Infinitely Wide Nets

Eugene Golikov

BayesGroup seminar, October 23, 2020

DeepPavlov.ai,
Neural Networks and Deep Learning Lab.,
Moscow Institute of Physics and Technology,
Moscow, Russia

Introduction

(Real) neural nets are hard to study theoretically:

1. Non-convex optimization landscape;
2. Non-deterministic training procedure;
3. Existence of poorly-generalizing minima [Zhang et al., 2016].

What can we do:

1. Come up with a theoretically-tractable proxy-model;
2. Relate a real net to this proxy.

Consider a neural net training process; **hyperparameters are:**

1. Learning rate;
2. Batch size;
3. Depth ($\propto$ number of dense/conv layers);
4. Width ($\propto$ number of hidden neurons);
5. ...

Taking a limit wrt to each of these hyperparameters may simplify the model:

1. Learning rate $\to 0 \Rightarrow$ continuous-time GD;
2. Batch size $\to \infty \Rightarrow$ deterministic GD;
3. Depth $\to \infty \Rightarrow$ ODENet (?) [Chen et al., 2018];
4. Width $\to \infty \Rightarrow$ **our topic today.**

There are **multiple infinite-width limits:**

1. A (constant) NTK limit: [Jacot et al., 2018];
2. A mean-field limit: multiple works.[1]

The cause of difference is **a hyperparameter scaling.**

Questions:

1. What are the properties of these limits (convergence/generalization)?
2. Other infinite-width limits?
3. Which of the limits is the best proxy-model for a finite-width net?

[1] [Mei et al., 2018, Mei et al., 2019, Rotskoff and Vanden-Eijnden, 2019, Chizat and Bach, 2018, Sirignano and Spiliopoulos, 2020, Yarotsky, 2018]

NTK limit

Consider a model $f(\mathbf{x}; \theta)$;
we minimize a loss $\mathcal{L}(\theta) = \mathbb{E}_{\mathbf{x},y} \ell(y, f(\mathbf{x}; \theta))$ with GD:

$$\dot{\theta}_t = -\eta \mathbb{E}_{\mathbf{x},y} \left. \frac{\partial \ell(y, z)}{\partial z} \right|_{z = f(\mathbf{x}; \theta_t)} \nabla_\theta f(\mathbf{x}; \theta_t); \qquad \theta_0 \sim \mathcal{P}_{init}.$$

This implies **a kernel gradient descent:**

$$\dot{f}_t(\mathbf{x}') = \nabla_\theta^T f(\mathbf{x}'; \theta_t) \dot{\theta}_t = -\eta \mathbb{E}_{\mathbf{x},y} \left. \frac{\partial \ell(y, z)}{\partial z} \right|_{z = f(\mathbf{x}; \theta_t)} K_t(\mathbf{x}', \mathbf{x}); \quad f_0 \sim \mathcal{F}_{init}.$$

Here we have introduced **a neural tangent kernel (NTK):**

$$K_t(\mathbf{x}', \mathbf{x}) = \nabla_\theta^T f(\mathbf{x}'; \theta_t) \nabla_\theta f(\mathbf{x}; \theta_t).$$

Note:

1. All info about the weights is "hidden" inside the kernel;
2. NTK is generally stochastic and evolves with time.

First consider a model with L hidden layers of width d in **default parameterization:**

$$f_{def}(\mathbf{x};\theta) = \sum_{r_L=1}^{d} \theta^L_{r_L} \phi\left(\dots \sum_{r_1=1}^{d} \theta^1_{r_2 r_1} \phi\left(\theta^{in,T}_{r_1} \mathbf{x}\right)\right).$$

The training process is:

$$\dot{\theta}_t = -\eta \mathbb{E}_{\mathbf{x},y} \left.\frac{\partial \ell(y,z)}{\partial z}\right|_{z=f_{def}(\mathbf{x};\theta_t)} \nabla_\theta f_{def}(\mathbf{x};\theta_t);$$

$$\theta^{in}_{r_1;0} \sim \mathcal{N}(0, I), \quad \theta^l_{r_{l+1},r_l;0} \sim \mathcal{N}(0, d^{-1}) \quad \forall l \in [L].$$

Up to a constant factor, the network is initialized with **He initialization scheme.**[2]

---

[2][He et al., 2015]

Consider then the same model in **NTK parameterization:**

$$f_{ntk}(\mathbf{x};\theta) = d^{-1/2}\sum_{r_L=1}^{d}\theta^{L}_{r_L}\phi\left(\dots d^{-1/2}\sum_{r_1=1}^{d}\theta^{1}_{r_2 r_1}\phi\left(\theta^{in,T}_{r_1}\mathbf{x}\right)\right).$$

The training process is:

$$\dot{\theta}_t = -\eta\mathbb{E}_{\mathbf{x},y}\left.\frac{\partial\ell(y,z)}{\partial z}\right|_{z=f_{ntk}(\mathbf{x};\theta_t)}\nabla_\theta f_{ntk}(\mathbf{x};\theta_t);$$

$$\theta^{in}_{r_1;0}\sim\mathcal{N}(0,I),\quad \theta^{l}_{r_{l+1},r_l;0}\sim\mathcal{N}(0,1)\quad \forall l\in[L].$$

Important:

1. The initialization does not depend on d now;
2. The initial model didn't change but the training process did: $f_{ntk;0} = f_{def;0}$ **but** $f_{ntk;t} \neq f_{def;t}\ \forall t > 0$;
3. **The NTK converges to a constant deterministic kernel:** $\lim_{d\to\infty} K_t(\mathbf{x}',\mathbf{x}) = \mathbb{E}\, K_0(\mathbf{x}',\mathbf{x})$.

