Landscape Connectivity and Dropout Stability of SGD Solutions for Over-parameterized Neural Networks

Alexander Shevchenko

Marco Mondelli

IST AUSTRIA

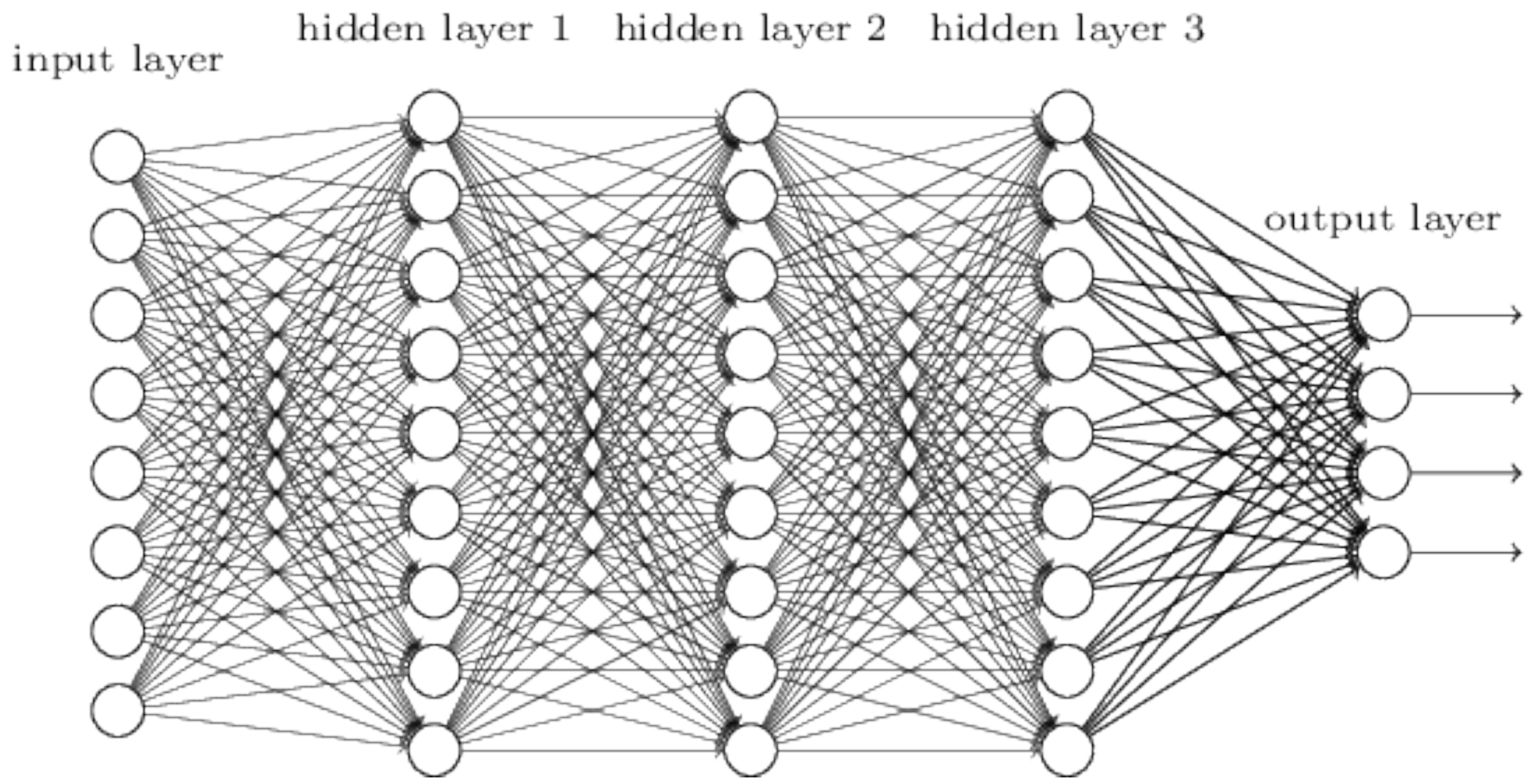
Institute of Science and Technology

Neural Network Training

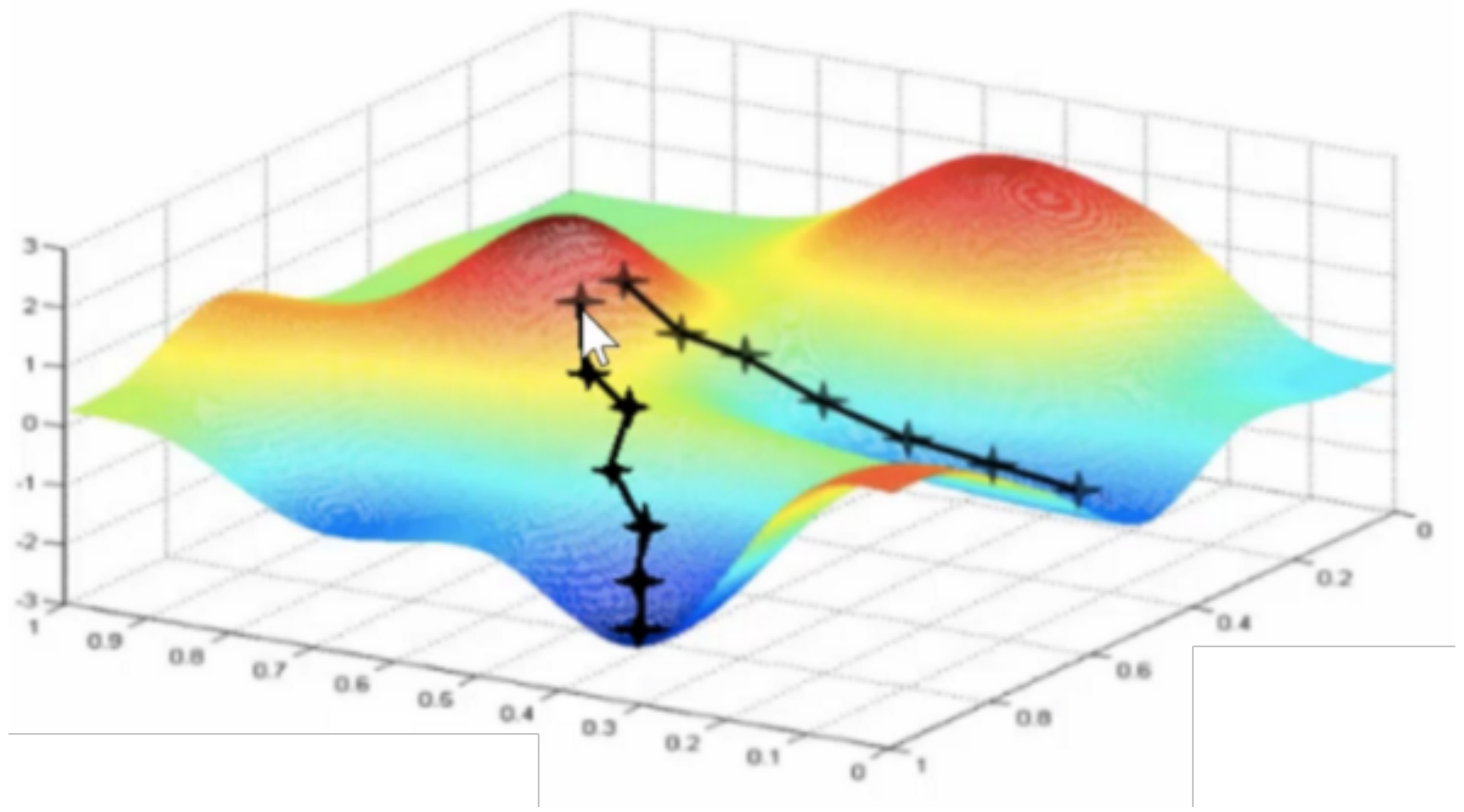
From theoretical perspective training of neural networks is difficult (NP-hardness, local/disconnected minima …), but in practice works remarkably well!

Two key ingredients of success:

Over-parameterization

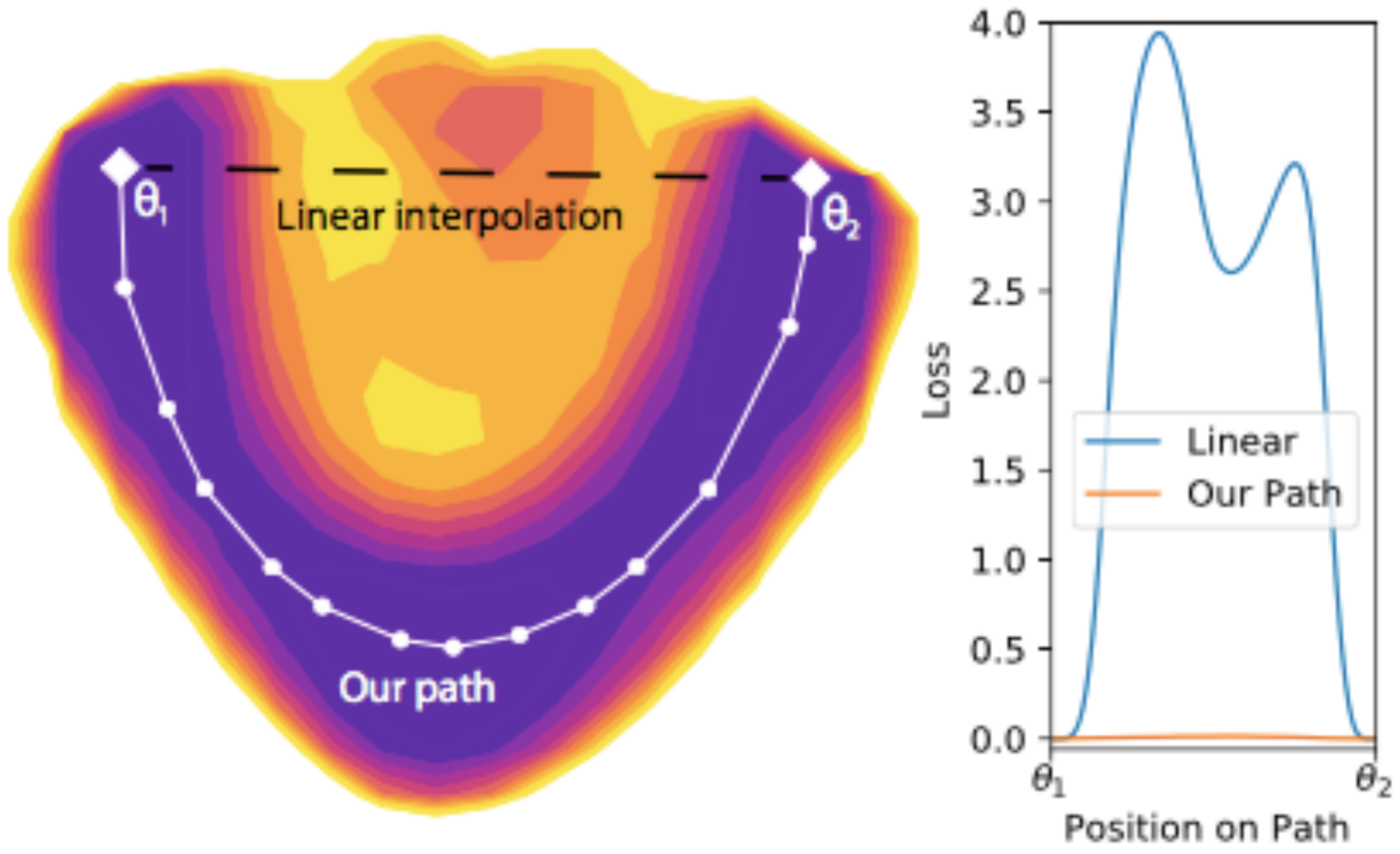


(Stochastic) gradient descent



Training Landscape is indeed NICE

- SGD minima connected via piecewise linear path with constant loss [Garipov et al., 2018; Draxler et al., 2018]
- Mode connectivity proved assuming properties of well-trained networks (dropout/noise stability) [Kuditipudi et al., 2019]



What do we show?

Theorem. (Informal) *As neural network grows wider the solutions obtained via SGD become increasingly more dropout stable and barriers between local optima disappear.*

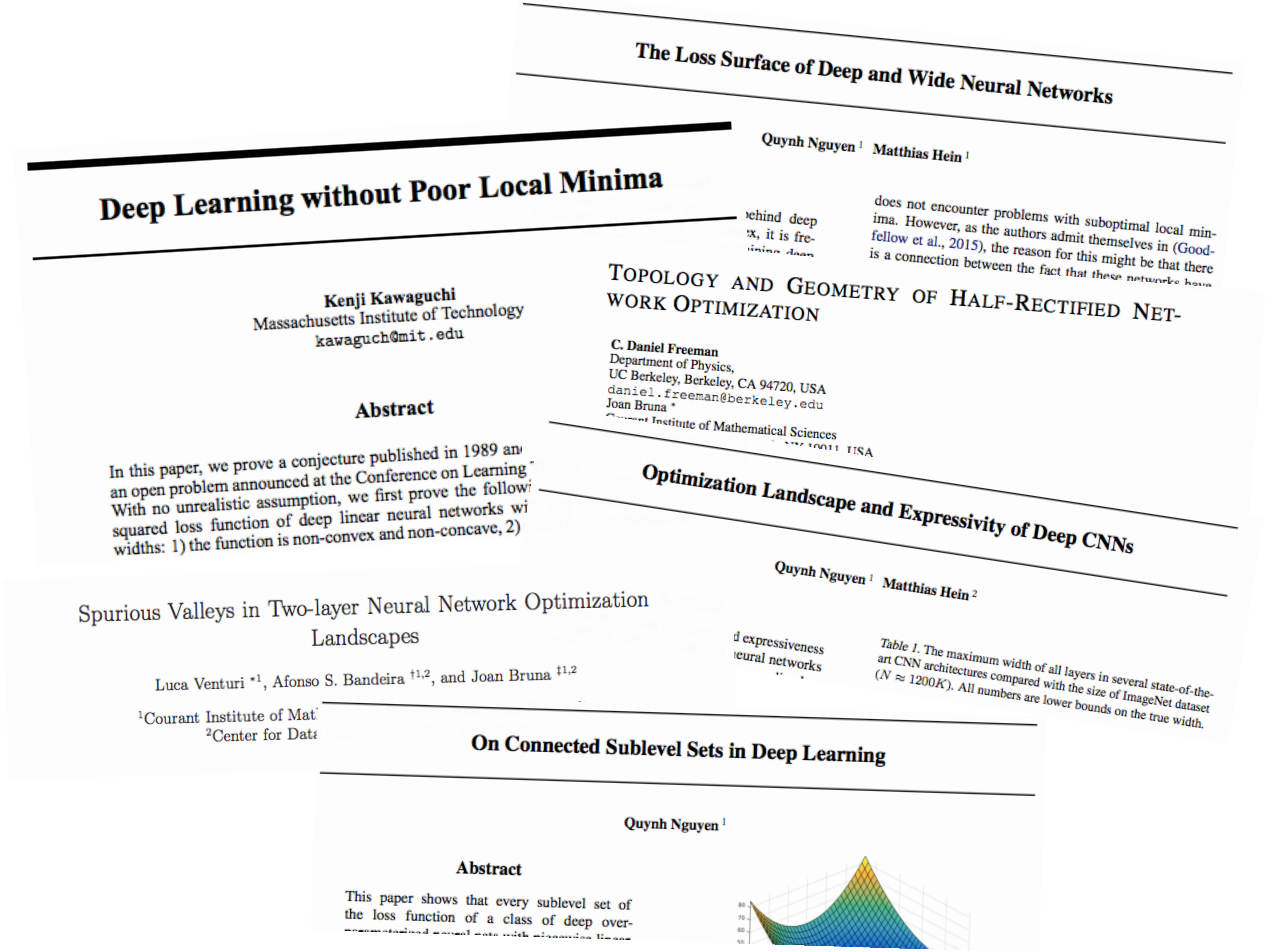
Mean-field view: Two layers [Mei et al., 2019] Multiple layers [Araujo et al., 2019]

Quantitative bounds:

- independent of input dimension for two-layer networks, scale linearly for multiple layers
- change in loss scales with network width as $\sqrt{\frac{1}{\text{width}}}$
- number of training samples is just required to scale faster than the $\sqrt{\log(\text{width})}$

Related Work

- Local minima are globally optimal for deep linear networks and networks with more neurons than training samples
- Connected landscape if the number of neurons grows large (two-layer networks, energy gap exponential in input dimension)



Strong assumptions on the model and poor scaling of parameters 😞

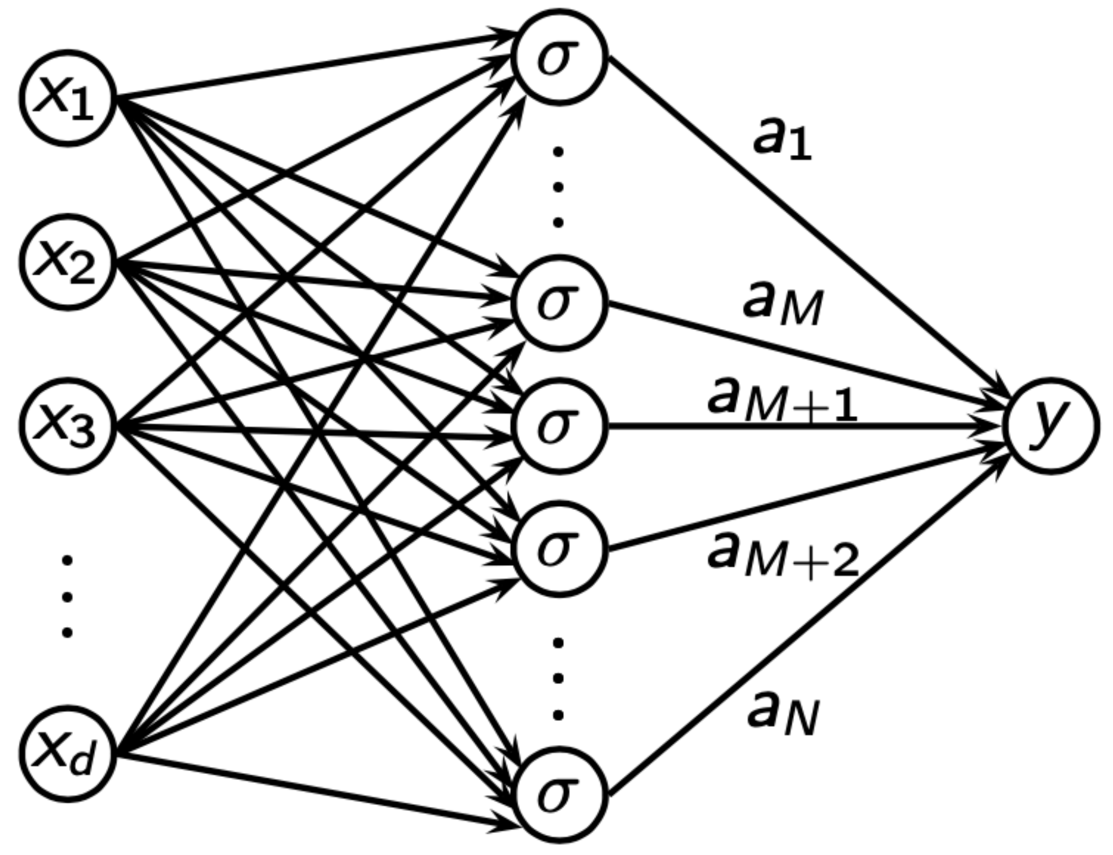
Warm-up: Two Layer Networks

Data: $\left\{ (\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n) \right\} \sim_{\text{i.i.d.}} \mathbb{P}\left(\mathbb{R}^d \times \mathbb{R}\right)$

