Neural Tangent Kernel

High-Dimensional Probability Analysis

Hooman Zolfaghari - Abdollah Zohrabi - Amirreza Velae

Sharif University of Technology

February 5, 2025

Key Idea

- **Dynamics** of training infinitely wide NNs ≈
- **convex optimization** in RKHS
- For $f_\theta : \mathbb{R}^d \to \mathbb{R}$ (NN), GD training induces:

$$\underbrace{\Theta(\theta)_{i,j}}_{\text{NTK}} := \nabla_\theta f_\theta(x_i) \nabla_\theta f_\theta(x_j)^\top \in \mathbb{R}^{n \times n}$$

Crucial Property: $\Theta(\theta^{(0)}) \to \Theta^\infty$ as width $\to \infty$

Modern Perspective

- **Feature Learning Gap**: NTK regime $\neq$ real NNs (finite-width trains via $\nabla\Theta \neq 0$)
- **Deep vs Shallow**: Depth induces **spectral bias** (eigenvalue decay of Θ^{∞})

$$\partial_t f_t = -\Theta^{\infty}(f_t - y) \quad \text{(grad flow ODE)}$$

Good News:

- Global convergence
- Linear rate for $\lambda_{\min}(\Theta^{\infty}) > 0$

Bad News:

- No feature learning
- Fails for transformers/attention

- Research focused on more practical assumptions.
- Two-layer ReLU network $f_{\mathbf{W},\mathbf{a}}(\mathbf{x}) = \frac{1}{\sqrt{m}} \sum_{r=1}^{m} a_r \mathrm{ReLU}(\mathbf{w}_r^\top \mathbf{x})$
- Least Squares Regression $C(W) := \frac{1}{2} \sum_{i=1}^{n} (y_i - f_{\mathbf{W},\mathbf{a}}(\mathbf{x}_i))$
- Resulting NTK gram matrix:

$$\mathbf{H}_{ij}^{\infty} = \mathbb{E}\mathbf{w} \sim \mathcal{N}(0, \mathbf{I}) \left[\mathbf{x}_i^\top \mathbf{x}_j \mathbb{I} \left\{ \mathbf{w}^\top \mathbf{x}_i \geq 0, \mathbf{w}^\top \mathbf{x}_j \geq 0 \right\} \right] \tag{1}$$

$$= \frac{\mathbf{x}_i^\top \mathbf{x}_j \left(\pi - \arccos(\mathbf{x}_i^\top \mathbf{x}_j) \right)}{2\pi}, \quad \forall i, j \in [n]. \tag{2}$$

- If H^{∞} is positive definite $\lambda_0 := \lambda_{\min}(H^{\infty}) > 0$, GD converges to 0 training loss if m is sufficiently large $\Omega(\frac{n^6}{\lambda_0^4})$.

- Eigen-decomposition $H^{\infty} = \sum_{i=1}^{n} \lambda_i v_i v_i^{\top}$,.
- Suppose $\lambda_0 = \lambda_{\min}(H^{\infty}) > 0, \kappa = O\left(\frac{\varepsilon_0 \delta}{\sqrt{n}}\right), \quad m = \Omega\left(\frac{n^7}{\lambda_0^4 \kappa^2 \delta^4 \varepsilon^2}\right), \quad \eta = O\left(\frac{\lambda_0}{n^2}\right)$. Then with probability at least $1-\delta$ over the random initialization, for all $k = 0, 1, 2, \ldots$ we have

$$\|y - u(k)\|_2 = \sqrt{\sum_{i=1}^{n} \left(1 - \eta \lambda_i\right)^{2k} \left(v_i^{\top} y\right)^2} \pm \varepsilon. \tag{8}$$

- A distribution D over $\mathbb{R}^d \times \mathbb{R}$ is called (λ_0, δ, n)-*non-degenerate* if for n i.i.d. samples $\{(x_i, y_i)\}_{i=1}^{n}$ from D, with probability at least $1-\delta$ we have

$$\lambda_{\min}\left(H^{\infty}\right) \geq \lambda_0 > 0.$$

Fix a failure probability $\delta \in (0,1)$. Suppose our data $S = \{(x_i, y_i)\}_{i=1}^n$ are i.i.d. samples from a $(\lambda_0, \delta/3, n)$-non-degenerate distribution D, and let

$$\kappa = O\Big(\frac{\lambda_0 \delta}{n}\Big), \quad m \geq \kappa^{-2}\,\mathrm{poly}\big(n, \lambda_0^{-1}, \delta^{-1}\big).$$

Consider any loss function $\ell : \mathbb{R} \times \mathbb{R} \to [0,1]$ that is 1-Lipschitz in its first argument and satisfies $\ell(y,y) = 0$. Then with probability at least $1-\delta$ (over the random initialization *and* the training samples), the two-layer neural network $f_{\mathbf{W}(k),a}$ trained by gradient descent for

$$k \geq \Omega\Big(\frac{1}{\eta\lambda_0}\log\frac{n}{\delta}\Big) \quad \text{iterations}$$

has population loss

$$L_D\big(f_{\mathbf{W}(k),a}\big) = \mathbb{E}_{(x,y)\sim D}\big[\ell\big(f_{\mathbf{W}(k),a}(x), y\big)\big] \leq \sqrt{\frac{2y^\top \big(H^\infty\big)^{-1} y}{n}} + O\Big(\sqrt{\frac{\log\big(\frac{n}{\lambda_0\delta}\big)}{n}}\Big). \tag{9}$$

