EAI Math Reading Group

Neural Tangent Kernels

07/05/2023

Outline

1. Kernel methods 101
2. Neural Tangent Kernel
3. Theoretical results

Kernel Functions

A (positive definite) *kernel function* on X is a symmetric function $k : X \times X \to \mathbb{R}$ such that

$$\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geq 0$$

for any $x_1, \ldots, x_n \in X$, $c_1, \ldots, c_n \in \mathbb{R}$

i.e. the matrix $[k(x_i, x_j)]_{ij}$ is positive definite

Reproducing Kernel Hilbert Spaces

Every p.d. kernel K induces a (unique) *Reproducing Kernel Hilbert Space* (RKHS) $(H, \langle . , . \rangle_H)$ of functions $X \to \mathbb{R}$ with

1. $K_x = K(x, .) \in H, \forall x \in X$
2. $\langle f, K_x \rangle_H = f(x), \forall f \in H, \forall x \in X$

Intuition: Kernel methods are essentially equivalent to projecting X to a Hilbert space, via some *feature map* $\Phi : X \to H$

$$K(x, y) = \langle \Phi(x), \Phi(y) \rangle_H$$

Gaussian Processes

A *Gaussian Process* is a stochastic process $X : \mathbb{X} \to \mathbb{R}^d$, such that for any $x_1, \ldots, x_n \in \mathbb{X}$,

$$[X(x_1), \ldots, X(x_n)] \sim \mathcal{N}(\mu_{x_1,\ldots,x_n}, \Sigma_{x_1,\ldots,x_n})$$

Intuition: "Gaussian distribution" over *functions*

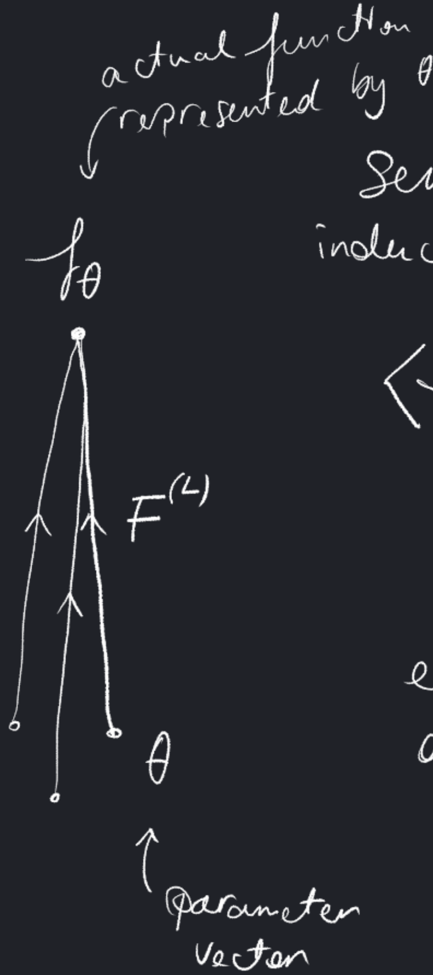
$$X \sim \mathcal{GP}(\mu, \Sigma)$$

NB. $\Sigma : \mathbb{X} \times \mathbb{X}$ is a p.d. kernel!

Realization function

Function Space

$\mathcal{F} \subset \{f : \mathbb{R}^{n_0} \to \mathbb{R}^{n_L}\}$



Seminorm on $\mathcal{F}$ $\|\cdot\|_{p_{in}}$ induced by

$$\langle f, g \rangle_{p_{in}} = \mathbb{E}_{x \sim p_{in}} \left[f(x)^T g(x) \right]$$

input distribution

empirical distribution $\frac{1}{N} \sum_{i=1}^{N} \delta_{x_i}$

datapoints

Parameter Space

$\Theta \subset \mathbb{R}^P$

NB. $\mathcal{F}$ only contains the functions $\{ f_\theta \mid \theta \in \Theta \}$

Neural Network

$$f_\theta(x) = (\varphi^{(L)} \circ \cdots \circ \varphi^{(0)})(x)$$

- $\varphi^{(\ell)}(x_\ell) = \sigma.\,(\tilde{\varphi}^{(\ell)})(x_\ell)$
- $\tilde{\varphi}^{(\ell)}(x_\ell) = \frac{1}{\sqrt{n_\ell}} W^{(\ell)} x_\ell + \beta b^{(\ell)}$
- $\theta = (W^{(0)}, b^{(0), \ldots, W^{(L)}, b^{(L)}})$
- $W_{ij}^{(\ell)}, b_i^{(\ell)} \sim \mathcal{N}(0,1)$

Tangent 1: Neural Network Gaussian Processes

The induced distribution in $\mathcal{F}$ is a *Gaussian Process*

$$f_\theta \sim \mathcal{GP}(0, K^L)$$

where K^L converges to a deterministic limit as $n_0, \ldots, n_L \to \infty$

Intuition: The pre-activations of each layer is a sum of Gaussian random variables (the parameters) weighted by the inputs.