Model: $\hat{y}_N(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} a_i \sigma\left(\boldsymbol{x}; \boldsymbol{w}_i\right)$

Goal: Minimize loss $L_N(\boldsymbol{\theta}) = \mathbb{E}\left\{ \left(y - \frac{1}{N} \sum_{i=1}^{N} a_i \sigma\left(\boldsymbol{x}; \boldsymbol{w}_i\right) \right)^2 \right\}, \boldsymbol{\theta} = (\boldsymbol{w}, a)$

Online SGD: $\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k + \alpha N \nabla_{\boldsymbol{\theta}^k} \left(\left(y_k - \frac{1}{N} \sum_{i=1}^{N} a_i^k \sigma\left(\boldsymbol{x}_k; \boldsymbol{w}_i^k\right) \right)^2 \right)$

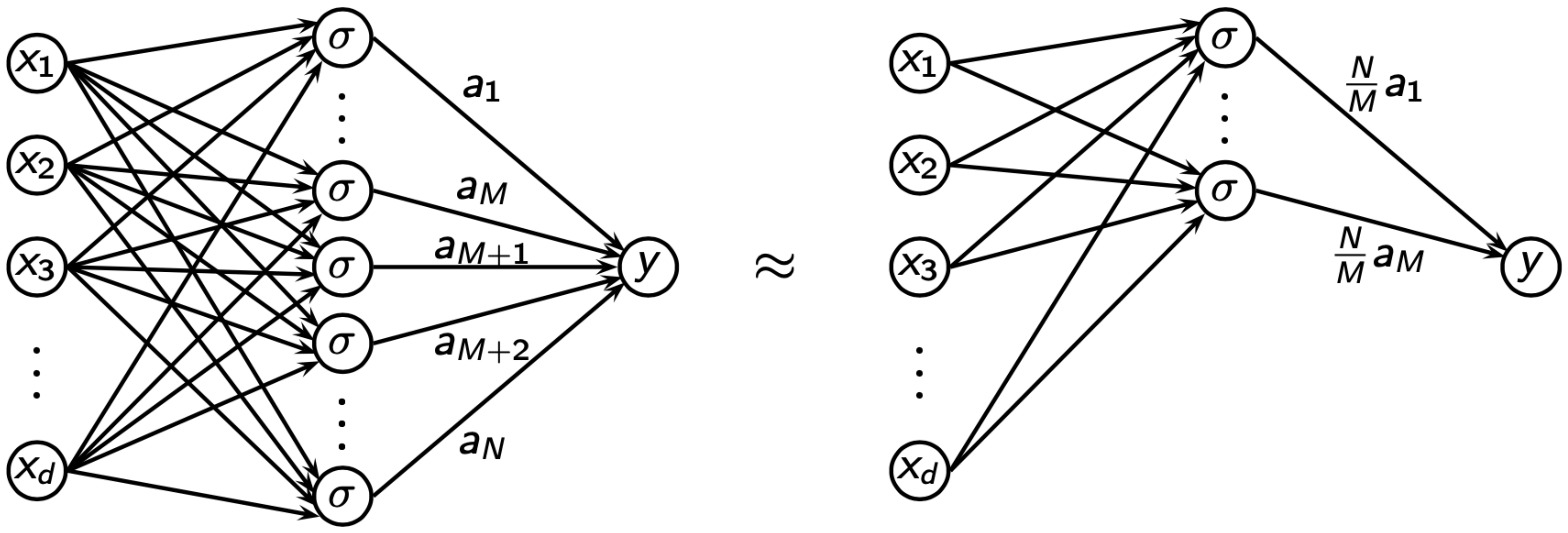


- y bounded, $\nabla_{\boldsymbol{w}} \sigma(\boldsymbol{x}, \boldsymbol{w})$ sub-gaussian
- σ bounded and differentiable, $\nabla \sigma$ bounded and Lipschitz
- initialization of a_i with bounded support

Recap: Dropout Stability

$$L_M(\boldsymbol{\theta}) = \mathbb{E}\left\{\left(y - \frac{1}{M}\sum_{i=1}^{M} a_i \sigma\left(\boldsymbol{x}; \boldsymbol{w}_i\right)\right)^2\right\}$$

$\boldsymbol{\theta}$ is ε_D - dropout stable if $|L_N(\boldsymbol{\theta}) - L_M(\boldsymbol{\theta})| \leq \varepsilon_D$

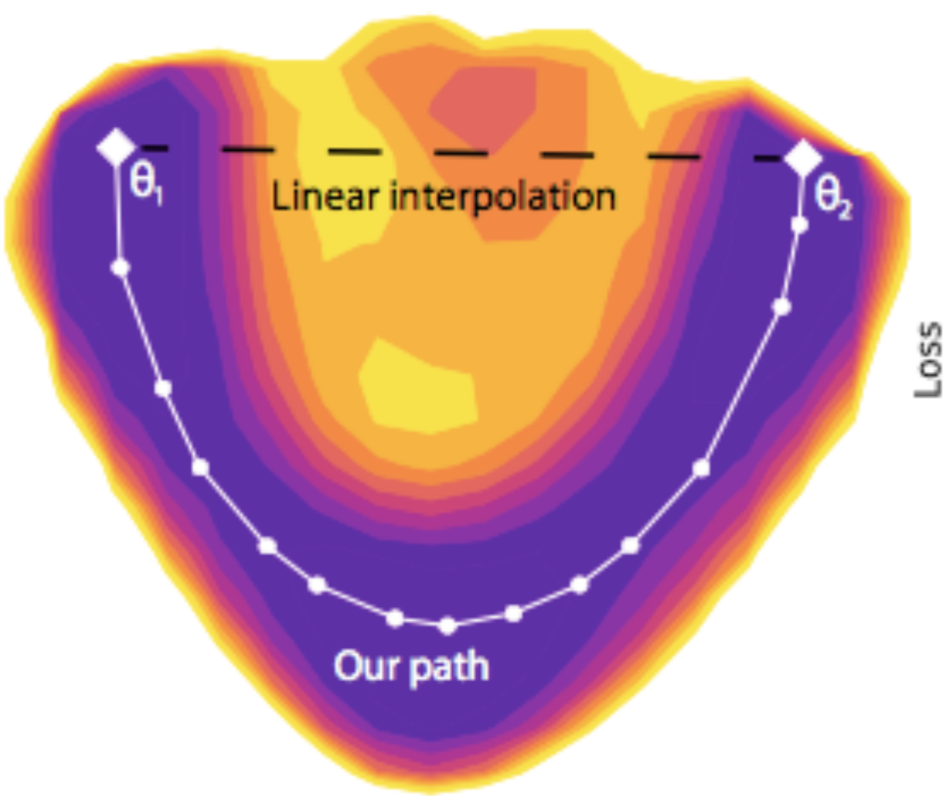


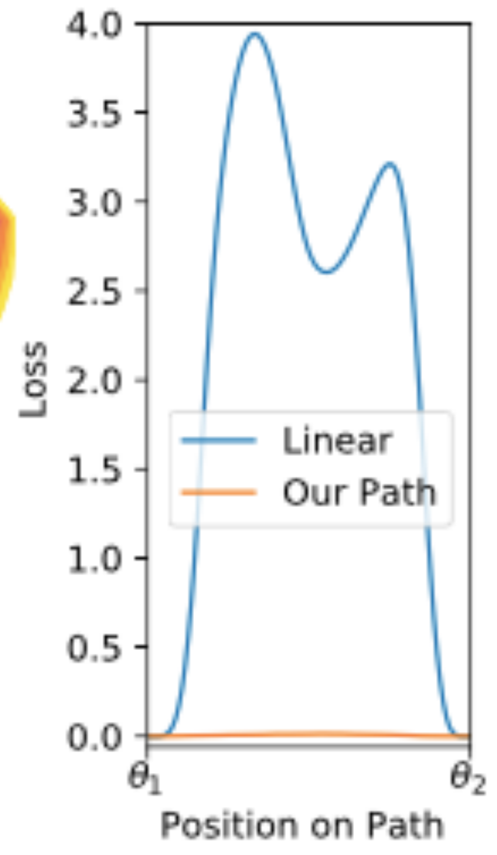
Recap: Dropout Stability and Connectivity

$$L_M(\boldsymbol{\theta}) = \mathbb{E}\left\{\left(y - \frac{1}{M}\sum_{i=1}^{M} a_i \sigma\left(\boldsymbol{x}; \boldsymbol{w}_i\right)\right)^2\right\}$$

