Review of NTK paper by A. Jacot (2018)

Andy Ko

September 2020

1 Notations and Preliminaries

Artificial Neural Network (ANN) with L layers has layer 0 (corresponding to input) to layer L (corresponding to output). For each layer, there are connection matrices and biases, denoted $W^{(l)} \in \mathbb{R}^{n_{l+1} \times n_l}, b^{(l)} \in \mathbb{R}^{n_{l+1}}$ for $l = 0, \dots, L-1$, which we call the parameters of network. All parameters are initialized as i.i.d. $\mathcal{N}(0,1)$. There are a total of $P = \sum_{i=0}^{L-1}(n_i + 1)n_{i+1}$ parameters as a result of counting number of elements in connection matrices and biases.
We can now define the realization function $F^L : \mathbb{R}^p \longrightarrow \mathcal{F}$ where $\mathcal{F}$ is the space of functions $\{f : \mathbb{R}^{n_0} \longrightarrow \mathbb{R}^{n_L}\}$:

$$F^{(L)} : \theta \longmapsto f_\theta$$
$$\theta \in \mathbb{R}^P, f_\theta : \mathbb{R}^{n_0} \longrightarrow \mathbb{R}^{n_L}$$

On $\mathcal{F}$, we impose inner product $\langle f, g \rangle_{p^{in}} \coloneqq \frac{1}{N}\sum_{i=1}^N f(x_i)^T g(x_i)$ which induces the seminorm $\| \cdot \|_{p^{in}} \coloneqq \sqrt{\langle f, f \rangle_{p^{in}}}$. This is a seminorm (and not a norm) because $\|f\|_{p^{in}} = 0$ only implies f is 0 at the given N data points and not that the function itself is a zero function. We consider deterministic finite dataset $x_1, \dots, x_N$ each of which is an element of $\mathbb{R}^{n_0}$. On the space of $\mathbb{R}^{n_0}$, we impose the empirical distribution $\frac{1}{N}\sum_{i=1}^N \delta_{x_i}$ where δ_x denotes the Dirac measure.
Neural network function f_θ is defined in the classic way:

$$f_\theta(x) \coloneqq \tilde{\alpha}^{(L)}(x; \theta)$$

where $\tilde{\alpha}^{(l)}(\cdot; \theta) : \mathbb{R}^{n_0} \longrightarrow \mathbb{R}^{n_l}$ and $\alpha^{(l)}(\cdot; \theta) : \mathbb{R}^{n_0} \longrightarrow \mathbb{R}^{n_l}$ are defined as:

$$\begin{aligned}
&\alpha^{(0)}(x, \theta) = x \\
&\tilde{\alpha}^{(l+1)}(x; \theta) = \frac{1}{\sqrt{n_l}} W^{(l)} \alpha^{(l)}(x; \theta) + \beta b^{(l)} \\
&\alpha^{(l)} x; \theta) = \sigma(\tilde{\alpha}^{(l)}(x; \theta)), \\
&l = 1, \dots, L-1
\end{aligned}$$

where σ is applied elementwise.

2 Kernel Gradient

Training a neural network involves optimizing over $f_\theta \in \mathcal{F}$ w.r.t. functional cost $C \coloneqq \mathcal{F} \longrightarrow \mathbb{R}$ such as regression or cross-entropy cost. That is if we plug in a deterministic function f to C, we get the cost associated with it. This is analogous to risk in decision theory except now there's no randomness in data.
While C (as a function of functions) might be convex, as in the case of mean squared error, $C \circ F^{(L)}$ (as a function of parameters) is usually not.
This paper shows gradient descent is equivalent to kernel gradient descent w.r.t. Neural Tangent Kernel

(NTK). We will soon define what we mean by kernel gradient descent with respect to a given kernel. First, we define a multidimensional kernel as $K : \mathbb{R}^{n_0} \times \mathbb{R}^{n_0} \longrightarrow \mathbb{R}^{n_L \times n_L}$ such that $K(x, x') = K(x', x)^T$ for all $(x, x') \in \mathbb{R}^{n_0} \times \mathbb{R}^{n_0}$.
Then, we can also define a bilinear map on $\mathcal{F}$:

$$\langle f, g\rangle_K = E_{x,x'\ p^{in}}[f(x)^T K(x, x')g(x')]$$

K is called positive definite w.r.t. $\| \cdot \|_{p^{in}}$ if $\|f\|_{p^{in}} > 0$ implies $\|f\|_K > 0$.
Now we can define the dual space of $\mathcal{F}$ with respect to p^{in} as $\mathcal{F}^* := \{\langle d, \cdot\rangle_{p^{in}} : d \in \mathcal{F}\}$. Note that by continuity of inner product, we have that every element of $\mathcal{F}^*$ is a continuous linear functional on $\mathcal{F}$. Then, we can appeal to Riesz representation theorem to deduce some properties. We state the theorem below for convenience.

