Computational Mathematics and AI

Lecture 4: Modern Theory of Stochastic Gradient Descent

Lars Ruthotto

Departments of Mathematics and Computer Science

lruthotto@emory.edu

larsruthotto

slido.com #CBMS25

Reading List

Historical Context: Modern understanding of SGD extends beyond convergence to implicit regularization, landscape structure, and continuous-time dynamics.

Key Readings:

1. Hardt et al. (2016) – Train Faster, Generalize Better: Stability of SGD. *ICML*

   Early stopping and implicit regularization.

2. Keskar et al. (2017) – Large-Batch Training: Generalization Gap and Sharp Minima. *ICLR*

   Batch size effects on generalization.

3. Cohen et al. (2021a) – Gradient Descent at the Edge of Stability. *ICLR*

   Neural networks train near stability boundary.

4. Jacot et al. (2018) – Neural Tangent Kernel. *NeurIPS*

   Infinite-width networks and lazy training.

5. Mei et al. (2018) – A Mean Field View of Two-Layers Neural Networks.
   Proves that SGD dynamics, in scaling limit, are governed by a nonlinear PDE.

Lecture Outline: SGD Properties $\rightarrow$ Learning Rate/Batch Size $\rightarrow$ Continuous-Time $\rightarrow$ Implicit Regularization $\rightarrow$ Landscape

Connection to Lecture 3

Lecture 3: SA/SAA, Gauss-Newton, SGD basics

- SA/SAA framework for optimization under uncertainty
- Backpropagation: Efficient gradient computation
- Computational challenges for Gauss-Newton
- Key observation: **Lazy regime works surprisingly well!**

Guiding question for this lecture:

Why does SGD in the lazy regime perform comparably to Gauss-Newton?

Roadmap: Modern theory of SGD

1. Flat vs. sharp minima: geometry and generalization
2. Implicit regularization of continuous-time SGD
3. Regularization effects of finite step size SGD
4. Over-parameterization: Neural tangent kernel and mean field perspectives

Flat vs. Sharp Minima

Flat vs. Sharp Minima: Geometry and Generalization

Definition: At a local minimum θ^*, let $H = \nabla^2 \mathcal{L}(\theta^*)$ be the Hessian.

- **Sharp minimum:** $\lambda_{\max}(H) \gg 0$ (high curvature, loss rises quickly)
- **Flat minimum:** $\lambda_{\max}(H) \approx 0$ (low curvature, loss rises slowly)

Generalization hypothesis: Based on empirical Evidence

- Flat minima correlate with better test error (robust to perturbations)
- Sharp minima correlate with overfitting
- Small-batch SGD finds flatter minima than large-batch Keskar et al. (2017)

Caveat: Diagonal reparameterization $\theta' = D\theta$ with $D = \text{diag}(\alpha_1, \ldots, \alpha_p)$:

$$\nabla^2_{\theta'} \mathcal{L} = D^{-1} \nabla^2_{\theta} \mathcal{L} \, D^{-1}$$

Eigenvalues scale independently $\Rightarrow$ curvature arbitrary without changing function!

The Curvature Measurement Problem

Reparameterization is necessary:

- $\lambda_{\max}(H)$ is **not invariant** to parameter rescaling
- Same function can appear arbitrarily sharp or flat depending on parameterization
- Hessian eigenvalues unreliable as generalization predictors Dinh et al. (2017)

Resolution: Fisher Information as metric Amari (1998)

Information Geometric Sharpness (IGS): Gradient norm in Fisher metric

$$\text{IGS}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \nabla_\theta \ell_i^\top \cdot \mathbf{F}(\theta)^\dagger \cdot \nabla_\theta \ell_i, \quad \text{with} \quad \mathbf{F}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{J}_i^\top \mathbf{J}_i$$

Using $\mathbf{J}_i = \nabla_\theta F_\theta(\mathbf{x}_i)$ from Lecture 3.