$\boldsymbol{\theta}$ is ε_D - dropout stable if $|L_N(\boldsymbol{\theta}) - L_M(\boldsymbol{\theta})| \leq \varepsilon_D$

$\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ are ε_C - connected if there exists a continuous path connecting them where the loss does not increase more than ε_C





Main Results: Dropout Stability

- $N = \#$ neurons of full network
- $M = \#$ neurons after dropout
- $\alpha =$ step size of SGD
- $D =$ dimension of weights

Theorem

Let $\boldsymbol{\theta}^k$ be obtained after k SGD iterations. Then, with probability $1 - e^{-z^2}$, for all $k \in [T/\alpha]$, $\boldsymbol{\theta}^k$ is ε_{D}-dropout stable with

$$\varepsilon_{\mathrm{D}} = Ke^{KT^3} \left(\frac{\sqrt{\log M} + z}{\sqrt{M}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right) \right).$$

Main Results: Dropout Stability

- $N = \#$ neurons of full network
- $M = \#$ neurons after dropout
- $\alpha =$ step size of SGD
- $D =$ dimension of weights

Theorem

Let $\boldsymbol{\theta}^k$ be obtained after k SGD iterations. Then, with probability $1 - e^{-z^2}$, for all $k \in [T/\alpha]$, $\boldsymbol{\theta}^k$ is ε_{D}-dropout stable with

$$\varepsilon_{\mathrm{D}} = Ke^{KT^3}\left(\frac{\sqrt{\log M} + z}{\sqrt{M}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right)\right).$$

Change in loss scales as $\sqrt{\frac{\log M}{M}} + \sqrt{\alpha(D + \log N)}$

Main Results: Dropout Stability

- $N = \#$ neurons of full network
- $M = \#$ neurons after dropout
- $\alpha =$ step size of SGD
- $D =$ dimension of weights

Theorem

Let $\boldsymbol{\theta}^k$ be obtained after k SGD iterations. Then, with probability $1 - e^{-z^2}$, for all $k \in [T/\alpha]$, $\boldsymbol{\theta}^k$ is ε_{D}-dropout stable with

$$\varepsilon_{\mathrm{D}} = Ke^{KT^3}\left(\frac{\sqrt{\log M} + z}{\sqrt{M}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right)\right).$$

- Loss change vanishes as $\alpha \ll \left(\sqrt{D + \log N}\right)^{-1}$ and $M \gg 1$
- M does not need to scale with N or D

Main Results: Connectivity

Theorem

Let $\boldsymbol{\theta}^k$ be obtained after k SGD iterations using $\{(\boldsymbol{x}_j, y_j)\}_{j=0}^{k} \sim \mathbb{P}$, and $(\boldsymbol{\theta}')^{k'}$ after k' SGD iterations using $\{(\boldsymbol{x}_j', y_j')\}_{j=0}^{k'} \sim \mathbb{P}$. Then, with probability $1 - e^{-z^2}$, for all $k \in [T/\alpha]$ and $k' \in [T'/\alpha]$, $\boldsymbol{\theta}^k$ and $(\boldsymbol{\theta}')^{k'}$ are ε_{C}-connected with

$$\varepsilon_{\mathrm{C}} = K e^{K \max(T, T')^3} \left(\frac{\sqrt{\log N} + z}{\sqrt{N}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right) \right).$$

Main Results: Connectivity

Theorem

Let $\boldsymbol{\theta}^k$ be obtained after k SGD iterations using $\{(\boldsymbol{x}_j, y_j)\}_{j=0}^{k} \sim \mathbb{P}$, and $(\boldsymbol{\theta}')^{k'}$ after k' SGD iterations using $\{(\boldsymbol{x}_j', y_j')\}_{j=0}^{k'} \sim \mathbb{P}$. Then, with probability $1 - e^{-z^2}$, for all $k \in [T/\alpha]$ and $k' \in [T'/\alpha]$, $\boldsymbol{\theta}^k$ and $(\boldsymbol{\theta}')^{k'}$ are ε_{C}-connected with

$$\varepsilon_{\mathrm{C}} = K e^{K \max(T,T')^3} \left(\frac{\sqrt{\log N} + z}{\sqrt{N}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right) \right).$$

- Change in loss scales as $\sqrt{\frac{\log N}{N}} + \sqrt{\alpha(D + \log N)}$

Main Results: Connectivity

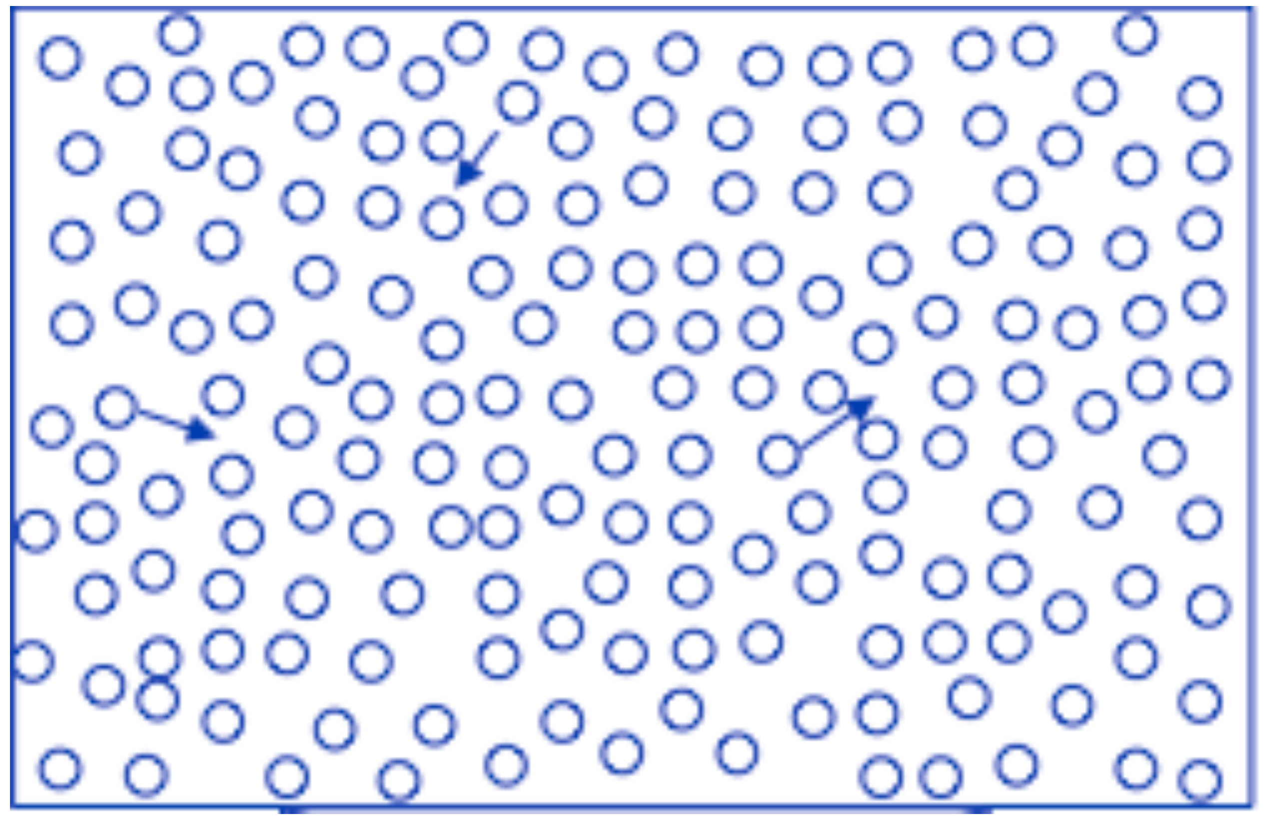
Theorem

Let $\boldsymbol{\theta}^k$ be obtained after k SGD iterations using $\{(\boldsymbol{x}_j, y_j)\}_{j=0}^k \sim \mathbb{P}$, and $(\boldsymbol{\theta}')^{k'}$ after k' SGD iterations using $\{(\boldsymbol{x}'_j, y'_j)\}_{j=0}^{k'} \sim \mathbb{P}$. Then, with probability $1 - e^{-z^2}$, for all $k \in [T/\alpha]$ and $k' \in [T'/\alpha]$, $\boldsymbol{\theta}^k$ and $(\boldsymbol{\theta}')^{k'}$ are ε_{C}-connected with

