Biases of gradient optimization in neural networks

▷ Mirror descent for wide and shallow nets

Guido Montúfar
montufar@math.ucla.edu

KIAS Center for AI and Natural Sciences 2024 Fall Workshop

UCLA

Max Planck Institute for Mathematics in the Sciences

NSF

DFG Deutsche Forschungsgemeinschaft German Research Foundation

Intuition

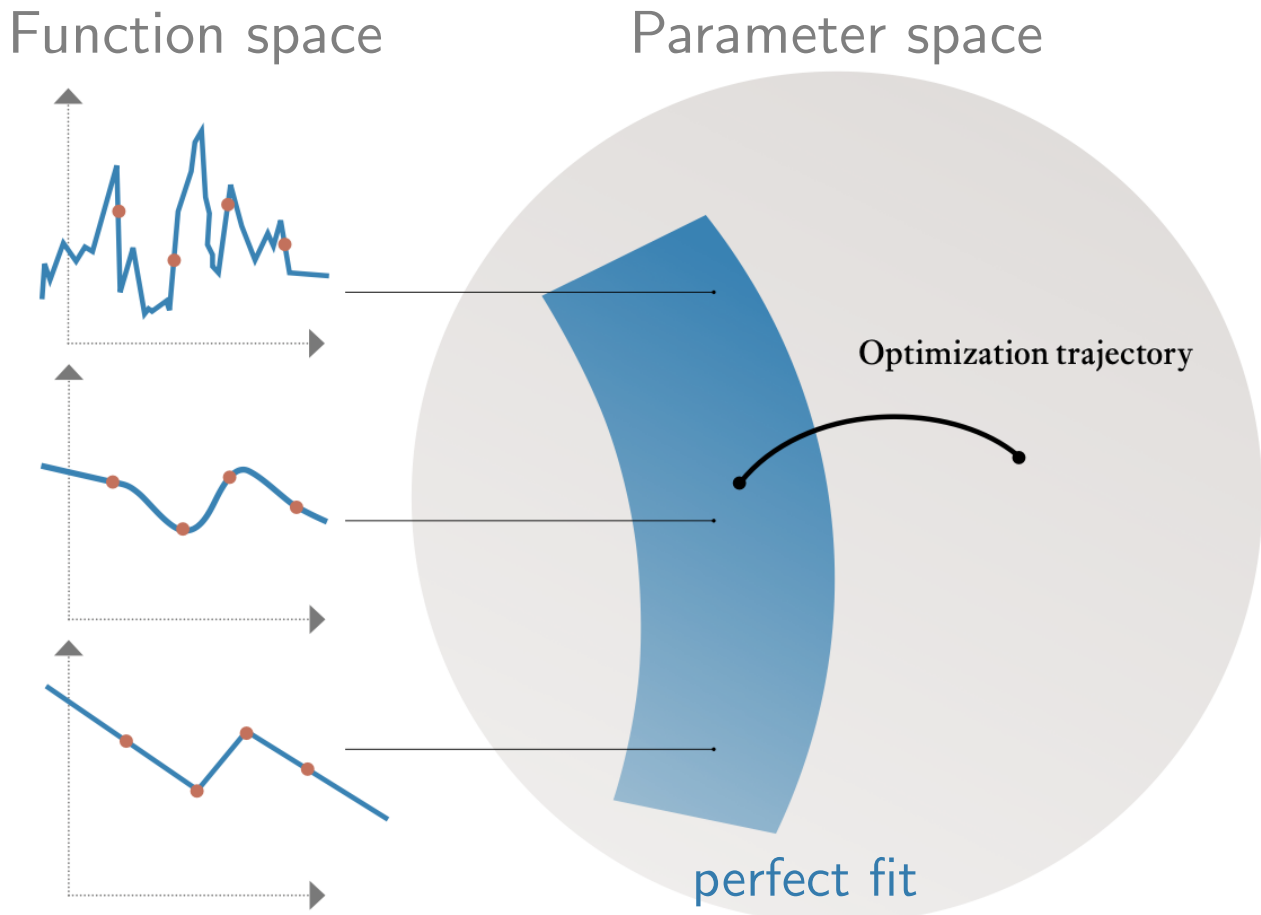


Figure 1: In an overparametrized model, there may be many different parameters and model functions that perfectly fit the training data

1. Implicit Bias and Algorithmic Regularization
2. Implicit Bias in Wide Shallow ReLU Networks
3. Implicit Bias of Mirror Descent for Wide Shallow Networks
4. Spectral Biases in Deep Nets

1. Implicit Bias and Algorithmic Regularization
2. Implicit Bias in Wide Shallow ReLU Networks
3. Implicit Bias of Mirror Descent for Wide Shallow Networks
4. Spectral Biases in Deep Nets

Motivation

- Neural networks are often overparameterized
  - Number of model parameters $>$ Number of training samples
  - Can fit random labels
- Many possible solutions fit the training data perfectly
  - Most of them generalize horribly
- Nevertheless, neural networks often generalize well, even without any explicit regularization
- The capacity of the model alone does not explain this

See Zhang et al. 2021

Motivation

- One possible explanation is that optimization algorithms are implicitly biased towards selecting simple solutions
- What kinds of solutions does an optimization algorithm return?
  - Examples: max margin classifier, smooth interpolation, sparse fct, ...
  - Depends on loss function, optimization algorithm, learning model, ...

Factors

- **Loss:** ℓ_2 loss (for regression); logistic loss (for classification)
- **Optimization algorithm:** gradient descent; stochastic gradient descent (SGD); mirror descent; steepest descent
- **Learning model:** linear models; linear neural networks; ReLU neural networks
- **Hyperparameters:** learning rate; initialization; mini-batch size

Gradient Descent

- Consider a linear model $f(x; w) = \langle x, w \rangle$ and a loss fct $\ell(\hat{y}, y)$.
- For training data $\{(x_i, y_i) : i = 1, \dots, n\}$ the training objective is

$$L(w) = \sum_{i=1}^{n} \ell(\langle x_i, w \rangle, y_i).$$

- Gradient descent iteration for initialization w_0 and step size η is

$$w_{t+1} = w_t - \eta \nabla L(w_t) = w_t - \eta \sum_{i=1}^{n} \nabla_1 \ell(\langle x_i, w_t \rangle, y_i) \cdot x_i.$$

Theorem 1 (Linear Model, Gunasekar et al. 2018a)

Consider a convex loss function ℓ with a unique finite minimizer, $\ell(\hat{y}, y) = 0$ iff $\hat{y} = y$. Assume that the gradient descent iteration converges to a global minimum of $L(w)$ with zero loss, i.e., $L(w_t) \to 0$. Then the algorithm returns the unique solution of following constrained optimization problem:

$$\min_w \|w - w_0\|_2 \quad s.t. \ \langle x_i, w \rangle = y_i, \quad i = 1, \dots, n. \tag{1}$$

Theorem 1 (Linear Model, Gunasekar et al. 2018a)

Consider a convex loss function ℓ with a unique finite minimizer, $\ell(\hat{y}, y) = 0$ iff $\hat{y} = y$. Assume that the gradient descent iteration converges to a global minimum of $L(w)$ with zero loss, i.e., $L(w_t) \to 0$. Then the algorithm returns the unique solution of following constrained optimization problem:

$$\min_w \|w - w_0\|_2 \quad s.t. \ \langle x_i, w \rangle = y_i, \quad i = 1, \dots, n. \tag{1}$$

The key idea is that the gradients are restricted to a n-dimensional subspace that is spanned by $\{x_i\}_{i=1}^n$ and is independent of w.

Given a strongly convex and differentiable potential ϕ, the mirror descent iteration is

$$w_{t+1} = \arg\min_{w} \eta\langle w, \nabla L(w_t)\rangle + D_\phi(w, w_t),$$

where $D_\phi(w, w') = \phi(w) - \phi(w') - \langle \nabla\phi(w'), w - w'\rangle$ is the Bregman divergence with respect to ϕ.

The first order optimality condition for the parameter update gives

$$\nabla\phi(w_{t+1}) = \nabla\phi(w_t) - \eta\nabla L(w_t).$$

Examples of ϕ:

- ℓ_2 norm: $\phi(w) = \frac{1}{2}\|w\|_2^2$, which leads to gradient descent;
- unnormalized negative entropy: $\phi(w) = \sum_i w_i \log w_i - w_i$.

Theorem 2 (Linear Model, Gunasekar et al. 2018a)

Consider any strongly convex potential ϕ. Assume that the mirror descent iteration converges to a global minimum of $L(w)$ with zero loss, i.e., $L(w_t) \to 0$. Then the algorithm returns the solution of following constrained optimization problem:

$$\min_w D_\phi(w, w_0) \quad s.t. \ \langle x_i, w \rangle = y_i, \quad i = 1, \dots, n. \tag{2}$$

Theorem 2 (Linear Model, Gunasekar et al. 2018a)

Consider any strongly convex potential ϕ. Assume that the mirror descent iteration converges to a global minimum of $L(w)$ with zero loss, i.e., $L(w_t) \to 0$. Then the algorithm returns the solution of following constrained optimization problem:

$$\min_w D_\phi(w, w_0) \quad s.t. \ \langle x_i, w \rangle = y_i, \quad i = 1, \dots, n. \tag{2}$$

The key idea is that $\nabla\phi(w_{t+1})$ (called dual iterates) are restricted to a n-dimensional manifold $\nabla\phi(w_0) + \text{span}(\{x_i\})$.

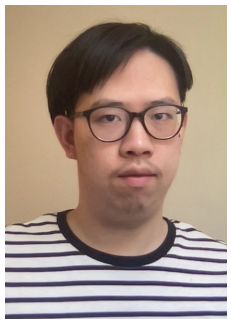
1 Implicit Bias and Algorithmic Regularization

2 Implicit Bias in Wide Shallow ReLU Networks

3 Implicit Bias of Mirror Descent for Wide Shallow Networks

4 Spectral Biases in Deep Nets

Hui Jin



(UCLA $\rightarrow$ Huawei)

- "Implicit Bias of Gradient Descent for Mean Squared Error Regression with Two-Layer Wide Neural Networks", JMLR 2023.

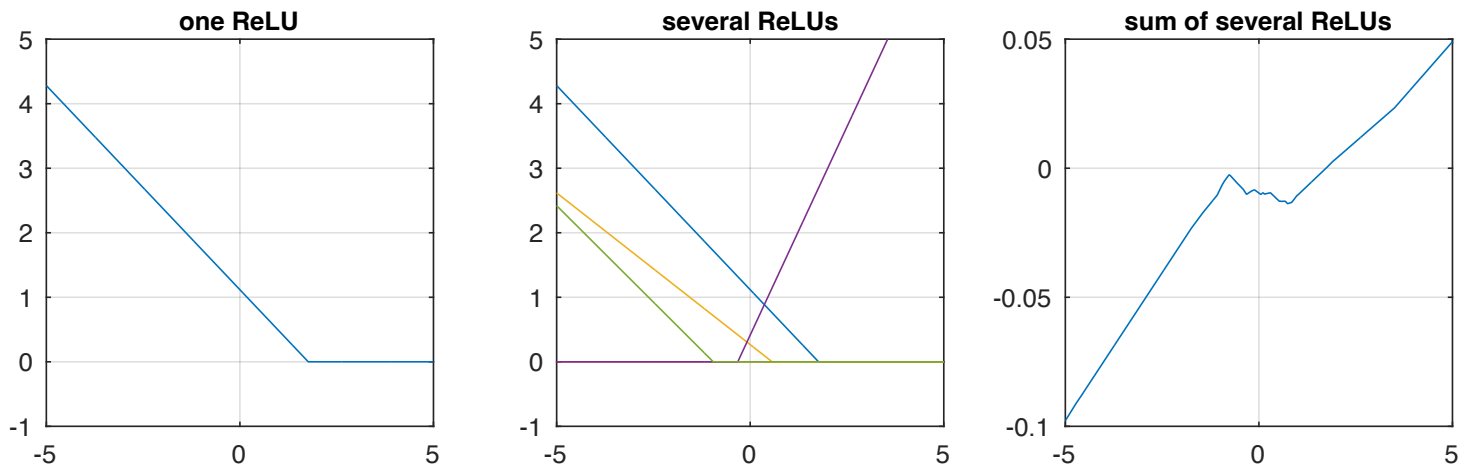
Regression with shallow ReLU net

- Consider a shallow ReLU network with n hidden units,

$$f(x,\theta) = \sum_{i=1}^{n} W_i^{(2)} [\langle W_i^{(1)}, x\rangle + b_i^{(1)}]_+ + b^{(2)}.$$

- Initialize the parameters by independent samples of $(\mathcal{W}, \mathcal{B})$.
- For data $\{(x_j, y_j)\}_{j=1}^M$, select a function by gradient descent minimization of the squared error $L(\theta) = \sum_{j=1}^M \|f(x_j,\theta) - y_j\|^2$.

- Consider first the univariate setting, $x \in \mathbb{R}$.
- A rectified linear unit $[w_i x + b_i]_+$ has breakpoint at $c_i = -b_i/w_i$.
- A weight-bias density $p_{\mathcal{W},\mathcal{B}}$ induces a breakpoint density $p_{\mathcal{C}}$.



