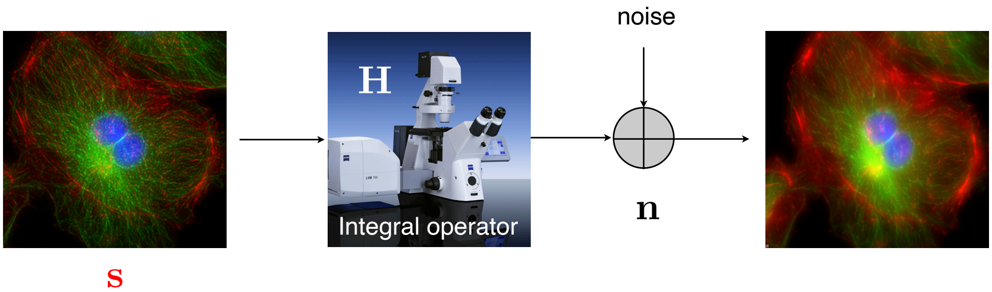
Splines and imaging:
From compressed sensing to deep neural nets

Michael Unser
Biomedical Imaging Group
EPFL, Lausanne, Switzerland

Plenary talk: Int. Conf. Signal Processing and Communications (SPCOM’20), IISc Bangalore, July 20-23, 2020

Variational formulation of inverse problems

- Linear forward model



$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}$$

Problem: recover $\mathbf{s}$ from noisy measurements $\mathbf{y}$

- Reconstruction as an optimization problem

$$\mathbf{s}_{\rm rec} = \arg\min_{\mathbf{s}\in\mathbb{R}^N} \underbrace{\|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2^2}_{\text{data consistency}} + \underbrace{\lambda\|\mathbf{L}\mathbf{s}\|_p^p}_{\text{regularization}}, \quad p = 1, 2$$

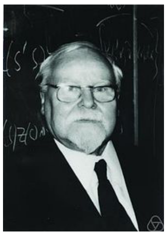
Linear inverse problems (20th century theory)

- Dealing with **ill-posed problems**: Tikhonov **regularization**

$\mathcal{R}(\mathbf{s}) = \|\mathbf{L}\mathbf{s}\|_2^2$: regularization (or smoothness) functional

$\mathbf{L}$: regularization operator (i.e., Gradient)

$$\min_{\mathbf{s}} \mathcal{R}(\mathbf{s}) \quad \text{subject to} \quad \|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2^2 \le \sigma^2$$



Andrey N. Tikhonov (1906-1993)

- Equivalent variational problem

$$\mathbf{s}^\star = \arg\min \underbrace{\|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2^2}_{\text{data consistency}} + \underbrace{\lambda \|\mathbf{L}\mathbf{s}\|_2^2}_{\text{regularization}}$$

Formal linear solution: $\mathbf{s} = (\mathbf{H}^T\mathbf{H} + \lambda \mathbf{L}^T\mathbf{L})^{-1}\mathbf{H}^T\mathbf{y} = \mathbf{R}_\lambda \cdot \mathbf{y}$

Interpretation: “filtered” backprojection

Learning as a (linear) inverse problem

but an infinite-dimensional one ...

Given the data points $(\boldsymbol{x}_m, y_m) \in \mathbb{R}^{N+1}$, find $f : \mathbb{R}^N \to \mathbb{R}$ s.t. $f(\boldsymbol{x}_m) \approx y_m$ for $m = 1, \dots, M$

- Introduce smoothness or **regularization** constraint (Poggio-Girosi 1990)

$R(f) = \|f\|_{\mathcal{H}}^2 = \|\mathrm{L}f\|_{L_2}^2 = \int_{\mathbb{R}^N} |\mathrm{L}f(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x}$: regularization functional

$$\min_{f \in \mathcal{H}} R(f) \quad \text{subject to} \quad \sum_{m=1}^{M} |y_m - f(\boldsymbol{x}_m)|^2 \le \sigma^2$$

- Regularized least-squares fit (theory of RKHS)

$$f_{\mathrm{RKHS}} = \arg\min_{f \in \mathcal{H}} \left(\sum_{m=1}^{M} |y_m - f(\boldsymbol{x}_m)|^2 + \lambda \|f\|_{\mathcal{H}}^2 \right)$$

$\Rightarrow$ kernel estimator

(Wahba 1990; Schölkopf 2001)