IGS changes in *predictions* per unit parameter change $\Rightarrow$ reparameterization-invariant

Continuous-Time Limit of SGD

Review: Gradient Flow ODE & Minimum Norm Bias

$$\frac{d\theta}{dt} = -\nabla\mathcal{L}(\theta(t)) \quad \text{as limit of GD} \quad \theta_{t+1} = \theta_t - \eta\nabla\mathcal{L}(\theta_t)$$

For Linear Models: $\mathcal{L}(\theta) = \|A\theta - b\|^2$ with $\theta(0) = 0$

1. **Time-dependent regularization**:

$$\theta(t) = \arg\min_{\theta} \|A\theta - b\|^2 + \frac{1}{t}\|\theta\|^2$$

   Early stopping at time $t \equiv$ regularization strength $1/t$
2. **Asymptotic limit** ($t \to \infty$): Minimum norm bias

$$\theta_\infty = \arg\min \{\|\theta\|_2 : \mathcal{L}(\theta) = 0\}$$

   Among all global minima, GD selects the one closest to initialization

Open problem: Extension to (Nonlinear) Neural Networks:

- What is the relevant "complexity measure"? Not simple ℓ_2 norm
- Connection to margin maximization, flatness, NTK regime

early stopping $\equiv$ implicit regularization; GD prefers simple solutions

From SGD to Stochastic Gradient Flow

Recall that SGD uses Monte Carlo estimates of the gradient:

$$\begin{aligned}\theta_{t+1} &= \theta_t - \eta \left[\frac{1}{b} \sum_{j=1}^{b} \nabla_\theta \ell(F_{\theta_t}(\mathbf{x}_j), \mathbf{y}_j) \right] \\ &= \theta_t - \eta \nabla \mathcal{L}(\theta_t) + \frac{\eta}{\sqrt{b}} \Sigma^{1/2}(\theta_t) \xi_t \quad \text{where } \xi_t \sim \mathcal{N}(0, I)\end{aligned}$$

Continuous limit yields Stochastic Gradient Flow (SGF):

$$d\theta_t = -\nabla \mathcal{L}(\theta_t)\, dt + \sqrt{2D(\theta_t)}\, dW_t, \quad \text{where} \quad D(\theta_t) = \frac{\eta}{2b} \Sigma(\theta_t) \text{ (diffusion matrix)}$$

Connection to implicit regularization for least squares Ali et al. (2020):

- **Mean trajectory**: $\mathbb{E}[\theta_{\mathsf{SGF}}(t)] = \theta_{\mathsf{GF}}(t)$
- SGF follows Tikhonov regularization path *in expectation*

But what does SGD converge to as $t \to \infty$?

Deriving the Stationary Distribution of SGF

Reminder (Feynman-Kac): SDE for $x(t)$ induces PDE for density $p(x,t)$.

$$dx = -g(x)\,dt + \sqrt{2D}\,dW \quad \Rightarrow \quad \frac{\partial p}{\partial t} = \nabla \cdot [g\,p + D\,\nabla p]$$

Langevin approximation: For $D(\theta) = \frac{1}{2}\epsilon I$ ($\epsilon = \eta/b$), SGF leads to Fokker Planck:

$$d\theta_t = -\nabla \mathcal{L}(\theta_t)\,dt + \sqrt{\epsilon}\,dW_t \quad \Rightarrow \quad \frac{\partial p}{\partial t} = \nabla \cdot \left[\nabla \mathcal{L}(\theta)\,p(\theta) + \frac{\epsilon}{2}\nabla p(\theta)\right]$$

Stationary condition: At equilibrium, detailed balance holds if

$$\nabla \mathcal{L}(\theta)\,p(\theta) + \frac{\epsilon}{2}\nabla p(\theta) = \mathbf{0} \quad \Rightarrow \quad \nabla p(\theta) = -\frac{2}{\epsilon}\nabla \mathcal{L}(\theta) p(\theta)$$

This holds for **Gibbs distribution**

$$p(\theta) = Z^{-1} \exp\left(-\frac{2\mathcal{L}(\theta)}{\epsilon}\right)$$

Interpreting the Stationary Distribution of SGF

Langevin SDE converges to Gibbs distribution with temperature ϵ:

$$p(\theta) \propto \exp\left(-\frac{2\mathcal{L}(\theta)}{\epsilon}\right), \quad \text{where} \quad \epsilon = \frac{\eta}{b}$$

Interpretation:

- **Flat minima** = wide basins = high probability at finite ϵ
- **Sharp minima** = narrow basins = low probability at finite ϵ
- Temperature $\epsilon \propto \eta/b$ controls smoothness of distribution

Consequences:

- **Small** ϵ (large batches): concentrate near global minimum
- **Large** ϵ (small batches): broader exploration, prefer flat minima

Continuous-time perspective shows implicit bias and sampling perspective

Limitation of Cont' Time: Edge of Stability Phenomenon

Classical stability condition: GD stable if $\eta\lambda_{\max}(H) < 2$
where $\lambda_{\max}(H)$ is maximum eigenvalue of Hessian $H = \nabla^2\mathcal{L}(\theta)$.