Implicit bias of GD in wide ReLU networks

Theorem 3 (Univariate regression)

- *Consider a feedforward network with a single input unit, a hidden layer of n ReLUs, skip connections, and a single linear output unit.*
- *Assume standard parametrization and that for each hidden unit the input weight and bias are initialized from a sub-Gaussian* $(\mathcal{W}, \mathcal{B})$ *with joint density* $p_{\mathcal{W},\mathcal{B}}$.
- *Then, for any finite data set* $\{(x_j, y_j)\}_{j=1}^M$ *and sufficiently large n optimization of the MSE by full-batch gradient descent with sufficiently small step size converges to a parameter* θ^* *for which the output function* $f(x, \theta^*)$ *attains zero training loss.*
- *Moreover,*

Implicit bias of GD in wide ReLU networks

Theorem 3 (Univariate regression)

letting

$$\zeta(x) = \int_{\mathbb{R}} |W|^3 p_{\mathcal{W},\mathcal{B}}(W, -Wx)\, \mathrm{d}W$$

and $S = \mathrm{supp}(\zeta) \cap [\min_i x_j, \max_i x_j]$, *we have*

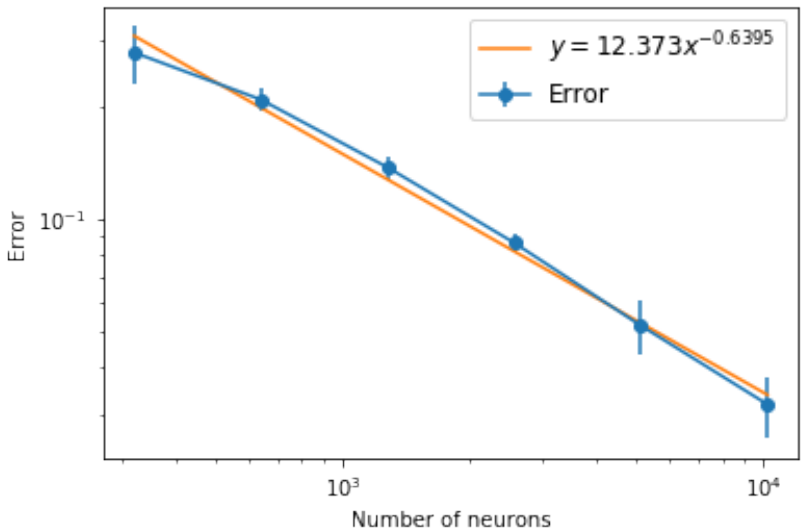
$$\|f(x,\theta^*) - g^*(x)\|_2 = O(n^{-\frac{1}{2}}), \quad x \in S$$

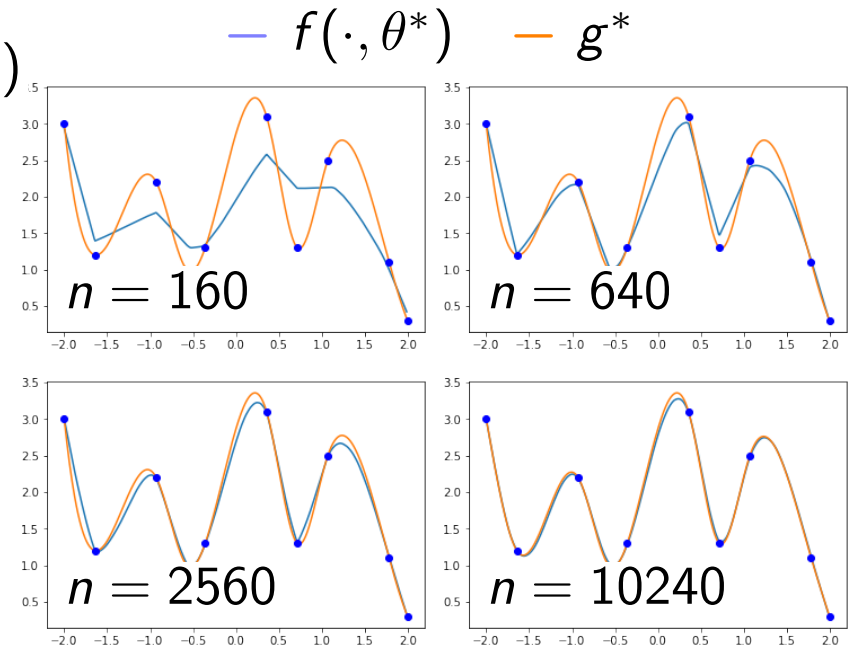
with high probability over the random initialization θ_0, *where* g^* *solves following variational problem:*

$$\begin{aligned} \min_{g \in C^2(S)} \quad & \int_S \frac{1}{\zeta(x)} (g''(x) - f''(x,\theta_0))^2 \, \mathrm{d}x \\ \text{subject to} \quad & g(x_j) = y_j, \quad j = 1, \ldots, M. \end{aligned} \tag{3}$$

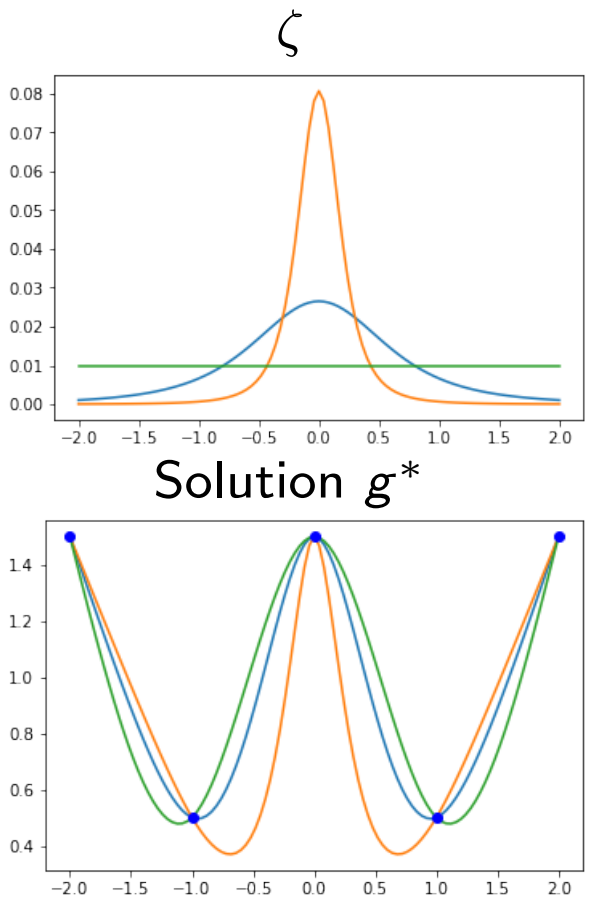
Numerical illustration

Uniform error between g^* and $f(\cdot, \theta^*)$





The curvature penalty function



- The curvature penalty is $1/\zeta(x) = 1/\int_{\mathbb{R}} |W|^3 p_{\mathcal{W},\mathcal{B}}(W, -Wx)\, \mathrm{d}W$.
- We obtain the explicit form of ζ for various initialization procedures.
- We obtain parameter initialization procedures leading to any desired ζ.

Explicit form of the curvature penalty

Theorem 4 (Curvature penalty for various initializations)

1. *Gaussian initialization. Assume that* $\mathcal{W}$ *and* $\mathcal{B}$ *are independent,* $\mathcal{W} \sim \mathcal{N}(0, \sigma_w^2)$ *and* $\mathcal{B} \sim \mathcal{N}(0, \sigma_b^2)$. *Then* ζ *is given by* $\zeta(x) = \frac{2\sigma_w^3 \sigma_b^3}{\pi(\sigma_b^2 + x^2\sigma_w^2)^2}$.
2. *Binary-uniform initialization. Assume that* $\mathcal{W}$ *and* $\mathcal{B}$ *are independent,* $\mathcal{W} \in \{-1, 1\}$ *and* $\mathcal{B} \sim \mathcal{U}(-a_b, a_b)$ *with* $a_b \geq L$. *Then* ζ *is constant on* $[-L, L]$.
3. *Uniform initialization. Assume that* $\mathcal{W}$ *and* $\mathcal{B}$ *are independent,* $\mathcal{W} \sim \mathcal{U}(-a_w, a_w)$ *and* $\mathcal{B} \sim \mathcal{U}(-a_b, a_b)$ *with* $\frac{a_b}{a_w} \geq L$. *Then* ζ *is constant on* $[-L, L]$.

Gaussian initialization

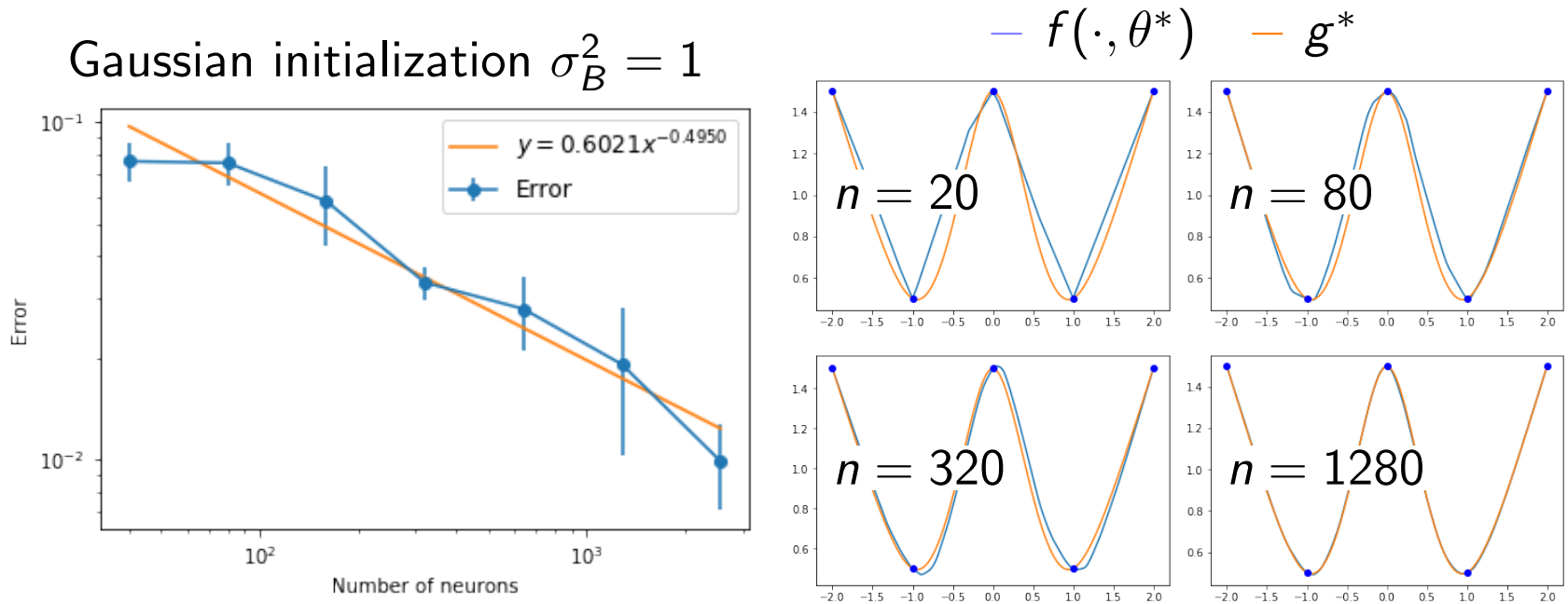


Figure 2: Initialization $W \sim N(0, 1)$ and $B \sim N(0, 1)$.

Sharp Gaussian initialization

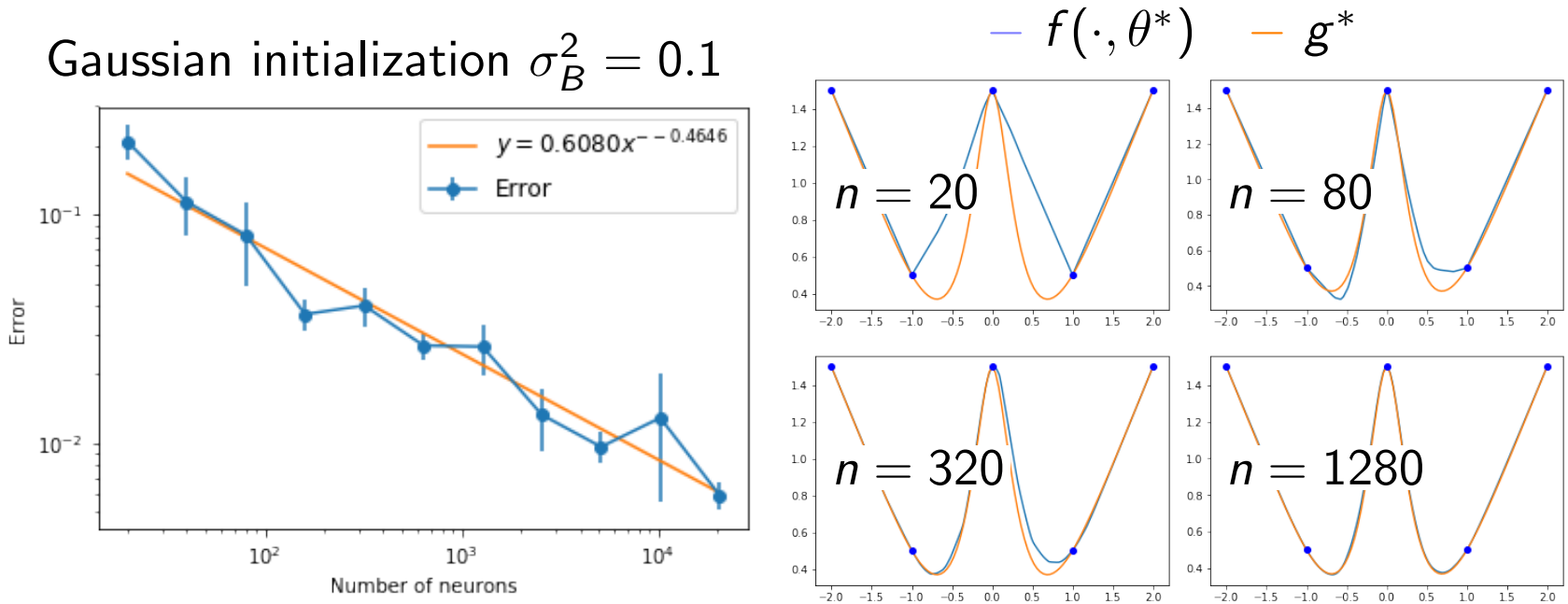


Figure 3: Initialization $W \sim N(0, 1)$ and $B \sim N(0, 0.1)$. In this case ζ that is more peaked at $x = 0$. Solutions more curvy around $x = 0$.

