On the Inductive Bias of Neural Tangent Kernels

Alberto Bietti   Julien Mairal - Inria Grenoble

Inria   erc   Microsoft Research - Inria JOINT CENTRE

Inductive Bias and Over-Parameterization

Optimization and Inductive Bias:
- Over-parameterized deep networks have great approximation power
- Optimization algorithm is plays a crucial role for generalization

Lazy Training: In certain regimes (over-parameterization, particular initialization), neural networks behave like their **linearization** near initialization

$$f(x;\theta) \approx f(x;\theta_0) + \langle \theta - \theta_0, \nabla_\theta f(x;\theta_0)\rangle$$

Neural Tangent Kernels (NTK): In this regime, generalization properties are controlled by the **limiting kernel** [Jacot et al., 2018]

$$\langle \nabla_\theta f(x;\theta_0), \nabla_\theta f(x',\theta_0)\rangle \to K(x,x').$$

In particular, with squared loss and infinite width, we get the interpolating solution with minimum RKHS norm.

Contributions:
- Derivation of NTK for **convolutional networks** with generic linear patch extraction/pooling operators;
- Study of **smoothness, stability, and approximation** properties of functions with finite RKHS norm;
- Comparison to other ReLU kernels (e.g. training only last layer): the NTK has **weaker smoothness** properties but **better approximation**

Neural Tangent Kernels for CNNs

Two-layer ReLU Networks: $f(x;\theta) = \sqrt{\frac{2}{m}} \sum_{j=1}^m v_j \sigma(w_j^\top x)$, NTK given by

$$K(x,x') = \|x\|\|x'\|\kappa\left(\frac{\langle x, x'\rangle}{\|x\|\|x'\|}\right),$$

where $\kappa(u) := u\kappa_0(u) + \kappa_1(u)$,

$$\kappa_0(u) = \frac{1}{\pi}(\pi - \arccos(u)), \qquad \kappa_1(u) = \frac{1}{\pi}\left(u\cdot(\pi - \arccos(u)) + \sqrt{1-u^2}\right).$$

Convolutional networks:
- Signals $x[u]$ in $\ell^2(\mathbb{Z}^d)$
- **Patch extraction** operators $P^k x[u] = |S_k|^{-1/2}(x[u+v])_{v\in S_k} \in \mathcal{H}^{|S_k|}$
- Linear **pooling** operators $A^k x[u] = \sum_{v\in\mathbb{Z}^d} h_k[u-v]x[v]$

Network: $f(x;\theta) = \sqrt{\frac{2}{m_n}}\langle w^{n+1}, a^n\rangle_{\ell^2}$, with

$$\tilde{a}^k[u] = \sqrt{2/m_{k-1}}\, W^k P^k a^{k-1}[u],$$
$$a^k[u] = A^k\sigma(\tilde{a}^k)[u], \quad k = 1,\dots,n,$$

NTK: Consider the non-linear operator

$$M(x,y)[u] = \begin{pmatrix} \varphi_0(x[u]) \otimes y[u] \\ \varphi_1(x[u]) \end{pmatrix},$$

where φ_0, φ_1 are kernel mappings for kernels κ_0 and κ_1.

Proposition (NTK feature map for CNN)

The NTK is given by

$$K(x,x') = \langle \Phi(x), \Phi(x')\rangle_{\ell^2(\mathbb{Z}^d)},$$

with $\Phi(x)[u] = A^n M(x_n, y_n)[u]$, $y_1[u] = x_1[u] = P^1 x[u]$ *and*

$$x_k[u] = P^k A^{k-1}\varphi_1(x_{k-1})[u]$$
$$y_k[u] = P^k A^{k-1} M(x_{k-1}, y_{k-1})[u],$$

with the notation $\varphi_1(x)[u] = \varphi_1(x[u])$ *for a signal* x.