Empirical observation Cohen et al. (2021b):
In deep learning, training often operates at

$$\boxed{\lambda_{\max}(H) \approx \frac{2}{\eta}}$$

This is *exactly at the stability boundary*!

The catapult mechanism:

1. GD moves toward minimum, curvature increases
2. When $\eta\lambda_{\max}(H) > 2$, GD overshoots
3. System "catapults" out of sharp region
4. Curvature settles back to $\lambda_{\max}(H) \approx 2/\eta$

In practice, $\eta \not\to 0$. Need to understand finite step effects...

Regularization of Finite Step Size SGD

Backward Error Analysis: Setup

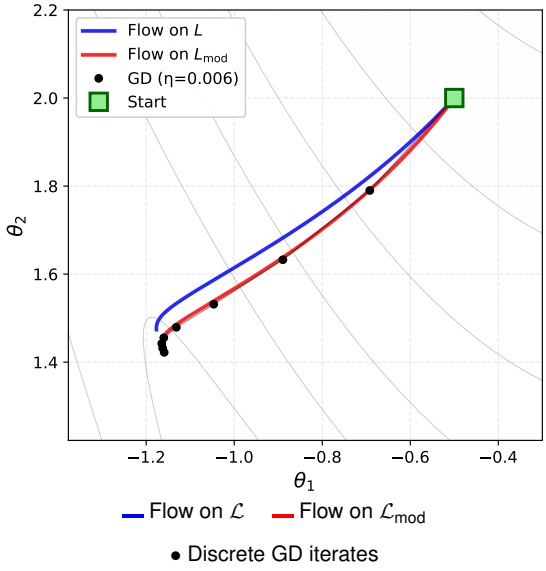
Edge of stability shows importance of finite step size:

$$\theta_{t+1} = \theta_t - \eta \nabla \mathcal{L}(\theta_t) \neq \frac{d\theta}{dt} = -\nabla \mathcal{L}(\theta)$$

Goal: Find $\mathcal{L}_{\text{mod}}$ such that discrete GD follows gradient flow on $\mathcal{L}_{\text{mod}}$

Tool: Backward Error Analysis.

Derive the modified loss that discrete GD optimizes



Derivation Part 1: Taylor Expansion in Time

Setup: Discrete trajectory satisfies $\theta(t+\eta) \approx \theta_{t+1}$ and (by Taylor in η)

$$\theta(t+\eta) = \theta(t) + \eta \frac{d\theta}{dt}\bigg|_t + \frac{\eta^2}{2}\frac{d^2\theta}{dt^2}\bigg|_t + O(\eta^3)$$

Setting this equal to discrete update $\theta(t+\eta) = \theta(t) - \eta\nabla\mathcal{L}(\theta(t))$ gives:

$$\eta\frac{d\theta}{dt} + \frac{\eta^2}{2}\frac{d^2\theta}{dt^2} = -\eta\nabla\mathcal{L}(\theta) \quad \Rightarrow \quad \boxed{\frac{d\theta}{dt} + \frac{\eta}{2}\frac{d^2\theta}{dt^2} = -\nabla\mathcal{L}(\theta)}$$

To find a modified GF $\frac{d\theta}{dt} = -\nabla\mathcal{L}_{\text{mod}}(\theta)$, note

$$\frac{d^2\theta}{dt^2} = -\frac{d}{dt}[\nabla\mathcal{L}_{\text{mod}}] = -\nabla^2\mathcal{L}_{\text{mod}} \cdot \frac{d\theta}{dt} = \nabla^2\mathcal{L}_{\text{mod}} \cdot \nabla\mathcal{L}_{\text{mod}}$$

Idea: Write $\mathcal{L}_{\text{mod}} = \mathcal{L} + \frac{\eta}{4}\|\nabla\mathcal{L}\|^2$, then (see Barrett and Dherin (2021))

$$\nabla\mathcal{L}_{\text{mod}} = \nabla\mathcal{L} + \frac{\eta}{2}\nabla^2\mathcal{L} \cdot \nabla\mathcal{L} + \mathcal{O}(\eta^2)$$

Finite steps implicitly add a gradient magnitude penalty!

