Invariance and Stability to Deformations of Deep Convolutional Representations

Alberto Bietti

Inria Grenoble

Microsoft Research, Redmond. June 18, 2019.



Success of deep convolutional networks



Convolutional Neural Networks (CNNs):

- Capture multi-scale and compositional structure in natural signals
- Provide some invariance
- Model local stationarity
- State-of-the-art in many applications

Understanding deep convolutional representations

- Are they stable to deformations?
- How can we achieve invariance to transformation groups?
- Do they preserve signal information?
- What are good measures of model complexity?

Kernels?

- Map data x to high-dimensional space, $\Phi(x) \in \mathcal{H}$ (\mathcal{H} : "RKHS")
- Non-linear $f \in \mathcal{H}$ takes linear form: $f(x) = \langle f, \Phi(x) \rangle$
- Learning with a positive definite kernel $\mathcal{K}(x,x') = \langle \Phi(x), \Phi(x') \rangle$

Kernels?

- Map data x to high-dimensional space, $\Phi(x) \in \mathcal{H}$ (\mathcal{H} : "RKHS")
- Non-linear $f \in \mathcal{H}$ takes linear form: $f(x) = \langle f, \Phi(x) \rangle$
- Learning with a positive definite kernel $\mathcal{K}(x,x') = \langle \Phi(x), \Phi(x') \rangle$
- Here, we construct specific kernels based on convolutional architectures, following Mairal (2016)

Kernels?

- Map data x to high-dimensional space, $\Phi(x) \in \mathcal{H}$ (\mathcal{H} : "RKHS")
- Non-linear $f \in \mathcal{H}$ takes linear form: $f(x) = \langle f, \Phi(x) \rangle$
- Learning with a positive definite kernel $K(x,x') = \langle \Phi(x), \Phi(x') \rangle$
- Here, we construct specific kernels based on convolutional architectures, following Mairal (2016)
 - ► Good empirical performance on image tasks (Mairal et al., 2014; Mairal, 2016)
 - ▶ RKHS contains CNNs, leads to good regularizers (Bietti et al., 2019)
 - ► Also related to neural tangent kernels for CNNs (Bietti and Mairal, 2019b)

Why? Separate learning from representation: $f(x) = \langle f, \Phi(x) \rangle$

- $\Phi(x)$: CNN architecture (stability, invariance, signal preservation)
- ${\ }$ $\bullet \ f:$ CNN model, learning, generalization through RKHS norm $\|f\|$

 $|f(x)-f(x')|\leq \|f\|\cdot\|\Phi(x)-\Phi(x')\|$

- ||f|| controls both stability and model complexity!
 - ightarrow discriminating small perturbations requires large $\|f\|$
 - \rightarrow learning stable functions may be "easier"

A signal processing perspective

- Consider images defined on a **continuous** domain $\Omega = \mathbb{R}^2$.
- $\tau: \Omega \to \Omega$: C^1 -diffeomorphism.
- $L_{\tau}x(u) = x(u \tau(u))$: action operator.
- Much richer group of transformations than translations.



A signal processing perspective

• Consider images defined on a **continuous** domain $\Omega = \mathbb{R}^2$.

- $\tau: \Omega \to \Omega$: C^1 -diffeomorphism.
- $L_{\tau}x(u) = x(u \tau(u))$: action operator.
- Much richer group of transformations than translations.

Definition of stability

• Representation $\Phi(\cdot)$ is **stable** (Mallat, 2012) if:

$$\|\Phi(L_ au x)-\Phi(x)\|\leq (C_1\|
abla au\|_\infty+C_2\| au\|_\infty)\|x\|_\infty$$

- $\|\nabla \tau\|_{\infty} = \sup_{u} \|\nabla \tau(u)\|$ controls deformation.
- $\|\tau\|_{\infty} = \sup_{u} |\tau(u)|$ controls translation.
- $C_2 \rightarrow 0$: translation invariance.

Outline

1 Construction of the Convolutional Representation

2 Invariance and Stability

3 Learning Aspects: Model Complexity of CNNs

4 Regularizing with the RKHS norm

• $x_0: \Omega \to \mathcal{H}_0$: initial (continuous) signal

- $u \in \Omega = \mathbb{R}^d$: location (d = 2 for images)
- $x_0(u) \in \mathcal{H}_0$: value ($\mathcal{H}_0 = \mathbb{R}^3$ for RGB images)

• $x_0 : \Omega \to \mathcal{H}_0$: initial (continuous) signal

- $u \in \Omega = \mathbb{R}^d$: location (d = 2 for images)
- $x_0(u) \in \mathcal{H}_0$: value ($\mathcal{H}_0 = \mathbb{R}^3$ for RGB images)

