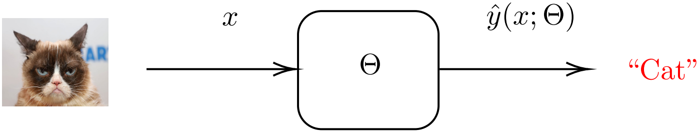
Mean field limit in multilayer neural network learning

Phan Minh Nguyen

Stanford University, 19 March 2020

Machine learning problem



- Data: $(x, y) \sim \mathcal{P}$
- Optimization problem to solve for Θ:

$$\inf_{\Theta} \mathbb{E}_{(x,y)\sim\mathcal{P}}\{\ell(\hat{y}(x;\Theta), y)\} \quad \equiv \quad \inf_{\Theta} \mathcal{L}(\hat{y}(\cdot;\Theta)),$$

  in which

$$\mathcal{L}(f(\cdot)) = \mathbb{E}_{(x,y)\sim\mathcal{P}}\{\ell(f(x), y)\}.$$

Arguably, the simplest (in regression):

- Linear model:

$$\hat{y}(x;\Theta) = \langle x, \Theta \rangle \qquad (x, \Theta \in \mathbb{R}^d).$$

- Convex loss, e.g.

$$\ell(\hat{y}, y) = (\hat{y} - y)^2.$$

- Convex optimization, gradient descent works (typically)

$$\frac{\mathrm{d}}{\mathrm{d}t}\Theta(t) = -\text{ gradient of } \mathcal{L}(\hat{y}(\cdot;\Theta(t))) \text{ w.r.t. } \Theta \; \checkmark$$

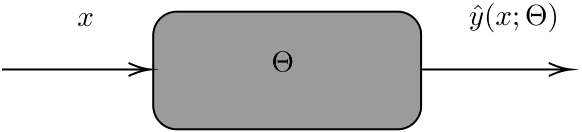
-

Machine learning problem

Arguably, the simplest (in regression):

- Linear model:
$$\hat{y}(x;\Theta) = \langle x, \Theta\rangle \qquad (x, \Theta \in \mathbb{R}^d).$$
- Convex loss, e.g.
$$\ell(\hat{y}, y) = (\hat{y} - y)^2.$$
- Convex optimization, gradient descent works (typically)
$$\frac{\mathrm{d}}{\mathrm{d}t}\Theta(t) = - \text{ gradient of } \mathcal{L}(\hat{y}(\cdot;\Theta(t))) \text{ w.r.t. } \Theta$$
- Modeling power ?

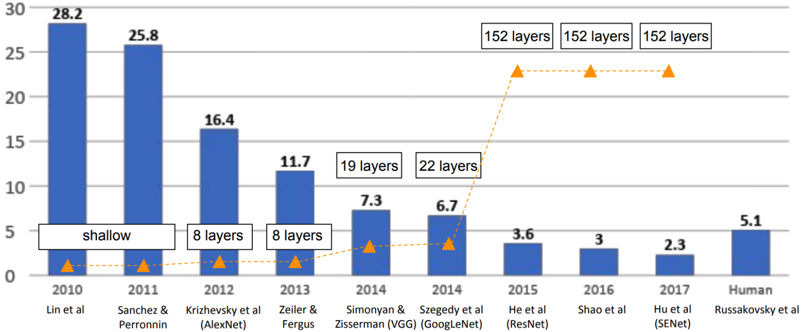
A model where $\Theta \mapsto \hat{y}(x;\Theta)$ is nonlinear?



... powerful, but more challenging to analyze.

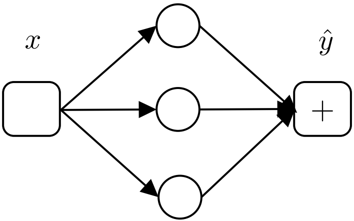
Deep neural network breakthrough...

ImageNet challenge winners: deep nets since 2012.