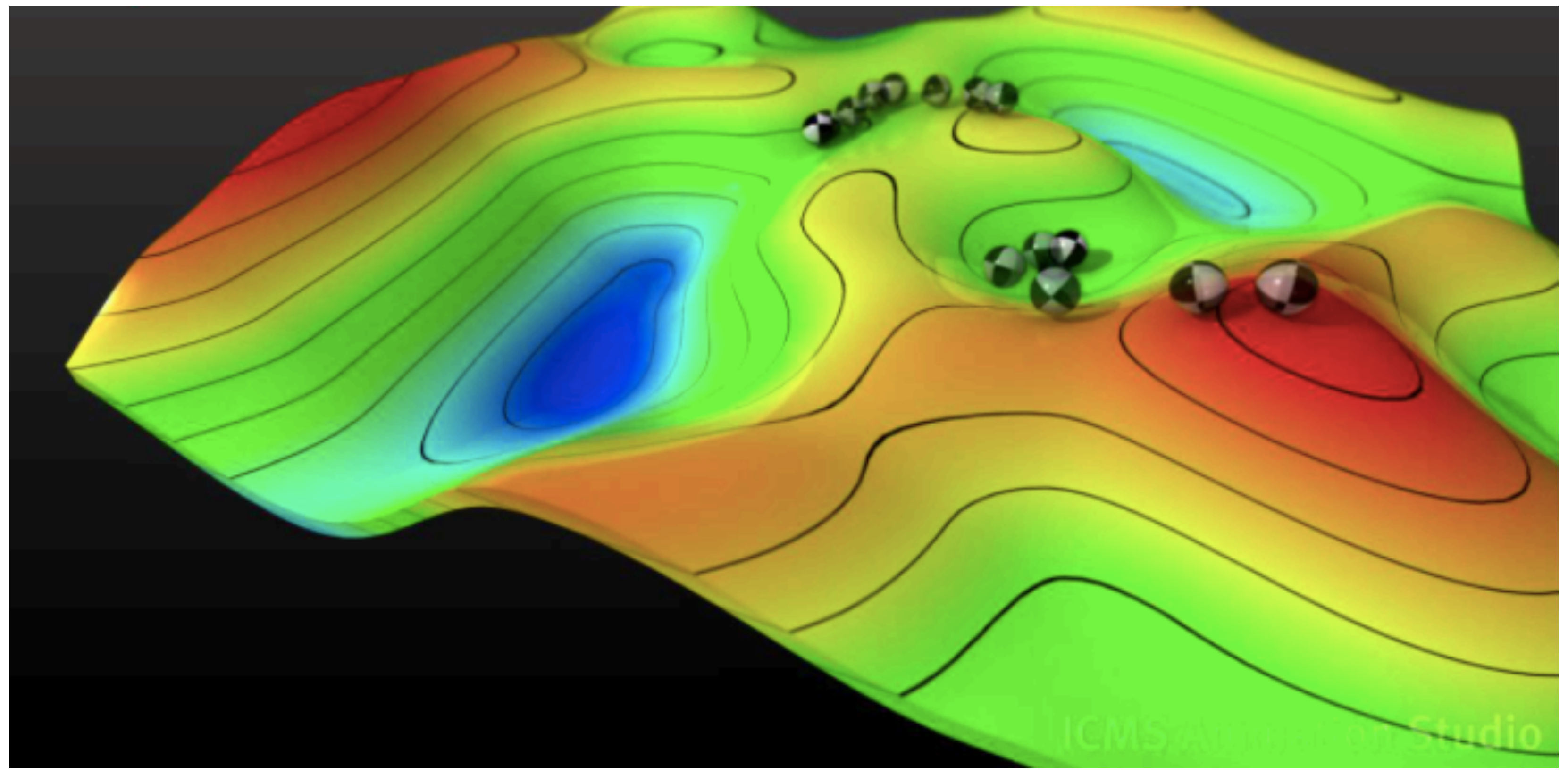
$$\varepsilon_{\mathrm{C}} = Ke^{K \max(T,T')^3} \left(\frac{\sqrt{\log N} + z}{\sqrt{N}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right) \right).$$

- Change in loss scales as $\sqrt{\frac{\log N}{N}} + \sqrt{\alpha(D + \log N)}$
- Can connect SGD solutions obtained from different training data (but same data distribution) and different initialization

Proof Idea



Discrete dynamics of SGD



Continuous dynamics of gradient flow

- $\boldsymbol{\theta}^k$ close to N i.i.d. particles that evolve with gradient flow
- $L_N(\boldsymbol{\theta})$ and $L_M(\boldsymbol{\theta})$ concentrate to the same limit
- Dropout stability with $M = N/2 \Rightarrow$ connectivity

Multilayer Case: Setup

Data: $\left\{ (\boldsymbol{x}_1, \boldsymbol{y}_1), \ldots, (\boldsymbol{x}_n, \boldsymbol{y}_n) \right\} \sim_{\text{i.i.d.}} \mathbb{P}\left(\mathbb{R}^{d_x} \times \mathbb{R}^{d_y}\right)$

Model: $\hat{\boldsymbol{y}}_N(\boldsymbol{x}, \boldsymbol{\theta}) = \dfrac{1}{N} \boldsymbol{W}_{L+1} \sigma_L \left(\cdots \left(\dfrac{1}{N} \boldsymbol{W}_2 \sigma_1 \left(\boldsymbol{W}_1 \boldsymbol{x} \right) \right) \cdots \right)$

Goal: Minimize loss $L_N(\boldsymbol{\theta}) = \mathbb{E} \left\{ \left\| \boldsymbol{y} - \hat{\boldsymbol{y}}_N(\boldsymbol{x}, \boldsymbol{\theta}) \right\|^2 \right\}$

Online SGD: $\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k + \alpha N^2 \nabla_{\boldsymbol{\theta}^k} \left\| \boldsymbol{y}_k - \hat{\boldsymbol{y}}_N \left(\boldsymbol{x}_k, \boldsymbol{\theta}^k \right) \right\|^2$

- $\boldsymbol{y}$ bounded
- σ_ℓ bounded and differentiable, $\nabla \sigma_\ell$ bounded and Lipschitz
- initialization with bounded support
- $\boldsymbol{W}_1$ and $\boldsymbol{W}_{L+1}$ stay fixed (random features)

Multilayer Case: Dropout Stability

Dropout stability: loss does not change much if we remove part of neurons from each layer (and suitably rescale remaining neurons).

Multilayer Case: Dropout Stability

$L_M(\boldsymbol{\theta}) :=$ loss when we keep at most M neurons per layer

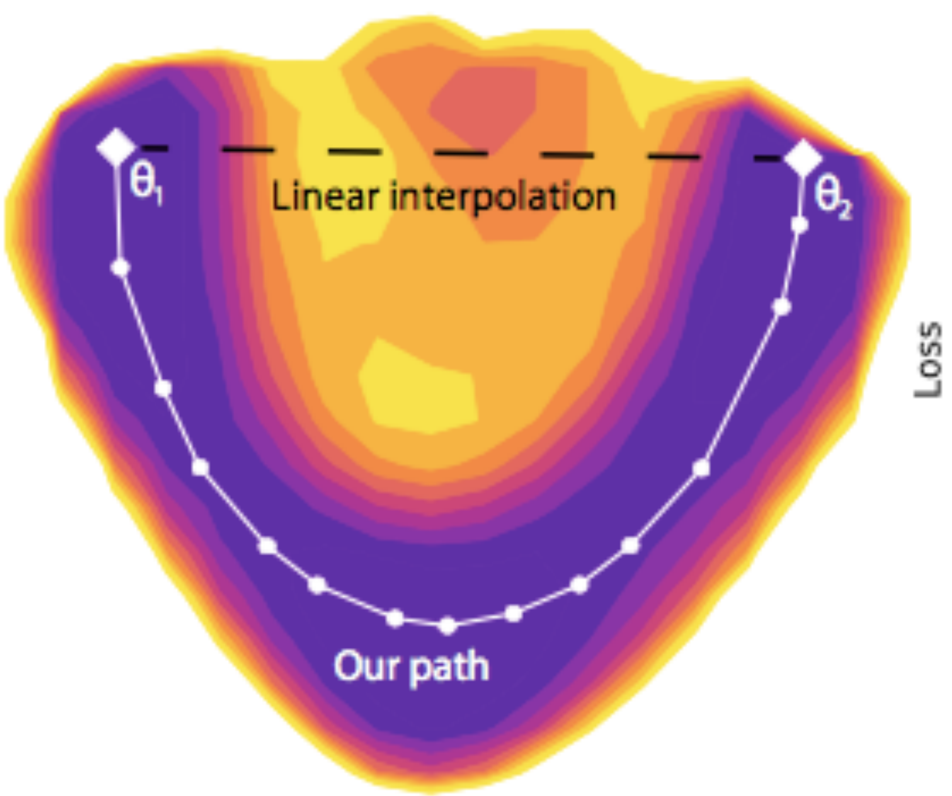
$\boldsymbol{\theta}$ is ε_D - dropout stable if $|L_N(\boldsymbol{\theta}) - L_M(\boldsymbol{\theta})| \leq \varepsilon_D$

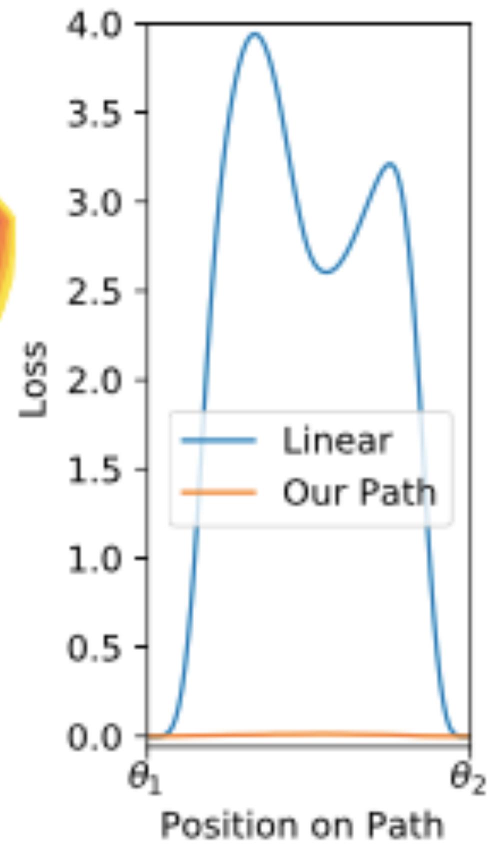
Multilayer Case: Dropout Stability and Connectivity