OUTLINE

- **Introduction** ✔
  - Image reconstruction as an inverse problem
  - Learning as an inverse problem
- **Continuous-domain theory of sparsity**
  - Splines and operators
  - gTV regularization: representer theorem for CS
- **From compressed sensing to deep neural networks**
  - Unrolling forward/backward iterations: FBPConv
- **Deep neural networks vs. deep splines**
  - Continuous piecewise linear (CPWL) functions / splines
  - New representer theorem for deep neural networks

Part I: Continuous-domain theory of sparsity

L_1 splines

(Fisher-Jerome 1975)

gTV optimality of splines for inverse problems

(U.-Fageot-Ward, *SIAM Review* 2017)

Splines are analog, but intrinsically sparse

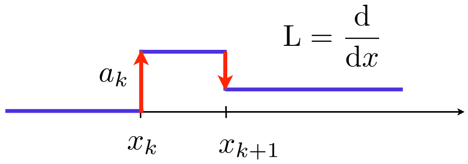
$\mathrm{L}\{\cdot\}$: differential operator (translation-invariant)

δ: Dirac distribution

Definition

The function $s : \mathbb{R}^d \to \mathbb{R}$ (possibly of slow growth) is a **nonuniform** L**-spline** with **knots** $\{\boldsymbol{x}_k\}_{k \in S}$

$$\Leftrightarrow \qquad \mathrm{L}s = \sum_{k \in S} a_k \delta(\cdot - \boldsymbol{x}_k) \; = w \; : \text{ spline's innovation}$$



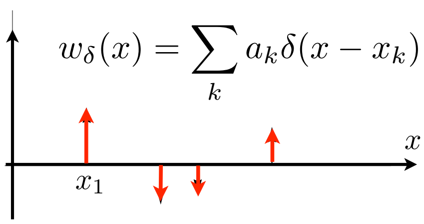
Spline theory: (Schultz-Varga, 1967)

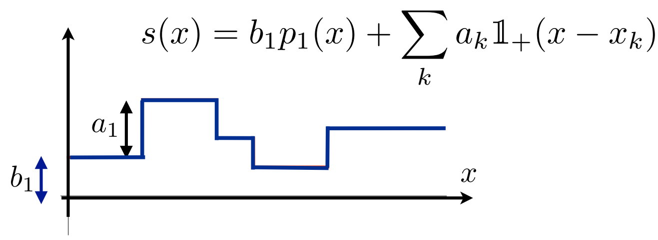
Spline synthesis: example

$$\mathrm{L} = \mathrm{D} = \frac{\mathrm{d}}{\mathrm{d}x}$$

Null space: $\mathcal{N}_{\mathrm{D}} = \mathrm{span}\{p_1\}, \quad p_1(x) = 1$

$\rho_{\mathrm{D}}(x) = \mathrm{D}^{-1}\{\delta\}(x) = \mathbb{1}_+(x)$: Heaviside function





Spline synthesis: generalization

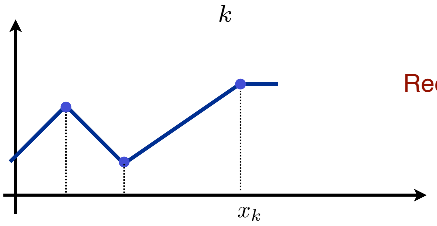
L: **spline-admissible** operator (LSI)

Finite-dimensional null space: $\mathcal{N}_{\mathrm{L}} = \mathrm{span}\{p_n\}_{n=1}^{N_0}$

Green's function of L: $\rho_{\mathrm{L}}(\boldsymbol{x}) = \mathrm{L}^{-1}\{\delta\}(\boldsymbol{x})$

Spline's innovation: $w_\delta(\boldsymbol{x}) = \sum_k a_k \delta(\boldsymbol{x} - \boldsymbol{x}_k)$

$$\Rightarrow \quad s(\boldsymbol{x}) = \sum_k a_k \rho_{\mathrm{L}}(\boldsymbol{x} - \boldsymbol{x}_k) + \sum_{n=1}^{N_0} b_n p_n(\boldsymbol{x})$$



Requires specification of boundary conditions

Proper continuous counterpart of $\ell_1(\mathbb{Z}^d)$

$\mathcal{S}(\mathbb{R}^d)$: Schwartz's space of smooth and rapidly decaying test functions on $\mathbb{R}^d$