(Source: CS231N lectures slides)

Two-layer neural network:



In formula:

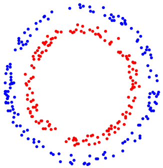
$$\hat{y}_N(x;\Theta) = \frac{1}{N}\sum_{i=1}^{N}\sigma_*(x;\theta_i).$$

A usual example:

$$\sigma_*(x;(a,w)) = a\sigma(\langle x, w\rangle).$$

An experiment:

- Two-class isotropic Gaussian data:

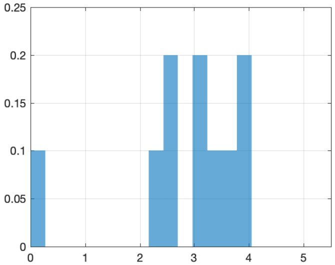


$$\text{w.p. } 1/2, \quad y = +1, \quad x \sim \mathsf{N}(0, (1+0.8)^2 \cdot I_d),$$
$$\text{w.p. } 1/2, \quad y = -1, \quad x \sim \mathsf{N}(0, (1-0.8)^2 \cdot I_d),$$

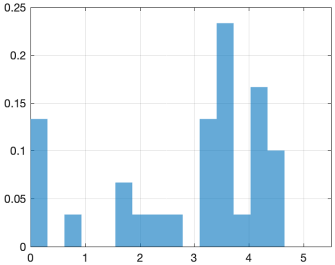
  with $d = 32$.

- Sigmoid-like activation $\sigma_*(x;\theta) = \sigma(\langle x, \theta\rangle)$.
- Run SGD with squared loss, $\theta_i(0) \sim \mathsf{N}(0, (0.8^2/d) \cdot I_d)$ i.i.d.
- Compute loss and $\{\|\theta_i\|_2\}_{i=1,\ldots,N}$ for varying N.

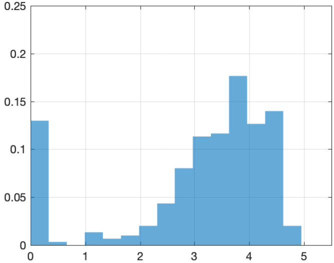
Histogram of $\{\|\theta_i\|_2\}_{i=1,\ldots,N}$, $N = 10$



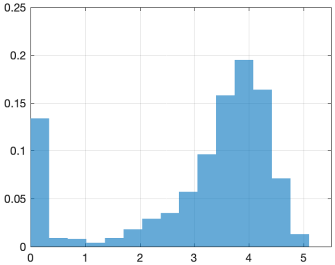
Histogram of $\{\|\theta_i\|_2\}_{i=1,\ldots,N}$, $N = 30$



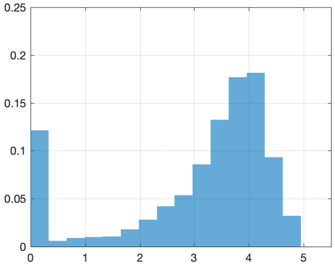
Histogram of $\{\|\theta_i\|_2\}_{i=1,\ldots,N}$, $N = 300$



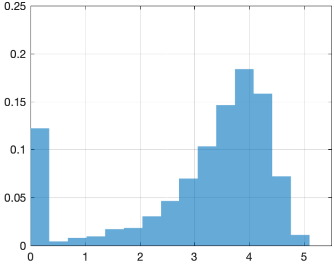
Histogram of $\{\|\theta_i\|_2\}_{i=1,\ldots,N}$, $N = 1000$