Exploiting the initialization

- With the presented bias description we can formulate heuristics for parameter initialization either to ease optimization or also to induce specific smoothness priors on the solutions.
- In particular, any curvature penalty $1/\zeta$ can be implemented by an appropriate choice of the initialization distribution.

Proposition 5 (Constructing any curvature penalty)

Given any function $\varrho\colon \mathbb{R} \to \mathbb{R}_{>0}$, *satisfying* $Z = \int_{\mathbb{R}} \frac{1}{\varrho} < \infty$, *if we set the density of* $\mathcal{C}$ *as* $p_{\mathcal{C}}(x) = \frac{1}{Z}\frac{1}{\varrho(x)}$ *and make* $\mathcal{W}$ *independent of* $\mathcal{C}$ *with non-vanishing second moment, then*

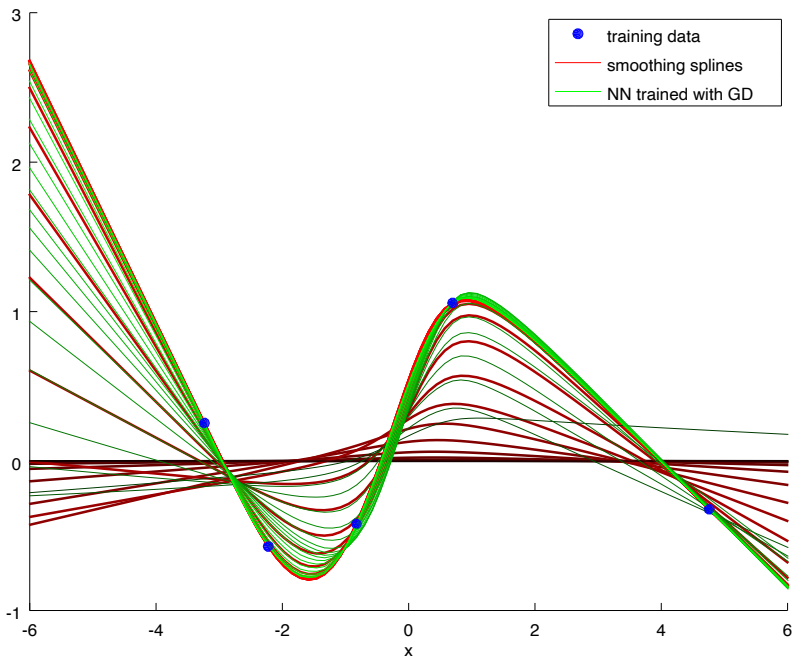
$$(\mathbb{E}(W^2|C=x)p_{\mathcal{C}}(x))^{-1} = (\mathbb{E}(W^2)p_{\mathcal{C}}(x))^{-1} \propto \varrho(x), \quad x \in \mathbb{R}.$$

Optimization trajectory in function space

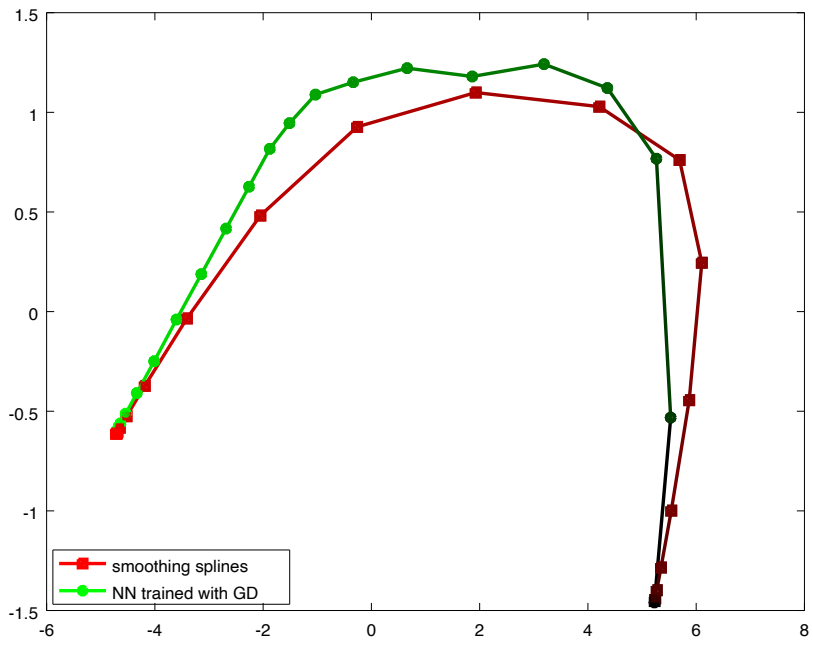
- Optimization trajectory described by smoothing splines

$$\min_{g \in C^2(S)} \quad \sum_{j=1}^{M} [g(x_j) - y_j]^2 + \frac{1}{\bar{\eta} t} \int_S \frac{1}{\zeta(x)} (g''(x) - f''(x, \theta_0))^2 \, \mathrm{d}x.$$

Trajectories of functions



2D PCA of the trajectories



Early stopping and spectral bias

- The training trajectory is approximated by a smoothing spline, meaning that the network will filter out high frequencies which are often associated with noise in the training data.
- This behavior is sometimes referred to as a spectral bias[1].
- - "Spectral Bias Outside the Training Set for Deep Networks in the Kernel Regime", NeurIPS 2022.
  - "Implicit Bias of MSE Gradient Optimization in Underparameterized Neural Networks", ICLR 2022.

Ben Bowman



(UCLA $\rightarrow$ AWS)

[1] Rahaman et al. 2019a.

Theorem 6 (Multivariate regression)

- *Use the same network setting as in Theorem 3 except that the number of input units changes to d.*
- *Assume that for each hidden unit the input weight and bias are initialized from a sub-Gaussian $(\boldsymbol{\mathcal{W}}, \mathcal{B})$ where $\boldsymbol{\mathcal{W}}$ is a d-dimensional random vector and $\mathcal{B}$ is a random variable.*
- *Let $\mathcal{U} = \|\boldsymbol{\mathcal{W}}\|_2$, $\boldsymbol{\mathcal{V}} = \boldsymbol{\mathcal{W}}/\|\boldsymbol{\mathcal{W}}\|_2$, $\mathcal{C} = -\mathcal{B}/\|\boldsymbol{\mathcal{W}}\|_2$, and let $p_{\boldsymbol{\mathcal{V}},\mathcal{C}}$ be the joint density of $(\boldsymbol{\mathcal{V}}, \mathcal{C})$.*
- *Then, for any finite data set $\{(\mathbf{x}_j, y_j)\}_{i=1}^M$ and sufficiently large n optimization of the MSE by full-batch gradient descent with sufficiently small step size converges to a parameter θ^* for which $f(\cdot, \theta^*)$ attains zero training error.*
- *Moreover,*

Theorem 6 (Multivariate regression)

letting $\zeta(\boldsymbol{V}, c) = p_{\boldsymbol{\mathcal{V}},\mathcal{C}}(\boldsymbol{V}, c)\mathbb{E}(\mathcal{U}^2|\boldsymbol{\mathcal{V}} = \boldsymbol{V}, \mathcal{C} = c)$, *we have*

$$\|f(\mathbf{x}, \theta^*) - g^*(\mathbf{x})\|_2 = O(n^{-\frac{1}{2}}), \quad \mathbf{x} \in \mathbb{R}^d$$

(the 2-norm over $\mathbb{R}^d$*) whp over* θ_0*, where* g^* *solves*

$$\min_{g \in C(\mathbb{R}^d)} \int_{\text{supp}(\zeta)} \frac{1}{\zeta(\boldsymbol{V}, c)} \left(\mathcal{R}\{(-\Delta)^{(d+1)/2}(g - f(\cdot, \theta_0))\}(\boldsymbol{V}, c)\right)^2 \, \mathrm{d}\boldsymbol{V}\mathrm{d}c$$

$$\text{s.t.} \quad g(\mathbf{x}_j) = y_j, \quad j = 1, \dots, M,$$

$$\mathcal{R}\{(-\Delta)^{(d+1)/2}(g - f(\cdot, \theta_0))\}(\boldsymbol{V}, c) = 0, \quad (\boldsymbol{V}, c) \notin \text{supp}(\zeta).$$

Here $\mathcal{R}$ *is the Radon transform,* $\mathcal{R}\{f\}(\boldsymbol{\omega}, b) := \int_{\langle \boldsymbol{\omega}, \mathbf{x} \rangle = b} f(\mathbf{x})\mathrm{d}s(\mathbf{x})$*, and the power of the negative Laplacian* $(-\Delta)^{(d+1)/2}$ *is the operator defined in Fourier domain by* $\widehat{(-\Delta)^{(d+1)/2} f}(\boldsymbol{\xi}) = \|\boldsymbol{\xi}\|^{d+1} \widehat{f}(\boldsymbol{\xi})$*.*

Generalization to other activation functions

Theorem 7 (Different activation functions)

- *Use the same setting as in Theorem 3 except that we use the activation function ϕ instead of ReLU.*
- *Suppose that ϕ is a Green's function of a linear operator L, i.e. $\mathrm{L}\phi = \delta$, where δ denotes the Dirac delta function.*
- *Assume that the activation function ϕ is homogeneous of degree k, i.e. $\phi(ax) = a^k\phi(x)$ for all $a > 0$.*
- *Then we can find a function p satisfying $\mathrm{L}p \equiv 0$ and adjust training data $\{(x_j, y_j)\}_{j=1}^M$ to $\{(x_j, y_j - p(x_j)\}_{j=1}^M$.*

Generalization to other activation functions

Theorem 7 (Different activation functions)

After that, the statement in Theorem 3 holds with

$$\min_{g \in C^2(S)} \quad \int_S \frac{1}{\zeta(x)} [\mathrm{L}(g(x) - f(x, \theta_0))]^2 \, \mathrm{d}x$$
$$\text{s.t.} \quad g(x_j) = y_j - p(x_j), \quad j = 1, \ldots, M,$$

where

$$\zeta(x) = p_{\mathcal{C}}(x) \mathbb{E}(\mathcal{W}^{2k} | \mathcal{C} = x)$$

and $S = \mathrm{supp}(\zeta) \cap [\min_i x_i, \max_i x_i]$.

From Theorem 3 we can extract generalization results such as

- In the univariate noisless model for a target g_0 on $[a, b]$, if ζ is uniform, then

$$\|g^* - g_0\|_\infty \leq C\|g_0^{(4)}\|_\infty h^4$$

  where $g^{(4)}$ is the fourth derivative of g_0 and $h = \max_i x_{i+1} - x_i$.

- For univariate noisy models with $y_j = g_0(x_j) + \epsilon_j$, ϵ_j independent zero mean with variance σ^2, if x_i is a uniform partition and ζ is uniform, using early stopping with $t = \Theta(M^{4/5})$, then

$$\mathbb{E}\|g^* - g_0\|_2^2 = O(M^{-4/5})$$

- Similar observations can be obtained in more general settings such as non-uniform training inputs, non-constant ζ

Summary

- Gradient descent training of overparametrized ReLU networks in the kernel regime is biased towards functions with low curvature.
- The parameter initialization procedure determines the curvature penalty function $1/\zeta$.
- Similar observations for multivariate ReLU networks, different activation functions, and optimization trajectories.

More on shallow ReLU nets

- Zhang et al. 2019 described the implicit bias of gradient descent in the kernel regime as minimizing a kernel norm from initialization, subject to fitting the training data.

More on shallow ReLU nets

- Zhang et al. 2019 described the implicit bias of gradient descent in the kernel regime as minimizing a kernel norm from initialization, subject to fitting the training data.
- Savarese et al. 2019 showed infinite-width networks with 2-norm weight regularization represent functions with smallest 1-norm of the second derivative, an example of which are linear splines.

More on shallow ReLU nets

- Zhang et al. 2019 described the implicit bias of gradient descent in the kernel regime as minimizing a kernel norm from initialization, subject to fitting the training data.
- Savarese et al. 2019 showed infinite-width networks with 2-norm weight regularization represent functions with smallest 1-norm of the second derivative, an example of which are linear splines.
- For univariate shallow ReLU nets:
  - Williams et al. 2019 showed a similar result training only the output layer from zero init.
  - Shevchenko, Kungurtsev, and Mondelli 2022 find SGD with noise is biased towards a piecewise linear map of the inputs.
  - Qiao et al. 2024 consider minima stability and find large learning rate training induces sparse linear spline fits.

More on shallow ReLU nets

- Zhang et al. 2019 described the implicit bias of gradient descent in the kernel regime as minimizing a kernel norm from initialization, subject to fitting the training data.
- Savarese et al. 2019 showed infinite-width networks with 2-norm weight regularization represent functions with smallest 1-norm of the second derivative, an example of which are linear splines.
- For univariate shallow ReLU nets:
  - Williams et al. 2019 showed a similar result training only the output layer from zero init.
  - Shevchenko, Kungurtsev, and Mondelli 2022 find SGD with noise is biased towards a piecewise linear map of the inputs.
  - Qiao et al. 2024 consider minima stability and find large learning rate training induces sparse linear spline fits.
- Feature learning regime not fully understood yet.

Further topics

- Spectral bias
- Implicit bias in mildly overparametrized nets
- Other optimization procedures, such as mirror descent
- Stability analysis
- Role of the data

1. Implicit Bias and Algorithmic Regularization
2. Implicit Bias in Wide Shallow ReLU Networks
3. **Implicit Bias of Mirror Descent for Wide Shallow Networks**
4. Spectral Biases in Deep Nets

Shuang Liang



- "Implicit Bias of Mirror Descent for Shallow Neural Networks in Univariate Regression", 2024

Mirror descent

- Mirror descent is a broad class of first-order optimization algorithms that generalize ordinary gradient descent.
- Precondition the gradient by the inverse Hessian $\nabla^2\Phi(\theta)^{-1}$ of a chosen convex potential Φ, adapting the update direction based on the local geometry determined by the potential.
- The choice of geometry in parameter space is crucial to
  - speed of convergence and generalization performance.
  - form of the implicit bias, as documented theoretically for linear models[2], matrix sensing[3], and locally for nonlinear models[4].
- No works characterizing the implicit bias of mirror decent for neural networks in function space...