Interpretation and Implications

Modified loss breakdown:

$$\mathcal{L}_{\text{mod}}(\theta) = \underbrace{\mathcal{L}(\theta)}_{\text{training loss}} + \underbrace{\frac{\eta}{4}\|\nabla\mathcal{L}(\theta)\|^2}_{\text{gradient penalty}} + O(\eta^2)$$

Flatness preference:

- Discrete GD prefers regions where $\|\nabla\mathcal{L}\| \approx 0$
- Not just low loss, but **flat loss landscape**!
- Penalty strength controlled by learning rate η

Larger $\eta \to$ stronger implicit regularization

- Small η: weak penalty, nearly pure GD
- Moderate η: balanced trade-off
- Large η: strong flatness bias (but may not converge!)

Finite η is a feature, not a bug: it creates implicit regularization!

Extension to Stochastic Gradient Descent

Recall from Block 2: SGD noise leads to Langevin dynamics

$$d\theta = -\nabla\mathcal{L}(\theta)dt + \sqrt{\epsilon}dW, \quad \epsilon \propto \eta/b$$

For discrete SGD: Similar backward error analysis applies

Key assumptions needed:

1. Gradient noise is approximately isotropic (common in practice)
2. Noise variance scales as σ^2/b (standard assumption)

Result: Same modified loss structure + temperature effects Smith et al. (2021)

$$\mathcal{L}_{\text{mod}}(\theta) = \mathcal{L}(\theta) + \frac{\eta}{4}\|\nabla\mathcal{L}(\theta)\|^2 + O(\eta/b) + O(\eta^2)$$

Note: For $b \gg 1$, $O(\eta/b) \ll O(\eta)$; for small batches ($b = O(1)$), both effects are $O(\eta)$

Temperature $\epsilon = \eta/b$ controls additional noise-driven exploration

- Small batches: more noise, broader exploration of flat regions
- Large batches: less noise, sharper convergence to nearest minimum

SGD combines gradient penalty (finite η) AND noise exploration (small b)

Finite Step Size Advantage

Recall from previous slide: Discrete GD optimizes

$$\mathcal{L}_{\text{mod}}(\theta) = \mathcal{L}(\theta) + \frac{\eta}{4}\|\nabla\mathcal{L}(\theta)\|^2$$

Consequence: Larger $\eta \to$ stronger flatness preference

Explains empirical observations [Evidence]:

- Moderate learning rates ($\eta \in [0.01, 0.1]$) generalize better than tiny η
- Finite η acts as implicit regularizer
- "Sweet spot" balances convergence speed vs. implicit regularization

Connection to generalization:

- Flat minima $\to$ small $\|\nabla L\|$ throughout basin
- Flat minima $\to$ robust to perturbations
- Robustness often correlates with test performance

take away: finite steps are a feature, not a bug!

Over-Parametrization

The Over-Parametrization Phenomenon

Modern neural networks: Parameters p far exceed training samples n

Examples:

- GPT-3: $p \approx 175$ billion parameters
- ResNet-50: $p \approx 25$ million on ImageNet ($n = 1.2$ million)
- Our peaks example: $p = 128 \times 2 + 128 + 5 \times 128 + 5 = 901$ on $n = 600$

Classical learning theory prediction:

- $p \gg n$ should lead to catastrophic overfitting
- Infinitely many interpolating solutions (training loss = 0)
- No reason to expect good generalization

Reality:

- Over-parametrization often *improves* generalization (recall double descent)
- Training converges reliably from random initialization
- Lazy regime works surprisingly well (Lecture 3)

Goal: Understand why the SGD finds good weights for huge networks

From Gauss-Newton to Neural Tangent Kernel

Goal: Find θ^* such that $f_{\theta^*}(\mathbf{x}) = \mathbf{y}$ for all training data pairs $(\mathbf{x}, \mathbf{y})$