For the sake of illustration, consider $L = 1$ with NTK parameterization:

$$f_{ntk}(\mathbf{x}; \mathbf{a}, W) = d^{-1/2} \sum_{r=1}^{d} a_r \phi(\mathbf{w}_r^T \mathbf{x}).$$

$$\dot{a}_{r;t} = -\eta \mathbb{E} \left. \frac{\partial \ell(y, z)}{\partial z} \right|_{z = f_{ntk}(\mathbf{x}; \mathbf{a}_t, W_t)} d^{-1/2} \phi(\mathbf{w}_{r;t}^T \mathbf{x}), \quad a_{r;0} \sim \mathcal{N}(0, 1);$$

$$\dot{\mathbf{w}}_{r;t} = -\eta \mathbb{E} \left. \frac{\partial \ell(y, z)}{\partial z} \right|_{z = f_{ntk}(\mathbf{x}; \mathbf{a}_t, W_t)} d^{-1/2} a_{r;t} \phi'(\mathbf{w}_{r;t}^T \mathbf{x}) \mathbf{x}, \quad \mathbf{w}_{r;0} \sim \mathcal{N}(0, I).$$

Note: $\dot{a}_{r;t}$ and $\dot{\mathbf{w}}_{r;t}$ go to zero as $d \to \infty$.

Hence **the weights do not evolve in the limit.**

$$
\begin{aligned}
K_t(\mathbf{x}', \mathbf{x}) = \nabla_{\mathbf{a}}^T f(\mathbf{x}'; \mathbf{a}_t, W_t) \nabla_{\mathbf{a}} f(\mathbf{x}; \mathbf{a}_t, W_t) + \\
+ \operatorname{tr}(\nabla_W^T f(\mathbf{x}'; \mathbf{a}_t, W_t) \nabla_W f(\mathbf{x}; \mathbf{a}_t, W_t)) = \\
= d^{-1} \sum_{r=1}^{d} \left(\phi(\mathbf{w}_{r;t}^T \mathbf{x}') \phi(\mathbf{w}_{r;t}^T \mathbf{x}) + |a_{r;t}|^2 \phi'(\mathbf{w}_{r;t}^T \mathbf{x}') \phi'(\mathbf{w}_{r;t}^T \mathbf{x}) \mathbf{x}'^{,T} \mathbf{x} \right) \to \\
\to \mathbb{E}_{(a, \mathbf{w}) \sim \mathcal{N}(0, I)} \left(\phi(\mathbf{w}^T \mathbf{x}') \phi(\mathbf{w}^T \mathbf{x}) + |a|^2 \phi'(\mathbf{w}^T \mathbf{x}') \phi'(\mathbf{w}^T \mathbf{x}) \mathbf{x}'^{,T} \mathbf{x} \right) \neq 0.
\end{aligned}
$$

The NTK converges to a constant deterministic kernel due to LLN.

For comparison consider $L = 1$ with default parameterization:

$$f_{def}(\mathbf{x}; \mathbf{a}, W) = \sum_{r=1}^{d} a_r \phi(\mathbf{w}_r^T \mathbf{x}).$$

$$\dot{a}_{r;t} = -\eta \mathbb{E}\ \left.\frac{\partial \ell(y,z)}{\partial z}\right|_{z=f_{def}(\mathbf{x};\mathbf{a}_t,W_t)} \phi(\mathbf{w}_{r;t}^T \mathbf{x}), \quad a_{r;0} \sim \mathcal{N}(0, d^{-1});$$

$$\dot{\mathbf{w}}_{r;t} = -\eta \mathbb{E}\ \left.\frac{\partial \ell(y,z)}{\partial z}\right|_{z=f_{def}(\mathbf{x};\mathbf{a}_t,W_t)} a_{r;t} \phi'(\mathbf{w}_{r;t}^T \mathbf{x}) \mathbf{x}, \quad \mathbf{w}_{r;0} \sim \mathcal{N}(0, I).$$

Now $\dot{a}_{r;t}$ and $\dot{\mathbf{w}}_{r;t}$ **do not go to zero** as $d \to \infty$.

$$K_t(\mathbf{x}', \mathbf{x}) = \sum_{r=1}^{d} \left(\phi(\mathbf{w}_{r;t}^T \mathbf{x}') \phi(\mathbf{w}_{r;t}^T \mathbf{x}) + |a_{r;t}|^2 \phi'(\mathbf{w}_{r;t}^T \mathbf{x}') \phi'(\mathbf{w}_{r;t}^T \mathbf{x}) \mathbf{x}'^{,T} \mathbf{x} \right).$$

The kernel diverges at initialization: $K_0(\mathbf{x}', \mathbf{x}) \to \infty$.

