Understanding deep learning through the study of wide neural networks

Goals of theory in machine learning:

1. Guarantees on system behavior, certification
2. Gain **understanding** of how systems work and mechanisms behind them. Identify underlying principles.

Develop theory that is guided by empirics & refined until it can be **predictive**.

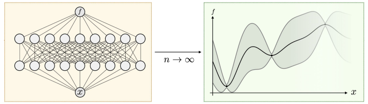
This poster: broadly applicable results for deep neural networks.

Consequences for theory of generalization, hyperparameter selection, architecture design.

Infinitely-wide deep NNs are Gaussian Processes

In infinitely-wide deep neural networks, the prior over functions is a Gaussian Process.

$$f_i^\ell(x) = b_i^l + \sum_{j=1}^{n} W_{ij}^l \phi(f_j^{\ell-1}(x)) \qquad f_i^\ell \sim \mathcal{GP}(0, K^\ell)$$



The kernel (covariance) function is defined by a recursion relation:

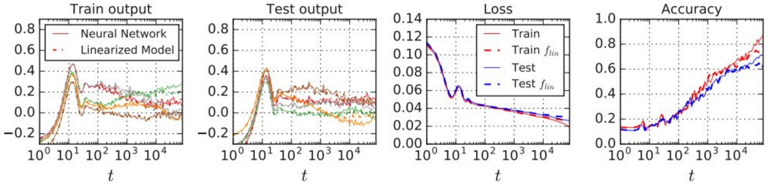
$$K^\ell(x,x') = \mathbb{E}\left[f_i^\ell(x) f_i^\ell(x')\right] = \sigma_b^2 + \sigma_w^2\, \mathcal{C}_\phi\left(K^{\ell-1}(x,x'), K^{\ell-1}(x,x), K^{\ell-1}(x',x')\right)$$

Bayesian inference in infinitely-wide deep networks can be done ***exactly***: $f(x_*)\big|\mathcal{D}, x_* \sim \mathcal{N}(\bar{\mu}, \bar{K})$

Infinitely-wide deep NNs under gradient descent are linear models

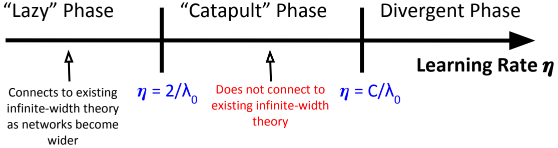
$$f(x,t) = f(x,0) + \nabla_\theta f(x,0)\cdot(\vec{\theta}(t) - \vec{\theta}(0))$$

Can obtain exact solutions for evolution under gradient descent.



References for poster: arxiv 1711.00165 (ICLR 2018), arxiv 1810.05148 (ICLR 2019), arxiv 1902.06720 (NeurIPS 2019), arxiv 2006.10540 (ICML 2020), arxiv 2003.02218 (under review).

The large learning rate phase of NNs under gradient descent



"Lazy" Phase

"Catapult" Phase

Divergent Phase

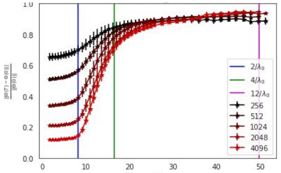
Learning Rate η

Connects to existing infinite-width theory as networks become wider

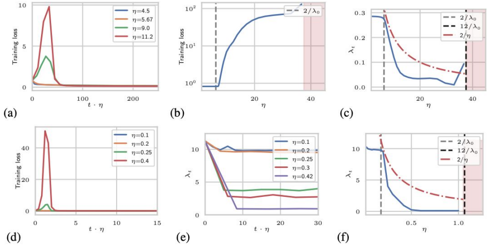
$\eta = 2/\lambda_0$

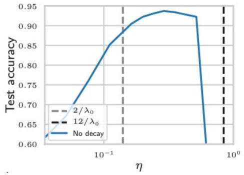
Does not connect to existing infinite-width theory

$\eta = C/\lambda_0$



Distinction becomes ***sharper*** as networks become ***wider***





Wide Resnet on CIFAR-10

There is a phase transition in gradient descent as a function of learning rate as neural networks become wider.

Dynamics are tractable in a two-layer NN. Setup: empirical risk minimization for square loss with gradient descent.

$$\mathcal{L} = \frac{1}{2m}\sum_{\alpha=1}^{m}(f(x_\alpha) - y_\alpha)^2 \qquad f(x) = \frac{1}{\sqrt{n}} v^T u\, x \qquad \Theta(x,x') = \frac{1}{m}\sum_{\mu=1}^{p}\frac{\partial f(x)}{\partial\theta_\mu}\frac{\partial f(x')}{\partial\theta_\mu}$$

Study evolution in the space of functions that is induced by gradient descent on parameters. Function evolution depends on Neural Tangent Kernel and other variables. In the simplest setting, we can get a ***closed dynamical system*** for the evolution of the function and the curvature (!). **(η = LR, n = neural network width)**

$$\boxed{f_{t+1} = \left(1 - \eta\lambda_t + \frac{\eta^2 f_t^2}{n}\right) f_t} \qquad \boxed{\lambda_{t+1} = \lambda_t + \frac{\eta f_t^2}{n}(\eta\lambda_t - 4)}$$

Lazy to catapult phase transition occurs at $2/\lambda_0$. Divergence happens at $4/\lambda_0$.