Second-order Taylor expansion: Concatenate all data into vectors $\mathbf{X}$, $\mathbf{Y}$

$$\mathbf{Y} = f_{\theta^*}(\mathbf{X}) = f_{\theta_0}(\mathbf{X}) + J_{\theta_0}(\mathbf{X})\,\delta\theta + \frac{1}{2}\delta\theta^T \nabla^2_{\tilde{\theta}} f(\mathbf{X})\,\delta\theta$$

where $\tilde{\theta}$ lies between θ_0 and θ^*, $\quad \delta\theta = \theta^* - \theta_0$, $\quad$ Jaobian $J = \nabla_\theta f(\mathbf{X})|_{\theta_0}$

Assume: Squared loss function and quadratic term is negligible

$$\mathbf{Y} - f_{\theta_0}(\mathbf{X}) \approx J_{\theta_0}(\mathbf{X})\,\delta\theta \quad \Longrightarrow \quad \text{solve } J_{\theta_0}(\mathbf{X})\,\delta\theta = \mathbf{R} \text{ where } \mathbf{R} = \mathbf{Y} - f_{\theta_0}(\mathbf{X})$$

Consequence: Gauss-Newton converges in one step.

Can we design network, so that we fit all data and the quadratic term vanishes?

The NTK Parameterization: A Worked Example

Architecture: Single hidden layer with width n

$$f(\mathbf{x}) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} a_i \sigma(\mathbf{w}_i^T \mathbf{x}) = \frac{1}{\sqrt{n}} \mathbf{a}^T \sigma(W\mathbf{x})$$

with $a_i \sim \mathcal{N}(0,1)$, $\mathbf{w}_i \sim \mathcal{N}(0, I_d/d)$, total parameters $p = n(1+d)$

Why $1/\sqrt{n}$? (architectural factor, NOT initialization variance)

- Each term $a_i \sigma(\mathbf{w}_i^T \mathbf{x}) = O(1)$ (i.i.d.)
- Sum of n terms would be $O(\sqrt{n})$ by CLT $\rightarrow$ too large!
- The $1/\sqrt{n}$ ensures $f(\mathbf{x}) = O(1)$ at initialization

Jacobian: $\dfrac{\partial f}{\partial a_i} = \dfrac{1}{\sqrt{n}} \sigma(\mathbf{w}_i^T \mathbf{x}) = O(1/\sqrt{n}), \quad \dfrac{\partial f}{\partial w_{ij}} = \dfrac{1}{\sqrt{n}} a_i \sigma'(\mathbf{w}_i^T \mathbf{x}) x_j = O(1/\sqrt{n})$

Hessian (of f, not loss!): $\dfrac{\partial^2 f}{\partial a_i \partial a_j} = 0, \quad \dfrac{\partial^2 f}{\partial w_{ik} \partial w_{jl}} = \dfrac{\delta_{ij}}{\sqrt{n}} a_i \sigma''(\mathbf{w}_i^T \mathbf{x}) x_k x_l = O(1/\sqrt{n})$

the $1/\sqrt{n}$ appears once in Jacobian, once in Hessian

The Dual Perspective: From Parameters to Functions

Assume overparameterization $(p > N)$: $J_{\theta_0}(\mathbf{X})\,\delta\theta = \mathbf{R}$ has infitely many solutions

Representer theorem: The minimum-norm solution has the form

$$\delta\theta^* = J_{\theta_0}(\mathbf{X})^T\alpha \quad \text{for some } \alpha \in \mathbb{R}^N$$

Substitute into $J_{\theta_0}(\mathbf{X})\,\delta\theta = \mathbf{R}$**:**

$$J_{\theta_0}(\mathbf{X})(J_{\theta_0}(\mathbf{X})^T\alpha) = \mathbf{R} \quad \Longrightarrow \quad (J_{\theta_0}(\mathbf{X})J_{\theta_0}(\mathbf{X})^T)\alpha = \mathbf{R}$$

Define: $K = J_{\theta_0}(\mathbf{X})J_{\theta_0}(\mathbf{X})^T \in \mathbb{R}^{N\times N}$ (the **NTK Gram matrix**)

Solution: $\alpha = K^{-1}\mathbf{R}$, so $\boxed{\delta\theta^* = J_{\theta_0}(\mathbf{X})^T K^{-1}(\mathbf{Y} - f_{\theta_0}(\mathbf{X}))}$