Consider a model f_d of width d with **NTK parameterization.**

Theorem (convergence to a limit model; [Jacot et al., 2018])
For sufficiently regular ϕ $K_{d,t} \to K_\infty = \mathbb{E}\, K_{d,0}$ and $f_{d,t} \to f_{\infty,t}$ as $d \to \infty$, where limit dynamics is given as:

$$\dot{f}_{\infty,t}(\mathbf{x}') = -\eta \mathbb{E}_{\mathbf{x},y} \left. \frac{\partial \ell(y,z)}{\partial z} \right|_{z=f_{\infty,t}(\mathbf{x})} K_\infty(\mathbf{x}',\mathbf{x}), \quad f_{\infty,0}(\mathbf{x}) \sim \mathcal{N}(0, \sigma_0^2(\mathbf{x})).$$

Question: what is the limit model for the default parameterization?
We shall discuss it later on.[3]

[3] or, see [Golikov, 2020a].

Suppose we have a train dataset of size n: $S_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$. Assume:

1. l_2 loss: $\ell(y, z) = \frac{1}{2}|y - z|^2$;
2. The Gramian $G = \{K_\infty(\mathbf{x}_i, \mathbf{x}_j)\}_{i,j=1}^n$ is positive definite.

Then $f_{\infty,t}$ **converges to a global minimum on the train dataset.**

Indeed, consider l_2-regression:

$$\dot{f}_{\infty,t}(\mathbf{x}) = \eta \frac{1}{n} \sum_{j=1}^{n} (y_j - f_{\infty,t}(\mathbf{x}_j)) K_\infty(\mathbf{x}, \mathbf{x}_j).$$

Denote $\mathbf{y} = \{y_i\}_{i=1}^n$, $\hat{\mathbf{y}}_t = \{f_{\infty,t}(\mathbf{x}_i)\}_{i=1}^n$.

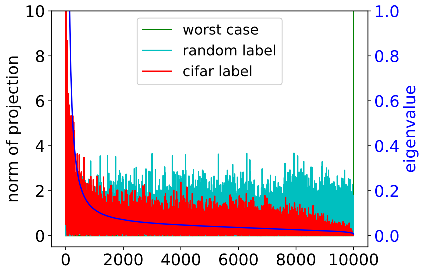
Let $\{(\lambda_i, \mathbf{v}_i)\}_{i=1}^n$ be a set of eigenvalue-eigenvector pairs for G. Then (see [Arora et al., 2019b]):

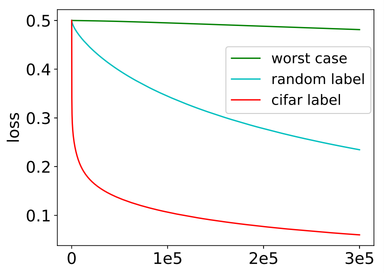
$$\|\hat{\mathbf{y}}_t - \mathbf{y}\|_2^2 = \sum_{i=1}^{n} ((\hat{\mathbf{y}}_0 - \mathbf{y})^T \mathbf{v}_i)^2 e^{-\frac{2\eta}{n} \lambda_i t}.$$

Important: assuming $\hat{\mathbf{y}}_0 = 0$, the speed of convergence is related to a **spectrum alignment** $\{(\mathbf{y}^T \mathbf{v}_i)^2\}_{i=1}^n$.

$$\|\hat{\mathbf{y}}_t - \mathbf{y}\|_2^2 = \sum_{i=1}^{n} (\mathbf{y}^T \mathbf{v}_i)^2 e^{-\frac{2\eta}{n}\lambda_i t}.$$

Norm of projection: $\mathbf{y}^T\mathbf{v}_i$; eigenvalue: λ_i.





So far, we have two results:

1. A finite-width model converges to a limit one as $d \to \infty$;
2. A limit model converges to a global minimum as $t \to \infty$ (**asymptotic convergence guarantee**).

Theorem (non-asymptotic conv. guarantee; [Du et al., 2018])
Consider a two-layered network with ReLU activations.

$\exists C$: *for* $\delta > 0$ *and* $d \geq C \frac{n^6}{\delta^3 \lambda_n^4}$ *(large but finite width)*

$$\|\hat{\mathbf{y}}_t - \mathbf{y}\|_2^2 \leq \exp\left(-\frac{2\eta}{n} \lambda_n t\right) \quad \textit{w.p.} \geq 1 - \delta.$$

[Song and Yang, 2019]: the same guarantee for $d \geq C \frac{n^4}{\lambda_n^4} \log^3\left(\frac{n}{\delta}\right)$.
[Arora et al., 2019b]: a similar guarantee for the spectrum alignment.

- Consider l_1 loss: $\ell(y, z) = |y - z|$.
- Assume $f_0 \equiv 0$.
- Suppose we have converged to a zero loss on the dataset $S_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ sampled from $\mathcal{D}$. Let $\hat{f}_n$ be the final network.

Theorem (non-asymptotic generalization guarantee; [Arora et al., 2019b])
Consider a two-layered network with ReLU activations.