[2] Gunasekar et al. 2018a; Pesme, Dragomir, and Flammarion 2024; Sun et al. 2023.

[3] Wu and Rebeschini 2021.

[4] Azizan, Lale, and Hassibi 2022.

Mirror flow for a wide ReLU network

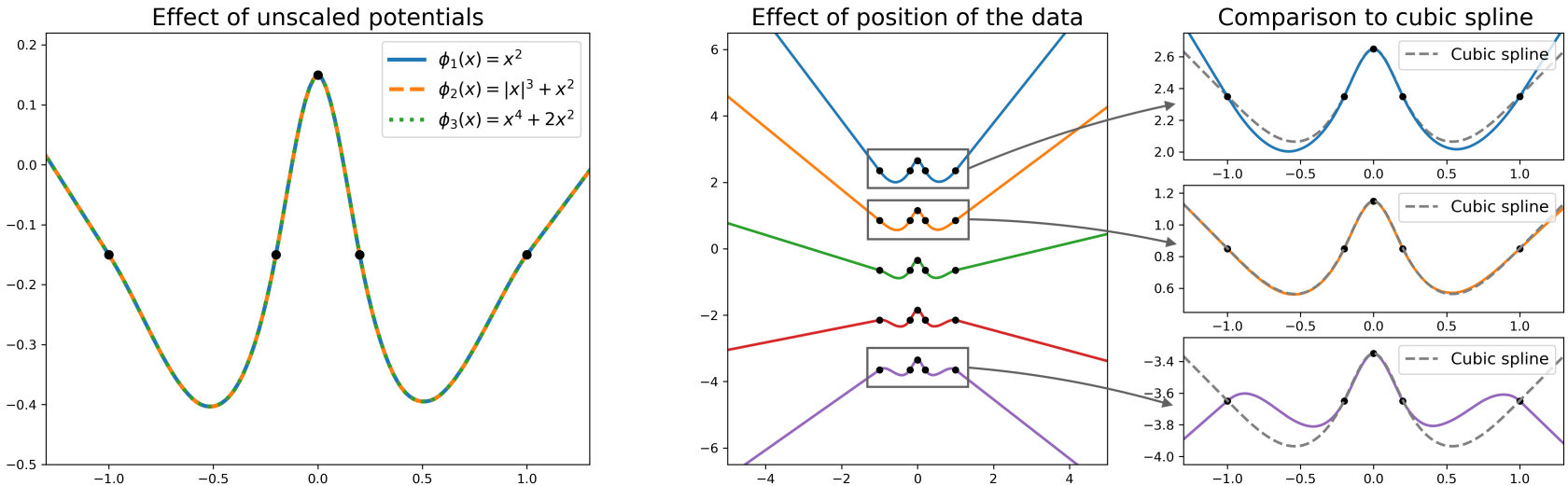


Figure 4: For any (unscaled) potential, the implicit bias of mirror flow is the same. Linear changes in data change the shape of the solution.

Mirror flow with scaled potentials for a shallow network

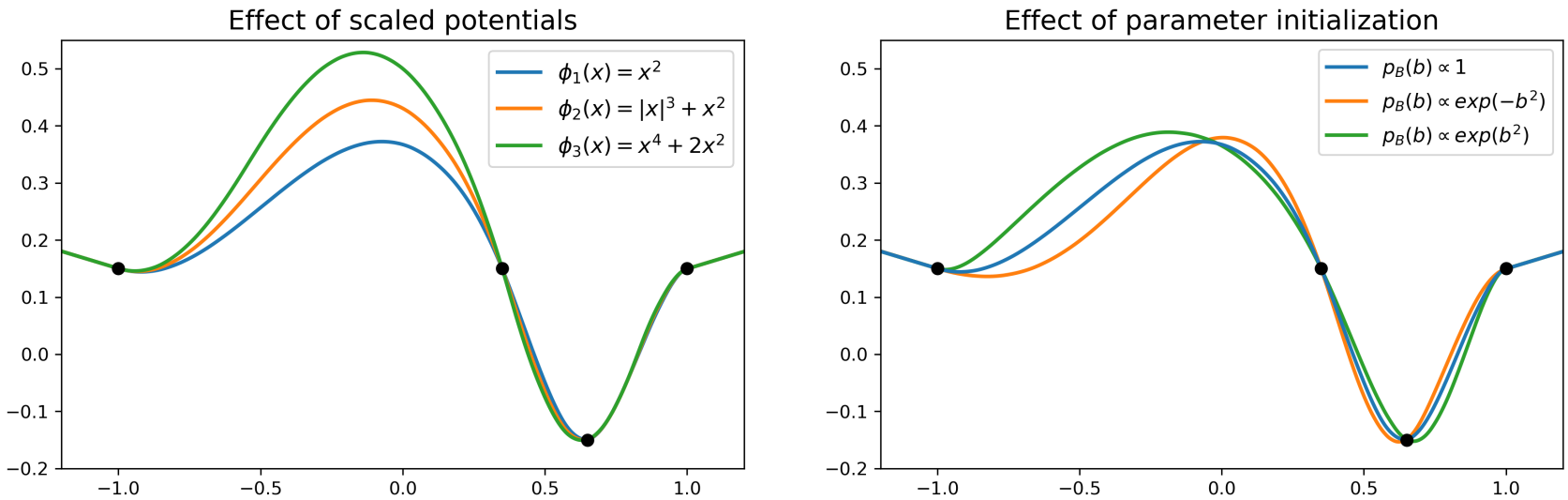


Figure 4: The potential determines how strongly different magnitudes of the curvature are penalized. The parameter initialization determines the how strongly the curvature is penalized at different input locations.

- Two-layer fully-connected network of width n and a single input:

$$f(x,\theta) = \sum_{k=1}^{n} a_k \sigma(w_k x - b_k) + c, \tag{4}$$

where $\theta = \mathrm{vec}(\mathbf{w}, \mathbf{b}, \mathbf{a}, c) \in \mathbb{R}^{3n+1}$; standard parametrization.

- Training objective for a finite data set $\{(x_i, y_i)\}_{i=1}^m \subset \mathbb{R} \times \mathbb{R}$:

$$L(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (f(x_i,\theta) - y_i)^2.$$

- Parameter initialization:

$$\begin{cases} w_k \overset{\text{iid}}{\sim} \mathcal{W} = \mathrm{Unif}(\{-1,1\}), \;\; b_k \overset{\text{iid}}{\sim} \mathcal{B}, \quad k = 1, \cdots, n; \\ a_k = 0, \;\; k = 1, \cdots, n; \quad c = 0. \end{cases} \tag{5}$$

Assume $\mathcal{B}$ is supported on $[-B_b, B_b]$ and has a density $p_{\mathcal{B}}$.

Assume $\mathcal{W}$ and $\mathcal{B}$ are independent and $(\mathcal{W}, \mathcal{B}) \overset{d}{=} (-\mathcal{W}, -\mathcal{B})$.

Mirror descent

- Given a strongly convex and twice differentiable *potential* Φ on the parameter space, the *Bregman divergence* is

$$D_\Phi(\theta, \theta') = \Phi(\theta) - \Phi(\theta') - \langle \nabla\Phi(\theta'), \theta - \theta' \rangle.$$

- Then the mirror descent parameter update[5] for $L(\theta)$ is

$$\theta(t+1) = \arg\min_{\theta'} \eta\langle \theta', \nabla L(\theta(t)) \rangle + D_\Phi(\theta', \theta(t)), \qquad (6)$$

  where $\eta > 0$ is the learning rate.

- Mirror flow

$$\frac{\mathrm{d}}{\mathrm{dt}}\theta(t) = -\eta(\nabla^2\Phi(\theta(t)))^{-1}\,\nabla L(\theta(t)). \qquad (7)$$

- Can be interpreted as Riemannian grad flow with metric $\nabla^2\Phi$.

[5] Beck and Teboulle 2003; Nemirovskij and Yudin 1983.

- Let $\mathbf{y} = (y_1, \cdots, y_m)^T \in \mathbb{R}^m$ denote the training labels
- Denote network predictions on input data at time t by

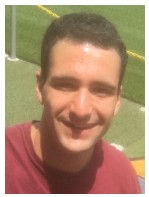
$$\mathbf{f}(t) = (f(x_1, \theta(t)), \ldots, f(x_m, \theta(t)))^T \in \mathbb{R}^m.$$

- The evolution of $\mathbf{f}(t)$ under mirror flow (7) is

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{f}(t) = -\eta n H(t)(\mathbf{f}(t) - \mathbf{y}), \quad \mathbf{f}(0) = 0, \tag{8}$$
$$\text{where } H(t) := n^{-1} J_\theta \mathbf{f}(t)(\nabla^2 \Phi(\theta(t)))^{-1}(J_\theta \mathbf{f}(t))^T,$$

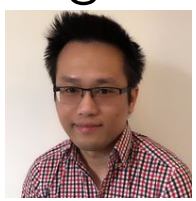
  where $J_\theta \mathbf{f}(t) = [\nabla_\theta f(x_1, \theta(t)), \ldots, \nabla_\theta f(x_m, \theta(t))]^T$ is the Jacobian matrix of the network predictions.
- For ordinary grad flow, NTK Gram $H(t) = J_\theta \mathbf{f}(t)(J_\theta \mathbf{f}(t))^T$.
- Different potentials induce different modifications of H and $\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{f}$.

- "Optimization Theory for ReLU Neural Networks Trained with Normalization Layers", ICML 2020. (Studies modified NTK)


Yoni


Quanquan

- "Tight Bounds on the Smallest Eigenvalue of the Neural Tangent Kernel for Deep ReLU Networks", ICML 2021.


Quynh


Marco

- *Bounds for the smallest eigenvalue of the NTK for arbitrary spherical data of arbitrary dimension*, NeurIPS 2024.


Kedar


Mike

Assumption 1

Assume the potential Φ is

(i) separable $\Phi(\theta) = \sum_k \phi(\theta_k)$;

(ii) twice continuously differentiable;

(iii) $\nabla^2\Phi(\theta)$ is positive definite for all θ.

(iv) $\Phi(0) = 0$, $\nabla\Phi(0) = 0$.

This is a relatively mild assumption and covers a broad class, e.g.,

- $\phi(x) = x^2$,
- $\phi(x) = |x|^p + ax^2$ with any $p > 2$, $a > 0$,
- $\phi(x) = \cosh(x)$.

In particular, no homogeneity requirement.

Theorem 8 (Implicit bias of mirror flow for ReLU network)

- *Consider a two-layer ReLU network* (4) *with n hidden units, n sufficiently large, initialization* (5) *with $p_{\mathcal{B}}(b)$ the density of $\mathcal{B}$.*
- *Consider any finite data set $\{(x_i, y_i)\}_{i=1}^m$, $\max_i |x_i| < B_b$, and mirror flow* (7) *with Φ satisfying Assumption 1 and $\eta = \Theta(1/n)$.*

Then ...

Theorem 8 (Implicit bias of mirror flow for ReLU network)

(i) Lazy training and convergence: *whp over the random parameter initialization, there exist* $C_1, C_2 > 0$ *such that for any* $t \geq 0$,

$$\begin{aligned} &\|\theta(t) - \theta(0)\|_\infty \leq C_1 n^{-1}, \ \lim_{n\to\infty} \|H(t) - H(0)\|_2 = 0, \\ &\|\mathbf{f}(t) - \mathbf{y}\|_2 \leq \exp(-\eta_0 C_2 t)\|\mathbf{y}\|_2. \end{aligned} \tag{9}$$

Theorem 8 (Implicit bias of mirror flow for ReLU network)

(ii) Description of the solution function: *For any given* $x \in \mathbb{R}$,

$$\lim_{n\to\infty} |f(x, \theta(\infty)) - h(x)| = 0,$$

where $h(\cdot)$ *is the solution to following variational problem:*

$$\min_{h\in\mathcal{F}_{\mathrm{ReLU}}} \mathcal{G}_1(h) + \mathcal{G}_2(h) + \mathcal{G}_3(h) \quad \text{s.t. } h(x_i) = y_i,\ \forall i \in [m],$$

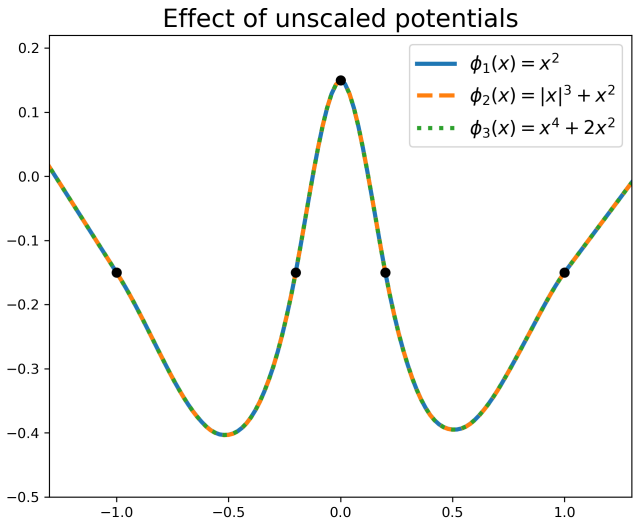
$$\begin{cases} \mathcal{G}_1(h) = \displaystyle\int_{-B_b}^{B_b} \frac{(h''(x))^2}{p_{\mathcal{B}}(x)} \mathrm{d}x \\ \mathcal{G}_2(h) = (h'(+\infty) + h'(-\infty))^2 \\ \mathcal{G}_3(h) = \dfrac{1}{\mathbb{E}[\mathcal{B}^2]} \Big(B_b(h'(+\infty) - h'(-\infty)) - (h(B_b) + h(-B_b)) \Big)^2. \end{cases} \tag{9}$$