Why K **converges** Jacot et al. (2018): $K = \sum_{i=1}^{p}(\nabla_{\theta_i} f)(\nabla_{\theta_i} f)^T$
Sum of p rank-1 matrices, each $O(1/p) \to$ deterministic (spd!) limit by LLN

shift from p-dimensional parameter space to N-dimensional function space

Why the Quadratic Term Vanishes

The key bound:

$$|\delta\theta^T \nabla^2_\theta f(\xi)\, \delta\theta| \leq \|\nabla^2_\theta f(\xi)\|_2 \cdot \|\delta\theta\|_2^2$$

1. Parameter change is bounded: $\|\delta\theta\|_2 = O(1)$

- Min-norm solution: $\delta\theta^* = J_{\theta_0}^T (J_{\theta_0} J_{\theta_0}^T)^{-1}\mathbf{R}$ (recall: GD $\to$ min-norm)
- $K = J_{\theta_0} J_{\theta_0}^T \in \mathbb{R}^{N\times N}$: $K_{ij} = \sum_{k=1}^{p} O(1/n) = O(1)$ (N fixed, $p = O(n)$)
- So $K, K^{-1}, \mathbf{r} = O(1) \Rightarrow \|\delta\theta^*\|_2^2 = \mathbf{r}^T K^{-1}\mathbf{r} = O(1)$

2. Hessian spectral norm vanishes: $\|\nabla^2_\theta f\|_2 = O(1/\sqrt{n})$

- From previous slide: each non-zero entry is $O(1/\sqrt{n})$
- Block-diagonal structure: spectral norm = max block norm = $O(1/\sqrt{n})$
- **Key:** The $1/\sqrt{n}$ factor appears in every second derivative

The conclusion:

$$\boxed{|\delta\theta^T \nabla^2_\theta f(\xi)\, \delta\theta| \leq O(1/\sqrt{n}) \cdot O(1) = O(1/\sqrt{n}) \to 0}$$

Hessian vanishes and change in weights stays bounded!

Connection to Kernel Methods & Lecture 3

Prediction at training data:

$$f^*(\mathbf{X}) = f_{\theta_0}(\mathbf{X}) + KK^{-1}(\mathbf{Y} - f_{\theta_0}(\mathbf{X})) = \mathbf{Y}$$

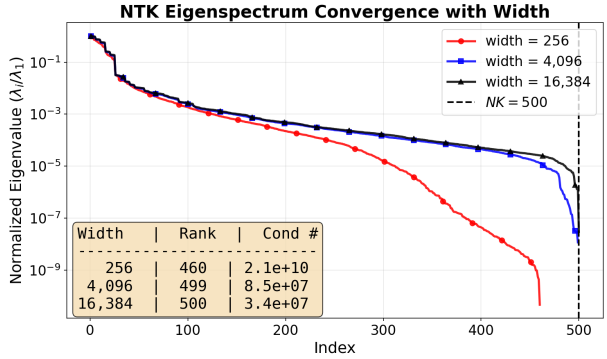
Prediction at new point $\mathbf{x}_{\text{new}}$:

$$f^*(\mathbf{x}_{\text{new}}) = f_{\theta_0}(\mathbf{x}_{\text{new}}) + K(\mathbf{x}_{\text{new}}, X)\, K^{-1}(\mathbf{Y} - f_{\theta_0}(\mathbf{X}))$$

where $K(\mathbf{x}_{\text{new}}, X) = J_{\theta_0}(\mathbf{x}_{\text{new}}) J_{\theta_0}(\mathbf{X})^T$

This IS kernel regression!

NTK: lazy training $\rightarrow$ kernel methods!



NTK eigenspectrum converges as width $\rightarrow \infty$

NTK Regime: Limitations

The NTK theory is elegant, but has important limitations:

1. No feature learning

- Kernel K is fixed at random initialization
- Network cannot adapt representations to the task
- Features are “frozen”: only linear combinations change

2. Infinite-width idealization

- Real networks have finite width and DO learn features
- Finite-width networks often outperform NTK predictions
- The “rich” or “feature learning” regime exists beyond lazy

3. Gap between theory and practice

- NTK explains convergence but not why learned features help
- Modern architectures (transformers) show clear feature learning
- Active research: when does feature learning emerge?