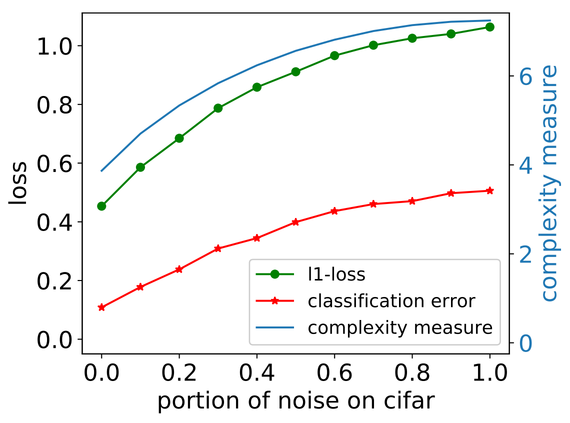
Then given $\delta \in (0, 1)$ for sufficiently large d w.p. $\geq 1 - \delta$ over S_n and initialization

$$\mathbb{E}_{\mathbf{x}, y \sim \mathcal{D}} \ell(y, \hat{f}_n(\mathbf{x})) \leq \sqrt{\frac{2\mathbf{y}^T G^{-1} \mathbf{y}}{n}} + O\left(\sqrt{\frac{\log \frac{n}{\lambda_n \delta}}{n}}\right).$$

Intuition: if we train a network on a dataset that aligns well with NTK then our network generalizes well w.h.p.

$$\mathbb{E}_{\mathcal{D}}\ell(y, \hat{f}_n(\mathbf{x})) \leq \sqrt{\frac{2\mathbf{y}^T G^{-1}\mathbf{y}}{n}} + O\left(\sqrt{\frac{\log \frac{n}{\lambda_n \delta}}{n}}\right) \text{ w.p. } \geq 1 - \delta.$$

A complexity measure: $\sqrt{2\mathbf{y}^T G^{-1}\mathbf{y}/n}$.



Recall the definition of NTK:

$$K_\infty(\mathbf{x}', \mathbf{x}) = \mathbb{E}_{\theta_0} K_0(\mathbf{x}', \mathbf{x}) = \mathbb{E}_{\theta_0} \nabla_\theta^T f(\mathbf{x}'; \theta_0) \nabla_\theta f(\mathbf{x}; \theta_0).$$

How to compute it?

1. Via Monte-Carlo [Lee et al., 2019]:
   - +: applicable to any architecture;
   - -: noisy.
2. Analytically [Arora et al., 2019a]:
   - +: exact and efficient;
   - -: available only for ReLU FC and Conv nets w/o BNs etc.

Depth	CNN	CNTK
3	63.81%	**70.47%**
4	**80.93%**	75.93%
6	**83.75%**	76.73%
11	**82.92%**	77.43%
21	**83.30%**	77.08%

Table 1: Comparing deep CNNs trained with square loss with their constant-kernel counterparts [Arora et al., 2019a]. **Dataset:** CIFAR10.

Conclusion: if we fix the kernel, performance gets worse.

Hence the kernel evolution is important.

Mean-field limit

Consider **a neural net with a single hidden layer:**

$$f_d(\mathbf{x}) = \frac{1}{d} \sum_{r=1}^{d} a_r \phi(\mathbf{w}_r^T \mathbf{x}).$$

Note the factor d^{-1} instead of $d^{-1/2}$ (NTK) or d^0 (default).

The training process is:

$$\dot{a}_{r;t} = -\eta \mathbb{E} \left. \frac{\partial \ell(y,z)}{\partial z} \right|_{z=f_{d,t}(\mathbf{x})} d^{-1} \phi(\mathbf{w}_{r;t}^T \mathbf{x}), \quad a_{r;0} \sim \mathcal{N}(0,1);$$

$$\dot{\mathbf{w}}_{r;t} = -\eta \mathbb{E} \left. \frac{\partial \ell(y,z)}{\partial z} \right|_{z=f_{d,t}(\mathbf{x})} d^{-1} a_{r;t} \phi'(\mathbf{w}_{r;t}^T \mathbf{x}) \mathbf{x}, \quad \mathbf{w}_{r;0} \sim \mathcal{N}(0,I).$$

Take $\eta = \eta^* d$. Then $\dot{a}_{r;t}$ and $\dot{\mathbf{w}}_{r;t}$ do not go to zero as $d \to \infty$.
Hence the weights evolve.

Consider **a weight-space measure:**

$$\mu_d = d^{-1} \sum_{r=1}^{d} \delta_{a_r} \otimes \delta_{\mathbf{w}_r} \in \mathcal{M}(\mathbb{R}^{1+d_{\mathbf{x}}}).$$

We can express the model in terms of this measure:

$$f_d(\mathbf{x}) = \int a\phi(\mathbf{w}^T \mathbf{x})\, \mu_d(da, d\mathbf{w}).$$

Also, express the training process [Rotskoff and Vanden-Eijnden, 2019]:

$$\dot{\mu}_{d,t} = -\eta^* \operatorname{div}(\mu_{d,t} \mathbf{v}_{d,t}), \quad \mu_{d,0} = d^{-1} \sum_{r=1}^{d} \delta_{a_{r;0}} \otimes \delta_{\mathbf{w}_{r;0}},$$

$$\mathbf{v}_{d,t}(a, \mathbf{w}) = \mathbb{E}_{\mathbf{x},y} \left.\frac{\partial \ell(y, z)}{\partial z}\right|_{z = f_{d,t}(\mathbf{x})} [\phi(\mathbf{w}^T \mathbf{x}), a\phi'(\mathbf{w}^T \mathbf{x})\mathbf{x}^T]^T.$$

Important: the weights are now "hidden" inside the measure.