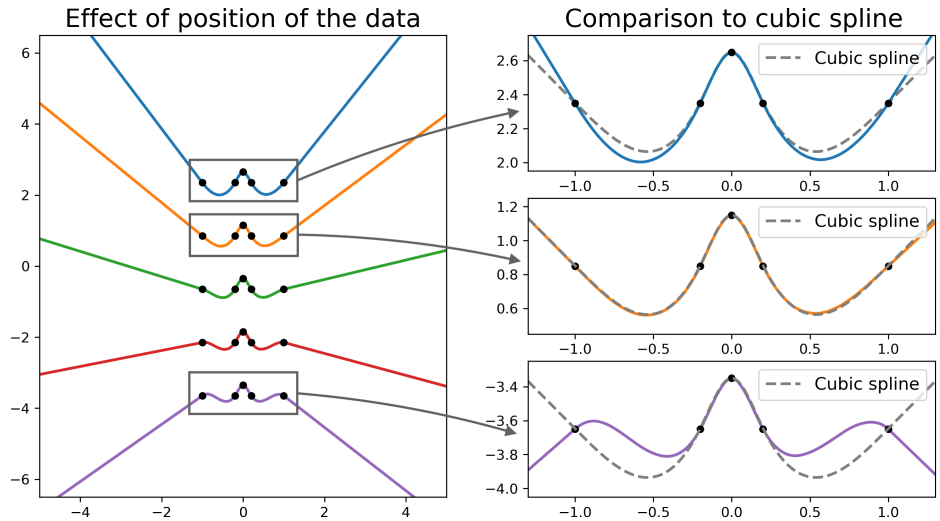
Here we use

$$\mathcal{F}_{\mathrm{ReLU}} = \{g(x) = \int \alpha(w, b)[wx - b]_+ \mathrm{d}\mu \colon \alpha \in C(\{1, -1\} \times [-B_b, B_b])\}.$$

Remarks

- Thus the solution does not depend on Φ. Diverse mirror flows, including ordinary gradient flow, induce the same bias.
  - This is a consequence of lazy training: parameter remains nearly unchanged, and so $\Phi \approx$ "2nd-order Taylor expansion at init".
  - This is not true for finite-dimensional parameter models.
- The three functionals serve different roles:
  - $\mathcal{G}_1$ is the $1/p_{\mathcal{B}}$-weighted ℓ_2 norm of the second derivative, where $p_{\mathcal{B}}$ is the ReLU breakpoint density at initialization; penalizes curvature.
  - $\mathcal{G}_2$ encourages opposite slopes at $\pm\infty$.
  - $\mathcal{G}_3$ encourages $\cup$ or $\cap$ shape if data is above or below x-axis.
- The result removes a previous requirement to either conduct an intractable adjustment of the data or add skip connections.
- $\mathcal{F}_{\text{ReLU}} = \mathcal{F}_{\text{Abs}} \oplus \{\text{affine functions}\}$





- All mirrors give the same solution.
- Solutions change when the data is translated.

Theorem 9 (Implicit bias of mirror flow for Abs network)

- *The same result as above holds for a network with absolute value activation function $\sigma(s) = |s|$, except for $\mathcal{G} = \mathcal{G}_1$.*
- *This is also the result for a ReLU network with skip connections.*
- *The solution to the variational problem with $\mathcal{G} = \mathcal{G}_1$ is the natural cubic spline interpolation of the training data.*

Scaled potential functions

Assumption 2

Assume the potential Φ is

(i) separable and scaled as $\Phi(x) = \frac{1}{n^2} \sum_k \phi(n(\theta_k - \bar{\theta}_k))$;

(ii) $\phi(x) = \psi(x) + ax^2$ for a convex $\psi \in C^3(\mathbb{R})$ and $a > 0$;

(iii) $\psi(0) = \psi'(0) = 0$.

The prefactor $\frac{1}{n^2}$ ensures $\nabla^2 \Phi$ is well conditioned as $n \to \infty$.

Theorem 10 (Implicit bias of scaled mirror flow for abs net)

- *Consider a two-layer network* (4) *with absolute value activation, n hidden units, n is sufficiently large, initialization* (5)*.*
- *Consider any finite data set $\{(x_i, y_i)\}_{i=1}^m$, and mirror flow* (7) *with scaled potential satisfying Assumption 2 and $\eta = \Theta(1/n)$. Specifically, $\phi(z) = \psi(z) + \gamma z^2$ with sufficiently large $\gamma > 0$.*

  Then ...

Theorem 10 (Implicit bias of scaled mirror flow for abs net)

(i) Lazy training and convergence: *whp over the random initialization, there exist* $C_1, C_2 > 0$ *such that for any* $t \geq 0$,

$$\|\theta(t)-\theta(0)\|_\infty \leq C_1 n^{-1}, \ \|\mathbf{f}(t)-\mathbf{y}\|_2 \leq \exp(-\eta_0 C_2 t)\|\mathbf{y}\|_2. \quad (10)$$

Theorem 10 (Implicit bias of scaled mirror flow for abs net)

(ii) Description of the solution function: *For any given* $x \in \mathbb{R}$,

$$\lim_{n\to\infty} |f(x, \theta(\infty)) - h(x)| = 0,$$

where $h(\cdot)$ *is the solution to following variational problem:*

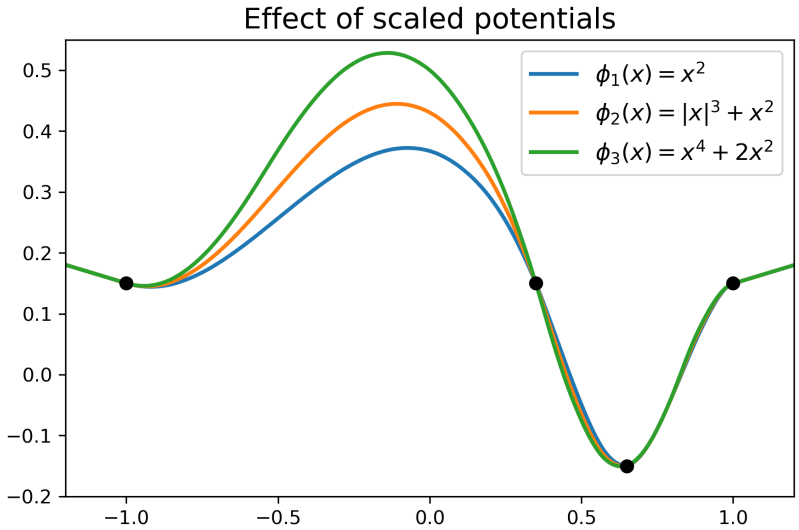
$$\min_{h\in\mathcal{F}_{\mathrm{Abs}}} \int_{-B_b}^{B_b} \phi\Big(\frac{h''(x)}{2p_{\mathcal{B}}(x)}\Big) p_{\mathcal{B}}(x)\mathrm{d}x \quad \text{s.t. } h(x_i) = y_i,\ \forall i \in [m]. \tag{10}$$

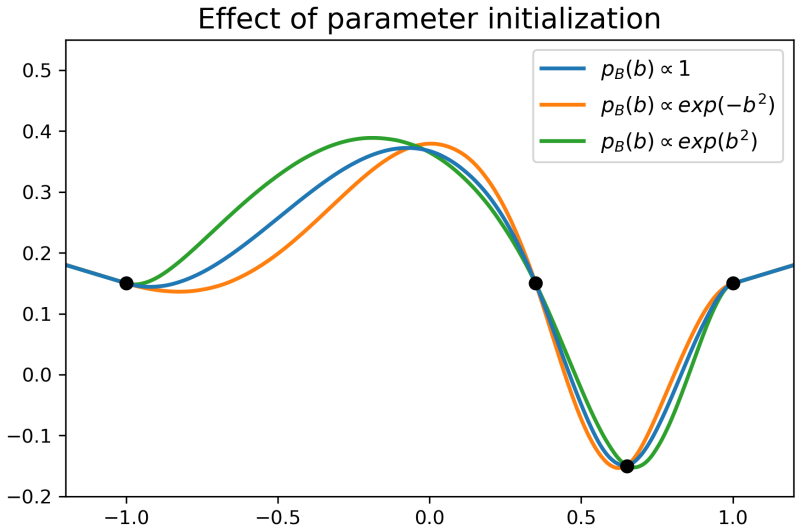
Here we use

$$\mathcal{F}_{\mathrm{Abs}} = \{g(x) = \int \alpha(w, b)|wx-b|\mathrm{d}\mu \colon \alpha \in C(\{-1, 1\}\times[-B_b, B_b]), \alpha \text{ even}\}.$$

Remarks

- Now the potential Φ has a significant impact on the solution.
- The previous functionals $\mathcal{G}_2$ and $\mathcal{G}_3$ vanish on $\mathcal{F}_{\text{Abs}}$.
- If ϕ is non-homogeneous, then the objective functional in Theorem 10 is not homogeneous and hence cannot be expressed as a function norm (e.g., RKHS norm) or squared function norm.
- In spite of lazy training, kernel matrix $H(t)$ does not necessarily stay close to $H(0)$ due to the scaling factor n in the potential.





- The mirror potential controls the different magnitudes of the curvature, via ϕ
- The parameter initialization controls the spatial distribution of the curvature, via $p_{\mathcal{B}}$

For instance, if $\phi(x) = \begin{cases} < x^2, & x \in [-1,1] \\ > x^2, & x \notin [-1,1] \end{cases}$, then mirror flow will be more tolerant of small curvature ($|h''| < 1$) but less tolerant of large curvature ($|h''| > 1$) than gradient flow.

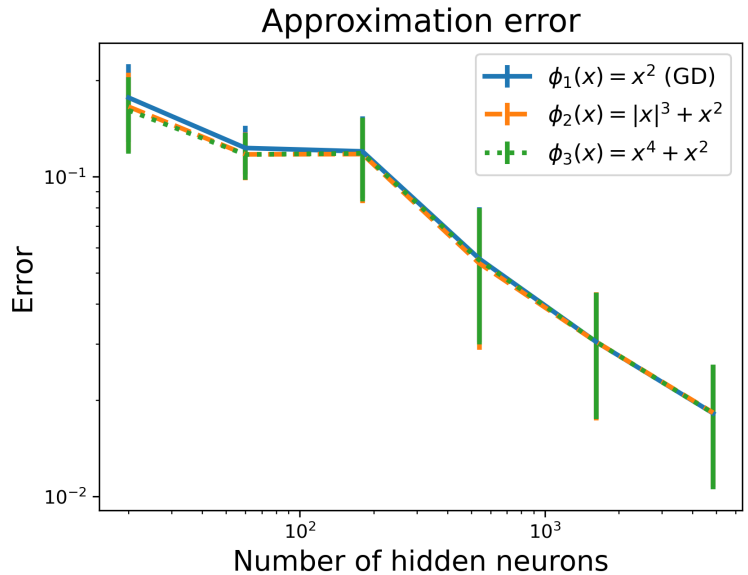


Figure 5: We did not provide a quantitative theoretical bound on the convergence as n increases, but here is what it looks like experimentally.

Proof sketch

- Step 1: Linearization
- Step 2: Implicit bias in parameter space
- Step 3: Implicit bias in function space

Linearization

- The first step of the proof is to show that, under mirror flow, functions obtained by training the network are close to those obtained by training only the output weights.
- The general procedure is well established for gradient flow and translates more or less directly for unscaled potentials.
- Let $\tilde{\theta}(t) = \mathrm{vec}(\mathbf{w}(0), \mathbf{b}(0), \tilde{\mathbf{a}}(t), c(0))$, where $\mathbf{w}, \mathbf{b}, c$ are kept at their initial values and

$$\frac{\mathrm{d}}{\mathrm{dt}}\tilde{\mathbf{a}}(t) = -\eta(\nabla^2\Phi(\tilde{\mathbf{a}}(t)))\nabla_{\tilde{\mathbf{a}}}L(\tilde{\theta}(t)), \quad \tilde{\mathbf{a}}(0) = \mathbf{a}(0). \tag{11}$$

- The evolution of $\tilde{\mathbf{f}}(t)$ can be given by

$$\frac{\mathrm{d}}{\mathrm{dt}}\tilde{\mathbf{f}}(t) = -n\eta\tilde{H}(t)(\tilde{\mathbf{f}}(t) - \mathbf{y}), \quad \tilde{\mathbf{f}}(0) = 0, \tag{12}$$
$$\text{where } \tilde{H}(t) \triangleq n^{-1}J_{\tilde{\mathbf{a}}}\tilde{\mathbf{f}}(t)(\nabla^2\Phi(\tilde{\mathbf{a}}(t)))^{-1}J_{\tilde{\mathbf{a}}}\tilde{\mathbf{f}}(t)^T.$$

- We aim to show that for any fixed $x \in \mathbb{R}$,

$$\lim_{n\to\infty}\sup_{t\geq 0}|f(x, \theta(t)) - f(x, \tilde{\theta}(t)| = 0. \tag{13}$$

- This can be established by showing the kernel matrix $H(t)$ remains sufficiently close to $\tilde{H}(t)$ during training.
- For unscaled potentials, we show that both $\theta(t)$ and $\tilde{\theta}(t)$ remain close to $\theta(0)$.[6]

  Then the closeness between $H(t)$ and $\tilde{H}(t)$ follows from the continuity of the Hessian map $\nabla^2\Phi(\cdot)$ and Assumption 1.
- For scaled potentials, we show taking large enough a that the Hessian map becomes sufficiently "flat" over a compact region containing the two trajectories $\{n\theta(t)\}_{t\geq 0}$ and $\{n\tilde{\theta}(t)\}_{t\geq 0}$.

  Based on this we then show that $n\theta(t)$ stays close to $n\tilde{\theta}(t)$.

[6] Using similar strategy as Du et al. 2018; Lee et al. 2019a.