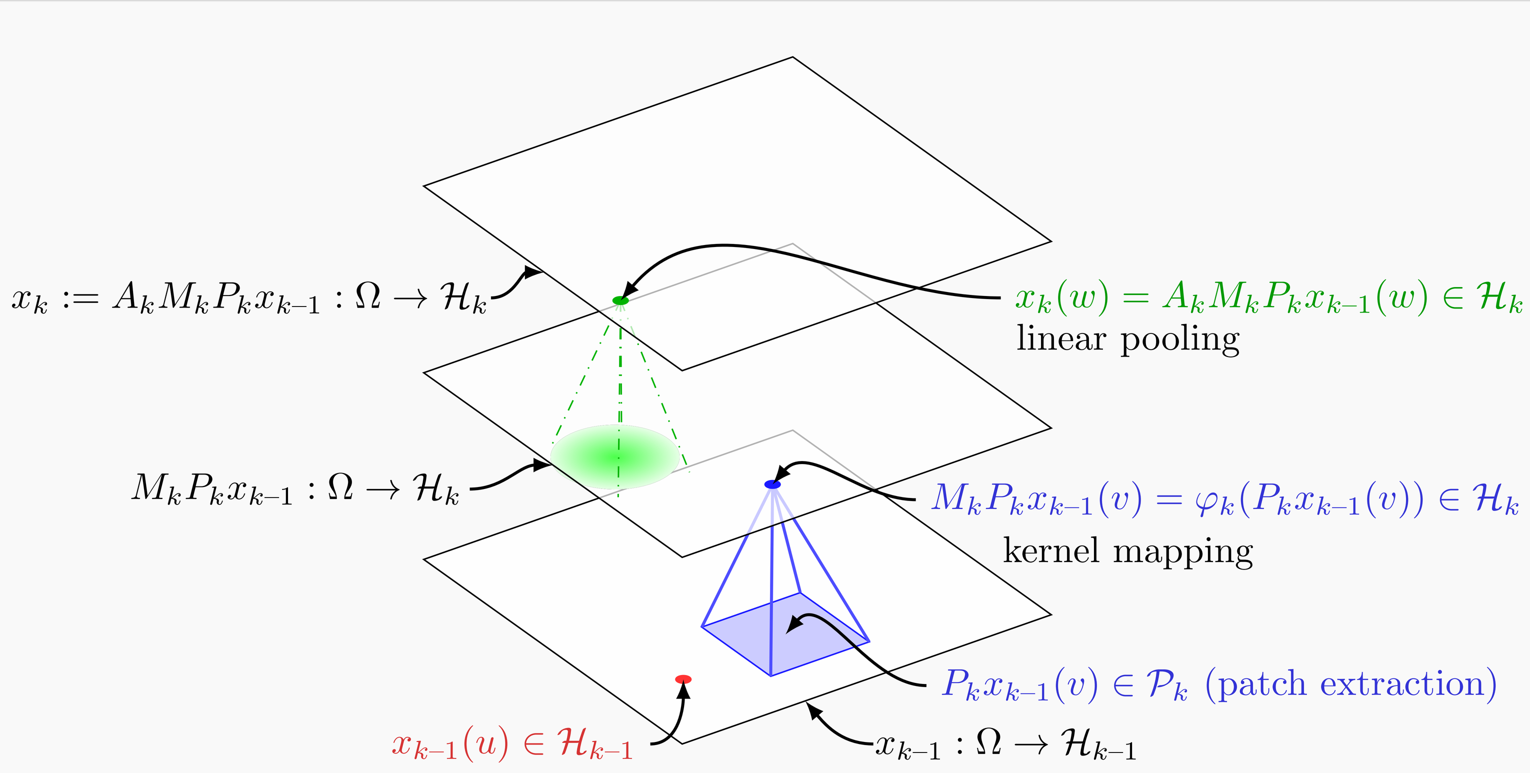


Figure: Illustration of feature maps construction for x_k.

Smoothness and Deformation Stability

Two-layer ReLU networks: The NTK (when training both layers) has **weaker smoothness** compared to training only the second layer.

Proposition (Non-Lipschitzness)

The kernel mapping $\Phi(\cdot)$ of the two-layer NTK is not Lipschitz:

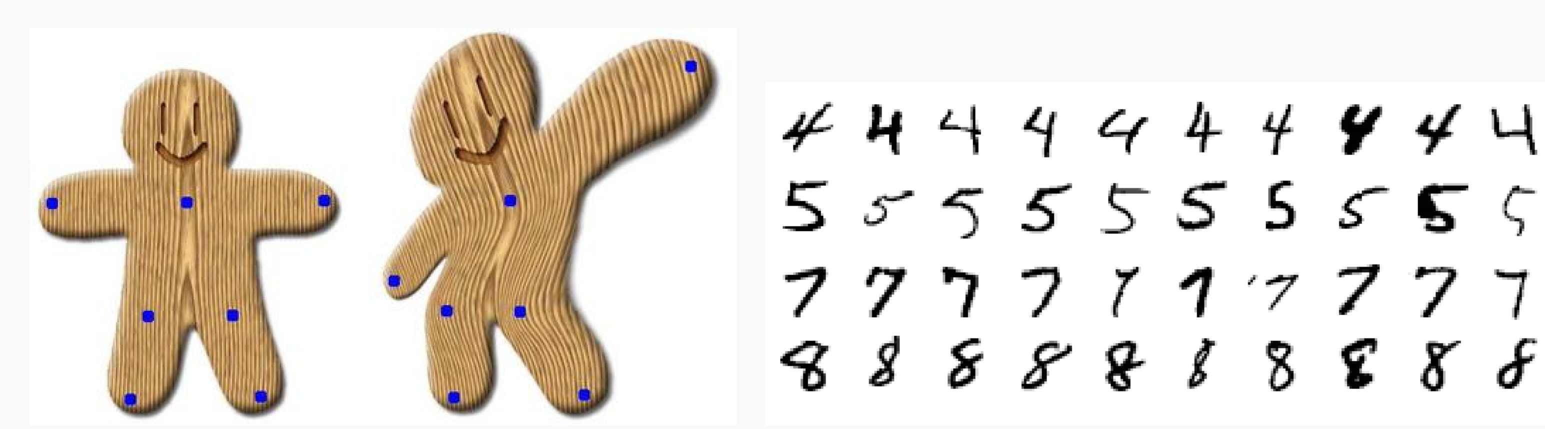
$$\sup_{x,y} \frac{\|\Phi(x) - \Phi(y)\|_{\mathcal{H}}}{\|x-y\|} \to +\infty.$$

It follows that the RKHS $\mathcal{H}$ contains unit-norm functions with arbitrarily large Lipschitz constant.

Proposition (Smoothness for ReLU NTK)

The kernel mapping Φ satisfies

$$\|\Phi(x) - \Phi(y)\| \leq \sqrt{\min(\|x\|,\|y\|)\|x-y\|} + 2\|x-y\|.$$



Deformation stability for deep ReLU CNNs: Similar assumptions to [Bietti and Mairal, 2019]
- **Continuous** signals $x(u)$ in $L^2(\mathbb{R}^d)$, deformations $L_\tau x(u) = x(u - \tau(u))$
- **Anti-aliasing** of the original signal: $A_0 x$ instead of x
- **Patch sizes** controlled at current resolution: $\sup_{v\in S_k}|v| \leq \beta\sigma_{k-1}$

Proposition (Stability of NTK)

Let $\Phi_n(x) = \Phi(A_0 x)$, *and assume* $\|\nabla\tau\|_\infty \leq 1/2$. *We have:*

$$\|\Phi_n(L_\tau x) - \Phi_n(x)\| \leq \left(C_\beta n^{7/4}\|\nabla\tau\|_\infty^{1/2} + C'_\beta n^2\|\nabla\tau\|_\infty + \sqrt{n+1}\frac{C''}{\sigma_n}\|\tau\|_\infty\right)\|x\|.$$

Worse dependence on $\|\nabla\tau\|_\infty$ for small deformations compared to training just the last layer!

Approximation Properties

Q: How rich is the RKHS for the NTK κ versus the simpler kernel κ_1 obtained by training just the second layer?

Mercer decomposition with spherical harmonics:

Proposition (Mercer decomposition)

For any $x, y \in \mathbb{S}^{p-1}$, we have the following decomposition of the NTK κ:

$$\kappa(\langle x, y\rangle) = \sum_{k=0}^{\infty} \mu_k \sum_{j=1}^{N(p,k)} Y_{k,j}(x) Y_{k,j}(y), \tag{1}$$

*where $Y_{k,j}$ are **spherical harmonic** polynomials of degree k, and the non-negative eigenvalues μ_k satisfy $\mu_0, \mu_1 > 0$, $\mu_k = 0$ if $k = 2j+1$ with $j \geq 1$, and otherwise $\mu_k \sim C(p)k^{-p}$ as $k \to \infty$.*

This gives an explicit characterization of the RKHS norm of a function.

Approximation results: (following [Bach 2017])
- The RKHS is "**larger**": **slower decay** compared to κ_1, for which $\mu_k = O(k^{-p-2})$;
- Contains functions with weaker requirements on derivatives;
- Better rates for approximating Lipschitz functions on the sphere.

Relevant References

F. Bach (2017).
Breaking the curse of dimensionality with convex neural networks.

A. Bietti and J. Mairal (2019).
Invariance and stability of deep convolutional representations.

A. Jacot, F. Gabriel and C. Hongler (2018).
Neural Tangent Kernel: convergence and generalization in neural networks.