$\mathcal{S}'(\mathbb{R}^d)$: Schwartz's space of tempered distributions

- Space of real-valued **bounded Radon measures** on $\mathbb{R}^d$

$$\mathcal{M}(\mathbb{R}^d) = \big(C_0(\mathbb{R}^d)\big)' = \Big\{w \in \mathcal{S}'(\mathbb{R}^d) : \|w\|_{\mathcal{M}} = \sup_{\varphi \in \mathcal{S}(\mathbb{R}^d): \|\varphi\|_\infty = 1} \langle w, \varphi\rangle < \infty\Big\},$$

where $w : \varphi \mapsto \langle w, \varphi\rangle \triangleq \int_{\mathbb{R}^d} \varphi(\boldsymbol{r}) w(\boldsymbol{r}) \mathrm{d}\boldsymbol{r}$

- Basic inclusions
  - $\forall f \in L_1(\mathbb{R}^d):\ \|f\|_{\mathcal{M}} = \|f\|_{L_1(\mathbb{R}^d)} \quad \Rightarrow \quad L_1(\mathbb{R}^d) \subseteq \mathcal{M}(\mathbb{R}^d)$
  - $\delta(\cdot - \boldsymbol{x}_0) \in \mathcal{M}(\mathbb{R}^d)$ with $\|\delta(\cdot - \boldsymbol{x}_0)\|_{\mathcal{M}} = 1$ for any $\boldsymbol{x}_0 \in \mathbb{R}^d$

Representer theorem for gTV regularization

- L: spline-admissible operator with null space $\mathcal{N}_{\mathrm{L}} = \mathrm{span}\{p_n\}_{n=1}^{N_0}$
- gTV semi-norm: $\|\mathrm{L}\{s\}\|_{\mathcal{M}} = \sup_{\|\varphi\|_\infty \le 1} \langle \mathrm{L}\{s\}, \varphi\rangle$
- Measurement functionals $h_m : \mathcal{M}_{\mathrm{L}}(\mathbb{R}^d) \to \mathbb{R}$ (weak$*$-continuous)

$$\mathcal{M}_{\mathrm{L}}(\mathbb{R}^d) = \{f \in \mathcal{S}'(\mathbb{R}^d) : \|\mathrm{L}f\|_{\mathcal{M}} < \infty\}$$

(P1) $$\arg \min_{f \in \mathcal{M}_{\mathrm{L}}(\mathbb{R}^d)} \left(\sum_{m=1}^{M} |y_m - \langle h_m, f\rangle|^2 + \lambda \|\mathrm{L}f\|_{\mathcal{M}} \right)$$

Convex loss function: $F : \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}$

$\boldsymbol{\nu} : \mathcal{M}_{\mathrm{L}} \to \mathbb{R}^M$ with $\boldsymbol{\nu}(f) = \big(\langle h_1, f\rangle, \dots, \langle h_M, f\rangle\big)$

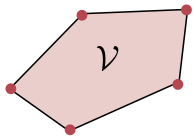
(P1') $$\arg \min_{f \in \mathcal{M}_{\mathrm{L}}(\mathbb{R}^d)} \big(F\big(\boldsymbol{y}, \boldsymbol{\nu}(f)\big) + \lambda \|\mathrm{L}f\|_{\mathcal{M}} \big)$$

Representer theorem for gTV-regularization

The extreme points of (P1') are **non-uniform** L**-spline** of the form

$$f_{\mathrm{spline}}(\boldsymbol{x}) = \sum_{k=1}^{K_{\mathrm{knots}}} a_k \rho_{\mathrm{L}}(\boldsymbol{x} - \boldsymbol{x}_k) + \sum_{n=1}^{N_0} b_n p_n(\boldsymbol{x})$$

with ρ_{L} such that $\mathrm{L}\{\rho_{\mathrm{L}}\} = \delta$, $K_{\mathrm{knots}} \le M - N_0$, and $\|\mathrm{L}f_{\mathrm{spline}}\|_{\mathcal{M}} = \|\mathbf{a}\|_{\ell_1}$.