Implicit bias in parameter space

- The second step is to characterize the implicit bias for wide networks in their parameter space.
- By the first part, it suffices to characterize the bias for networks where only output weights are trained. We can use

Theorem 11 (Gunasekar et al. 2018a)

If mirror flow (11) *converges to zero loss, then the final parameter is the solution to following constrained optimization problem:*

$$\min_{\tilde{\mathbf{a}} \in \mathbb{R}^n} D_\Phi(\tilde{\mathbf{a}}, \tilde{\mathbf{a}}(0)) \quad \text{s.t. } f(x_i, \tilde{\theta}) = y_i, \ \forall i \in [m]. \tag{14}$$

- We reformulate problem (14) in a way that allows us to consider the infinite width limit, i.e., $n \to \infty$.
- Consider the empirical distribution μ_n on $\mathbb{R} \times \mathbb{R}$ induced by the sampled input weights and biases $\{(w_k(0), b_k(0))\}_{k=1}^n$.[7]

  We reparametrize the finite-width networks as follow:

$$f(x, \tilde{\theta}) = g_n(x, \alpha) = \int \alpha(w, b)\sigma(wx - b)\mathrm{d}\mu_n. \qquad (15)$$

- Noticing μ_n converges weakly to μ, we write infinitely wide networks as follows:

$$g(x, \alpha) = \int \alpha(w, b)\sigma(wx - b)\mathrm{d}\mu, \quad \alpha \in \Theta. \qquad (16)$$

  Here $\Theta \subseteq C(\{-1, 1\} \times [-B_b, B_b])$.

[7] $\mu_n(A) = \frac{1}{n}\sum_{k=1}^n \mathbb{I}_A((w_k(0), b_k(0)))$ for any measurable set $A \subset \{-1, 1\} \times [-B_b, B_b]$, where $\mathbb{I}_A$ denotes the indicator function of set A.

- We aim to identify a cost functional $\mathcal{C}_\Phi : \Theta \to \mathbb{R}$ such that, as the width of the network goes to infinity, the solution to (14) "converges to" the solution to the following problem

$$\min_{\alpha \in \Theta} \mathcal{C}_\Phi(\alpha) \quad \text{s.t. } g(x_i, \alpha) = y_i, \ \forall i \in [m]. \tag{17}$$

- For unscaled potentials, we show that $\mathcal{C}_\Phi(\alpha) = \int (\alpha(w, b))^2 \mathrm{d}\mu$ regardless of the choice of Φ.
- In contrast, for scaled potentials, $\mathcal{C}_\Phi(\alpha) = \int \phi(\alpha(w, b)) \mathrm{d}\mu$.

Implicit bias in function space

- The last step is to characterize the solution to problem (17) in function space

$$\mathcal{F} = \left\{ g(x, \alpha) = \int \alpha(w, b)\sigma(wx - b)\mathrm{d}\mu \colon \alpha \in \Theta \right\}.$$

- Inspired by Savarese et al. 2019, we consider the map from $\mathcal{F}$ to Θ induced by the minimal representation, i.e., for any $h \in \mathcal{F}$,

$$T_\Phi(h) \triangleq \arg\min_{\alpha \in \Theta} \mathcal{C}_\Phi(\alpha) \quad \text{s.t. } g(x, \alpha) = h(x), \ \forall x \in \mathbb{R}. \tag{18}$$

- $T_\Phi(h)$ is well defined if $\mathcal{C}_\Phi$ is strictly convex on Θ.

- We show that T_Φ properly pulls back the cost functional to the function space, which allows us to obtain a function space description of the implicit bias.

Proposition 12

Assume that Θ is a vector subspace in $C(\{-1,1\} \times [-B_b, B_b])$ and that C_Φ is strictly convex on Θ. Consider the map T_Φ defined in (18). If $\bar{\alpha}$ is the solution to problem (17), then $g(\cdot, \bar{\alpha})$ solves the following variational problem:

$$\min_{h \in \mathcal{F}} \mathcal{C}_\Phi \circ T_\Phi(h) \quad \text{s.t. } h(x_i) = y_i, \forall i \in [m]. \tag{19}$$

Summary

Summary:

- Implicit bias in function space for shallow wide nets.
- Lazy training; kernel regime for unscaled potentials non-kernel regime for scaled potentials
- Potential controls magnitudes of curvature, whereas initialization controls location in input space

Summary

Summary:

- Implicit bias in function space for shallow wide nets.
- Lazy training; kernel regime for unscaled potentials non-kernel regime for scaled potentials
- Potential controls magnitudes of curvature, whereas initialization controls location in input space

Questions:

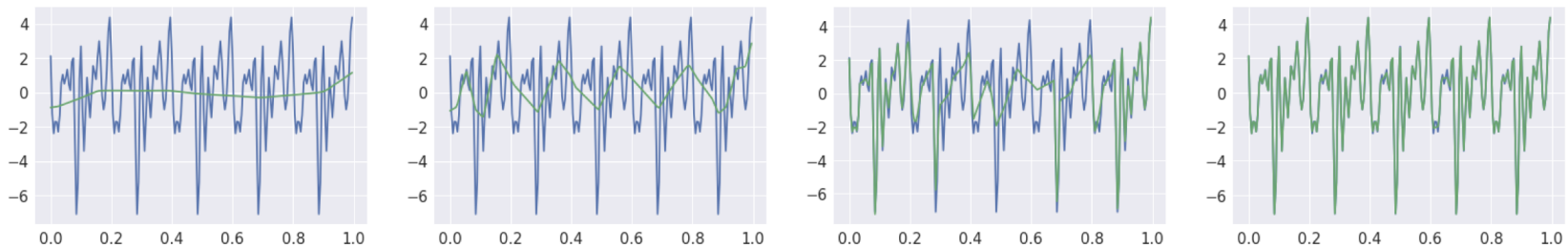
- Rich feature learning regime and friends?
- Which mirrors are better for which tasks? Instantiate generalization bounds, run experiments.
- Other parameter initializations for mirror descent?
- Multivariate case, can we leverage approach from GD?
- We can compare shallow without and with skip connections. Other architectures, particularly deep?

1. Implicit Bias and Algorithmic Regularization
2. Implicit Bias in Wide Shallow ReLU Networks
3. Implicit Bias of Mirror Descent for Wide Shallow Networks
4. Spectral Biases in Deep Nets

Benjamin Bowman



- "Spectral Bias Outside the Training Set for Deep Networks in the Kernel Regime"
- "Implicit Bias of MSE Gradient Optimization in Underparameterized Neural Networks"



(a) Iteration 100 (b) Iteration 1000 (c) Iteration 10000 (d) Iteration 80000

Figure 6: Learned function (green) as training progresses[8].

[8] Figure from Rahaman et al. 2019b.

Spectral biases

- For shallow univariate ReLU networks the dominant eigenfunctions of the Neural Tangent Kernel are smoother[9]
- ReLU nets in the kernel regime are biased towards smooth interpolants[10]
- "Spectral Bias" can be interpreted to mean bias towards learning the top eigenfunctions of the NTK
- By looking at empirical approximations to the eigenfunctions, spectral bias was demonstrated to hold on the training set[11]

[9] Basri et al. 2019, 2020.

[10] Jin and Montúfar 2023; Williams et al. 2019.

[11] Arora et al. 2019a; Basri et al. 2020; Cao et al. 2021.

Overview

- We provide quantitative bounds measuring the L^2 difference in function space between the trajectory of a

  finite-width network trained on finitely many samples

  idealized kernel dynamics of infinite width and infinite data

Overview

- We provide quantitative bounds measuring the L^2 difference in function space between the trajectory of a

  finite-width network trained on finitely many samples

  idealized kernel dynamics of infinite width and infinite data

- As an implication, eigenfunctions of the NTK integral operator (not just their empirical approximations) are learned at rates corresponding to their eigenvalues

Overview

- We provide quantitative bounds measuring the L^2 difference in function space between the trajectory of a

  finite-width network trained on finitely many samples

  idealized kernel dynamics of infinite width and infinite data

- As an implication, eigenfunctions of the NTK integral operator (not just their empirical approximations) are learned at rates corresponding to their eigenvalues

- The network inherits bias of the kernel at beginning of training even when the width only grows linearly with the training sample

Spectrum is skewed

- The dependence of the convergence rate on σ_i is particularly relevant as the NTK tends to have a very skewed spectrum

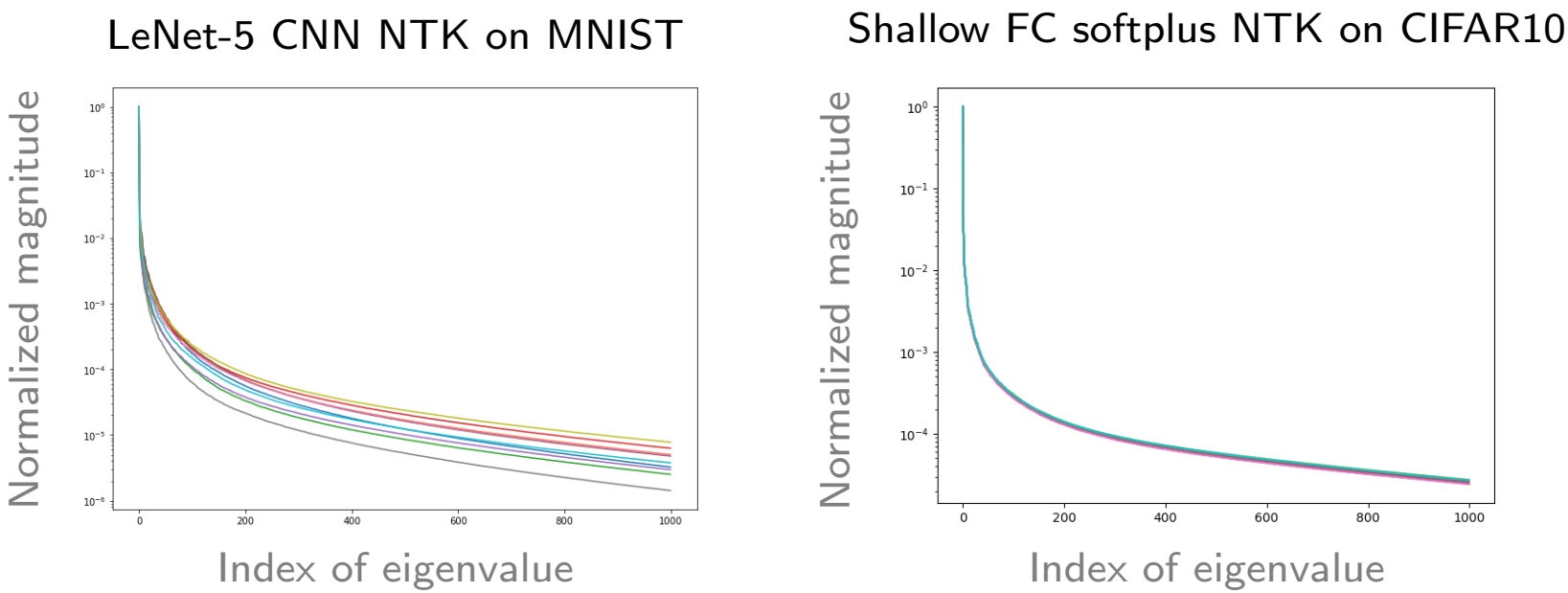


Figure 7: Normalized NTK spectrum λ_k/λ_1 on MNIST and CIFAR10 for two networks using 10 random parameter initializations and data batches.

More on spectrum of NTK: Murray et al. 2023; Least eigenvalue: Karhadkar, Murray, and Montúfar 2024.

Settings

- Neural network: $f(x;\theta)$ taking inputs $x \in X \subset \mathbb{R}^d$, parameterized by $\theta \in \mathbb{R}^p$.
- Training data: $\{(x_1, y_1), \ldots, (x_n, y_n)\} \subset \mathbb{R}^d \times \mathbb{R}$, $y_i = f^*(x_i)$.
- Residual error on training set: $\hat{r}(\theta) \in \mathbb{R}^n$, $\hat{r}(\theta)_i := f(x_i;\theta) - y_i$.
- Squared error loss:

$$\Phi(\theta) := \tfrac{1}{2n} \|\hat{r}(\theta)\|_2^2 = \tfrac{1}{2} \|\hat{r}(\theta)\|_{\mathbb{R}^n}^2$$

- Gradient flow:

$$\partial_t \theta_t = -\partial_\theta \Phi(\theta)$$

$\langle \bullet, \bullet \rangle$ and $\|\bullet\|_2$ Euclidean inner product and norm. $\langle \bullet, \bullet \rangle_{\mathbb{R}^n} = \frac{1}{n} \langle \bullet, \bullet \rangle$ and $\|\bullet\|_{\mathbb{R}^n} := \sqrt{\langle \bullet, \bullet \rangle_{\mathbb{R}^n}}$. Let $L^p(X, \nu)$ denote the L^p space over domain X with measure ν.

- Analytical NTK:

$$K^{\infty}(x, x') := \mathbb{E}_{\theta_0 \sim \mu} \left[\langle \nabla_\theta f(x; \theta_0), \nabla_\theta f(x'; \theta_0) \rangle \right],$$

with expectation taken over the parameter initialization $\theta_0 \sim \mu$.

- Analytical NTK:

$$K^{\infty}(x, x') := \mathbb{E}_{\theta_0 \sim \mu}\left[\langle \nabla_\theta f(x; \theta_0), \nabla_\theta f(x'; \theta_0)\rangle\right],$$

with expectation taken over the parameter initialization $\theta_0 \sim \mu$.

- Integral operator: The kernel K^{∞} induces

$$T_{K^{\infty}} : L^2(X, \rho) \to L^2(X, \rho); \quad g(x) \mapsto \int_X K^{\infty}(x, s) g(s) d\rho(s), \tag{20}$$

where X is our input space and ρ is the input distribution.

- Analytical NTK:

$$K^{\infty}(x, x') := \mathbb{E}_{\theta_0 \sim \mu}\left[\langle \nabla_\theta f(x; \theta_0), \nabla_\theta f(x'; \theta_0)\rangle\right],$$

with expectation taken over the parameter initialization $\theta_0 \sim \mu$.