• $x_k : \Omega \to \mathcal{H}_k$: feature map at layer k

 $P_k x_{k-1}$

• P_k : patch extraction operator, extract small patch of feature map x_{k-1} around each point u

• $x_0: \Omega \to \mathcal{H}_0$: initial (continuous) signal

- $u \in \Omega = \mathbb{R}^d$: location (d = 2 for images)
- $x_0(u) \in \mathcal{H}_0$: value ($\mathcal{H}_0 = \mathbb{R}^3$ for RGB images)
- $x_k : \Omega \to \mathcal{H}_k$: feature map at layer k

$$M_k P_k x_{k-1}$$

- P_k : patch extraction operator, extract small patch of feature map x_{k-1} around each point u
- M_k : non-linear mapping operator, maps each patch to a new point with a pointwise non-linear function $\varphi_k(\cdot)$

• $x_0: \Omega \to \mathcal{H}_0$: initial (continuous) signal

- $u \in \Omega = \mathbb{R}^d$: location (d = 2 for images)
- $x_0(u) \in \mathcal{H}_0$: value ($\mathcal{H}_0 = \mathbb{R}^3$ for RGB images)
- $x_k : \Omega \to \mathcal{H}_k$: feature map at layer k

$$x_k = A_k M_k P_k x_{k-1}$$

- P_k : patch extraction operator, extract small patch of feature map x_{k-1} around each point u
- ► M_k : non-linear mapping operator, maps each patch to a new point with a pointwise non-linear function $\varphi_k(\cdot)$
- A_k : (linear, Gaussian) **pooling** operator at scale σ_k



Patch extraction operator P_k



Patch extraction operator P_k

$$P_k x_{k-1}(u) := (v \in S_k \mapsto x_{k-1}(u+v)) \in \mathcal{P}_k = \mathcal{H}_{k-1}^{S_k}$$

- S_k : patch shape, e.g. box
- P_k is linear, and preserves the L^2 norm: $||P_k x_{k-1}|| = ||x_{k-1}||$

Non-linear mapping operator M_k



Non-linear mapping operator M_k

$$M_k P_k x_{k-1}(u) := \varphi_k (P_k x_{k-1}(u)) \in \mathcal{H}_k$$

φ_k : P_k → H_k pointwise non-linearity on patches (kernel map)
We assume non-expansivity: for z, z' ∈ P_k

$$\|\varphi_k(z)\| \le \|z\|$$
 and $\|\varphi_k(z) - \varphi_k(z')\| \le \|z - z'\|$

• M_k then satisfies, for $x,x'\in L^2(\Omega,\mathcal{P}_k)$

$$\|M_k x\| \le \|x\|$$
 and $\|M_k x - M_k x'\| \le \|x - x'\|$

Non-linear mapping operator M_k

$$M_k P_k x_{k-1}(u) := \varphi_k (P_k x_{k-1}(u)) \in \mathcal{H}_k$$

φ_k : P_k → H_k pointwise non-linearity on patches
We assume: for z, z' ∈ P_k

 $\|arphi_k(z)\|\leq
ho_k\|z\|$ and $\|arphi_k(z)-arphi_k(z')\|\leq
ho_k\|z-z'\|$

• M_k then satisfies, for $x,x'\in L^2(\Omega,\mathcal{P}_k)$

$$\|M_k x\| \le \rho_k \|x\|$$
 and $\|M_k x - M_k x'\| \le \rho_k \|x - x'\|$

• (at the cost of paying $\prod_k \rho_k$ later)

φ_k from kernels

• Kernel mapping of homogeneous dot-product kernels:

$$\mathcal{K}_k(z,z') = \|z\| \|z'\| \kappa_k \Big(rac{\langle z,z'
angle}{\|z\|\|z'\|}\Big) = \langle arphi_k(z),arphi_k(z')
angle.$$

•
$$\kappa_k(u) = \sum_{j=0}^\infty b_j u^j$$
 with $b_j \ge 0$, $\kappa_k(1) = 1$

- Commonly used for hierarchical kernels
- $\|\varphi_k(z)\| = K_k(z,z)^{1/2} = \|z\|$
- $\| arphi_k(z) arphi_k(z') \| \leq \| z z' \|$ if $\kappa_k'(1) \leq 1$
- \implies non-expansive

φ_k from kernels

• Kernel mapping of homogeneous dot-product kernels:

$$\mathcal{K}_k(z,z') = \|z\| \|z'\| \kappa_k igg(rac{\langle z,z'
angle}{\|z\| \|z'\|} igg) = \langle arphi_k(z),arphi_k(z')
angle.$$

•
$$\kappa_k(u) = \sum_{j=0}^\infty b_j u^j$$
 with $b_j \ge 0$, $\kappa_k(1) = 1$