Histogram of $\{\|\theta_i\|_2\}_{i=1,\ldots,N}$, $N = 2000$

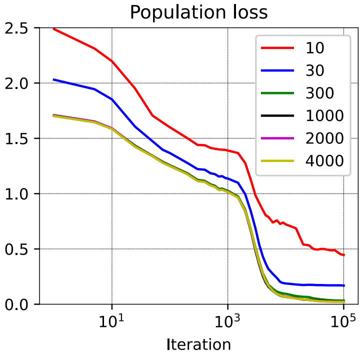


Histogram of $\{\|\theta_i\|_2\}_{i=1,\dots,N}$, $N = 4000$



Mean field limit

$$\frac{1}{N}\sum_{i=1}^{N}\delta_{\theta_i} \to \text{some limiting measure?}$$


Population loss
2.5
2.0
1.5
1.0
0.5
0.0
10^1
10^3
10^5
Iteration
10
30
300
1000
2000
4000

$\mathcal{L}(\hat{y}_N(\cdot;\Theta(t)))$ during training $\to$ some limiting loss evolution curve?

A limiting behavior? Can we prove it?

Yes: under a suitable scaling, there is a limiting characterization, which we call the mean field limit.

- MF limit:

$$\hat{y}(x;\rho) = \int \sigma_*(x;\theta)\rho(\mathrm{d}\theta)$$

- Neural net:

$$\hat{y}_N(x;\Theta) = \frac{1}{N}\sum_{i=1}^{N}\sigma_*(x;\theta_i)$$

- Identification:

$$\rho = \frac{1}{N}\sum_{i=1}^{N}\delta_{\theta_i} \quad \Rightarrow \quad \hat{y}(x;\rho) = \hat{y}_N(x;\Theta),$$

hence the MF limit can realize neural net of any size...

What about gradient descent?

- Squared loss:

$$
\begin{aligned}
\mathcal{L}(\hat{y}_N(\cdot;\Theta)) &= \mathbb{E}_{\mathcal{P}}\{(\hat{y}_N(x;\Theta) - y)^2\} \\
&= \mathbb{E}_{\mathcal{P}}\{y^2\} + \frac{2}{N}\sum_{i=1}^{N} V(\theta_i) + \frac{1}{N^2}\sum_{i,j=1}^{N} U(\theta_i, \theta_j)
\end{aligned}
$$

$$
\begin{aligned}
V(\theta) &= -\mathbb{E}_{\mathcal{P}}\{y\sigma_*(x;\theta)\}, \\
U(\theta, \theta') &= \mathbb{E}_{\mathcal{P}}\{\sigma_*(x;\theta)\sigma_*(x;\theta')\}.
\end{aligned}
$$

- Neural net with continuous-time GD:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\theta_i(t) &= -\boldsymbol{N} \cdot \text{gradient of loss w.r.t. } \theta_i \\ &= -\nabla V(\theta_i(t)) + \frac{1}{N}\sum_{j=1}^{N} \nabla_1 U(\theta_i(t), \theta_j(t)).\end{aligned}$$

  with initialization: $\{\theta_i(0)\}_{i=1,\ldots,N} \sim \rho(0,\cdot)$ i.i.d.

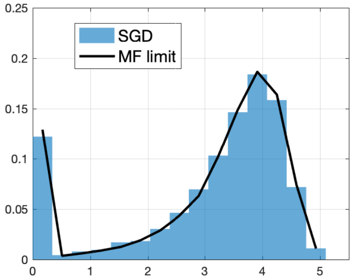
- MF limiting dynamics for $\rho(t,\theta)$:

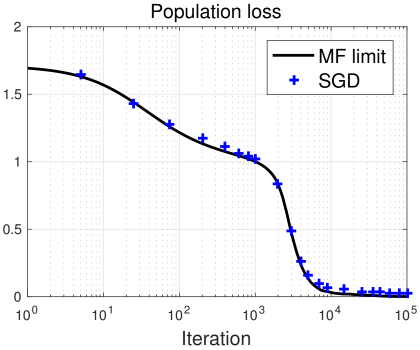
$$\partial_t \rho(t,\theta) = \mathrm{div}\Big(\rho(t,\theta)\cdot\Big[\nabla V(\theta) + \int \nabla_1 U(\theta,\theta')\rho(t,\mathrm{d}\theta')\Big]\Big).$$

  with initialization $\rho(0,\cdot)$.