The initial measure is random:

$$\mu_{d,0} = d^{-1} \sum_{r=1}^{d} \delta_{a_{r;0}} \otimes \delta_{\mathbf{w}_{r;0}}, \quad \delta_{a_{r;0}} \sim \mathcal{N}(0,1),\ \delta_{\mathbf{w}_{r;0}} \sim \mathcal{N}(0, I_{d_{\mathbf{x}}}) \quad \forall r \in [d].$$

However **it converges to a deterministic one:**

$$\lim_{d \to \infty} \mu_{d,0} = \mu_{\infty,0} = \mathcal{N}(0, I_{1+d_{\mathbf{x}}}).$$

This gives the limit dynamics:

$$\dot{\mu}_{\infty,t} = -\eta^* \operatorname{div}(\mu_{\infty,t} \mathbf{v}_{\infty,t}), \quad \mu_{\infty,0} = \mathcal{N}(0, I_{1+d_{\mathbf{x}}}),$$

$$\mathbf{v}_{\infty,t}(a, \mathbf{w}) = \mathbb{E}_{\mathbf{x},y} \left. \frac{\partial \ell(y,z)}{\partial z} \right|_{z=f_{\infty,t}(\mathbf{x})} [\phi(\mathbf{w}^T \mathbf{x}), a\phi'(\mathbf{w}^T \mathbf{x}) \mathbf{x}^T]^T,$$

$$f_{\infty,t}(\mathbf{x}) = \int a\phi(\mathbf{w}^T \mathbf{x})\, \mu_{\infty,t}(da, d\mathbf{w}).$$

This limit is referred as **the mean-field limit.**

What happens to the kernel in the mean-field limit?

$$K_t(\mathbf{x}', \mathbf{x}) = \eta \nabla_{\mathbf{a}}^T f(\mathbf{x}'; \mathbf{a}_t, W_t) \nabla_{\mathbf{a}} f(\mathbf{x}; \mathbf{a}_t, W_t) +$$

$$+ \eta \operatorname{tr}(\nabla_W^T f(\mathbf{x}'; \mathbf{a}_t, W_t) \nabla_W f(\mathbf{x}; \mathbf{a}_t, W_t)) =$$

$$= \eta^* d^{-1} \sum_{r=1}^{d} \left(\phi(\mathbf{w}_{r;t}^T \mathbf{x}') \phi(\mathbf{w}_{r;t}^T \mathbf{x}) + |a_{r;t}|^2 \phi'(\mathbf{w}_{r;t}^T \mathbf{x}') \phi'(\mathbf{w}_{r;t}^T \mathbf{x}) \mathbf{x}'^{,T} \mathbf{x} \right) \to$$

$$\to \eta^* \int \left(\phi(\mathbf{w}^T \mathbf{x}') \phi(\mathbf{w}^T \mathbf{x}) + |a|^2 \phi'(\mathbf{w}^T \mathbf{x}') \phi'(\mathbf{w}^T \mathbf{x}) \mathbf{x}'^{,T} \mathbf{x} \right) \, d\mu_{\infty,t}(da, d\mathbf{w}).$$

It converges, but evolves with time.

A mean-field limit for multi-layered nets?

- Not obvious, how to express a finite-width dynamics in terms of the measure.
- Still, a limit dynamics can be expressed as a measure evolution [Araújo et al., 2019].
- Heuristic: if $\phi(0) = 0$ and initialization is zero-centered, then a limit model vanishes if the number of hidden layers is at least three [Golikov, 2020b].

Open questions:

- Non-asymptotic convergence guarantees, as for the NTK limit?
- Generalization guarantees?

A general treatment

Consider a network with a single hidden layer:

$$f_d(\mathbf{x}) = \sigma(d) \sum_{r=1}^{d} a_r \phi(\mathbf{w}_r^T \mathbf{x}), \quad a_{r;0} \sim \mathcal{N}(0,1),\ \mathbf{w}_{r;0} \sim \mathcal{N}(0,I) \quad \forall r \in [d].$$

- A scaling $\sigma \propto d^{-1/2}$, $\eta = \mathrm{const}$ leads to the **NTK limit** as $d \to \infty$.
- A scaling $\sigma \propto d^{-1}$, $\eta \propto d$ leads to the **mean-field limit** as $d \to \infty$.

Questions:

1. What is a limit dynamics for the default parameterization?
2. Do other hyperparameter scalings lead to "well-defined" limits?
3. Which limit dynamics describe the finite-width one best?

Setup (following [Golikov, 2020a]):

- **A model:**

$$f(\mathbf{x}; W, \mathbf{a}) = \sum_{r=1}^{d} a_r \phi(\mathbf{w}_r^T \mathbf{x}),$$

where ϕ is real analytic.