- Commonly used for hierarchical kernels
- $\|\varphi_k(z)\| = K_k(z,z)^{1/2} = \|z\|$
- $\| arphi_k(z) arphi_k(z') \| \leq \| z z' \|$ if $\kappa_k'(1) \leq 1$
- \implies non-expansive
- Examples:

•
$$\kappa_{\exp}(\langle z, z' \rangle) = e^{\langle z, z' \rangle - 1}$$
 (Gaussian kernel on the sphere)

- $\kappa_{\text{inv-poly}}(\langle z, z' \rangle) = \frac{1}{2 \langle z, z' \rangle}$
- ▶ arc-cosine kernel of degree 1 (random features with ReLU activation)

φ_k from kernels: CKNs approximation

Convolutional Kernel Networks approximation (Mairal, 2016):

φ_k from kernels: CKNs approximation

Convolutional Kernel Networks approximation (Mairal, 2016):

- Approximate $\varphi_k(z)$ by **projection** on $span(\varphi_k(z_1), \ldots, \varphi_k(z_p))$ (Nystrom)
- Leads to **tractable**, *p*-dimensional representation $\psi_k(z)$
- Norm is preserved, and projection is non-expansive:

$$\begin{split} \|\psi_k(z) - \psi_k(z')\| &= \|\Pi_k \varphi_k(z) - \Pi_k \varphi_k(z')\| \\ &\leq \|\varphi_k(z) - \varphi_k(z')\| \leq \|z - z'\| \end{split}$$

• Anchor points z_1, \ldots, z_p (\approx filters) can be **learned from data** (K-means or backprop)

φ_k from kernels: CKNs approximation



Pooling operator A_k



Pooling operator A_k

$$x_k(u) = A_k M_k P_k x_{k-1}(u) = \int_{\mathbb{R}^d} h_{\sigma_k}(u-v) M_k P_k x_{k-1}(v) dv \in \mathcal{H}_k$$

- h_{σ_k} : pooling filter at scale σ_k
- $h_{\sigma_k}(u) := \sigma_k^{-d} h(u/\sigma_k)$ with h(u) Gaussian
- linear, non-expansive operator: $\|A_k\| \leq 1$

Recap: P_k , M_k , A_k



Multilayer construction

Assumption on x₀

- x_0 is typically a **discrete** signal aquired with physical device.
- Natural assumption: $x_0 = A_0 x$, with x the original continuous signal, A_0 local integrator with scale σ_0 (anti-aliasing).

Multilayer construction

Assumption on x₀

- x_0 is typically a **discrete** signal aquired with physical device.
- Natural assumption: $x_0 = A_0 x$, with x the original continuous signal, A_0 local integrator with scale σ_0 (anti-aliasing).

Multilayer representation

$$\Phi_n(x) = A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x_0 \in L^2(\Omega, \mathcal{H}_n).$$

• S_k , σ_k grow exponentially in practice (i.e., fixed with subsampling).

Multilayer construction

Assumption on x₀

- x_0 is typically a **discrete** signal aquired with physical device.
- Natural assumption: $x_0 = A_0 x$, with x the original continuous signal, A_0 local integrator with scale σ_0 (anti-aliasing).

Multilayer representation

$$\Phi_n(x) = A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x_0 \in L^2(\Omega, \mathcal{H}_n).$$

• S_k , σ_k grow exponentially in practice (i.e., fixed with subsampling). **Prediction layer**

• e.g., linear
$$f(x) = \langle w, \Phi_n(x) \rangle$$
.

• "linear kernel" $\mathcal{K}(x,x') = \langle \Phi_n(x), \Phi_n(x') \rangle = \int_{\Omega} \langle x_n(u), x'_n(u) \rangle du$.



• \bar{x}_k : subsampling factor s_k after pooling with scale $\sigma_k \approx s_k$:

 $\bar{x}_k[n] = A_k M_k P_k \bar{x}_{k-1}[ns_k]$

• \bar{x}_k : subsampling factor s_k after pooling with scale $\sigma_k \approx s_k$:

$$\bar{x}_k[n] = A_k M_k P_k \bar{x}_{k-1}[ns_k]$$

• Claim: We can recover \bar{x}_{k-1} from \bar{x}_k if subsampling $s_k \leq$ patch size

• \bar{x}_k : subsampling factor s_k after pooling with scale $\sigma_k \approx s_k$:

$$\bar{x}_k[n] = A_k M_k P_k \bar{x}_{k-1}[ns_k]$$

■ Claim: We can recover x
_{k-1} from x
_k if subsampling s_k ≤ patch size
■ How? Kernels! Recover patches with linear functions (contained in RKHS)

$$\langle f_w, M_k P_k x(u) \rangle = f_w(P_k x(u)) = \langle w, P_k x(u) \rangle$$