- We can see that the bound depends on:
  - Distribution $(x, y) \sim \mathscr{D}$ such that,
  - $\mathbf{y}^\top H^\infty \mathbf{y} \leq \|(H^\infty)^{-1}\| \|\mathbf{y}\|_2 = \lambda_{\min}(H^\infty) \|\mathbf{y}\|_2$
- To be able to provably learn (e.g. PAC-Learning), the bound must converge to 0 as $n \to \infty$.
- The paper mentions the case of $y = g(x)$ for some function g.
- So we focus on bounding $\lambda_{\min}(H^\infty)$ and $\|\mathbf{y}\|_2$

- Data scaling assumption. The data distribution P_X satisfies the following properties:
  1. $\int \|x\|_2 \, dP_X(x) = \Theta(\sqrt{d})$.
  2. $\int \|x\|_2^2 \, dP_X(x) = \Theta(d)$.
  3. $\int \left\| x - \int x' \, dP_X(x') \right\|_2^2 dP_X(x) = \Omega(d)$.
- These are just scaling conditions on the data vector x or its centered counterpart $x - \mathbb{E}x$. We remark that the data can have any scaling, but in this paper we fix it to be of order d for convenience. We further assume the following condition on the data distribution.

- Lipschitz concentration assumption. The data distribution P_X satisfies the Lipschitz concentration property. Namely, for every Lipschitz continuous function $f : \mathbb{R}^d \to \mathbb{R}$, there exists an absolute constant $c > 0$ such that, for all $t > 0$,

$$\mathbb{P}\left(|f(x) - \int f(x')\, dP_X(x')| \; > \; t \right) \; \leq \; 2\, e^{-c\, t^2 \,/\, \|f\|_{\text{Lip}}^2}.$$

- In general, this assumption covers the whole family of distributions that satisfy the log-Sobolev inequality with a dimension-independent constant (or distributions with log-concave densities).

Theorem (Smallest eigenvalue of limiting NTK)

Let $\{x_i\}_{i=1}^N$ be a set of i.i.d. data points from P_X, where P_X has zero mean and satisfies Assumptions 2.1 and 2.2. Let $K^{(L)}$ be the limiting NTK recursively defined in (9). Then, for any even integer constant $r \geq 2$, we have with probability at least

$$1 - N e^{-\Omega(d)} - N^2 e^{-\Omega\left(dN^{-2/(r-0.5)}\right)}$$

that

$$\mathrm{LO}(d) \geq \lambda_{\min}\left(K^{(L)}\right) \geq \mu_r(\sigma)^2 \, \Omega(d),$$

where $\mu_r(\sigma)$ is the r-th Hermite coefficient of the ReLU function given by (8).

- Most papers and our main reference assume $\|x\| = 1$ and $|y| \leq 1$, for simplicity.
- Therefore, the diagonal $H^{\infty}_{ii} = \frac{1}{1}$.
- Now denote $\rho = \max_{i,j\neq i} |x_i^\top x_j|$. since $f(x) = \frac{x(\pi - \arccos(x))}{2\pi}$, is as below, we get:

$$H^{\infty}_{i,j\neq i} \leq \frac{\rho(\pi - \arccos(\rho))}{2\pi} \leq \frac{1}{2}$$

- We can use the **Gershgorin circle** theorem and get

$$|\lambda_{\min}(H^{\infty}) - \frac{1}{2}| \leq (n-1)\frac{\rho(\pi - \arccos(\rho))}{2\pi}$$

- Thus, the bound depends on the maximum "correlation" between two distinct i.i.d x_i in a sample of size n. This will depend on the distribution of x on S^{d-1}.

- So the bound is good when x_i are almost orthogonal with high probability. We can show the following examples:
- $x_i \sim \text{Unif}(S^{d-1})$:

$$\max_{1 \le i < j \le n} \left| \langle X_i, X_j \rangle \right| \le C \sqrt{\frac{\log\left(\frac{n}{\delta}\right)}{d}}.$$

- x_i are isotropic, mean-zero, sub-Gaussian vectors:

$$\max_{1 \le i < j \le n} |\langle X_i, X_j \rangle| \le C \sqrt{\frac{\log\left(\frac{n}{\delta}\right)}{d}} \max_{1 \le i \le n} \|X_i\|.$$

Bernstein Result

[1] L. Weng, "Some math behind neural tangent kernel," *Lil'Log*, Sep 2022.

[2] A. Jacot, F. Gabriel, and C. Hongler, "Neural tangent kernel: Convergence and generalization in neural networks," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.

[3] J. Lee, L. Xiao, S. S. Schoenholz, Y. Bahri, R. Novak, J. Sohl-Dickstein, and J. Pennington, "On exact computation with an infinitely wide neural net," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.

[4] Y. Cao and Q. Gu, "Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks," *Proceedings of the 36th International Conference on Machine Learning (ICML)*, vol. 97, pp. 1047–1055, 2019.

[5] N. Rahaman, D. Arpit, F. Draxler, M. Lin, F. A. Hamprecht, Y. Bengio, and A. Courville, "Spectral bias outside the training set for deep networks in the kernel regime," *International Conference on Learning Representations (ICLR)*, 2019.

[6] G. Meanti, L. Rosasco, A. Rudi, and L. Carratino, “Scaling neural tangent kernels via sketching and random features,” *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 14536–14546, 2020.

[7] S. Arora, S. S. Du, W. Hu, Z. Li, and R. Wang, “Overparametrized multi-layer neural networks: Uniform concentration of neural tangent kernel and convergence of stochastic gradient descent,” *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.