- **A training procedure**:

$$a_r^{(k+1)} = a_r^{(k)} - \eta_a \mathbb{E}_{\mathbf{x},y} \left(\nabla_{f_d}^{(k)} \ell(\mathbf{x}, y) \phi(\mathbf{w}_r^{(k),T} \mathbf{x}) \right),$$

$$\mathbf{w}_r^{(k+1)} = \mathbf{w}_r^{(k)} - \eta_w \mathbb{E}_{\mathbf{x},y} \left(\nabla_{f_d}^{(k)} \ell(\mathbf{x}, y) a_r^{(k)} \phi'(\mathbf{w}_r^{(k),T} \mathbf{x}) \mathbf{x} \right),$$

$$a_r^{(0)} \sim \mathcal{N}(0, \sigma_a^2), \quad \mathbf{w}_r^{(0)} \sim \mathcal{N}(0, \sigma_w^2 I), \quad \forall r \in [d],$$

where $\nabla_{f_d}^{(k)} \ell(\mathbf{x}, y) = \left. \frac{\partial \ell(y,z)}{\partial z} \right|_{z = f(W^{(k)}, \mathbf{a}^{(k)}, \mathbf{x})}$.

Introduce scaled quantities:

$$\hat{\mathbf{w}}_r = \frac{\mathbf{w}_r}{\sigma_w}, \quad \hat{a}_r = \frac{a_r}{\sigma_a}, \quad \hat{\eta}_w = \frac{\eta_w}{\sigma_w^2}, \quad \hat{\eta}_a = \frac{\eta_a}{\sigma_a^2}.$$

The model becomes:

$$f_d^{(k)}(\mathbf{x}) = \sigma_a \sum_{r=1}^{d} \hat{a}_r^{(k)} \phi(\sigma_w \hat{\mathbf{w}}_r^{(k),T} \mathbf{x}) = \sigma \sum_{r=1}^{d} \hat{a}_r^{(k)} \phi(\hat{\mathbf{w}}_r^{(k),T} \mathbf{x}),$$

where $\sigma = \sigma_a$ and take $\sigma_w = 1$ w.l.o.g.

The training procedure becomes:

$$\hat{\mathbf{w}}_r^{(k+1)} = \hat{\mathbf{w}}_r^{(k)} - \hat{\eta}_w \sigma \mathbb{E}_{\mathbf{x},y} \left(\nabla_{f_d}^{(k)} \ell(\mathbf{x}, y) \hat{a}_r^{(k)} \phi'(\hat{\mathbf{w}}_r^{(k),T} \mathbf{x}) \mathbf{x} \right),$$

$$\hat{a}_r^{(k+1)} = \hat{a}_r^{(k)} - \hat{\eta}_a \sigma \mathbb{E}_{\mathbf{x},y} \left(\nabla_{f_d}^{(k)} \ell(\mathbf{x}, y) \phi(\hat{\mathbf{w}}_r^{(k),T} \mathbf{x}) \right),$$

$$\hat{a}_r^{(0)} \sim \mathcal{N}(0,1) \quad \hat{\mathbf{w}}_r^{(0)} \sim \mathcal{N}(0, I), \quad \forall r \in [d].$$

This dynamics is driven by three hyperparameters: σ, $\hat{\eta}_a$, $\hat{\eta}_w$.

Assume power-law dependencies with respect to width d:

$$\sigma = \sigma^* (d/d^*)^{q_\sigma}, \quad \hat{\eta}_a = \hat{\eta}_a^* (d/d^*)^{\tilde{q}_a}, \quad \hat{\eta}_w = \hat{\eta}_w^* (d/d^*)^{\tilde{q}_w}.$$

$$\sigma = \sigma^* (d/d^*)^{q_\sigma}, \quad \hat{\eta}_a = \hat{\eta}_a^* (d/d^*)^{\tilde{q}_a}, \quad \hat{\eta}_w = \hat{\eta}_w^* (d/d^*)^{\tilde{q}_w}.$$

Available scalings:

1. **NTK:** $q_\sigma = -\frac{1}{2}$, $\tilde{q}_a = \tilde{q}_w = 0$.
2. **Mean-field:** $q_\sigma = -1$, $\tilde{q}_a = \tilde{q}_w = 1$.
3. **"Default":**
   - He initialization: $\sigma = \sigma_a \propto d^{-1/2}$.
   - Constant learning rates: $\eta_a \propto 1 \Rightarrow \hat{\eta}_a = \eta_a \sigma_a^{-2} \propto d$, $\hat{\eta}_w = \eta_w \propto 1$.

   Hence $q_\sigma = -\frac{1}{2}$, $\tilde{q}_a = 1$, $\tilde{q}_w = 0$.
4. **"Sym-default":** $q_\sigma = -\frac{1}{2}$, $\tilde{q}_a = \tilde{q}_w = \frac{1}{2}$. Almost the same dynamics as for the default scaling but $\tilde{q}_a = \tilde{q}_w$.

Assume $\tilde{q}_a = \tilde{q}_w = \tilde{q}$.

Question: can we have a "well-defined" limit model evolution for other scalings?

What do we mean by "well-defined" by the way?