Signal recovery: example in 1D


Outline

1) Construction of the Convolutional Representation

(2) Invariance and Stability

3 Learning Aspects: Model Complexity of CNNs

4 Regularizing with the RKHS norm

Stability to deformations: definitions

- $\tau: \Omega \to \Omega$: C^1 -diffeomorphism
- $L_{\tau}x(u) = x(u \tau(u))$: action operator
- Much richer group of transformations than translations



44444444444 777717777 88888888888

• Studied for wavelet-based scattering transform (Mallat, 2012; Bruna and Mallat, 2013)

Stability to deformations: definitions

• Representation $\Phi(\cdot)$ is **stable** (Mallat, 2012) if:

 $\|\Phi(L_{\tau}x)-\Phi(x)\|\leq (C_1\|\nabla\tau\|_{\infty}+C_2\|\tau\|_{\infty})\|x\|$

- $\|\nabla \tau\|_{\infty} = \sup_{u} \|\nabla \tau(u)\|$ controls deformation
- $\|\tau\|_{\infty} = \sup_{u} |\tau(u)|$ controls translation
- $C_2 \rightarrow 0$: translation invariance

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x.$$

• Translation: $L_c x(u) = x(u-c)$

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x.$$

- Translation: $L_c x(u) = x(u-c)$
- Equivariance all operators commute with L_c : $\Box L_c = L_c \Box$

$$\begin{aligned} \|\Phi_n(L_c x) - \Phi_n(x)\| &= \|L_c \Phi_n(x) - \Phi_n(x)\| \\ &\leq \|L_c A_n - A_n\| \cdot \|x\| \end{aligned}$$

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x.$$

- Translation: $L_c x(u) = x(u-c)$
- Equivariance all operators commute with L_c : $\Box L_c = L_c \Box$

$$\begin{aligned} \|\Phi_n(L_c x) - \Phi_n(x)\| &= \|L_c \Phi_n(x) - \Phi_n(x)\| \\ &\leq \|L_c A_n - A_n\| \cdot \|x\| \end{aligned}$$

• Mallat (2012): $\|L_{ au}A_n - A_n\| \leq rac{C_2}{\sigma_n} \| au\|_\infty$

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x.$$

- Translation: $L_c x(u) = x(u-c)$
- Equivariance all operators commute with L_c : $\Box L_c = L_c \Box$

$$\|\Phi_n(L_c x) - \Phi_n(x)\| = \|L_c \Phi_n(x) - \Phi_n(x)\|$$

$$\leq \|L_c A_n - A_n\| \cdot \|x\|$$

• Mallat (2012): $\|L_c A_n - A_n\| \leq \frac{C_2}{\sigma_n} c$

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x.$$

• Patch extraction P_k and pooling A_k do not commute with L_{τ} !

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x.$$

• Patch extraction P_k and pooling A_k do not commute with L_{τ} !

• $||A_k L_{\tau} - L_{\tau} A_k|| \le C_1 ||\nabla \tau||_{\infty}$ (from Mallat, 2012)

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x.$$

• Patch extraction P_k and pooling A_k do not commute with L_{τ} !

• $\|[A_k, L_{\tau}]\| \leq C_1 \|\nabla \tau\|_{\infty}$ (from Mallat, 2012)

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x.$$

- Patch extraction P_k and pooling A_k do not commute with L_{τ} !
- $\|[A_k, L_{\tau}]\| \leq C_1 \|\nabla \tau\|_{\infty}$ (from Mallat, 2012)
- But: $[P_k, L_{\tau}]$ is **unstable** at high frequencies!

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 A_0 x.$$

- Patch extraction P_k and pooling A_k do not commute with L_{τ} !
- $\|[A_k, L_{ au}]\| \le C_1 \|
 abla au\|_\infty$ (from Mallat, 2012)
- But: $[P_k, L_{\tau}]$ is **unstable** at high frequencies!
- Adapt to current layer resolution, patch size controlled by σ_{k-1} :

$$\|[P_k A_{k-1}, L_{\tau}]\| \le C_{1,\beta} \|\nabla \tau\|_{\infty} \qquad \sup_{u \in S_k} |u| \le \beta \sigma_{k-1}$$

• Representation:

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 A_0 x.$$

- Patch extraction P_k and pooling A_k do not commute with L_{τ} !
- $\|[A_k, L_{ au}]\| \le C_1 \|
 abla au\|_\infty$ (from Mallat, 2012)
- But: $[P_k, L_{\tau}]$ is **unstable** at high frequencies!
- Adapt to current layer resolution, patch size controlled by σ_{k-1} :

$$\|[P_k A_{k-1}, L_{\tau}]\| \le C_{1,\beta} \|\nabla \tau\|_{\infty} \qquad \sup_{u \in S_k} |u| \le \beta \sigma_{k-1}$$

• $C_{1,\beta}$ grows as $\beta^{d+1} \implies$ more stable with small patches (e.g., 3x3, VGG et al.)