Training

Training a model involves minimizing some cost $\mathcal{C} : \mathcal{F} \to \mathbb{R}$

Problem:
even if $\mathcal{C}$ is convex in $\mathcal{F}$,
the parametrized function $\mathcal{C} \circ F^L$ might not be

Multi-dimensional kernel

symmetric function $K : \mathbb{R}^{n_0} \times \mathbb{R}^{n_0} \to \mathbb{R}^{n_L \times n_L}$.

Induced bilinear map on $\mathcal{F}$

$$\langle f, g \rangle_K = \mathbb{E}_{x,y \sim p_{in}}[f(x)^T K(x,y) g(y)]$$

K is p.d. with respect to $\|.\|_{p_{in}}$ if

$$\|f\|_{p_{in}} > 0 \Rightarrow \|f\|_K = \sqrt{\langle f, f \rangle_K} > 0$$

Dual space

$\mathcal{F}^* = \mathcal{L}(\mathcal{F}, \mathbb{R})$
$\mu \in \mathcal{F}^* \Rightarrow \exists d \in \mathcal{F}$ s.t.

$$\mu(f) = \langle d, f \rangle_{p_{in}}$$

NB. due to p_{in}, it is *finite* dimensional

Mapping from $\mathcal{F}^*$ to $\mathcal{F}$

For $i \in 1, \ldots, n_L$, $x \in \mathbb{R}^{n_0}$, $K_{i,.}(x, .\,) \in \mathcal{F}$

Define $\Phi_K : \mathcal{F}^* \to \mathcal{F}; \mu \mapsto f_\mu$

$$f_{\mu,i}(x) = \mu(K_{i,.}(x, .\,)) = \langle d, K_{i,.}(x, .\,)\rangle_{p_{in}}$$

Tangent 2: Functional Derivatives

Let $f : X \to Y$ be a map between normed spaces. f is *differentiable* at $x_0 \in X$ if there exists $L \in \mathcal{L}(X, Y)$ s.t. $\forall \varepsilon > 0$, $\exists \delta > 0$

$$\|x - x_0\|_X < \delta \;\Rightarrow\; \frac{\|f(x) - f(x_0) - L(x - x_0)\|_Y}{\|x - x_0\|} < \varepsilon$$

$L = Df(x_0)$

Special case: $f : X \to \mathbb{R}$

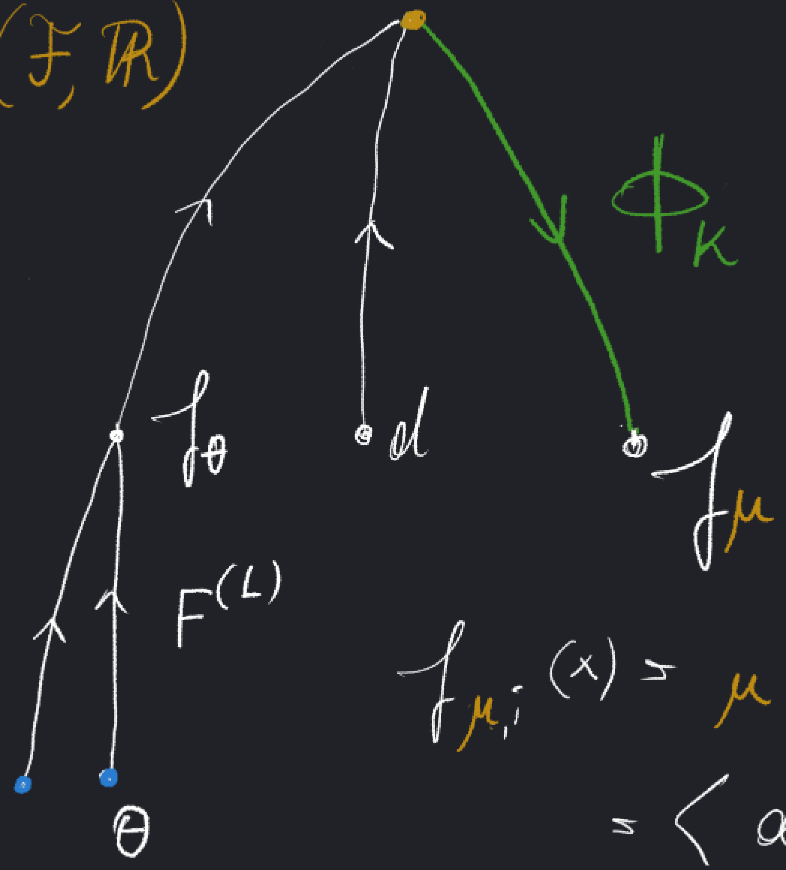
$Df(x_0) \in \mathcal{L}(X, \mathbb{R}) = X^*$ (dual space of X)