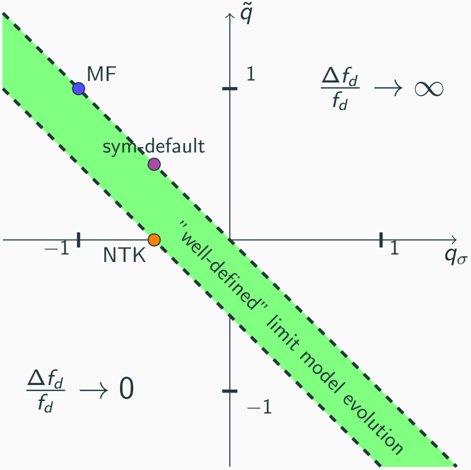
Definition (well-definiteness; informal)
We say "a scaling $(q_\sigma, \tilde{q})$ defines a well-defined limit model" if

$$\exists k^* : \; \forall k \geq k^* \quad \frac{\Delta f_d^{(k)}}{f_d^{(k^*)}} = \Theta_{d \to \infty}(1).$$

Here $\Delta f_d^{(k)} = f_d^{(k+1)} - f_d^{(k)}$.

In other words, the change of logits should be comparable to logits themselves.

Can we have a "well-defined" limit model evolution for other scalings?



Note: MF, NTK, and sym-default scalings are special (later).

Possible properties of limit models:

1. A limit model at initialization is finite;
2. Tangent kernels at initialization are finite;
3. Tangent kernels and a limit model are of the same order at initialization;
4. Tangent kernels start to evolve.

Note: a finite-width model satisfies all of these properties.

Consequence: these properties are necessary for a limit model to approximate a finite-width net.

Each property can be expressed in terms of a scaling:

1. A limit model at initialization is finite:

$$f_d^{(0)} = \Theta_{d\to\infty}(1) \;\Rightarrow\; q_\sigma + 1/2 = 0;$$

2. Tangent kernels at initialization are finite:

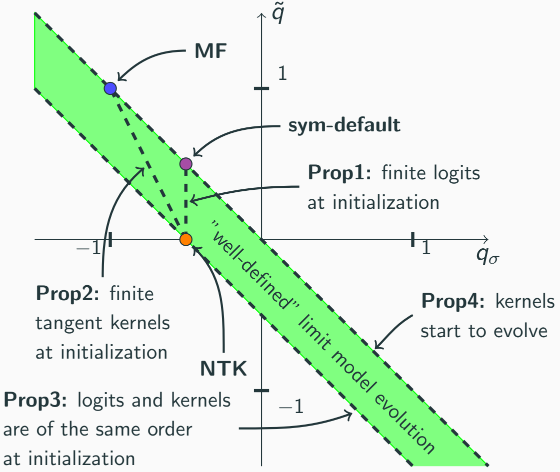
$$K_d^{(0)} = \Theta_{d\to\infty}(1) \;\Rightarrow\; 2q_\sigma + \tilde{q} + 1 = 0;$$

3. Tangent kernels and a limit model are of the same order at initialization:

$$K_d^{(0)} = \Theta_{d\to\infty}(f_d^{(0)}) \;\Rightarrow\; q_\sigma + \tilde{q} + 1/2 = 0;$$

4. Tangent kernels start to evolve:

$$\Delta K_d^{(0)} = \Theta_{d\to\infty}(K_d^{(0)}) \;\Rightarrow\; q_\sigma + \tilde{q} = 0.$$



- NTK, MF, and sym-default limits satisfy the maximal number of properties of finite-width models.
- Each region in the $(q_\sigma, \tilde{q})$-plane corresponds to a distinct limit model. Hence **the number of possible limit models are finite.**

How to satisfy all of these properties in the limit?

Start with a MF-scaling:

$$f_{mf,d}(\mathbf{x}) = \sigma^*(d/d^*)^{-1} \sum_{r=1}^{d} \hat{a}_r \phi(\hat{\mathbf{w}}_r^T \mathbf{x}).$$

It violates property 1: $f_d^{(0)} \to 0$ as $d \to \infty$.

Modify a model:

$$f_{icmf,d}(\mathbf{x}) = \sigma^*(d/d^*)^{-1} \sum_{r=1}^{d} \hat{a}_r \phi(\hat{\mathbf{w}}_r^T \mathbf{x}) + \sigma^*(d/d^*)^{-1/2} \sum_{r=1}^{d} \hat{a}_r^{(0)} \phi(\hat{\mathbf{w}}_r^{(0),T} \mathbf{x}).$$

We call the corresponding limit model an **initialization-corrected mean-field** limit (IC-MF).

Important: IC-MF limit satisfies all of the properties considered above.

Hypothesis: the IC-MF limit approximates the finite-width model better than other limit models.

How to test it?

Consider a "reference" network of width d^*. Assume:

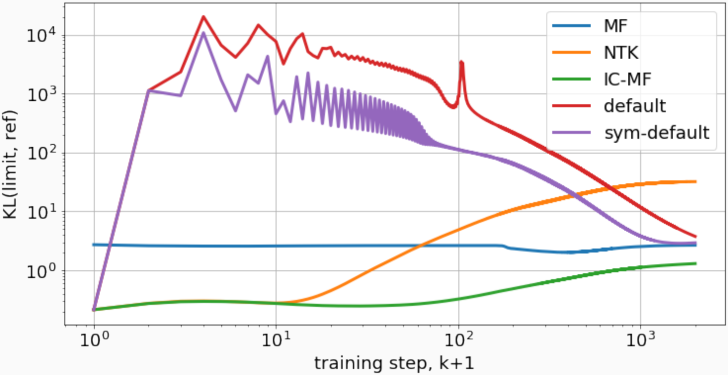
$$\sigma(d) = \sigma^*(d/d^*)^{q_\sigma}, \quad \hat{\eta}_{a/w}(d) = \hat{\eta}^*_{a/w}(d/d^*)^{\tilde{q}_{a/w}}.$$