Stability to deformations: final result

Theorem

If $\|\nabla \tau\|_{\infty} \leq 1/2$,

$$\|\Phi_n(L_{\tau}x)-\Phi_n(x)\| \leq \left(C_{1,\beta}\left(n+1\right)\|\nabla\tau\|_{\infty}+\frac{C_2}{\sigma_n}\|\tau\|_{\infty}\right)\|x\|$$

- translation invariance: large σ_n
- stability: small patch sizes
- ullet signal preservation: subsampling factor \approx patch size
- ullet \Longrightarrow needs several layers

Stability to deformations: final result

Theorem

If $\|
abla au \|_\infty \leq 1/2$,

$$\|\Phi_n(L_{\tau}x) - \Phi_n(x)\| \leq \prod_k \rho_k \left(C_{1,\beta} \left(n + 1 \right) \|\nabla \tau\|_{\infty} + \frac{C_2}{\sigma_n} \|\tau\|_{\infty} \right) \|x\|$$

- translation invariance: large σ_n
- stability: small patch sizes
- ${\scriptstyle \bullet}$ signal preservation: subsampling factor \approx patch size
- \implies needs several layers
- (also valid for generic CNNs with ReLUs: multiply by $\prod_k \rho_k = \prod_k ||W_k||$, but no direct signal preservation).

Beyond the translation group

Global invariance to other groups?

- Rotations, reflections, roto-translations, ...
- Group action $L_g x(u) = x(g^{-1}u)$
- Equivariance in inner layers + (global) pooling in last layer
- Similar construction to Cohen and Welling (2016); Kondor and Trivedi (2018)

G-equivariant layer construction

- Feature maps x(u) defined on $u \in G$ (G: locally compact group)
 - ▶ Input needs special definition when $G \neq \Omega$
- Patch extraction:

$$Px(u) = (x(uv))_{v \in S}$$

- Non-linear mapping: equivariant because pointwise!
- **Pooling** (μ : left-invariant Haar measure):

$$Ax(u) = \int_G x(uv)h(v)d\mu(v) = \int_G x(v)h(u^{-1}v)d\mu(v)$$

Group invariance and stability

Roto-translation group $G = \mathbb{R}^2 \rtimes SO(2)$ (translations + rotations)

- **Stability** w.r.t. translation group
- Global invariance to rotations (only global pooling at final layer)
 - ► Inner layers: patches and pooling only on translation group
 - ► Last layer: global pooling on rotations
 - Cohen and Welling (2016): pooling on rotations in inner layers hurts performance on Rotated MNIST

Outline

1) Construction of the Convolutional Representation

2 Invariance and Stability

③ Learning Aspects: Model Complexity of CNNs

4 Regularizing with the RKHS norm

RKHS of patch kernels K_k

$$\mathcal{K}_k(z,z') = \|z\| \|z'\| \kappa\left(\frac{\langle z,z'\rangle}{\|z\|\|z'\|}\right), \qquad \kappa(u) = \sum_{j=0}^{\infty} b_j u^j$$

• RKHS contains homogeneous functions:

$$f: z \mapsto \|z\|\sigma(\langle g, z \rangle / \|z\|)$$

Homogeneous version of (Zhang et al., 2016, 2017)

Alberto Bietti

RKHS of patch kernels K_k

$$\mathcal{K}_k(z,z') = \|z\| \|z'\| \kappa\left(\frac{\langle z,z'\rangle}{\|z\|\|z'\|}\right), \qquad \kappa(u) = \sum_{j=0}^{\infty} b_j u^j$$

• RKHS contains homogeneous functions:

$$f: z \mapsto \|z\|\sigma(\langle g, z \rangle / \|z\|)$$

- Smooth activations: $\sigma(u) = \sum_{j=0}^{\infty} a_j u^j$
- Norm: $\|f\|_{\mathcal{H}_k}^2 \leq C_{\sigma}^2(\|g\|^2) = \sum_{j=0}^{\infty} \frac{a_j^2}{b_j} \|g\|^2 < \infty$

Homogeneous version of (Zhang et al., 2016, 2017)

RKHS of patch kernels K_k

Examples:



Alberto Bietti

Constructing a CNN in the RKHS $\mathcal{H}_\mathcal{K}$

- Consider a CNN with filters $W_k^{ij}(u), u \in S_k$
- ${\, \bullet \,}$ "Smooth homogeneous" activations σ
- $\bullet\,$ The CNN can be constructed hierarchically in $\mathcal{H}_{\mathcal{K}}$
- Norm upper bound:

$$\|f_{\sigma}\|_{\mathcal{H}}^{2} \leq \|W_{n+1}\|_{2}^{2} C_{\sigma}^{2}(\|W_{n}\|_{2}^{2} C_{\sigma}^{2}(\|W_{n-1}\|_{2}^{2} C_{\sigma}^{2}(\dots)))$$

Constructing a CNN in the RKHS $\mathcal{H}_\mathcal{K}$

- Consider a CNN with filters $W_k^{ij}(u), u \in S_k$
- ${\, \bullet \,}$ "Smooth homogeneous" activations σ
- $\bullet\,$ The CNN can be constructed hierarchically in $\mathcal{H}_{\mathcal{K}}$
- Norm upper bound (linear layers):

$$\|f_{\sigma}\|_{\mathcal{H}}^{2} \leq \|W_{n+1}\|_{2}^{2} \cdot \|W_{n}\|_{2}^{2} \cdot \|W_{n-1}\|_{2}^{2} \dots \|W_{1}\|_{2}^{2}$$

• Linear layers: product of spectral norms

Link with generalization

• Simple bound on Rademacher complexity for linear/kernel methods:

$$\mathcal{F}_B = \{ f \in \mathcal{H}_{\mathcal{K}}, \| f \|_{\mathcal{H}} \leq B \} \implies \operatorname{\mathsf{Rad}}_N(\mathcal{F}_B) \leq O\left(\frac{BR}{\sqrt{N}}\right)$$

Link with generalization

• Simple bound on Rademacher complexity for linear/kernel methods:

$$\mathcal{F}_B = \{ f \in \mathcal{H}_{\mathcal{K}}, \| f \|_{\mathcal{H}} \leq B \} \implies \mathsf{Rad}_N(\mathcal{F}_B) \leq O\left(\frac{BR}{\sqrt{N}}\right)$$

- Leads to margin bound $O(\|\hat{f}_N\|_{\mathcal{H}} R/\gamma \sqrt{N})$ for a learned CNN \hat{f}_N with margin (confidence) $\gamma > 0$
- Related to generalization bounds for neural networks based on **product of spectral norms** (*e.g.*, Bartlett et al., 2017; Neyshabur et al., 2018)

Outline

1 Construction of the Convolutional Representation

2 Invariance and Stability

3 Learning Aspects: Model Complexity of CNNs

④ Regularizing with the RKHS norm

Deep learning struggles with small datasets and adversarial examples.

(Riatti Mialon Chan and Mairal 2010) Alberto Bietti

Can we obtain better models through regularization?

(Riotti Mislon Chan and Mairal 2010) Alberto Bietti

Can we obtain better models through regularization?

• Controlling upper bounds: spectral norm penalties/constraints

Can we obtain better models through regularization?

- Controlling upper bounds: spectral norm penalties/constraints
- Controlling lower bounds using $\|f\|_{\mathcal{H}} = \sup_{\|u\|_{\mathcal{H}} \leq 1} \langle f, u \rangle$

Can we obtain better models through regularization?

- Controlling upper bounds: spectral norm penalties/constraints
- ${\ }$ Ontrolling lower bounds using $\|f\|_{\mathcal H} = \sup_{\|u\|_{\mathcal H} \leq 1} \langle f, u \rangle$
- \implies consider tractable subsets of the unit ball

Can we obtain better models through regularization?

- Controlling upper bounds: spectral norm penalties/constraints
- ${\ }$ Ontrolling lower bounds using $\|f\|_{\mathcal H} = \sup_{\|u\|_{\mathcal H} \leq 1} \langle f, u \rangle$
- \implies consider tractable subsets of the unit ball

$$\|f\|_{\mathcal{H}} \geq \sup_{x, \|\delta\| \leq 1} \langle f, \Phi(x + \delta) - \Phi(x)
angle_{\mathcal{H}}$$
 (adversarial perturbations)

Can we obtain better models through regularization?

- Controlling upper bounds: spectral norm penalties/constraints
- ${\ }$ Ontrolling lower bounds using $\|f\|_{\mathcal H} = \sup_{\|u\|_{\mathcal H} \leq 1} \langle f, u \rangle$
- $\bullet \implies {\sf consider \ tractable \ subsets \ of \ the \ unit \ ball}$

$$\|f\|_{\mathcal{H}} \geq \sup_{x, \|\delta\| \leq 1} f(x + \delta) - f(x)$$
 (adversarial perturbations)

(Riatti Mialon Chan and Mairal 2010) Alberto Bietti

Can we obtain better models through regularization?