(U.-Fageot-Ward, *SIAM Review* 2017)

Example: 1D inverse problem with TV(2) regularization

$$s_{\mathrm{spline}} = \arg \min_{s \in \mathcal{M}_{\mathrm{D}^2}(\mathbb{R})} \left(\sum_{m=1}^{M} |y_m - \langle h_m, s\rangle|^2 + \lambda \mathrm{TV}^{(2)}(s) \right)$$

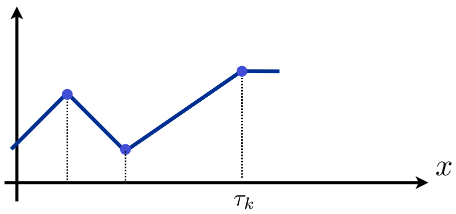
- Total 2nd-variation: $\mathrm{TV}^{(2)}(s) = \sup_{\|\varphi\|_\infty \le 1} \langle \mathrm{D}^2 s, \varphi\rangle = \|\mathrm{D}^2 s\|_{\mathcal{M}}$

$\mathrm{L} = \mathrm{D}^2 = \dfrac{\mathrm{d}^2}{\mathrm{d}x^2}$ $\quad \rho_{\mathrm{D}^2}(x) = (x)_+$: ReLU $\quad \mathcal{N}_{\mathrm{D}^2} = \mathrm{span}\{1, x\}$

- Generic form of the solution

$$s_{\mathrm{spline}}(x) = b_1 + b_2 x + \sum_{k=1}^{K} a_k (x - \tau_k)_+$$

no penalty



with $K < M$ and free parameters b_1, b_2 and $(a_k, \tau_k)_{k=1}^{K}$

Other spline-admissible operators

- $\mathrm{L} = \mathrm{D}^n$ (pure derivatives) (Schoenberg 1946)
  $\Rightarrow$ polynomial splines of degree $(n-1)$
- $\mathrm{L} = \mathrm{D}^n + a_{n-1}\mathrm{D}^{n-1} + \cdots + a_0\mathrm{I}$ (ordinary differential operator) (Dahmen-Micchelli 1987)
  $\Rightarrow$ exponential splines
- Fractional derivatives: $\mathrm{L} = \mathrm{D}^\gamma \quad \overset{\mathcal{F}}{\longleftrightarrow} \quad (\mathrm{j}\omega)^\gamma$ (U.-Blu 2000)
  $\Rightarrow$ fractional splines
- Fractional Laplacian: $(-\Delta)^{\frac{\gamma}{2}} \quad \overset{\mathcal{F}}{\longleftrightarrow} \quad \|\boldsymbol{\omega}\|^\gamma$ (Duchon 1977)
  $\Rightarrow$ polyharmonic splines
- Elliptical differential operators; e.g, $\mathrm{L} = (-\Delta + \alpha\mathrm{I})^\gamma$ (Ward-U. 2014)
  $\Rightarrow$ Sobolev splines

Recovery with sparsity constraints: discretization

- Constrained optimization formulation

  Auxiliary **innovation** variable: $\mathbf{u} = \mathbf{L}\mathbf{s}$

$$\mathbf{s}_{\mathrm{sparse}} = \arg\min_{\mathbf{s}\in\mathbb{R}^N} \left(\frac{1}{2}\|\mathbf{y}-\mathbf{H}\mathbf{s}\|_2^2 + \lambda\|\mathbf{u}\|_1\right) \text{ subject to } \mathbf{u} = \mathbf{L}\mathbf{s}$$

- Augmented Lagrangian method

  Quadratic penalty term: $\frac{\mu}{2}\|\mathbf{L}\mathbf{s}-\mathbf{u}\|_2^2$

  Lagrange multipler vector: $\boldsymbol{\alpha}$

$$\mathcal{L}_{\mathcal{A}}(\mathbf{s},\mathbf{u},\boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{y}-\mathbf{H}\mathbf{s}\|_2^2 + \lambda\sum_n |[\mathbf{u}]_n| + \boldsymbol{\alpha}^T(\mathbf{L}\mathbf{s}-\mathbf{u}) + \frac{\mu}{2}\|\mathbf{L}\mathbf{s}-\mathbf{u}\|_2^2$$

(Ramani-Fessler, *IEEE TMI* 2011)