- Integral operator: The kernel K^{∞} induces

$$T_{K^{\infty}} : L^2(X, \rho) \to L^2(X, \rho); \quad g(x) \mapsto \int_X K^{\infty}(x, s) g(s) d\rho(s), \tag{20}$$

where X is our input space and ρ is the input distribution.

- Spectral decomposition: By Mercer's theorem we have

$$K^{\infty}(x, x') = \sum_{i=1}^{\infty} \sigma_i \phi_i(x) \phi_i(x'),$$

where $\{\phi_i\}$ is an orthonormal basis for $L^2(X, \rho)$ and $\{\sigma_i\}$ is a nonincreasing sequence of positive values.

- Discretization: Training sample $x_1, \ldots, x_n$ introduces

$$T_n : g(x) \mapsto \frac{1}{n} \sum_{i=1}^{n} K^{\infty}(x, x_i) g(x_i) = \int_X K^{\infty}(x, s) g(s) d\widehat{\rho}(s), \tag{21}$$

where $\widehat{\rho} = \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i}$ is the empirical measure.

- Discretization: Training sample $x_1, \ldots, x_n$ introduces

$$T_n : g(x) \mapsto \frac{1}{n} \sum_{i=1}^{n} K^{\infty}(x, x_i) g(x_i) = \int_X K^{\infty}(x, s) g(s) d\widehat{\rho}(s), \tag{21}$$

  where $\widehat{\rho} = \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i}$ is the empirical measure.
- Time dependent NTK:

$$K_t(x, x') := \langle \nabla_\theta f(x; \theta_t), \nabla_\theta f(x'; \theta_t) \rangle$$

  has an associated time-dependent operator T_n^t

$$T_n^t g(x) := \frac{1}{n} \sum_{i=1}^{n} K_t(x, x_i) g(x_i) = \int_X K_t(x, s) g(s) d\widehat{\rho}(s). \tag{22}$$

- Discretization: Training sample $x_1, \dots, x_n$ introduces

$$T_n : g(x) \mapsto \frac{1}{n} \sum_{i=1}^{n} K^{\infty}(x, x_i) g(x_i) = \int_X K^{\infty}(x, s) g(s) d\widehat{\rho}(s), \tag{21}$$

  where $\widehat{\rho} = \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i}$ is the empirical measure.
- Time dependent NTK:

$$K_t(x, x') := \langle \nabla_\theta f(x; \theta_t), \nabla_\theta f(x'; \theta_t) \rangle$$

  has an associated time-dependent operator T_n^t

$$T_n^t g(x) := \frac{1}{n} \sum_{i=1}^{n} K_t(x, x_i) g(x_i) = \int_X K_t(x, s) g(s) d\widehat{\rho}(s). \tag{22}$$

- Update rule: under gradient flow the residual is given by

$$\partial_t r_t(x) = -\frac{1}{n} \sum_{i=1}^{n} K_t(x, x_i) r_t(x_i) = -T_n^t r_t.$$

Idealized infinite width and infinite data

- Infinite width limit: As the network width tends to infinity the time-dependent NTK becomes constant so that

$$K_t(x, x') = K^{\infty}(x, x') \quad \text{and} \quad T_n^t = T_n$$

Idealized infinite width and infinite data

- Infinite width limit: As the network width tends to infinity the time-dependent NTK becomes constant so that

$$K_t(x, x') = K^{\infty}(x, x') \quad \text{and} \quad T_n^t = T_n$$

- Infinite data limit: Similarly, heuristically as $n \to \infty$ we have

$$T_n \to T_{K^{\infty}}$$

Idealized infinite width and infinite data

- Infinite width limit: As the network width tends to infinity the time-dependent NTK becomes constant so that

$$K_t(x, x') = K^{\infty}(x, x') \quad \text{and} \quad T_n^t = T_n$$

- Infinite data limit: Similarly, heuristically as $n \to \infty$ we have

$$T_n \to T_{K^\infty}$$

- In this idealized setting the update rule is $\partial_t r_t = -T_{K^\infty} r_t$, which has the solution $r_t = \exp(-T_{K^\infty} t) r_0$ defined via

$$\langle r_t, \phi_i \rangle_\rho = \exp(-\sigma_i t) \langle r_0, \phi_i \rangle_\rho. \tag{23}$$

Idealized infinite width and infinite data

- Infinite width limit: As the network width tends to infinity the time-dependent NTK becomes constant so that

$$K_t(x, x') = K^{\infty}(x, x') \quad \text{and} \quad T_n^t = T_n$$

- Infinite data limit: Similarly, heuristically as $n \to \infty$ we have

$$T_n \to T_{K^\infty}$$

- In this idealized setting the update rule is $\partial_t r_t = -T_{K^\infty} r_t$, which has the solution $r_t = \exp(-T_{K^\infty} t) r_0$ defined via

$$\langle r_t, \phi_i \rangle_\rho = \exp(-\sigma_i t) \langle r_0, \phi_i \rangle_\rho. \tag{23}$$

- Thus in this idealized setting the network learns eigenfunctions ϕ_i at rates determined by their eigenvalues σ_i.

Spectral bias outside the training set

- We will see that the bias at the beginning of training can be described entirely through T_{K^∞} and its eigenfunctions.
- This depends only on the model architecture, parameter initialization distribution μ, and input distribution ρ.

Spectral bias outside the training set

- We will see that the bias at the beginning of training can be described entirely through T_{K^∞} and its eigenfunctions.
- This depends only on the model architecture, parameter initialization distribution μ, and input distribution ρ.

- Architectures: deep nets with fully connected, convolutional, residual layers, minimum width m.
- Initialization: ASI
- Assumptions: Twice continuously differentiable activation, bounded target

Theorem 13

- *Let $K(x, x')$ fixed continuous, symmetric, positive definite kernel*
- *Let $P_k : L^2(X, \rho) \to L^2(X, \rho)$ denote the orthogonal projection onto the span of the top k eigenfcts of the operator T_K*
- *Let $\sigma_k > 0$ denote the k-th eigenvalue of T_K*

Then $m = \tilde{\Omega}(T^4/\epsilon^2)$ and $n = \tilde{\Omega}(T^2/\epsilon^2)$ suffices to ensure with probability $1 - O(mn)\exp(-\Omega(\log^2 m)$ over the parameter initilization and the training samples that for all $t \leq T$ and $k \in \mathbb{N}$

$$\|P_k(r_t - \exp(-T_K t) r_0)\|^2_{L^2(X,\rho)}$$
$$\leq \left[\frac{1 - \exp(-\sigma_k t)}{\sigma_k}\right]^2 \cdot \left[4 \|f^*\|^2_\infty \|K - K_0\|^2_{L^2(X^2,\rho\otimes\rho)} + \epsilon\right]$$

and

$$\|r_t - \exp(-T_K t) r_0\|^2_{L^2(X,\rho)} \leq t^2 \cdot \left[4 \|f^*\|^2_\infty \|K - K_0\|^2_{L^2(X^2,\rho\otimes\rho)} + \epsilon\right].$$

Interpretation

- Theorem 13 compares the dynamics of

$r_t(x) := f(x;\theta_t) - f^*(x)$
finite-width model trained on finitely many samples

$\exp(-T_K t) r_0$
idealized kernel method with infinite data

Interpretation

- Theorem 13 compares the dynamics of

$r_t(x) := f(x;\theta_t) - f^*(x)$
finite-width model trained on finitely many samples

$\exp(-T_K t) r_0$
idealized kernel method with infinite data

- $\exp(-T_K t) r_0$ learns projection along ϕ_i linearly at rate σ_i, by (23),

$$\langle r_t, \phi_i \rangle_\rho = \exp(-\sigma_i t) \langle r_0, \phi_i \rangle_\rho.$$

Whenever the NTK at initialization K_0 concentrates around K, the residual r_t will inherit this bias of the kernel dynamics.

Interpretation

- Theorem 13 compares the dynamics of

$r_t(x) := f(x;\theta_t) - f^*(x)$
finite-width model trained on finitely many samples

$\exp(-T_K t) r_0$
idealized kernel method with infinite data

- $\exp(-T_K t) r_0$ learns projection along ϕ_i linearly at rate σ_i, by (23),

$$\langle r_t, \phi_i \rangle_\rho = \exp(-\sigma_i t) \langle r_0, \phi_i \rangle_\rho.$$

  Whenever the NTK at initialization K_0 concentrates around K, the residual r_t will inherit this bias of the kernel dynamics.

- Furthermore, the bound for the projected differences is smaller when σ_k is larger. Therefore the bias appears more pronounced along eigendirections with large eigenvalues.

Consequences for the special case $K = K^\infty$

In infinite width limit, K_0 approaches K^∞ for general architectures[12]
The typical rate is $|K_0(x,x') - K^\infty(x,x')| = \tilde{O}(1/\sqrt{m})$ whp[13][14], so

Assumption 3

$m = \tilde{\Omega}(\epsilon^{-2})$ suffices to ensure that $\|K_0 - K^\infty\|^2_{L^2(X\times X, \rho\otimes\rho)} \leq \epsilon$ holds whp $1-\delta(m)$ over the initialization θ_0, where $\delta(m) = o(1)$.

[12] Yang 2020.

[13] Du et al. 2019; Du et al. 2018; Huang and Yau 2020, for fixed x, x'.

[14] Bowman and Montufar 2022; Buchanan, Gilboa, and Wright 2021, uniformly over x, x'.

Consequences for the special case $K = K^\infty$

Corollary 14

Under Assumption 3, setting $K = K^\infty$, we have $m = \tilde{\Omega}(T^4/\epsilon^2)$ and $n = \tilde{\Omega}(T^2/\epsilon^2)$ suffices to ensure with probability $1 - O(mn)\exp(-\Omega(\log^2 m) - \delta(m)$ that for all $t \leq T$ and $k \in \mathbb{N}$

$$\|P_k(r_t - \exp(-T_{K^\infty} t) r_0)\|^2_{L^2(X,\rho)} \leq \left[\frac{1 - \exp(-\sigma_k t)}{\sigma_k}\right]^2 \cdot \epsilon$$

and

$$\|r_t - \exp(-T_{K^\infty} t) r_0\|^2_{L^2(X,\rho)} \leq t^2 \cdot \epsilon.$$

Consequences for the special case $K = K^\infty$

- Corollary 14 states that up to time T, $r_t \approx \exp(-T_{K^\infty} t) r_0$
- Given that K^∞ tends to have a highly skewed spectrum, the magnitude of σ_i is particularly relevant on the convergence rate
- The bound on projected difference is smaller when σ_k is large. Thus bias along top eigenfunctions is particularly pronounced

Observation 15

At the beginning of training the network learns projections along eigenfunctions of NTK integral operator T_{K^∞} at rates given by the eigenvalues; particularly so for eigenfcts with large eigenvalues.

Summary

- Quantitative bounds on the L^2 difference in function space between a finite-width network trained on finite samples and the corresponding kernel method with infinite width and data.
- The network inherits the bias of the kernel at the beginning of training even when the width scales linearly with the sample size.
- Bias is not only over training data but over entire input space.

References I


Allen-Zhu, Zeyuan, Yuanzhi Li, and Zhao Song (2019). "A Convergence Theory for Deep Learning via Over-Parameterization". In: *Proceedings of the 36th International Conference on Machine Learning*. Vol. 97. Proceedings of Machine Learning Research. PMLR, pp. 242–252. URL: https://proceedings.mlr.press/v97/allen-zhu19a.html.


Arora, Sanjeev et al. (2019a). "Fine-Grained Analysis of Optimization and Generalization for Overparameterized Two-Layer Neural Networks". In: *Proceedings of the 36th International Conference on Machine Learning*. Vol. 97. Proceedings of Machine Learning Research. PMLR, pp. 322–332. URL: https://proceedings.mlr.press/v97/arora19a.html.


Arora, Sanjeev et al. (2019b). "Implicit regularization in deep matrix factorization". In: *Advances in Neural Information Processing Systems* 32.

References II

- Arora, Sanjeev et al. (2019c). "On Exact Computation with an Infinitely Wide Neural Net". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2019/file/dbc4d84bfcfe2284ba11beffb853a8c4-Paper.pdf.
- Azizan, Navid, Sahin Lale, and Babak Hassibi (2022). "Stochastic Mirror Descent on Overparameterized Nonlinear Models". In: *IEEE Transactions on Neural Networks and Learning Systems* 33.12, pp. 7717–7727. DOI: 10.1109/TNNLS.2021.3087480.
- Basri, Ronen et al. (2019). "The Convergence Rate of Neural Networks for Learned Functions of Different Frequencies". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2019/file/5ac8bb8a7d745102a978c5f8ccdb61b8-Paper.pdf.

References III

- Basri, Ronen et al. (2020). "Frequency Bias in Neural Networks for Input of Non-Uniform Density". In: *Proceedings of the 37th International Conference on Machine Learning*. Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 685–694. URL: https://proceedings.mlr.press/v119/basri20a.html.
- Beck, Amir and Marc Teboulle (2003). "Mirror Descent and Nonlinear Projected Subgradient Methods for Convex Optimization". In: *Operations Research Letters* 31.3, pp. 167–175.
- Bowman, Benjamin and Guido Montufar (2022). "Implicit Bias of MSE Gradient Optimization in Underparameterized Neural Networks". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=VLgmhQDVBV.

References IV


Bowman, Benjamin and Guido Montúfar (2022). "Spectral Bias Outside the Training Set for Deep Networks in the Kernel Regime". In: *Advances in Neural Information Processing Systems*. Vol. 35. Curran Associates, Inc., pp. 30362–30377. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/c4006ff54a7bbda74c09bad6f7586f5b-Paper-Conference.pdf.


Buchanan, Sam, Dar Gilboa, and John Wright (2021). "Deep Networks and the Multiple Manifold Problem". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=O-6Pm_d_Q-.

References V


Cao, Yuan et al. (Aug. 2021). "Towards Understanding the Spectral Bias of Deep Learning". In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*. Main Track. International Joint Conferences on Artificial Intelligence Organization, pp. 2205–2211. DOI: 10.24963/ijcai.2021/304. URL: https://doi.org/10.24963/ijcai.2021/304.