- Controlling upper bounds: spectral norm penalties/constraints
- ${\ }$ Ontrolling lower bounds using $\|f\|_{\mathcal H} = \sup_{\|u\|_{\mathcal H} \leq 1} \langle f, u \rangle$
- ullet \implies consider tractable subsets of the unit ball

$$\|f\|_{\mathcal{H}} \ge \sup_{x, \|\delta\| \le 1} f(x + \delta) - f(x)$$
 (adversarial perturbations)
 $\|f\|_{\mathcal{H}} \ge \sup_{x, \|\tau\| \le C} f(L_{\tau}x) - f(x)$ (adversarial deformations)

(Righti Mislon Chan and Mairal 2010) Alberto Bietti

Can we obtain better models through regularization?

- Controlling upper bounds: spectral norm penalties/constraints
- ${\ }$ Ontrolling lower bounds using $\|f\|_{\mathcal H} = \sup_{\|u\|_{\mathcal H} \leq 1} \langle f, u \rangle$
- \implies consider tractable subsets of the unit ball

$$\begin{split} \|f\|_{\mathcal{H}} &\geq \sup_{x, \|\delta\| \leq 1} f(x+\delta) - f(x) \quad (\text{adversarial perturbations}) \\ \|f\|_{\mathcal{H}} &\geq \sup_{x, \|\tau\| \leq C} f(\mathcal{L}_{\tau}x) - f(x) \quad (\text{adversarial deformations}) \\ \|f\|_{\mathcal{H}} &\geq \sup_{x} \|\nabla f(x)\|_2 \quad (\text{gradient penalty}) \end{split}$$

(Riatti Mislan Chan and Mairal 2010)
Regularizing with the RKHS norm in practice

Can we obtain better models through regularization?

- Controlling upper bounds: spectral norm penalties/constraints
- ${\ }$ Ontrolling lower bounds using $\|f\|_{\mathcal H} = \sup_{\|u\|_{\mathcal H} \leq 1} \langle f, u \rangle$
- \implies consider tractable subsets of the unit ball

$$\begin{split} \|f\|_{\mathcal{H}} &\geq \sup_{x, \|\delta\| \leq 1} f(x+\delta) - f(x) \quad (\text{adversarial perturbations}) \\ \|f\|_{\mathcal{H}} &\geq \sup_{x, \|\tau\| \leq C} f(\mathcal{L}_{\tau}x) - f(x) \quad (\text{adversarial deformations}) \\ \|f\|_{\mathcal{H}} &\geq \sup_{x} \|\nabla f(x)\|_2 \quad (\text{gradient penalty}) \end{split}$$

 $\bullet\,$ Best performance by combining upper + lower bound approaches

(Riatti Mialon Chan and Mairal 2010) Alberto Bietti

Regularizing with the RKHS norm in practice

Can we obtain better models through regularization?

Table 2. Regularization on 300 or 1000 examples from MNIST, using deformations from Infinite MNIST. (*) indicates that random deformations were included as training examples, while $||f||_{\tau}^2$ and $||D_{\tau}f||^2$ use them as part of the regularization penalty.

Method	300 VGG	1k VGG
Weight decay	89.32	94.08
SN projection	90.69	95.01
grad_2	93.63	96.67
$ f _{\delta}^2$ penalty	94.17	96.99
$\ \nabla f\ ^2$ penalty	94.08	96.82
Weight decay (*)	92.41	95.64
grad- ℓ_2 (*)	95.05	97.48
$ D_{\tau}f ^2$ penalty	94.18	96.98
$ f _{\tau}^2$ penalty	94.42	97.13
$\ f\ _{\tau}^2 + \ \nabla f\ ^2$	94.75	97.40
$ f _{\tau}^2 + f _{\delta}^2$	95.23	97.66
$ f _{\tau}^2 + f _{\delta}^2 (*)$	95.53	97.56
$ f _{\tau}^{2} + f _{\delta}^{2} + SN \text{ proj}$	95.20	97.60
$ f _{\tau}^2 + f _{\delta}^2 + \text{SN proj}(*)$	95.40	97.77

(Riatti Mislan Chan and Mairal 2010)

Regularizing with the RKHS norm in practice

Can we obtain better models through regularization?

Table 3. Regularization on protein homology detection tasks, with or without data augmentation (DA). Fixed hyperparameters are selected using the first half of the datasets, and we report the average auROC50 score on the second half.