$$\mu = \langle d, \cdot \rangle_{\mathcal{P}_{in}} \qquad \mathcal{D}_{in} C(f_0) = \langle d|_{f_0}, \cdot \rangle$$

$\mathcal{F}^* = \mathcal{L}(\mathcal{F}, \mathbb{R})$
dual space

$\mathcal{F}$
function space

Θ
parameter space



Kernel Gradient
$\nabla_K C(f_0)$

$$f_{\mu,i}(x) = \mu K_{i,\cdot}(x, \cdot)$$
$$= \langle d, K_{i,\cdot}(x, \cdot) \rangle_{\mathcal{P}_{in}}$$

Kernel Gradient

$$\nabla_K \mathcal{C}|_{f_0} = \Phi_K(D_{in}\mathcal{C}(f_0))$$

On empirical dataset:

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

Kernel gradient descent

$f(t) \in \mathcal{F}$ follows the *kernel gradient descent* with respect to kernel K iff

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

The cost $\mathcal{C}(f(t))$ evolves as

$$\partial_t \mathcal{C}|_{f(t)} = -\langle d|_{f(t)}, \nabla_K \mathcal{C}|_{f(t)} \rangle_{p_{in}} = \|d|_{f(t)}\|_K^2$$

- If K is p.d. with respect to $\|.\|_{p_{in}}$, then f converges to a critical point of $\mathcal{C}$ (which is decreasing).
- If $\mathcal{C}$ is convex and bounded below, f converges to a global minimum.

Example: Random Functions approximation

Given a kernel K, we can approximate it by sampling P random functions $f^{(p)}$ from a distribution whose covariance is given by K:

$$\mathbb{E}[f_k^{(p)}(x)f_{k'}^{(p)}(x')] = K_{kk'}(x, x')$$

Random Linear parametrization F^{lin}:

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

Example: Random Fourier Features

Example: Random Functions approximation

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

Optimizing $C \circ F^{lin}$ via gradient descent yields

$$\partial_t \theta_p(t) = -\frac{1}{\sqrt{P}} D_{in} C(f_{\theta(t)}) f^{(p)} = -\frac{1}{\sqrt{P}} \langle d|_{f_{\theta(t)}}, f^{(p)} \rangle_{p_{in}}$$

Example: Random Functions approximation

$$\partial_t f_{\theta(t)} = \frac{1}{\sqrt{P}} \sum_{p=1}^{P} \partial_t \theta_p(t) f^{(p)} = -\frac{1}{P} \sum_{p=1}^{P} \langle d|_{f_{\theta(t)}}, f^{(p)} \rangle_{p_{in}} f^{(p)}$$

Tangent Kernel

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

Neural Tangent Kernel

For neural networks, the network function follows the kernel gradient descent

$$\partial_t f_{\theta(t)} = -\nabla_{\Theta^{(L)}} \mathcal{C}|_{f_{\theta(t)}}$$

with the *neural tangent kernel*

$$\Theta^{(L)}(\theta) = \sum_{p=1}^{P} \partial_{\theta_p} F^{(L)}(\theta) \otimes F^{(L)}(\theta)$$

which corresponds to the feature map $x \mapsto \nabla_\theta f_\theta(x)$

Infinite width limit

As $n_1, \dots, n_L \to \infty$, σ Lipschitz

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

scalar kernel

$$\boxed{\begin{aligned}\Theta_\infty^{(1)}(x,x') &= \Sigma^{(1)}(x,x')\\ \Theta_\infty^{(\ell+1)}(x,x') &= \Theta_\infty^{(\ell)}(x,x')\,\dot{\Sigma}^{(\ell+1)}_{(x,x')} + \Sigma^{(\ell+1)}(x,x')\end{aligned}}$$

$$\mathbb{E}_{f\sim \mathcal{GP}(0,\Sigma^{(\ell)})}\left[\sigma'(f(x))\,\sigma'(f(x'))\right]$$

$$\Sigma^{(1)}(x,x') = \frac{1}{n_0} x^T x' + \beta^2 \quad \text{(input covariance)}$$

$$\Sigma^{(\ell+1)}(x,x') = \mathbb{E}_{f\sim \mathcal{GP}(0,\Sigma^{(\ell)})}\left[\sigma(f(x))\,\sigma(f(x'))\right] + \beta^2$$