$L_M(\boldsymbol{\theta}) :=$ loss when we keep at most M neurons per layer

$\boldsymbol{\theta}$ is ε_D - dropout stable if $|L_N(\boldsymbol{\theta}) - L_M(\boldsymbol{\theta})| \leq \varepsilon_D$

$\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ are ε_C - connected if there exists a continuous path connecting them where the loss does not increase more than ε_C





Multilayer Case: Results

- $N = \#$ neurons per layer of full network
- $M =$ max. $\#$ neurons per layer after dropout
- $\alpha =$ step size of SGD
- $D = \max(d_x, d_y)$

Theorem

Let $\boldsymbol{\theta}^k$ be obtained after k SGD iterations, with $k = T/\alpha$. Then, w. p. $1 - e^{-z^2}$, $\boldsymbol{\theta}^k$ is ε_{D}-dropout stable with

$$\varepsilon_{\mathrm{D}} = K(T, L)\left(\frac{\sqrt{D} + z}{\sqrt{M}} + \frac{\sqrt{\log N}}{\sqrt{N}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right)\right).$$

Multilayer Case: Results

- $N = \#$ neurons per layer of full network
- $M =$ max. $\#$ neurons per layer after dropout
- $\alpha =$ step size of SGD
- $D = \max(d_x, d_y)$

Theorem

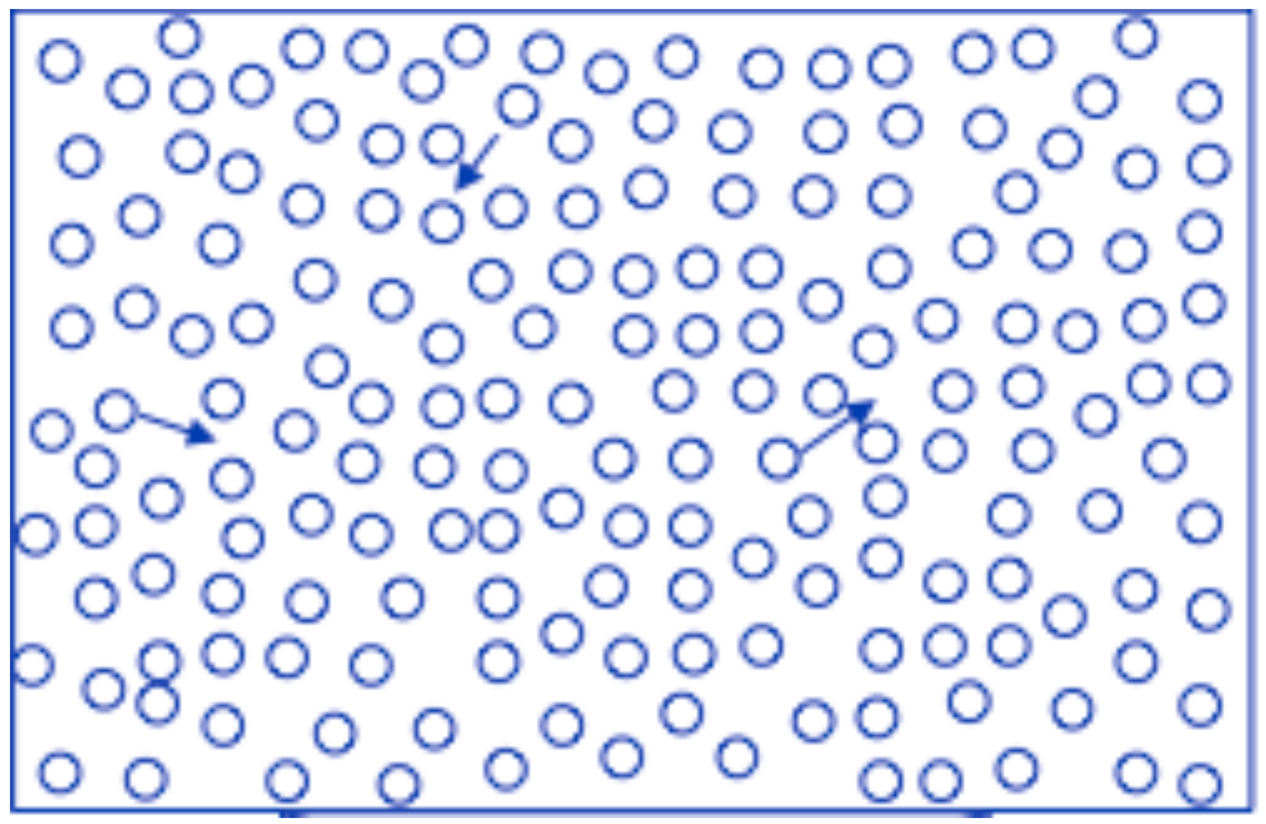
Let $\boldsymbol{\theta}^k$ be obtained after k SGD iterations, with $k = T/\alpha$. Then, w. p. $1 - e^{-z^2}$, $\boldsymbol{\theta}^k$ is ε_{D}-dropout stable with

$$\varepsilon_{\mathrm{D}} = K(T, L)\left(\frac{\sqrt{D} + z}{\sqrt{M}} + \frac{\sqrt{\log N}}{\sqrt{N}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right)\right).$$

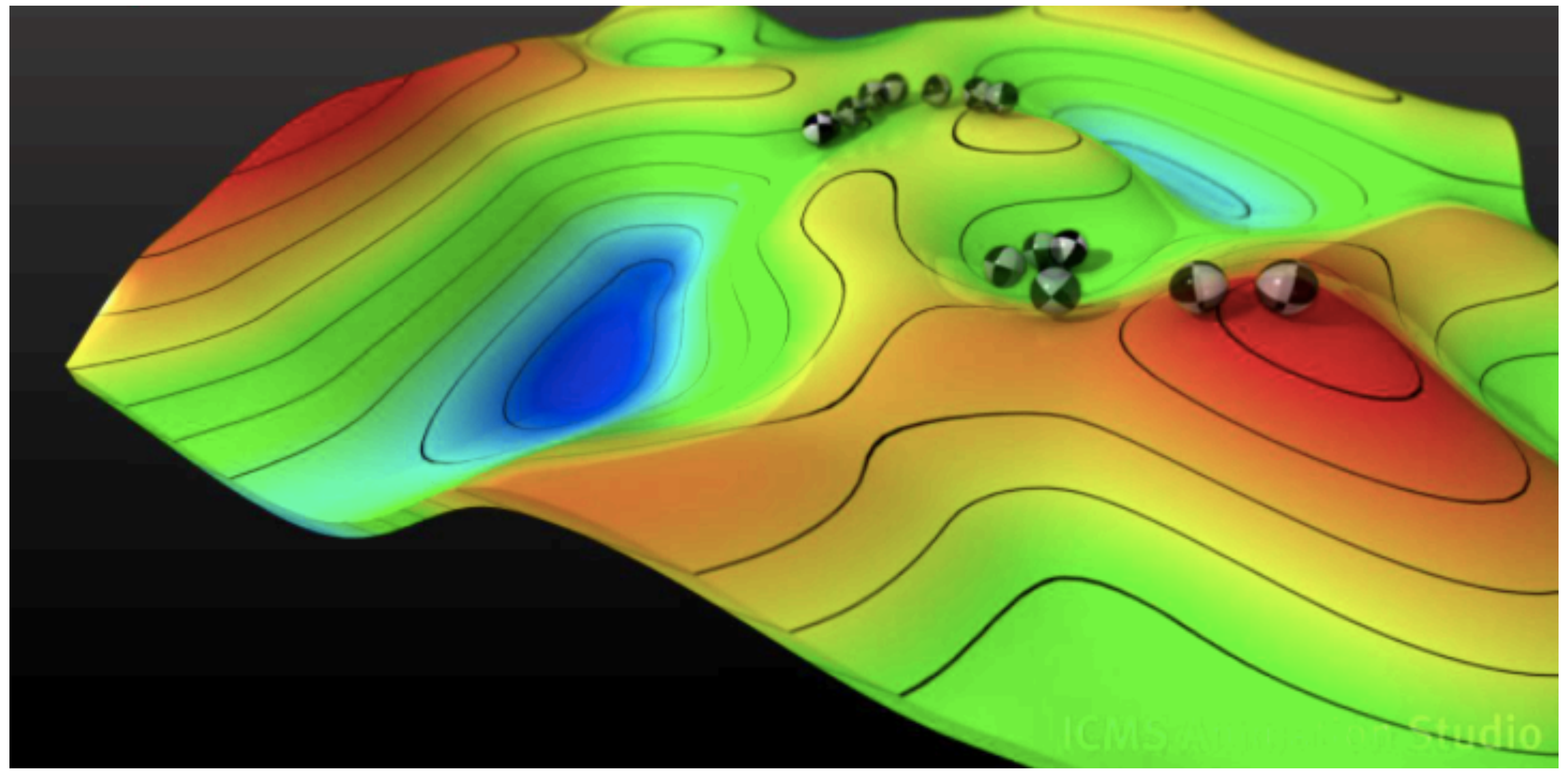
Let $(\boldsymbol{\theta}')^{k'}$ be obtained after k' SGD iterations, with $k' = T'/\alpha$. Then, w. p. $1 - e^{-z^2}$, $\boldsymbol{\theta}^k$ and $(\boldsymbol{\theta}')^{k'}$ are ε_{C}-connected with

$$\varepsilon_{\mathrm{C}} = K(T, T', L)\left(\frac{\sqrt{D + \log N} + z}{\sqrt{N}} + \sqrt{\alpha}\left(\sqrt{D + \log N} + z\right)\right).$$