Method	No DA	DA
No weight decay	0.446	0.500
Weight decay	0.501	0.546
SN proj	0.591	0.632
PGD- ℓ_2	0.575	0.595
grad- ℓ_2	0.540	0.552
$\ f\ _{\delta}^2$	0.600	0.608
$\ \nabla f\ ^2$	0.585	0.611
PGD- ℓ_2 + SN proj	0.596	0.627
grad- ℓ_2 + SN proj	0.592	0.624
$ f _{\delta}^2$ + SN proj	0.630	0.644
$\ \nabla f\ ^2$ + SN proj	0.603	0.625

(Riatti Mialan Chan and Mairal 2010)

Regularization for robustness

• Robust optimization yields another lower bound (hinge/logistic loss)

$$\frac{1}{N}\sum_{i=1}^{N}\sup_{\|\delta\|_{2}\leq\epsilon}\ell(y_{i},f(x_{i}+\delta))\leq\frac{1}{N}\sum_{i=1}^{N}\ell(y_{i},f(x_{i}))+\epsilon\|f\|_{\mathcal{H}}$$

- Controlling $||f||_{\mathcal{H}}$ allows a more **global** form of robustness
- Leads to margin bounds for adversarial generalization with ℓ_2 perturbations
 - Using $\|f\|_{\mathcal{H}} \ge \|f\|_{Lip}$ near the margin
- But, may cause a loss in accuracy in practice

(Bietti, Mialon, Chen, and Mairal, 2019)

Regularization for robustness

Robust vs standard accuracy trade-offs



(Bietti, Mialon, Chen, and Mairal, 2019)

Regularization for robustness

Upper vs lower bounds



Deep convolutional representations: conclusions

Study of generic properties

- Deformation stability with small patches, adapted to resolution
- $\, \bullet \,$ Signal preservation when subsampling \leq patch size
- Group invariance by changing patch extraction and pooling

Deep convolutional representations: conclusions

Study of generic properties

- Deformation stability with small patches, adapted to resolution
- $\, \bullet \,$ Signal preservation when subsampling \leq patch size
- Group invariance by changing patch extraction and pooling

Applies to learned models

- Same quantity ||f|| controls stability and complexity:
 - "higher capacity" is needed to discriminate small deformations
 - Learning may be "easier" with stable functions
- Better regularization of generic CNNs using RKHS norm

Deep convolutional representations: conclusions

Study of generic properties

- Deformation stability with small patches, adapted to resolution
- $\, \bullet \,$ Signal preservation when subsampling \leq patch size
- Group invariance by changing patch extraction and pooling

Applies to learned models

- Same quantity ||f|| controls stability and complexity:
 - "higher capacity" is needed to discriminate small deformations
 - ► Learning may be "easier" with stable functions
- Better regularization of generic CNNs using RKHS norm
- Links with optimization (Bietti and Mairal, 2019b)
 - Similar kernel (NTK) arises from optimization in a certain regime
 - Weaker stability guarantees, but better approximation properties

References I

- P. Bartlett, D. J. Foster, and M. Telgarsky. Spectrally-normalized margin bounds for neural networks. In Advances in Neural Information Processing Systems (NIPS), 2017.
- A. Bietti and J. Mairal. Invariance and stability of deep convolutional representations. In Advances in Neural Information Processing Systems (NIPS), 2017.
- A. Bietti and J. Mairal. Group invariance, stability to deformations, and complexity of deep convolutional representations. *Journal of Machine Learning Research (JMLR)*, 20(25):1–49, 2019a.
- A. Bietti and J. Mairal. On the inductive bias of neural tangent kernels. *arXiv preprint* arXiv:1905.12173, 2019b.
- A. Bietti, G. Mialon, D. Chen, and J. Mairal. A kernel perspective for regularizing deep neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2019.
- J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE Transactions on pattern analysis and machine intelligence (PAMI)*, 35(8):1872–1886, 2013.
- T. Cohen and M. Welling. Group equivariant convolutional networks. In *International Conference on Machine Learning (ICML)*, 2016.

References II

- R. Kondor and S. Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2018.
- J. Mairal. End-to-End Kernel Learning with Supervised Convolutional Kernel Networks. In Advances in Neural Information Processing Systems (NIPS), 2016.
- J. Mairal, P. Koniusz, Z. Harchaoui, and C. Schmid. Convolutional kernel networks. In Advances in Neural Information Processing Systems (NIPS), 2014.
- S. Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10): 1331–1398, 2012.
- B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro. A PAC-Bayesian approach to spectrally-normalized margin bounds for neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.
- Y. Zhang, J. D. Lee, and M. I. Jordan. ℓ_1 -regularized neural networks are improperly learnable in polynomial time. In *International Conference on Machine Learning (ICML)*, 2016.
- Y. Zhang, P. Liang, and M. J. Wainwright. Convexified convolutional neural networks. In International Conference on Machine Learning (ICML), 2017.