NTK explains lazy regime; finite-width networks can do more

Beyond Lazy: The Mean-Field Viewpoint

Key insight: Different scaling leads to different infinite-width limits

NTK approach (lazy regime):

- Initialize $w_j \sim \mathcal{N}(0, 1/n), a_j \sim \mathcal{N}(0, 1)$, output scaled by $1/\sqrt{n}$
- Kernel K fixed at initialization $\Rightarrow$ no feature learning

Mean-field approach (feature learning regime):

- Goal: Force features to evolve for every n
- Initialize $w_j \sim \mathcal{N}(0, 1)$, keep $a_j = 1$, output scaled by $1/n$
- Track the *distribution* of weights $w_t \sim \mu_t$ rather than individual parameters

Particle interpretation:

- Each neuron is a "particle" in weight space
- The population of particles evolves collectively
- Width sufficiently large $\rightarrow$ no need to track particles individually

mean-field scaling allows feature learning in the infinite-width limit

Distributional Dynamics (DD)

Setup: Two-layer network with mean-field scaling

$$f(x;\theta) = \frac{1}{n}\sum_{j=1}^{n} \sigma(w_j \cdot x)$$

Key result Mei et al. (2018) and Chizat and Bach (2018):
As $n \to \infty$, SGD dynamics converge to a **PDE on measure space**:

$$\partial_t \mu_t = \nabla_w \cdot \left(\mu_t \nabla_w \frac{\delta \mathcal{L}}{\delta \mu}(\mu_t) \right)$$

where $\frac{\delta \mathcal{L}}{\delta \mu}$ is the functional derivative of the loss.

Interpretation: This is a **Wasserstein gradient flow** of the loss functional.

With SGD noise $\to$ Fokker-Planck equation:

$$\partial_t \mu_t = \nabla_w \cdot \left(\mu_t \nabla_w \frac{\delta \mathcal{L}}{\delta \mu} \right) + \frac{1}{\beta} \Delta_w \mu_t$$

SGD on parameters $\to$ gradient flow on probability measures

Mean-Field Theory: Implications

What mean-field theory provides:

1. Global convergence: Proven for 2-layer networks

- Gradient flow on μ_t converges to global optimum
- Loss landscape has “no bad traps” in distribution space

2. Stochastic attractivity

- SGD noise drives system toward simpler solutions
- Implicit bias toward low-complexity subnetworks

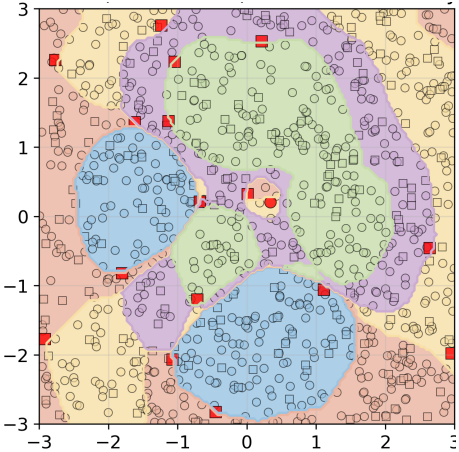
Limitations: Open problems

- Rigorous results mainly for shallow (2-layer) networks
- Extension to deep networks is **active research**
- Gap between mean-field limit and practical finite-width behavior

mean-field: rigorous foundation for feature learning; deep theory remains open

Mean Field Network (width=4096): Adam

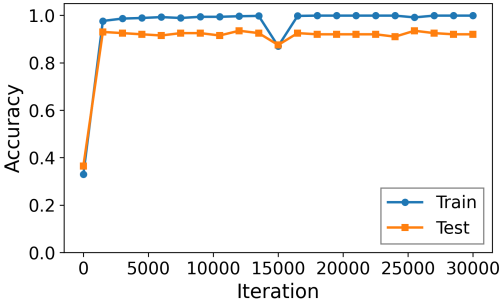
Predicted decision boundary:

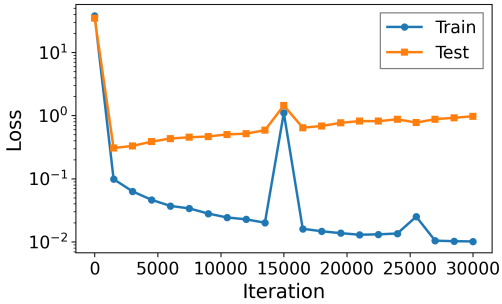


Final accuracy:

Train: 99.87% — Test: 92.00%

Convergence dynamics:





NTK vs Mean-Field: Comparison

Feature	NTK (Lazy)	Mean-Field
Infinite-width limit	Fixed kernel	Weight distribution evolves
Parameter behavior	Stay near initialization	Particles move freely
Dynamics	Linear (kernel fixed)	Nonlinear PDE
Feature learning	No	Yes
Math framework	Kernel regression	Wasserstein gradient flow
Proven results	Convergence to RKHS	Global optima (shallow)

The relationship:

- NTK is a *special case*: the "zero learning" limit
- Mean-field captures dynamics when features are allowed to evolve
- Real networks operate *between* these two regimes

NTK = lazy limit; mean-field = feature learning limit

Summary and Outlook

Σ: Modern SGD Theory

1. **Convergence Properties and Sampling Perspective:**
   - Unbiased gradient estimates, Robbins-Monro conditions
   - CLT for SGD noise, variance
2. **Implicit Regularization in Continuous Time:**
   - Early stopping $\leftrightarrow$ minimum norm bias
   - Langevin dynamics: noise enables exploration
3. **Finite Step Reality:**
   - Backward error: finite η penalizes $\frac{\eta}{4}\|\nabla L\|^2$
   - Effective temperature: $T_{\text{eff}} \propto \eta/b$
   - Penalty and noise prefer flat minima $\Rightarrow$ implicit regularization
4. **Over-parametrization:**
   - NTK regime: lazy training, kernel fixed, convergence guaranteed
   - Mean-field: feature learning possible, distributional dynamics

Outlook: Other Important Topics and Open Questions

Topics we mentioned but didn't cover:

- **Loss landscape geometry:** Mode connectivity, solution manifolds
- **Saddle point escape:** GD avoids strict saddles, perturbed GD escapes fast
- **Sharp vs. flat minima:** Hessian spectrum, PAC-Bayes bounds
- **Large-batch training:** Warmup schedules, critical batch size

Open research questions:

- **Finite width:** Beyond NTK/mean-field infinite-width limits
- **Deep architectures:** Theory mostly for shallow networks
- **Feature learning dynamics:** When and how features emerge

This lecture: Why SGD works (theory + mechanisms)

Next lecture: How to make optimization faster and more efficient

References I

- Ali, Alnur et al. (2020). “The Implicit Regularization of Stochastic Gradient Flow for Least Squares”. In: *International Conference on Machine Learning (ICML)*. PMLR, pp. 233–244.
- Amari, S.-I. (1998). “Natural Gradient Works Efficiently in Learning”. In: *Neural Computation* 10.2, pp. 251–276.
- Barrett, David GT and Benoit Dherin (2021). “Implicit gradient regularization”. In: *International Conference on Learning Representations*.
- Chizat, Lénaïc and Francis Bach (2018). “On the Global Convergence of Gradient Descent for Over-parameterized Models using Optimal Transport”. In: *Advances in Neural Information Processing Systems*.
- Cohen, J. et al. (2021a). “Gradient Descent on Neural Networks Typically Occurs at the Edge of Stability”. In: *International Conference on Learning Representations (ICLR)*.
- Cohen, Jeremy et al. (2021b). “Gradient descent on neural networks typically occurs at the edge of stability”. In: *International Conference on Learning Representations*.

References II

- Dinh, L. et al. (2017). “Sharp Minima Can Generalize For Deep Nets”. In: *International Conference on Machine Learning (ICML)*.
- Hardt, M. et al. (2016). “Train Faster, Generalize Better: Stability of Stochastic Gradient Descent”. In: *International Conference on Machine Learning (ICML)*, pp. 1225–1234.
- Jacot, A. et al. (2018). “Neural Tangent Kernel: Convergence and Generalization in Neural Networks”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 31.
- Keskar, N. S. et al. (2017). “On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima”. In: *International Conference on Learning Representations (ICLR)*.
- Mei, S. et al. (2018). “A Mean Field View of the Landscape of Two-Layer Neural Networks”. In: *Proceedings of the National Academy of Sciences* 115.33, E7665–E7671.

References III

Smith, Samuel L et al. (2021). “On the Origin of Implicit Regularization in Stochastic Gradient Descent”. In: *International Conference on Learning Representations*.