- Regularity: ∇V, $\nabla_1 U$ bounded Lipschitz, σ_* bounded, $\nabla_\theta \sigma_*(x;\theta)$ subgaussian.

Histogram of $\{\|\theta_i\|_2\}_{i=1,\ldots,N}$, $N = 4000$




Population loss
MF limit
SGD
2
1.5
1
0.5
0
10^0
10^1
10^2
10^3
10^4
10^5
Iteration

Mean field limit: two-layer network

Theorem (Mei, Montanari, Nguyen – PNAS 2018)

For any bounded Lipschitz test function f,

$$\sup_{t\le T}\left|\frac{1}{N}\sum_{i=1}^{N} f(\theta_i(t)) - \int f(\theta)\rho(t,\mathrm{d}\theta)\right| \le Ke^{KT}\mathrm{err}_{N,d}(z),$$

$$\sup_{t\le T}|\mathcal{L}(\hat{y}_N(\cdot;\Theta(t))) - \mathcal{L}(\hat{y}(\cdot;\rho(t,\cdot)))| \le Ke^{KT}\mathrm{err}_{N,d}(z),$$

with probability at least $1-4e^{-z^2/2}$, *where*

$$\mathrm{err}_{N,D}(z) = \frac{1}{\sqrt{N}}(\sqrt{d}+z).$$

Remark:

- The full theorem applies to SGD.
- Chizat & Bach 2018 proves a non-quantitative result, but for general convex losses.

- Non-convex optimization on Θ:

$$\inf_{\Theta} \mathbb{E}_{\mathcal{P}}\left\{\left(\frac{1}{N}\sum_{i=1}^{N}\sigma_*(x;\theta_i)-y\right)^2\right\}$$

- Convex optimization on ρ:

$$\inf_{\rho} \mathbb{E}_{\mathcal{P}}\left\{\left(\int \sigma_*(x;\theta)\rho(\mathrm{d}\theta)-y\right)^2\right\}$$

"convex neural network"
(Bengio et al 2006)

- Same observation for generic convex losses.

- Is it trivialized? No: dynamics on $\rho(t,\cdot)$ is not gradient descent.

Theorem (Chizat & Bach 2018)

Assume (essentially) the setting:

1. *convex loss*,
2. $\sigma_*(x,(a,w)) = a\sigma(\langle x, w\rangle)$ *with some regularity,*
3. *full support of* $\rho(0,\cdot)$ *for the first layer* w*,* $\leftarrow$*(diversity)*
4. $\rho(t,\cdot)$ *converges in* W_2 *as* $t \to \infty$.

Then:

$$\boxed{\mathcal{L}(\hat{y}(\cdot;\rho(t))) \to \inf_{\rho} \mathcal{L}(\hat{y}(\cdot;\rho)) \ \textit{as } t \to \infty.}$$

Remark: Global convergence for noisy GD in Mei, Montanari, Nguyen – PNAS 2018.

Noisy GD:

- Regularized loss:

$$\mathcal{L}_\lambda(\hat{y}_N(\cdot;\Theta)) = \mathbb{E}_{\mathcal{P}}\{(\hat{y}(x;\Theta) - y)^2\} + \frac{\lambda}{N}\sum_{i=1}^{N}\|\theta_i\|_2^2.$$

- Neural net with continuous-time GD:

$$\theta_i(t) = -\int_0^t \left[\nabla V(\theta_i(s)) + \sum_{j=1}^{N} \nabla_1 U(\theta_i(s), \theta_j(s)) + \lambda\theta_i(s)\right] \mathrm{d}s + \int_0^t \sqrt{\frac{1}{\beta}} \mathrm{d}B(s).$$

- MF limiting dynamics for $\rho(t,\theta)$:

$$\partial_t \rho(t,\theta) = \mathrm{div}\Big(\rho(t,\theta)\cdot\Big[\nabla V(\theta) + \int \nabla_1 U(\theta,\theta')\rho(t,\mathrm{d}\theta') + \lambda\theta\Big]\Big) + \frac{1}{\beta}\Delta_\theta \rho(t,\theta).$$