Theorem $-$ Let H be a Hilbert space whose inner product is antlinear in its first argument and let $\varphi \in H^*$ be a continuous linear functional. Then there exists (a unique) $f_\varphi \in H$ such that for any $x \in H$, $\varphi(x) = \langle f_\varphi, x\rangle$. Moreover $\|f_\varphi\|_H = \|\varphi\|_{H^*}$

To apply the theorem, we must first check that $\mathcal{F}$ of neural networks is indeed a Hilbert space. That is, the functions form a complete metric space with respect to the inner product $\langle \cdot, \cdot\rangle_{p^{in}}$. Clearly, this is not the case since the inner product only involves the N data points so that the norm of a function is 0 as long as the function is 0 at those N points. To resolve this issue, we treat two functions to be equivalent if they agree on all N points, i.e. f we can easily show any Cauchy sequence of functions in $\mathcal{F}$ converge to a function in $\mathcal{F}$ (Each component of the limit function is simply the limit of corresponding component of function sequence). More precisely, given a Cauchy sequence of functions $f_1, f_2, \dots \in \mathcal{F}$, for any given $\varepsilon > 0$, there exists M such that for all $n, m \geq M$, $\|f_n - f_m\|^2_{p^{in}} < \varepsilon$. That is, $\frac{1}{N}\sum_{i=1}^N (f_n(x_i) - f_m(x_i))^2 < \varepsilon$. Hence, we get that $(f_n(x_i) - f_m(x_i))^2 < \varepsilon$ for all $i = 1, \dots, N$ and $n, m \geq M$. Thus for every x_i, $(f_n(x_i))_{n\in\mathbb{N}}$ is a Cauchy sequence of real numbers that converges to a real number. Since we can always find a neural network that outputs given N real numbers for each input x_i, this shows $\mathcal{F}$ is a Hilbert space.
Then, theorem above shows $\mathcal{F}^*$ includes all continuous (equivalently, bounded) linear functionals on $\mathcal{F}$ since any continuous linear functional can be written in the form $\langle d, \cdot\rangle_{p^{in}}$. Moreover, by the last statement of theorem above, the dual norm for $\varphi \in \mathcal{F}^*$ given by $\|\varphi\|_{\mathcal{F}^*} := \sup\{|\varphi(f)| : f \in \mathcal{F}, \|f\|_{p^{in}} \leq 1\}$ has the property that

$$\begin{aligned}\|\varphi\|_{\mathcal{F}^*} &= \|\langle d, \cdot\rangle_{p^{in}}\|_{\mathcal{F}^*} \\ &= \|d\|_{\mathcal{F}} \\ &= \|d\|_{p^{in}}\end{aligned}$$

where d is the unique element in $\mathcal{F}$ depending only on φ.
Now note $K_{i,\cdot}(x, \cdot) \in \mathcal{F}$ since it is a mapping from $\mathbb{R}^{n_0}$ to $\mathbb{R}^{n_L}$, and define the mapping $\Phi_K : \mathcal{F}^* \longrightarrow \mathcal{F}$ as

$$\Phi_K(\langle d, \cdot\rangle_{p^{in}})(x) := \langle d, K_{i,\cdot}(x, \cdot)\rangle_{p^{in}},\ x \in \mathbb{R}^{n_0}$$

