Stat 946 - Topics in Probability and Statistics: Mathematical Foundations of Deep Learning *Lecture 5*

Instructor: *Mufan Li*, Scribe: *Sidharth Bajaj*

September 13, 2025

1 NTK for network with 1 Hidden Layer of width n

Recap: The network function $f(\cdot\, ;\, \theta)$ *is parameterized by the weights* $\theta = \{\underbrace{W_0}_{n\times n_0}, \underbrace{W_1}_{1\times n}\}$ *where* $W_{l,ij}$ *are i.i.d. realizations of* $N(0,1)$.

The network's output f^α for input $\underbrace{x^\alpha}_{n_0\times 1}$ is defined as follows:

$$f^\alpha = (1/\sqrt{n}) \cdot W_1 \cdot \psi(h_1^\alpha)$$

where $\underbrace{h_1^\alpha}_{n\times 1} = W_0 \cdot x^\alpha$ refers to the corresponding hidden layer for input x^α and ψ is an activation function applied to each component in the column input individually.

The network function f is trained on a data-set, $\mathcal{D}$, of size m:

$$\mathcal{D} = \{(x^\alpha, y^\alpha)\}_{\alpha=1}^m$$

and the loss observed by the network f as a function of the parameters θ is specified below:

$$\mathcal{L}(\theta) = (1/(2\cdot m)) \cdot \Sigma_{\alpha=1}^m (f^\alpha - y^\alpha)^2$$

An application of chain rule allows us to model the gradient flow (i.e. $\underbrace{\partial_t}_{\partial/\partial_t = d/d_t} \theta(t)$) using the following differential equation:

$$\begin{aligned} \partial_t \theta(t) &= -\nabla_\theta \mathcal{L}(\theta(t)) \\ &= -(1/m) \cdot \Sigma_{\alpha=1}^m (f^\alpha - y^\alpha) \cdot \nabla_\theta f^\alpha \end{aligned} \tag{1}$$

We can write out the change in the residual, $\partial_t(f^\alpha - y^\alpha)$, for input x^α as follows:

$$\begin{aligned}\partial_t(f^\alpha - y^\alpha) &= \partial_t f^\alpha \\ &= \langle \nabla_\theta f^\alpha, \partial_t \theta(t)\rangle \\ &= -(1/m) \cdot \Sigma_{\beta=1}^{m} \underbrace{\langle \nabla_\theta f^\alpha, \nabla_\theta f^\beta \rangle}_{\mathcal{K}_t^n(x^\alpha, x^\beta)} \cdot (f^\beta - y^\beta)\end{aligned} \tag{2}$$

The second equality is another application of the chain-rule. The last equality is obtained by substituting ${}_theta_t$ into $\partial_t\theta(t)$. The random tangent Kernel, $\mathcal{K}_t^n(\cdot\,;\,\cdot)$, which is implicitly parameterized by θ is also referred to as the NTK. Note that $\partial_t(f^\alpha - y^\alpha)$ can be viewed as a mixture of the rest of the residuals (i.e. $\{f^\beta - y^\beta\}_{\beta=1,\cdots,m}$).

Jacot et al. (2020) prove the following:

Theorem 1. *In the infinite-width limit (i.e n $\to \infty$), by the law of large numbers (LLN), the random tangent kernel, K_t^n, tends to a deterministic kernel, $\mathcal{K} = [\mathcal{K}(x^\alpha, x^\beta)]_{\alpha,\beta=1}^m$, which stays constant during the entire training process.*

Now we can apply Theorem 1 and rewrite the change in the residuals, Eq. (2), using the following linear DE:

$$\partial_t\underbrace{(f - y)}_{m\times 1} = -(1/m) \cdot \underbrace{\mathcal{K}}_{m\times m} \cdot (f - y) \tag{3}$$

We can view $(1/m)\cdot\mathcal{K}$ as the *pre-conditioner* on the *co-ordinate* $(f - y)$.

The differential equation expressed as Eq. (3) has the following solution:

$$(f - y) = \exp\{-(1/m)\cdot\mathcal{K}\cdot t\} \cdot (f(\theta(0)) - y) \tag{4}$$

where the exponential operation on a matrix A refers to:

$$\exp\{A\} = \mathcal{I} + A + A^2/2! + A^3/3! + \cdots$$

Let's generalize the loss function on our training set of size m:

$$\mathcal{L}^*(\theta) = (1/m) \cdot \Sigma_{\alpha=1}^m l(f^\alpha, x^\alpha)$$

Now we can model the evolution of the generalized loss function using the following differential equation:

$$\begin{aligned}\partial_t\mathcal{L}^*(\theta(t)) &= (1/m) \cdot \Sigma_{\alpha=1}^m \partial_{f^\alpha} l(f^\alpha, y^\alpha) \cdot \partial_t f^\alpha \\ &= (-1/m^2) \cdot \Sigma_{\alpha=1}^m \partial_{f^\alpha} l(f^\alpha, y^\alpha) \cdot \Sigma_{\beta=1}^m \mathcal{K}(x^\alpha, y^\beta) \cdot \partial_{f^\beta} l(f^\beta, y^\beta) \\ &= (-1/m^2) \cdot (\partial_x l(f, y))^T \cdot \mathcal{K} \cdot \partial_x l(f, y)\end{aligned} \tag{5}$$

where $\partial_x l(f, y)$ is an $m \times 1$ column-vector whose i^{th} entry refers to $\partial_{f^i} l(f^i, y^i)$ and the second equality is an exercise for the reader.