Theorem (Mei, Montanari, Nguyen – PNAS 2018, informal statement)

Neural net (noisy GD) $\longleftrightarrow$ *MF limit (PDE).*

Theorem (Mei, Montanari, Nguyen – PNAS 2018)

Fix $\eta > 0$ *and* $\delta > 0$. *There exists* $\beta_0 = \beta_0(\eta, d, U, V)$ *and* $C_0 = C_0(\eta, U, V, \delta)$ *such that the following holds.*
For $N \geq C_0 d \log d$ *and* $\beta \geq \beta_0$, *there exists* $T = T(d, U, V, \beta, \eta)$ *such that for* $t \in [T, 10T]$,

$$\mathcal{L}(\hat{y}_N(\cdot; \Theta(t))) \leq \inf_{\rho} \mathcal{L}_\lambda(\hat{y}(\cdot; \rho)) + \eta,$$

with probability at least $1 - \delta$.

Recap on two-layer nets:

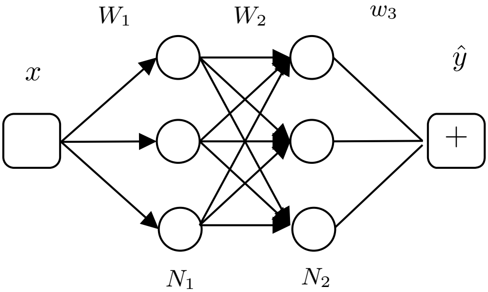
- Neural net $\approx$ MF limit (under scaling)
- Nonlinear, nontrivial behavior: e.g. global convergence

More than two layers?

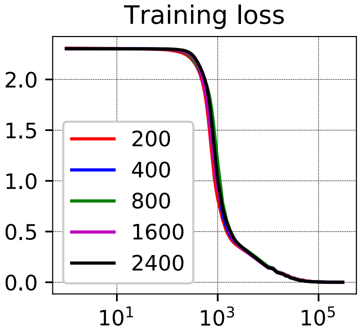
Convexity? Global convergence?

Three-layer neural network:

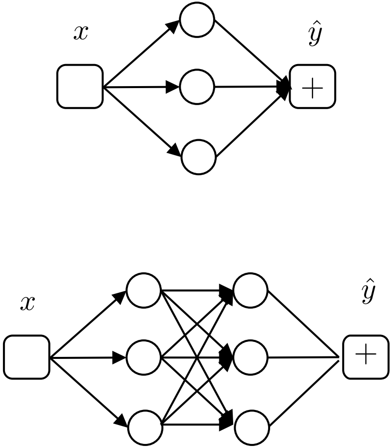
$$\hat{y}_N(x;\Theta) = \frac{1}{N_2}\langle w_3, \sigma(h)\rangle, \qquad h = \frac{1}{N_1} W_2 \sigma(W_1 x)$$

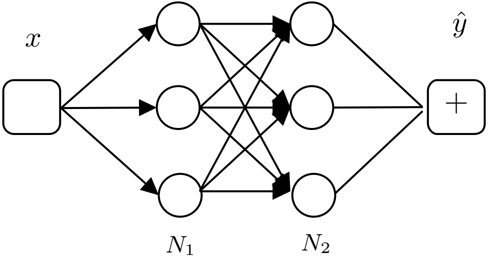


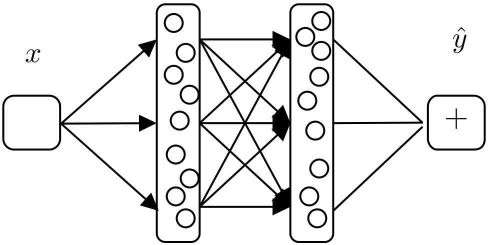
Three-layer neural nets, MNIST classification, $N_1 = N_2$.