Here, observe $\langle d, \cdot\rangle_{p^{in}} \in \mathcal{F}^*$ represents an arbitrary element in $\mathcal{F}^*$
In doing gradient descent, cost functional C only depends on the values of $f \in \mathcal{F}$ at the N data points. Also the (functional) total derivative of C at a point (which is a function, say f_0), denoted $\partial^{in}_f C|_{f_0}$, is nothing but another functional that is linear:

$$\partial^{in}_f C|_{f_0} : \mathcal{F} \longrightarrow \mathbb{R}$$

Assuming C is continuously differentiable, $\partial^{in}_f C|_{f_0}$ is then a continuous linear functional. Thus, there exists $d|_{f_0} \in \mathcal{F}$ such that $\partial_f C|_{f_0} = \langle d|_{f_0}(x_i), \cdot\rangle_{p^{in}}$.
With this setup, kernel gradient is defined as $\nabla_K C|_{f_0} := \Phi_K(\partial^{in}_f C|_{f_0}) \in \mathcal{F}$ so that

$$\nabla_K C|_{f_0}(x) = \frac{1}{N}\sum_{i=1}^N K(x, x_i)d|_{f_0}(x_i)$$

We say time-dependent function f follows kernel gradient descent w.r.t. K if

$$\partial_t f(t) = -\nabla_K C|_{f(t)}$$

Note $f(t)$ itself is a function: in our setting, it is a neural network trained up to time step t. If this is the case, we can write

$$\begin{aligned}
\partial_t C|_{f(t)} &= \partial_f^{in} C|_{f(t)} \partial_t f(t) \\
&= \partial_f^{in} C|_{f(t)} (-\nabla_K C|_{f(t)}) \\
&= -\langle d|_{f(t)}, \nabla_K C|_{f(t)} \rangle_{p^{in}} \\
&= -\frac{1}{N} \sum_{i=1}^{N} d|_{f(t)}(x_i)^T \left(\frac{1}{N} \sum_{j=1}^{N} K(x_i, x_j) d|_{f(t)}(x_j) \right) \\
&= -\frac{1}{N} \sum_{i=1}^{N} \frac{1}{N} \sum_{j=1}^{N} d|_{f(t)}(x_i)^T K(x_i, x_j) d|_{f(t)}(x_j) \\
&= -\|d|_{f(t)}\|_K^2
\end{aligned}$$

Therefore, once we show f follows kernel gradient descent w.r.t. K, and if we're given that K is positive definite with respect to $\| \cdot \|_{p^{in}}$, convergence of C to a critical point is guaranteed because then the cost is strictly decreasing except at points where $\|d_{f(t)}\|_{p^{in}}$=0. In particular, if C is convex and bounded from below, we obtain convergence to global minimum.

3 Example with random functions approximation

As an attempt to understand the training dynamics of ANN, we first consider a simpler function class. Suppose that $\mathcal{F}$ is equipped with an arbitrary probability measure and that we have P random, independent samples from the distribution, $f^{(1)}, \dots, f^{(P)}$. Define the mapping $F^{lin} : \mathbb{R}^P \longrightarrow \mathcal{F}$ as:

$$\theta \mapsto f_\theta^{lin} = \frac{1}{\sqrt{P}} \sum_{p=1}^{P} \theta_p f^{(p)}$$

so that

$$\partial_{\theta_p} F^{lin}(\theta) = \frac{1}{P} f^{(p)}$$

Optimizing cost $C \circ F^{lin}$ through gradient descent (in parameter space), the parameters follow the ODE:

$$\partial_t \theta_p(t) = -\partial_{\theta_p} \left(C \circ F^{lin} \right) (\theta(t)) = -\frac{1}{\sqrt{P}} \partial_f^{in} C|_{f_{\theta(t)}^{lin}} f^{(p)} = -\frac{1}{\sqrt{P}} \left\langle d|_{f_{\theta(t)}^{lin}} f^{(p)} \right\rangle_{p^{in}}$$

Using this,

$$
\begin{aligned}
\partial_t f^{lin}_{\theta(t)} &= \frac{1}{\sqrt{P}} \sum_{p=1}^{P} \partial_t \theta_p(t) f^{(p)} \\
&= -\frac{1}{P} \sum_{p=1}^{P} \left\langle d|_{f^{lin}_{\theta(t)}}, f^{(p)} \right\rangle f^{(p)} \\
&= -\frac{1}{P} \sum_{p=1}^{P} \frac{1}{N} \sum_{j=1}^{N} d|_{f^{lin}_{\theta(t)}}(x_j)^T f^{(p)}(x_j) f^{(p)} \\
&= -\frac{1}{N} \sum_{j=1}^{N} \frac{1}{P} \sum_{p=1}^{P} f^{(p)} f^{(p)}(x_j)^T d|_{f^{lin}_{\theta(t)}}(x_j) \\
&= -\frac{1}{N} \tilde{K}(\cdot, x_j) d|_{f^{lin}_{\theta(t)}}(x_j) \\
&= -\nabla_{\tilde{K}} C|_{f^{lin}_{\theta(t)}}
\end{aligned}
$$

