}_t$, **faster interpolation**

more benefits and generalizations see Dou & L. '18+, to be posted.

CONCLUSION

- Minimax optimal rates **does not** explain the empirical success of neural networks.
- One needs new **adaptive** (to properties of data), **data-dependent** framework/understanding.
- Requires new insights on **regularization** and **interpolation**.

CONCLUSION

- Minimax optimal rates **does not** explain the empirical success of neural networks.
- One needs new **adaptive** (to properties of data), **data-dependent** framework/understanding.
- Requires new insights on **regularization** and **interpolation**.

image credit to Internet

Thank you!

Liang, T. (2018). — On How Well Generative Adversarial Networks Learn Densities: Nonparametric and Parametric Results. *available on arXiv:1811.03179* *under review*

Liang, T. & Rakhlin, A. (2018). — Just Interpolate: Kernel "Ridgeless" Regression Can Generalize. *available on arXiv:1808.00387* *revision invited*

Liang, T. & Stokes, J. (2018). — Interaction Matters: A Note on Non-asymptotic Local Convergence of Generative Adversarial Networks. *available on arXiv:1802.06132* *AISTATS 2019, to appear*

Liang, T., Poggio, T., Rakhlin, A. & Stokes, J. (2017). — Fisher-Rao Metric, Geometry, and Complexity of Neural Networks. *available on arXiv:1711.01530* *AISTATS 2019, to appear*

Dou, X. & **Liang, T.** (2018+). — Training Neural Networks as Learning Data-adaptive Kernels: Provable Representation and Approximation Benefits. *available on here*