(Setup: SGD, ReLU activation, cross entropy loss.)


x
$\hat{y}$
$+$
x
$\hat{y}$
$+$


x
$\hat{y}$
$+$
N_1
N_2


x
$\hat{y}$
$+$

An idea about an embedding...

Let us build a "neural net" with "arbitrary sizes" (MF limit).

- Fix a probability space P on $\Omega_1 \times \Omega_2$, from which two random variables C_1 and C_2 are drawn.
- MF limit:

$$\begin{aligned}
\hat{y}(x; f_1, f_2, f_3) &= \mathbb{E}_{C_2}\{f_3(C_2) \cdot \sigma(h(C_2))\}, \\
h(c_2) &= \mathbb{E}_{C_1}\{f_2(c_2, C_1) \cdot \sigma(\langle f_1(C_1), x\rangle)\}
\end{aligned}$$

  in which

$$f_1 : \Omega_1 \to \mathbb{R}^d, \quad f_2 : \Omega_1 \times \Omega_2 \to \mathbb{R}, \quad f_3 : \Omega_2 \to \mathbb{R}.$$

Let us build a (N_1, N_2)-sized neural net.

- Sample independently:

$$C_1(j), \quad j = 1, ..., N_1,$$
$$C_2(i), \quad i = 1, ..., N_2.$$

- Expectation $\longleftrightarrow$ Expectation w.r.t. empirical distribution:

$$\hat{y}(x; f_1, f_2, f_3) = \mathbb{E}_{C_2}\{f_3(C_2) \cdot \sigma(h(C_2))\},$$

$$h(c_2) = \mathbb{E}_{C_1}\{f_2(c_2, C_1) \cdot \sigma(\langle f_1(C_1), x\rangle)\}$$

Let us build a (N_1, N_2)-sized neural net.

- Sample independently:

$$\begin{aligned} C_1(j), &\quad j = 1, ..., N_1, \\ C_2(i), &\quad i = 1, ..., N_2. \end{aligned}$$

- Expectation $\longleftrightarrow$ Expectation w.r.t. empirical distribution:

$$\hat{y}(x; f_1, f_2, f_3) = \mathbb{E}_{C_2}\{f_3(C_2) \cdot \sigma(h(C_2))\},$$

$$h(c_2) = \frac{1}{N_1} \sum_{j=1}^{N_1} f_2(c_2, C_1(j)) \cdot \sigma(\langle f_1(C_1(j)), x \rangle)$$

Let us build a (N_1, N_2)-sized neural net.

- Sample independently:

$$C_1(j), \quad j = 1, ..., N_1,$$
$$C_2(i), \quad i = 1, ..., N_2.$$

- Expectation $\longleftrightarrow$ Expectation w.r.t. empirical distribution:

$$\hat{y}(x; f_1, f_2, f_3) = \frac{1}{N_2} \sum_{i=1}^{N_2} f_3(C_2(i)) \cdot \sigma(h(C_2(i))),$$

$$h(c_2) = \frac{1}{N_1} \sum_{j=1}^{N_1} f_2(c_2, C_1(j)) \cdot \sigma(\langle f_1(C_1(j)), x\rangle)$$

- Expectation $\longleftrightarrow$ Expectation w.r.t. empirical distribution:

$$\hat{y}(x; f_1, f_2, f_3) = \frac{1}{N_2} \sum_{i=1}^{N_2} f_3(C_2(i)) \cdot \sigma(h(C_2(i))),$$

$$h(c_2) = \frac{1}{N_1} \sum_{j=1}^{N_1} f_2(c_2, C_1(j)) \cdot \sigma(\langle f_1(C_1(j)), x \rangle)$$

- Three-layer neural network:

$$\hat{y}_N(x; \Theta) = \frac{1}{N_2} \langle w_3, \sigma(h) \rangle, \qquad h = \frac{1}{N_1} W_2 \sigma(W_1 x)$$

- Identification:

$$\begin{aligned} W_{1,j} &= f_1(C_1(j)), \\ W_{2,ij} &= f_2(C_2(i), C_1(j)), \\ w_{3,i} &= f_3(C_2(i)). \end{aligned}$$

MF limit (independent of N_1, N_2) $\longleftrightarrow$ (N_1, N_2)-sized neural net.