where $\tilde{K} := \frac{1}{P} \sum_{p=1}^{P} f^{(p)} \otimes f^{(p)}$ is a n_L-dimensional kernel with values $\tilde{K}_{ii'}(x, x') = \frac{1}{P} \sum_{p=1}^{P} f_i^{(p)}(x) f_{i'}^{(p)}(x')$. We have thus shown that $f^{lin}_{\theta(t)}$ follows kernel gradient descent with tangent kernel $\tilde{K}$. Note that kernel $\tilde{K}$ is random since each $f^{(p)}$ were sampled randomly.
Let's now consider the asymptotic regime where $P \to \infty$. By law of large numbers, $\tilde{K}$ should tend to the limit $E_{\mathcal{F}}[f^{(p)} \otimes f^{(p)}]$. Therefore, this method can be interpreted as an approximation of kernel gradient descent with respect to limiting kernel K.

4 Neural Tangent Kernel

Now we finally consider gradient descent on $C \circ F^{(L)}$ where $F^{(L)}$ maps parameters to neural networks (cf. Section 1). Notation-wise, we write $\theta(t)$ to denote a P-dimensional vector that contains all the parameters of a neural network including weights and biases and $\theta_p \in \mathbb{R}$ to denote a single parameter in a network. Similarly as in previous section, training via gradient descent follows the following dynamics:

$$
\partial_t \theta_p(t) = -\partial_{\theta_p} \left(C \circ F^{(L)} \right)(\theta(t)) = -\partial_f C|_{f_{\theta(t)}} \circ \partial_{\theta_p} F^{(L)}(\theta(t)) = -\left\langle d|_{f_{\theta(t)}}, \partial_{\theta_p} F^{(L)}(\theta(t)) \right\rangle_{p^{in}} \tag{1}
$$

Recall from our previous discussions that $\partial_f C|_{f_{\theta(t)}}$ is a continuous linear functional and $\partial_{\theta_p} F^{(L)}(\theta(t)) \in \mathcal{F}$. Then it follows

$$
\begin{aligned}
\partial_t f_{\theta(t)} &= \partial_t F^{(L)}(\theta(t)) \\
&= \sum_{p=1}^{P} \partial_{\theta_p} F^{(L)}(\theta(t)) \partial_t \theta_p(t) \\
&= -\sum_{p=1}^{P} \frac{1}{N} \sum_{j=1}^{N} \partial_{\theta_p} F^{(L)}(\theta(t)) \partial_{\theta_p} F^{(L)}(\theta(t))(x_j)^T d|_{f_{\theta(t)}}(x_j) \text{ by (1)} \\
&= -\frac{1}{N} \sum_{j=1}^{N} \sum_{p=1}^{P} \partial_{\theta_p} F^{(L)}(\theta(t)) \partial_{\theta_p} F^{(L)}(\theta(t))(x_j)^T d|_{f_{\theta(t)}}(x_j) \\
&= -\nabla_{\Theta^{(L)}(\theta(t))} C|_{f_{\theta(t)}}
\end{aligned}
$$

where $\Theta^L(\theta(t)) := \sum_{p=1}^{P} \partial_{\theta_p} F^{(L)}(\theta(t)) \otimes \partial_{\theta_p} F^{(L)}(\theta(t))$ is called the neural tangent kernel (NTK). In contrast to F^{lin} in Section 3, $F^{(L)}$ is not linear in $\theta(t)$. Hence $\partial_{\theta_p} F^{(L)}(\theta(t))$ and the NTK depend on $\theta(t)$ and since

we randomly initialize all parameters as i.i.d. $\mathcal{N}(0,1)$, NTK varies during training.
This paper's main results show that the NTK becomes deterministic at initialization, say $\Theta_\infty^{(L)}$, and remains constant during training in the regime of infinite-width limit. If this is shown, one great advantage we gain is that since the asymptotic behavior of f_θ during training is understood by $-\nabla_{\Theta_\infty^{(L)}} C|_{f_{\theta(t)}}$ and the kernel is independent of parameters, we just need to study the behavior of cost C in terms of function space $\mathcal{F}$. What this implies is that by the argument at the end of section 2, we can obtain convergence guarantees for training of neural network under some mild assumptions if we show that the limiting NTK is positive definite.