Consider a metric: $\mathbb{E}_{\mathbf{x}\sim\mathcal{D}} D_{logits}(f_\infty^{(k)}(\mathbf{x}) \,||\, f_{d^*}^{(k)}(\mathbf{x}))$, where

$$D_{logits}(\xi \,||\, \xi^*) = \mathrm{KL}(\mathcal{N}(\mathbb{E}\,\xi, \mathbb{V}\mathrm{ar}\,\xi) \,||\, \mathcal{N}(\mathbb{E}\,\xi^*, \mathbb{V}\mathrm{ar}\,\xi^*)).$$

We measure: $\mathbb{E}_{\mathbf{x}\sim\mathcal{D}} D_{logits}(f_\infty^{(k)}(\mathbf{x}) \,||\, f_{d^*}^{(k)}(\mathbf{x}))$, where

$$D_{logits}(\xi \,||\, \xi^*) = \mathrm{KL}(\mathcal{N}(\mathbb{E}\,\xi, \mathbb{V}\mathrm{ar}\,\xi) \,||\, \mathcal{N}(\mathbb{E}\,\xi^*, \mathbb{V}\mathrm{ar}\,\xi^*)).$$



How do limit dynamics look like:

- **NTK limit:** dynamics in a function space driven by a constant deterministic kernel;
- **MF limit:** deterministic dynamics in a measure space;
- **Sym-default limit:** deterministic dynamics in a measure space too [Golikov, 2020a];
- **Default limit:** again, deterministic dynamics in a measure space.

Take-aways:

1. One can consider an infinite-width limit as a proxy-model for a finite-width net;
2. There are good optimization and generalization guarantees for the NTK limit;
3. The NTK can be computed exactly for simple deep nets;
4. Mean-field and NTK limits are not the only possible ones;
5. There are a finite number of possible infinite-width limits depending on parameter scaling;
6. The NTK limit is not a perfect proxy for finite-width nets;
7. For shallow nets the IC-MF limit is a better proxy than the NTK one.

Bibliography

Araújo, D., Oliveira, R. I., and Yukimura, D. (2019).
A mean-field limit for certain deep neural networks.
arXiv preprint arXiv:1906.00193.

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R. R., and Wang, R. (2019a).
On exact computation with an infinitely wide neural net.
In *Advances in Neural Information Processing Systems*, pages 8139–8148.

Arora, S., Du, S. S., Hu, W., Li, Z., and Wang, R. (2019b).
Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks.
arXiv preprint arXiv:1901.08584.

Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. (2018).
Neural ordinary differential equations.
In *Advances in neural information processing systems*, pages 6571–6583.

Chizat, L. and Bach, F. (2018).
On the global convergence of gradient descent for over-parameterized models using optimal transport.
In *Advances in neural information processing systems*, pages 3036–3046.

Du, S. S., Zhai, X., Poczos, B., and Singh, A. (2018).
Gradient descent provably optimizes over-parameterized neural networks.
arXiv preprint arXiv:1810.02054.

Golikov, E. A. (2020a).
Dynamically stable infinite-width limits of neural classifiers.
arXiv preprint arXiv:2006.06574.

Golikov, E. A. (2020b).
Towards a general theory of infinite-width limits of neural classifiers.
arXiv preprint arXiv:2003.05884.

He, K., Zhang, X., Ren, S., and Sun, J. (2015).
Delving deep into rectifiers: Surpassing human-level performance on imagenet classification.
In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034.

Jacot, A., Gabriel, F., and Hongler, C. (2018).
Neural tangent kernel: Convergence and generalization in neural networks.
In *Advances in neural information processing systems*, pages 8571–8580.

Lee, J., Xiao, L., Schoenholz, S., Bahri, Y., Novak, R., Sohl-Dickstein, J., and Pennington, J. (2019).
Wide neural networks of any depth evolve as linear models under gradient descent.
In *Advances in neural information processing systems*, pages 8570–8581.

Mei, S., Misiakiewicz, T., and Montanari, A. (2019).

Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit.
arXiv preprint arXiv:1902.06015.

Mei, S., Montanari, A., and Nguyen, P.-M. (2018).
A mean field view of the landscape of two-layer neural networks.
Proceedings of the National Academy of Sciences, 115(33):E7665–E7671.

Rotskoff, G. M. and Vanden-Eijnden, E. (2019).
Trainability and accuracy of neural networks: an interacting particle system approach.
stat, 1050:30.

Sirignano, J. and Spiliopoulos, K. (2020).
Mean field analysis of neural networks: A law of large numbers.
SIAM Journal on Applied Mathematics, 80(2):725–752.

Song, Z. and Yang, X. (2019).

Quadratic suffices for over-parametrization via matrix chernoff bound.
arXiv preprint arXiv:1906.03593.

Yarotsky, D. (2018).
Collective evolution of weights in wide neural networks.
arXiv preprint arXiv:1810.03974.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016).
Understanding deep learning requires rethinking generalization.
arXiv preprint arXiv:1611.03530.