This connection is facilitated by an embedding,
realized by the probability space P.

$$\begin{aligned} W_{1,j} &= f_1(C_1(j)), \\ W_{2,ij} &= f_2(C_2(i), C_1(j)), \\ w_{3,i} &= f_3(C_2(i)). \end{aligned}$$

Then one can write the MF limiting dynamics for GD of neural net...

Let us state the result formally...

- Fix a probability space P for C_1 and C_2.
- Run MF dynamics, i.e. continuous-time evolution of

$$f_1(t,\cdot),\ f_2(t,\cdot,\cdot),\ f_3(t,\cdot),$$
$$\hat{y}(x; f_1(t,\cdot), f_2(t,\cdot,\cdot), f_3(t,\cdot)),$$

initialized with $f_1(0,\cdot)$, $f_2(0,\cdot,\cdot)$, $f_3(0,\cdot)$.

- Sample independently:

$$C_1(j), \quad j = 1, ..., N_1,$$
$$C_2(i), \quad i = 1, ..., N_2.$$

- Run continuous-time GD on neural net of size (N_1, N_2), i.e. continuous-time evolution of

$$W_1(t),\ W_2(t),\ w_3(t),$$
$$\hat{y}_N(x; \Theta(t)),$$

initialized by the identification:

$$W_{1,j}(0) = f_1(0, C_1(j)),$$
$$W_{2,ij}(0) = f_2(0, C_1(j), C_2(i)),$$
$$w_{3,j}(0) = f_3(0, C_2(j)).$$

- Setup: smooth σ, Lipschitz loss ℓ.

Theorem (Nguyen, Pham 2020)

With probability at least $1-\delta$,

$$\sup_{t \le T} \left| \mathcal{L}(\hat{y}_N(\cdot;\Theta(t))) - \mathcal{L}(\hat{y}(\cdot; f_1(t), f_2(t), f_3(t))) \right| = \tilde{O}\left(\frac{1}{\sqrt{\min(N_1, N_2)}} \right)$$

assuming that there exists $(P, f_1(0), f_2(0), f_3(0))$ that accommodates the initialization law of the neural net.

($\tilde{O}$ hides factors of $\log(1/\delta)$, $\log(\max(N_1, N_2))$ and dependency on T).

Remark: The full theorem is proven for an arbitrary number of layers, general stochastic learning dynamics and operations in Hilbert spaces.

$$\begin{aligned}\hat{y}(x; f_1, f_2, f_3) &= \mathbb{E}_{C_2}\{f_3(C_2)\sigma(h(C_2))\},\\ h(c_2) &= \mathbb{E}_{C_1}\{f_2(c_2, C_1)\sigma(\langle f_1(C_1), x\rangle)\}\end{aligned}$$

No convexity.

Global convergence? ✓

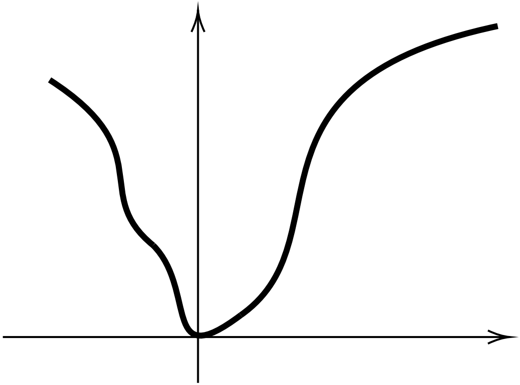
Theorem (Nguyen, Pham 2020)