Proof Challenges



Discrete dynamics of SGD

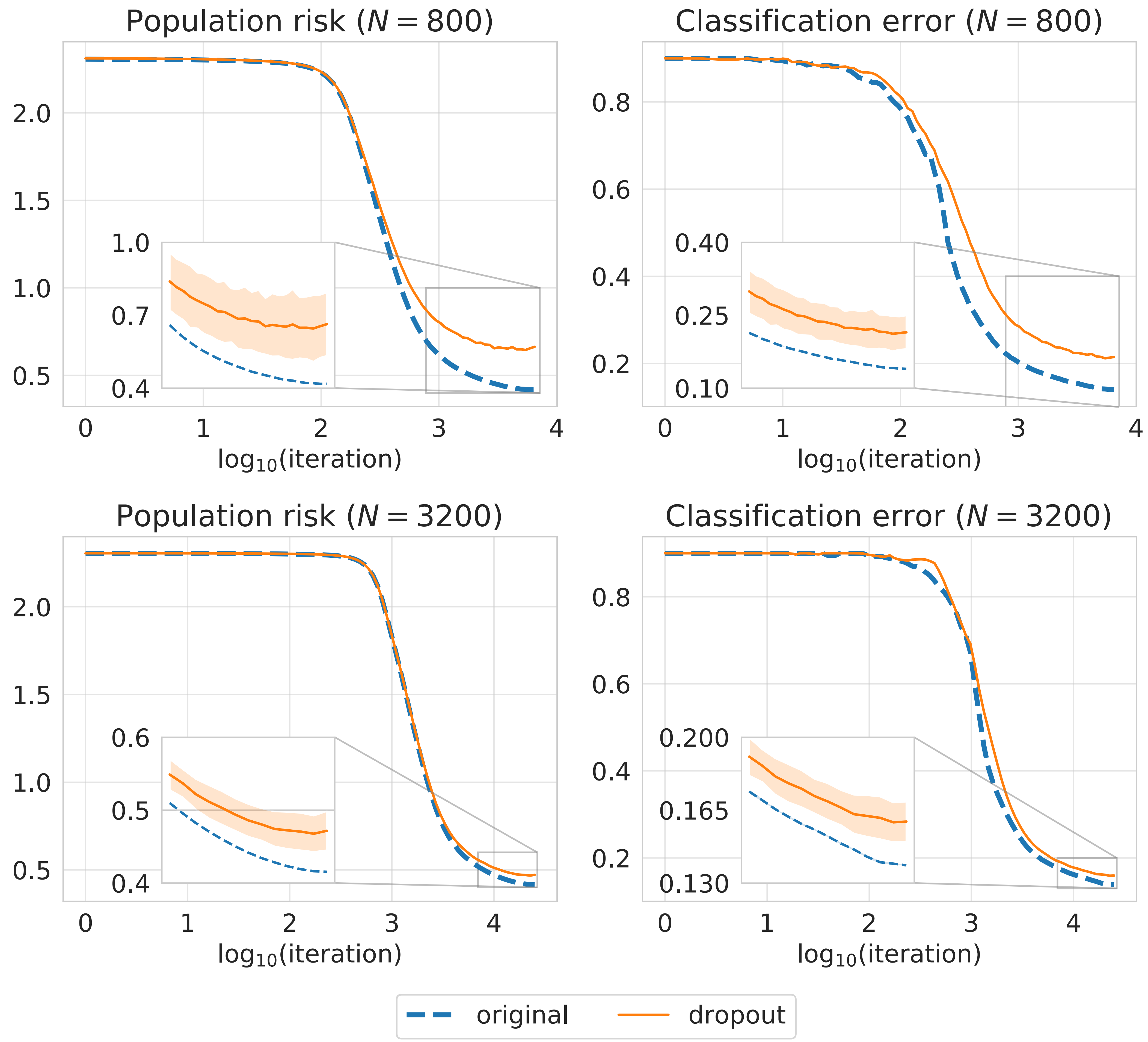


Continuous dynamics of gradient flow

- Ideal particles are no longer independent (weights in different layers are correlated)
- Bound on norm of weights during the training
- Bound maximum distance between SGD and ideal particles ([Araujo et al., 2019] bounds the average distance)

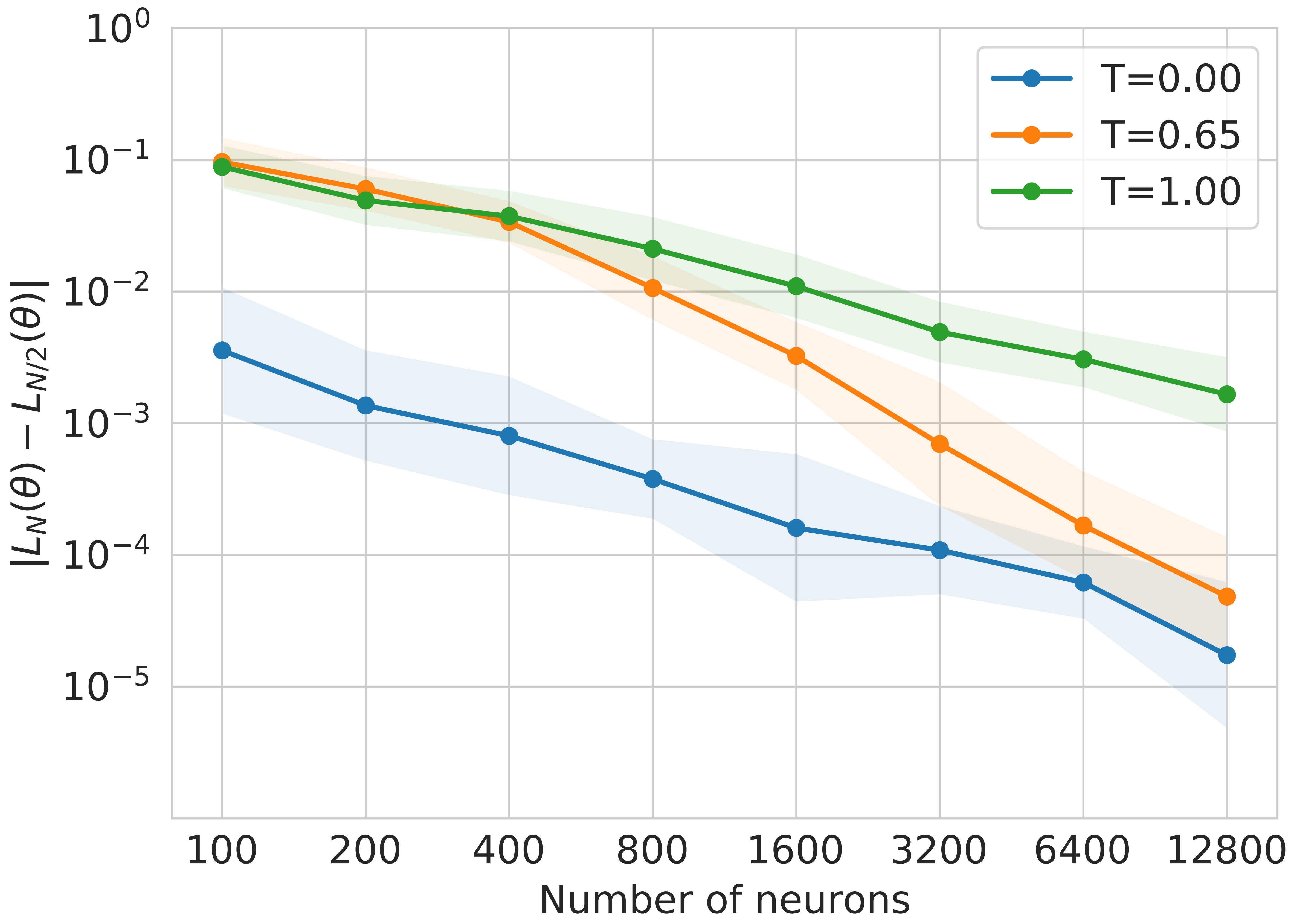
Numerical Results

- CIFAR-10 dataset
- Pretrained VGG-16 features
- # layers = 3
- Keep half of neurons