5 Main Results

We now aim to show that in the infinite-width limit, NTK becomes deterministic at initialization and stays constant during training.

Proposition 5.1. For a network of depth L with a Lipschitz nonlinearity σ, and in the limit as $n_1, \ldots, n_{L-1} \to \infty$, the output functions $f_{\theta,k}$ for $k = 1, \ldots, n_L$, tend (in law) to iid centered Gaussian processes of covariance $\Sigma^{(L)}$, where $\Sigma^{(L)}$ is defined recursively by:

$$\Sigma^{(1)}(x,x') = \frac{1}{n_0} x^T x' + \beta^2$$
$$\Sigma^{(L+1)}(x,x') = \mathbb{E}_{f\sim\mathcal{N}\left(0,\Sigma^{(L)}\right)}\left[\sigma(f(x))\sigma\left(f\left(x'\right)\right)\right] + \beta^2$$

taking the expectation with respect to a centered Gaussian process f of covariance $\Sigma^{(L)}$.

First theorem shows that the NTK becomes deterministic at initialization in the infinite-width limit.

Theorem 5.1. For a network of depth L at initialization with a Lipschitz nonlinearity σ, and in the limit as the layers width $n_1, \ldots, n_{L-1} \to \infty$, the NTK $\Theta^{(L)}$ converges in probability to a deterministic limiting kernel:

$$\Theta^{(L)} \to \Theta_\infty^{(L)} \otimes Id_{n_L}$$

The scalar kernel $\Theta_\infty^{(L)} : \mathbb{R}^{n_0} \times \mathbb{R}^{n_0} \to \mathbb{R}$ is defined recursively by

$$\Theta_\infty^{(1)}(x,x') = \Sigma^{(1)}(x,x')$$
$$\Theta_\infty^{(L+1)}(x,x') = \Theta_\infty^{(L)}(x,x')\,\dot{\Sigma}^{(L+1)}(x,x') + \Sigma^{(L+1)}(x,x'),$$

where

$$\dot{\Sigma}^{(L+1)}(x,x') = \mathbb{E}_{f\sim\mathcal{N}\left(0,\Sigma^{(L)}\right)}\left[\dot{\sigma}(f(x))\dot{\sigma}\left(f\left(x'\right)\right)\right]$$

taking the expectation with respect to a centered Gaussian process f of covariance $\Sigma^{(L)}$, and where $\dot{\sigma}$ denotes the derivative of σ.

Second theorem shows that this limiting kernel at initialization is indeed constant during training.

Theorem 5.2. Assume that σ is a Lipschitz, twice differentiable nonlinearity function, with bounded second derivative. For any T such that the integral $\int_0^T \|d_t\|_{p^{in}}\,dt$ stays stochastically bounded, as $n_1, \ldots, n_{L-1} \to \infty$, we have, uniformly for $t \in [0,T]$

$$\Theta^{(L)}(t) \to \Theta_\infty^{(L)} \otimes Id_{n_L}$$

As a consequence, in this limit, the dynamics of f_θ is described by the differential equation

$$\partial_t f_{\theta(t)} = \Phi_{\Theta_\infty^{(L)}\otimes Id_{n_L}}\left(\langle d_t, \cdot\rangle_{p^{in}}\right)$$

Now recall from Section 3 that when the kernel is positive definite with respect to $\|\cdot\|_{p^{in}}$, then convergence of kernel gradient descent to a critical point of C is guaranteed. We can further observe that the limiting NTK is positive definite if the span of the derivatives $\partial_{\theta_p} F^{(L)}$ becomes dense in $\mathcal{F}$. Below is a proposition that shows the limiting NTK is indeed positive definite when data is restricted to unit sphere and nonlinearity σ is non-polynomial.

Proposition 5.2. For a non-polynomial Lipschitz nonlinearity σ, for any input dimension n_0, the restriction of the limiting NTK $\Theta_{\infty}^{(L)}$ to the unit sphere $\mathbb{S}^{n_0-1} = \left\{x \in \mathbb{R}^{n_0} : x^T x = 1\right\}$ is positive definite if $L \geq 2$