Du, Simon et al. (2019). "Gradient Descent Finds Global Minima of Deep Neural Networks". In: *Proceedings of the 36th International Conference on Machine Learning*. Vol. 97. Proceedings of Machine Learning Research. PMLR, pp. 1675–1685. URL: https://proceedings.mlr.press/v97/du19c.html.


Du, Simon S et al. (2018). "Gradient descent provably optimizes over-parameterized neural networks". In: *arXiv preprint arXiv:1810.02054*.

References VI


Dukler, Yonatan, Quanquan Gu, and Guido Montufar (2020). "Optimization Theory for ReLU Neural Networks Trained with Normalization Layers". In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 2751–2760. URL: https://proceedings.mlr.press/v119/dukler20a.html.


E, Weinan, Chao Ma, and Lei Wu (2020). "A comparative analysis of optimization and generalization properties of two-layer neural network and random feature models under gradient descent dynamics". In: *Science China Mathematics* 63, pp. 1235–1258.


Fan, Zhou and Zhichao Wang (2020). "Spectra of the Conjugate Kernel and Neural Tangent Kernel for Linear-Width Neural Networks". In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. NIPS'20. Vancouver, BC, Canada: Curran Associates Inc. ISBN: 9781713829546.

References VII

- Gunasekar, Suriya et al. (2017). "Implicit regularization in matrix factorization". In: *Advances in Neural Information Processing Systems* 30.
- Gunasekar, Suriya et al. (2018a). "Characterizing Implicit Bias in Terms of Optimization Geometry". In: *Proceedings of the 35th International Conference on Machine Learning*. Vol. 80. PMLR, pp. 1832–1841. URL: http://proceedings.mlr.press/v80/gunasekar18a.html.
- Gunasekar, Suriya et al. (2018b). "Implicit bias of gradient descent on linear convolutional networks". In: *Advances in Neural Information Processing Systems* 31.
- Huang, Jiaoyang and Horng-Tzer Yau (2020). "Dynamics of Deep Neural Networks and Neural Tangent Hierarchy". In: *Proceedings of the 37th International Conference on Machine Learning*. Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 4542–4551. URL: https://proceedings.mlr.press/v119/huang20l.html.

References VIII

- Jacot, Arthur, Franck Gabriel, and Clement Hongler (2018). "Neural Tangent Kernel: Convergence and Generalization in Neural Networks". In: *Advances in Neural Information Processing Systems*. Vol. 31. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2018/file/5a4be1fa34e62bb8a6ec6b91d2462f5a-Paper.pdf.
- Ji, Ziwei and Matus Telgarsky (2019a). "The implicit bias of gradient descent on nonseparable data". In: *Proceedings of the Thirty-Second Conference on Learning Theory*. Vol. 99. Proceedings of Machine Learning Research. PMLR, pp. 1772–1798. URL: https://proceedings.mlr.press/v99/ji19a.html.
- — (2019b). "The implicit bias of gradient descent on nonseparable data". In: *Proceedings of the Thirty-Second Conference on Learning Theory*. Ed. by Alina Beygelzimer and Daniel Hsu. Vol. 99. Proceedings of Machine Learning Research. Phoenix, USA: PMLR, pp. 1772–1798. URL: http://proceedings.mlr.press/v99/ji19a.html.

References IX


Ji, Ziwei and Matus Telgarsky (2020). "Polylogarithmic width suffices for gradient descent to achieve arbitrarily small test error with shallow ReLU networks". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=HygegyrYwH.


Jin, Hui and Guido Montufar (2023). "Implicit Bias of Gradient Descent for Mean Squared Error Regression with Two-Layer Wide Neural Networks". In: *Journal of Machine Learning Research* 24.137, pp. 1–97. URL: http://jmlr.org/papers/v24/21-0832.html.


Jin, Hui and Guido Montúfar (2023). "Implicit Bias of Gradient Descent for Mean Squared Error Regression with Two-Layer Wide Neural Networks". In: *Journal of Machine Learning Research* 24.137, pp. 1–97. URL: http://jmlr.org/papers/v24/21-0832.html.

References X


Karakida, Ryo, Shotaro Akaho, and Shun-ichi Amari (July 2021). "Pathological Spectra of the Fisher Information Metric and Its Variants in Deep Neural Networks". In: *Neural Computation* 33.8. _eprint: https://direct.mit.edu/neco/article-pdf/33/8/2274/1930880/neco_a_01411.pdf, pp. 2274–2307. ISSN: 0899-7667. DOI: 10.1162/neco_a_01411. URL: https://doi.org/10.1162/neco_a_01411.


Karhadkar, Kedar, Michael Murray, and Guido Montúfar (2024). *Bounds for the smallest eigenvalue of the NTK for arbitrary spherical data of arbitrary dimension*. arXiv: 2405.14630 [stat.ML]. URL: https://arxiv.org/abs/2405.14630.


Karhadkar, Kedar et al. (2024). *Benign overfitting in leaky ReLU networks with moderate input dimension*. arXiv: 2403.06903 [cs.LG]. URL: https://arxiv.org/abs/2403.06903.

References XI

- Lee, Jaehoon et al. (2019a). "Wide Neural Networks of Any Depth Evolve as Linear Models Under Gradient Descent". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., pp. 8572–8583.
- — (2019b). "Wide Neural Networks of Any Depth Evolve as Linear Models Under Gradient Descent". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2019/file/0d1a9651497a38d8b1c3871c84528bd4-Paper.pdf.
- Li, Mingchen, Mahdi Soltanolkotabi, and Samet Oymak (2020). "Gradient Descent with Early Stopping is Provably Robust to Label Noise for Overparameterized Neural Networks". In: *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*. Vol. 108. Proceedings of Machine Learning Research. PMLR, pp. 4313–4324. URL: https://proceedings.mlr.press/v108/li20j.html.

References XII

- Li, Yuanzhi, Tengyu Ma, and Hongyang Zhang (2018). "Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations". In: *Conference On Learning Theory*. PMLR, pp. 2–47.
- Liang, Shuang and Guido Montúfar (2024). "Implicit Bias of Mirror Descent for Shallow Neural Networks in Univariate Regression". In: *TBD*.
- Luo, Tao et al. (2022). "On the Exact Computation of Linear Frequency Principle Dynamics and Its Generalization". In: *SIAM Journal on Mathematics of Data Science* 4.4, pp. 1272–1292. DOI: 10.1137/21M1444400. eprint: https://doi.org/10.1137/21M1444400. URL: https://doi.org/10.1137/21M1444400.
- Murray, Michael et al. (2023). "Characterizing the spectrum of the NTK via a power series expansion". In: *The Eleventh International Conference on Learning Representations*. URL: https://openreview.net/forum?id=Tvms8xrZHyR.

References XIII


Nemirovskij, Arkadij Semenovič and David Borisovich Yudin (1983). *Problem Complexity and Method Efficiency in Optimization*. A Wiley-Interscience publication. Wiley. ISBN: 9780471103455. URL: https://books.google.com.tw/books?id=6ULvAAAAMAAJ.


Nguyen, Quynh (2021). "On the Proof of Global Convergence of Gradient Descent for Deep ReLU Networks with Linear Widths". In: *Proceedings of the 38th International Conference on Machine Learning*. Vol. 139. Proceedings of Machine Learning Research. PMLR, pp. 8056–8062. URL: https://proceedings.mlr.press/v139/nguyen21a.html.


Nguyen, Quynh and Marco Mondelli (2020). "Global Convergence of Deep Networks with One Wide Layer Followed by Pyramidal Topology". In: *Advances in Neural Information Processing Systems*. Vol. 33. Curran Associates, Inc., pp. 11961–11972. URL: https://proceedings.neurips.cc/paper/2020/file/8abfe8ac9ec214d68541fcb888c0b4c3-Paper.pdf.

References XIV


Nguyen, Quynh, Marco Mondelli, and Guido F Montufar (2021). "Tight Bounds on the Smallest Eigenvalue of the Neural Tangent Kernel for Deep ReLU Networks". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, pp. 8119–8129. URL: https://proceedings.mlr.press/v139/nguyen21g.html.


Oymak, Samet and Mahdi Soltanolkotabi (2020). "Toward Moderate Overparameterization: Global Convergence Guarantees for Training Shallow Neural Networks". In: *IEEE Journal on Selected Areas in Information Theory* 1.1, pp. 84–105.


Oymak, Samet et al. (2020). *Generalization Guarantees for Neural Networks via Harnessing the Low-rank Structure of the Jacobian*. URL: https://openreview.net/forum?id=ryl5CJSFPS.

References XV

- Pennington, Jeffrey and Yasaman Bahri (2017). "Geometry of Neural Network Loss Surfaces via Random Matrix Theory". In: *Proceedings of the 34th International Conference on Machine Learning*. Vol. 70. Proceedings of Machine Learning Research. PMLR, pp. 2798–2806. URL: https://proceedings.mlr.press/v70/pennington17a.html.
- Pennington, Jeffrey and Pratik Worah (2018). "The Spectrum of the Fisher Information Matrix of a Single-Hidden-Layer Neural Network". In: *Advances in Neural Information Processing Systems*. Vol. 31. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2018/file/18bb68e2b38e4a8ce7cf4f6b2625768c-Paper.pdf.
- Pesme, Scott, Radu-Alexandru Dragomir, and Nicolas Flammarion (2024). "Implicit Bias of Mirror Flow on Separable Data". In: *arXiv preprint arXiv:2406.12763*.

References XVI

- Qiao, Dan et al. (2024). *Stable Minima Cannot Overfit in Univariate ReLU Networks: Generalization by Large Step Sizes*. arXiv: 2406.06838 [cs.LG]. URL: https://arxiv.org/abs/2406.06838.
- Rahaman, Nasim et al. (2019a). "On the spectral bias of neural networks". In: *International Conference on Machine Learning*. PMLR, pp. 5301–5310.
- — (2019b). "On the Spectral Bias of Neural Networks". In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. Long Beach, California, USA: PMLR, pp. 5301–5310. URL: http://proceedings.mlr.press/v97/rahaman19a.html.
- Razin, Noam and Nadav Cohen (2020). "Implicit regularization in deep learning may not be explainable by norms". In: *Advances in neural information processing systems* 33, pp. 21174–21187.

References XVII


Savarese, Pedro et al. (2019). "How do infinite width bounded norm networks look in function space?" In: *Proceedings of the Thirty-Second Conference on Learning Theory*. Ed. by Alina Beygelzimer and Daniel Hsu. Vol. 99. Proceedings of Machine Learning Research. Phoenix, USA: PMLR, pp. 2667–2690. URL: http://proceedings.mlr.press/v99/savarese19a.html.


Shevchenko, Alexander, Vyacheslav Kungurtsev, and Marco Mondelli (2022). "Mean-field Analysis of Piecewise Linear Solutions for Wide ReLU Networks". In: *Journal of Machine Learning Research* 23.130, pp. 1–55. URL: http://jmlr.org/papers/v23/21-1365.html.


Soudry, Daniel et al. (2018). "The implicit bias of gradient descent on separable data". In: *Journal of Machine Learning Research* 19.1, pp. 2822–2878.

References XVIII

- Su, Lili and Pengkun Yang (2019). "On Learning Over-parameterized Neural Networks: A Functional Approximation Perspective". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2019/file/253f7b5d921338af34da817c00f42753-Paper.pdf.
- Sun, Haoyuan et al. (2022). "Mirror Descent Maximizes Generalized Margin and Can Be Implemented Efficiently". In: *Advances in Neural Information Processing Systems*. Ed. by Alice H. Oh et al. URL: https://openreview.net/forum?id=0SVOleKNRAU.
- Sun, Haoyuan et al. (2023). "A Unified Approach to Controlling Implicit Regularization via Mirror Descent". In: *arXiv preprint arXiv:2306.13853*.

References XIX

- Velikanov, Maksim and Dmitry Yarotsky (2021). "Explicit loss asymptotics in the gradient descent training of neural networks". In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 2570–2582. URL: https://proceedings.neurips.cc/paper/2021/hash/14faf969228fc18fcd4fcf59437b0c97-Abstract.html.
- Williams, Francis et al. (2019). "Gradient dynamics of shallow univariate ReLU networks". In: *Advances in Neural Information Processing Systems*, pp. 8378–8387.
- Wu, Fan and Patrick Rebeschini (2021). "Implicit Regularization in Matrix Sensing via Mirror Descent". In: *Advances in Neural Information Processing Systems* 34, pp. 20558–20570.
- Yang, Greg (2020). *Tensor Programs II: Neural Tangent Kernel for Any Architecture*. DOI: 10.48550/ARXIV.2006.14548. URL: https://arxiv.org/abs/2006.14548.

References XX

- Yang, Greg and Hadi Salman (2019). *A Fine-Grained Spectral Perspective on Neural Networks*. DOI: 10.48550/ARXIV.1907.10599. URL: https://arxiv.org/abs/1907.10599.
- Zhang, Chiyuan et al. (2021). "Understanding deep learning (still) requires rethinking generalization". In: *Communications of the ACM* 64.3, pp. 107–115.
- Zhang, Yaoyu et al. (2019). "A type of generalization error induced by initialization in deep neural networks". In: *arXiv preprint arXiv:1905.07777*.
- — (2020). "A type of generalization error induced by initialization in deep neural networks". In: *Proceedings of The First Mathematical and Scientific Machine Learning Conference*. Vol. 107. Proceedings of Machine Learning Research. PMLR, pp. 144–164. URL: https://proceedings.mlr.press/v107/zhang20a.html.

References XXI


Zou, Difan and Quanquan Gu (2019). "An Improved Analysis of Training Over-parameterized Deep Neural Networks". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2019/file/6a61d423d02a1c56250dc23ae7ff12f3-Paper.pdf.


Zou, Difan et al. (2020). "Gradient descent optimizes over-parameterized deep ReLU networks". In: *Machine learning* 109, 467–492.