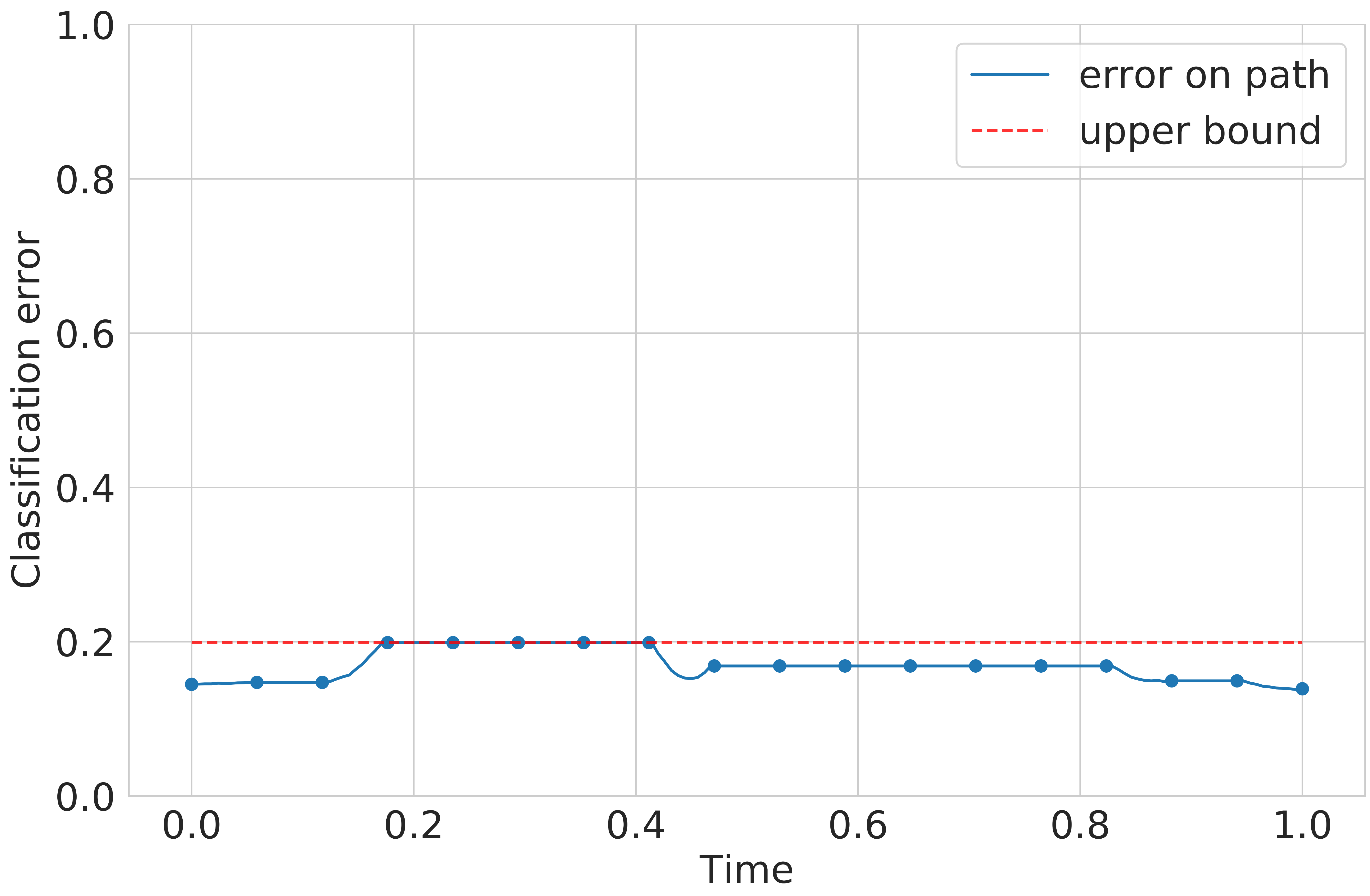
Numerical Results

- CIFAR-10 dataset
- Pretrained VGG-16 features
- # layers = 3
- Keep half of neurons

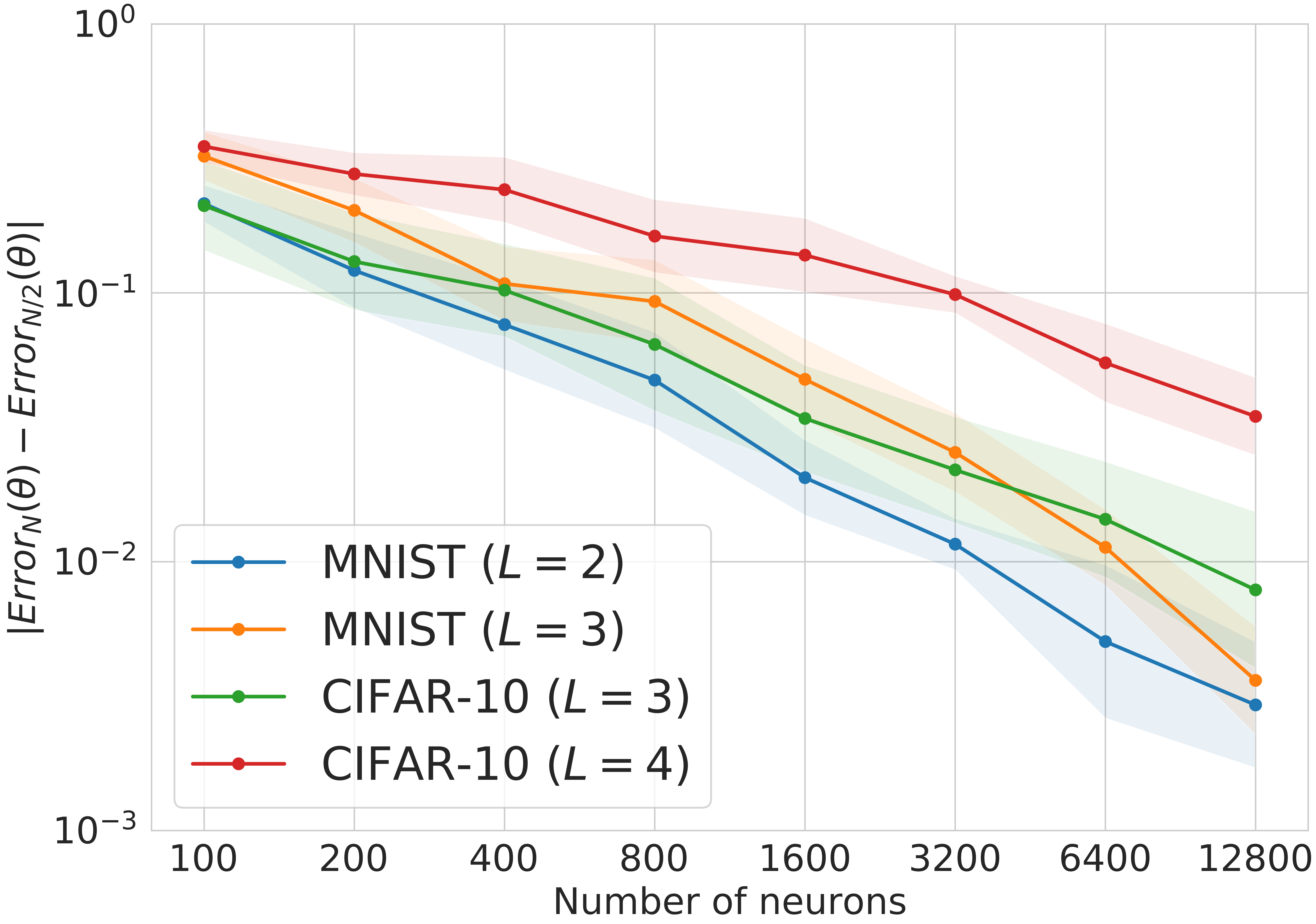


Numerical Results

- CIFAR-10 dataset
- Pretrained VGG-16 features
- # layers = 3
- Keep half of neurons

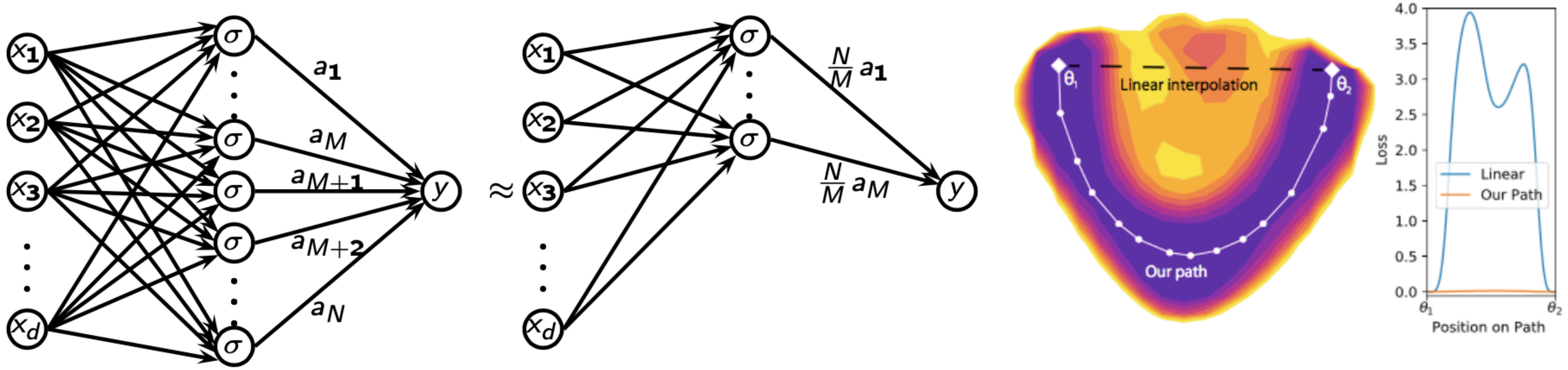


Numerical Results



Conclusion

Over-parameterization + SGD $\Rightarrow$ dropout stability & connectivity



Thank You for Your Attention

Over-parameterization + SGD $\Rightarrow$ dropout stability & connectivity