Training

Suppose parameters are updated in some training direction $d_t \in \mathcal{F}$:

$$\partial_t \theta_p(t) = \left\langle \partial_{\theta_p} F^{(L)}(\theta(t)), d_t \right\rangle$$

such that $\int_0^T \|d_t\|_{p_{in}} dt$ is *stochastically bounded*, then as $n_1, \ldots, n_L \to \infty$

$$\Theta^{(L)}(t) \to \Theta^{(L)}_\infty \otimes I_{n_L}$$

NB. σ Lipschitz, with bounded second derivative

Positive Definiteness

The NTK is already positive *semidefinite*.

For positive definiteness, we need the span of $\partial_{\theta_p} F^{(L)}$ to be dense in $(\mathcal{F}, \|.\|_{p_{in}})$

NB. The pre-activations of the last layer appear in $\partial_{\theta_p} F^{(L)}$ *are* dense for many p_{in} and activation functions (by Universal Approximation Theorems)

Example: least-square regression

$$\mathcal{E}(f) = \frac{1}{2}\|f^* - f\|^2_{\varphi_{in}}$$

$$\partial_t f_t = \Phi_K\left(\langle f^* - f_t, \cdot \rangle_{\varphi_{in}}\right) = \underbrace{\Pi}_{\text{linear operator}}(f^* - f_t)$$

$\hookrightarrow$ NTK

analytical solution:

$$f_t = f^* + \underbrace{e^{-t\Pi}}_{\text{exponential operator}}(f^* - f_0)$$

$$e^{-t\Pi} = \sum_{k=0}^{\infty} \frac{(-t)^k \Pi^k}{k!}$$

If $\Pi = \sum_i \lambda_i \langle f^{(i)}, \cdot \rangle f^{(i)}$; $e^{-t\Pi} = \sum_i e^{-\lambda_i t} \langle f^{(i)}, \cdot \rangle f^{(i)}$

finite dataset: $\Pi f_k(x) = \frac{1}{N} \sum_{i=1}^{N} \sum_{k'=1}^{n_L} f_{k'}(x_i) K_{kk'}(x_i, x)$

$\rightsquigarrow$ at most $N n_L$ eigenfunctions $\rightsquigarrow$ Kernel PCA

$$(f^* - f_0) = \Delta_f^0 + \Delta_f^1 + \dots + \Delta_f^{N n_L}$$

$$f_t = f^* + \underbrace{\Delta_f^0}_{\text{projection in } \ker(\Pi)} + \sum_{i=1}^{N n_L} e^{-t\lambda_i} \Delta_f^i$$

Probabilistic interpretation

taking $t \to \infty$: $f_\infty = f^* + \Delta_f^0 = f_0 - \sum_i \Delta_f^i$

$$f_{\infty,k}(x) = \underbrace{k_{x,k}^T \tilde{K}^{-1} y^*}_{\substack{\text{MAP estimate for} \\ f_k \sim GP(0,\, \Theta_\infty^{(L)}) \\ = \text{Kernel Regression estimate}}} + \underbrace{\left(f_0(x) - k_{x,k}^T \tilde{K}^{-1} y_0 \right)}_{\text{centered Gaussian}}$$

$$\boxed{\begin{aligned} y^* &= \left(f_k^*(x_i) \right)_{k,i} \\ k_{x,k} &= \left(K_{kk'}(x, x_i) \right)_{i,k'} \\ \tilde{K} &= \left(K_{kk'}(x_i, x_j) \right)_{ik,jk'} \end{aligned}}$$

Summary

- Can use (Neural) Tangent Kernel to describe model evolution during training
- Constant limit at infinite width: prove convergence with positive definiteness
- Direct Link between NNs and Kernel methods

Going Further

- NTK for other architectures (see Tensor Programs)
- Predict "maximum effective depth" for given architecture
- Designing activation functions to achieve particular NTK
- ...