Assume the setup:

1. $\partial_1 \ell(\hat{y}, y) = 0$ *implies* $\ell(\hat{y}, y) = 0$,
2. *full support of the distribution of* $f_1(0, C_1)$, $\leftarrow$ *(diversity)*
3. *the set* $\{x \mapsto \sigma(\langle x, w\rangle)\}_{w \in \mathcal{X}}$ *is dense in* $L_2(\mathcal{P}_x)$, $\leftarrow$ *(universal approx.)*
4. $y = y(x)$,
5. $f_1(t)$, $f_2(t)$, $f_3(t)$ *converge in appropriate sense as* $t \to \infty$.

Then:

$$\boxed{\mathcal{L}(\hat{y}(\cdot; f_1(t), f_2(t), f_3(t))) \to 0 \text{ as } t \to \infty.}$$



The loss ℓ does not have to be convex.

Why?

Infinitely-wide neural nets are universal approximators. ✓

High-level idea:

- At convergence, gradient update $= 0$:

$$\mathbb{E}_{\mathcal{P}}\{\partial_1 \ell(\hat{y}(x), y(x)) \cdot \text{something} \cdot \sigma(\langle f_1(c_1), x\rangle)\} = 0.$$

- Universal approximation of $\{x \mapsto \sigma(\langle w_1, x\rangle)\}_{\text{indexed by } w_1}$:

$$\forall w_1, \;\; \mathbb{E}_{\mathcal{P}}\{g(x)\sigma(\langle w_1, x\rangle)\} = 0 \quad \Leftrightarrow \quad g = 0 \quad a.e.\ x.$$

- So if there is sufficient diversity and 'something' is nice,

$$\partial_1 \ell(\hat{y}(x), y(x)) = 0 \quad a.e.\ x.$$

- Hence global convergence by assumption.

And so, we move away from convex paradigm
to something truly "neural net"...

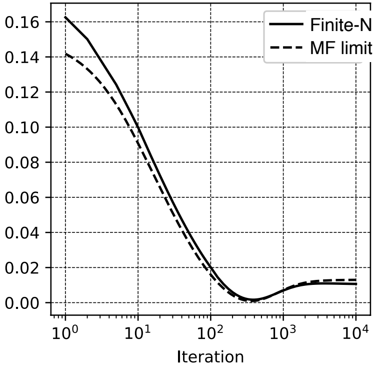
- Formulation of MF limit for two-layer networks. Global convergence.
- Formulation of MF limit for three-layer networks. Extend naturally to any number of layers.
- Global convergence for three-layer networks. No more need of convexity.

"A mean field view of the landscape of two-layers neural networks", S. Mei, A. Montanari, P.-M. Nguyen, PNAS 2018.

"On the global convergence of gradient descent for over-parameterized models using optimal transport", L. Chizat and F. Bach, NeurIPS 2018.

"A rigorous framework for the mean field limit of multilayer neural networks", P.-M. Nguyen and H. T. Pham, 2020. arXiv:2001.11443.

Two-layer autoencoder, MNIST data.



"A mean-field analysis of weight-tied autoencoders", A. Montanari and P.-M. Nguyen, in preparation.

A different MF formulation for multilayer neural networks:

"Mean field limit of the learning dynamics of multilayer neural networks", P.-M. Nguyen, 2019. arXiv:1902.02880.

Other works

"On random deep weight-tied autoencoders: Exact asymptotic analysis, phase transitions, and implications to training", P. Li and P.-M. Nguyen, ICLR 2019.

"State evolution for approximate message passing with non-separable functions", R. Berthier, A. Montanari, P.-M. Nguyen, Information and Inference: A Journal of the IMA (2019).

"Universality of the elastic net error", A. Montanari and P.-M. Nguyen, ISIT 2017.

"Capacity of the energy-harvesting channel with a finite battery", D. Shaviv, P.-M. Nguyen, A. Ozgur, IEEE IT Trans. 2016.