Let's revisit our original loss function $\mathcal{L}(\theta) = (1/(2 \cdot m)) \cdot ||f - y||^2$. We can apply similar logic as above to arrive at the following expression:

$$\partial_t \mathcal{L}(\theta(t)) = -(1/m^2) \cdot (f - y)^T \cdot \mathcal{K} \cdot (f - y)$$

Now we state the following lemma:

Lemma 2. *Let $M \in \mathbb{R}^{m \times m}$ be a symmetric and PSD matrix. Let $\lambda^*(M)$ refer to the minimum eigenvalue of matrix M. For any $\mu \in \mathbb{R}^m$, $\lambda^*(M) \cdot ||u||^2 \leq u^T \cdot M \cdot u$.*

Proof.

$$\begin{aligned} u^t \cdot M \cdot u &= u^T \cdot P \cdot D \cdot P^T \cdot u \\ &\geq \lambda^*(M) \cdot u^T \cdot P \cdot P^T \cdot u \\ &= \lambda^*(M) \cdot ||u||^2 \end{aligned}$$

□

The first inequality makes use of the non-negativity of the eigenvalues of the matrix and the last equality makes use of the orthogonal diagonalizability of symmetric matrices.

Now we can apply Lemma 2 with respect to our PSD kernel matrix $\mathcal{K}$ to derive the following differential inequality:

$$\begin{aligned} \partial_t \mathcal{L}(\theta(t)) &\leq -(\lambda^*(\mathcal{K}))/m^2) \cdot ||f - y||^2 \\ &= -(\lambda^*(\mathcal{K}))/m^2) \cdot 2 \cdot m \cdot \mathcal{L}(\theta(t)) \\ &= -2 \cdot (\lambda^*(\mathcal{K})/m) \cdot \mathcal{L}(\theta(t)) \\ &\leq -(1/2) \cdot (\lambda^*(\mathcal{K})/m) \cdot \mathcal{L}(\theta(t)) \end{aligned}$$

We apply *Grönwall's Inequality*, to get the following solution to the above differential inequality:

$$\mathcal{L}(\theta(t)) \leq \exp\{-\frac{\lambda^*(\mathcal{K}) \cdot t}{2 \cdot m}\} \cdot \mathcal{L}(\theta(0)) \tag{6}$$

A consequence of Eq. (6) *is that if $\lambda^*(\mathcal{K}) > 0$, then NN training should converge!* Note that we can show that $\lambda^*(\mathcal{K}) > 0$ using RMT.

Exercises

Show the following chain of equalities:

1.

$$\mathcal{K}_t^n(x^\alpha, x^\beta) = (1/n) \cdot \underbrace{\langle \psi(h_1^\alpha), \psi(h_1^\beta) \rangle}_{\text{GP Kernel } \psi(x^\alpha, x^\beta)} + (1/n) \cdot \langle \text{diag}(\underbrace{\psi^{'}(h_1^\alpha)}_{n \times n}) \cdot (\underbrace{x_\alpha \cdot W_1}_{n \times n_0})^T, \text{diag}(\psi^{'}(h_1^\beta)) \cdot (x_\beta \cdot W_1)^T \rangle$$

2.

$$\partial_t(\mathcal{K}_t^n(x^\alpha, x^\beta)) = -(1/m) \cdot \Sigma_{\gamma=1}^m (f^\alpha - y^\alpha) \cdot [(\nabla_\theta(f^\beta))^T \cdot \nabla_\theta^2(f^\alpha) \cdot \nabla_\theta(f^\gamma) + (\nabla_\theta(f^\alpha))^T \cdot \nabla_\theta^2(f^\beta) \cdot \nabla_\theta(f^\gamma)]$$

3.

$$\begin{aligned}(\nabla_\theta(f^\beta))^T \cdot \nabla_\theta^2(f^\alpha) \cdot \nabla_\theta(f^\gamma) &= (\langle x^\alpha, x^\beta \rangle / \sqrt{n}) \cdot (1/n) \cdot \Sigma_{i=1}^n W_{1,i} \cdot \psi(h_{1,i}^\beta) \cdot \psi^{'}(h_{1,i}^\alpha) \cdot \psi^{'}(h_{1,i}^\gamma) \\ &+ (\langle x^\alpha, x^\beta \rangle / \sqrt{n}) \cdot (1/n) \cdot \Sigma_{i=1}^n W_{1,i} \cdot \psi(h_{1,i}^\gamma) \cdot \psi^{'}(h_{1,i}^\alpha) \cdot \psi^{'}(h_{1,i}^\beta) \\ &+ (\langle x^\alpha, x^\beta \rangle \cdot \langle x^\alpha, x^\beta \rangle / \sqrt{n}) \cdot \underbrace{(1/n) \cdot \Sigma_{i=1}^n W_{1,i}^3 \cdot \psi^{'}(h_i^\beta) \cdot \psi^{''}(h_i^\alpha) \cdot \psi^{'}(h_i^\gamma)}_{O(1)}\end{aligned}$$

A consequence of the above equalities is that $\partial_t(\mathcal{K}_t^n(x^\alpha, x^\beta)) = O(1/\sqrt{n}) \to 0$. Hence, the limiting kernel $\mathcal{K}_t$ is also stationary (i.e. $\mathcal{K}_t = \mathcal{K}$) in the infinite-width limit.

2 References

Jacot, A., Gabriel, F., and Hongler, C. (2020). Neural tangent kernel: Convergence and generalization in neural networks.