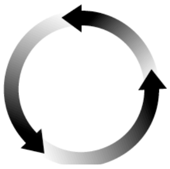
Discretization: compatible with CS paradigm

$$\mathbf{s}_{\text{sparse}} = \arg\min_{\mathbf{s}\in\mathbb{R}^K} \left(\frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2^2 + \lambda\|\mathbf{u}\|_1 \right) \text{ subject to } \mathbf{u} = \mathbf{L}\mathbf{s}$$

$$\mathcal{L}_{\mathcal{A}}(\mathbf{s}, \mathbf{u}, \boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{s}\|_2^2 + \lambda \sum_n |[\mathbf{u}]_n| + \boldsymbol{\alpha}^T(\mathbf{L}\mathbf{s} - \mathbf{u}) + \frac{\mu}{2}\|\mathbf{L}\mathbf{s} - \mathbf{u}\|_2^2$$

ADMM algorithm

For $k = 0, \dots, K$



Linear step

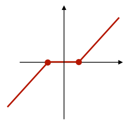
$$\mathbf{s}^{k+1} = \left(\mathbf{H}^T\mathbf{H} + \mu\mathbf{L}^T\mathbf{L}\right)^{-1}\left(\mathbf{z}_0 + \mathbf{z}^{k+1}\right)$$

with $\mathbf{z}^{k+1} = \mathbf{L}^T\left(\mu\mathbf{u}^k - \boldsymbol{\alpha}^k\right)$

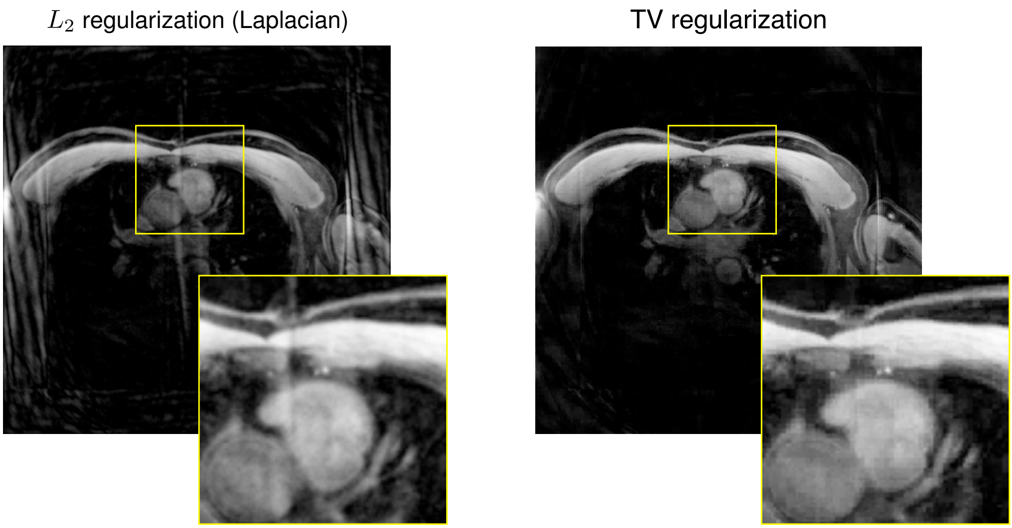
$$\boldsymbol{\alpha}^{k+1} = \boldsymbol{\alpha}^k + \mu\left(\mathbf{L}\mathbf{s}^{k+1} - \mathbf{u}^k\right)$$

Proximal step = pointwise non-linearity

$$\mathbf{u}^{k+1} = \text{prox}_{|\cdot|}\left(\mathbf{L}\mathbf{s}^{k+1} + \tfrac{1}{\mu}\boldsymbol{\alpha}^{k+1}; \tfrac{\lambda}{\mu}\right)$$



Example: ISMRM reconstruction challenge



M. Guerquin-Kern, M. Häberlin, K.P. Pruessmann, M. Unser, *IEEE Trans. Medical Imaging*, 2011.

OUTLINE

- Introduction ✔
- Continuous-domain theory of sparsity ✔
- **From compressed sensing to deep neural networks**
  - Unrolling forward/backward iterations: FBPConv
- **Deep neural networks vs. deep splines**
  - Continuous piecewise linear (CPWL) functions / splines
  - New representer theorem for deep neural networks

Discretization: compatible with CS paradigm

$$\mathbf{s}_{\mathrm{sparse}} = \arg\min_{\mathbf{s}\in\mathbb{R}^K} \left(\frac{1}{2}\|\mathbf{y}-\mathbf{H}\mathbf{s}\|_2^2 + \lambda\|\mathbf{u}\|_1 \right) \quad \text{subject to } \mathbf{u} = \mathbf{L}\mathbf{s}$$

$$\mathcal{L}_{\mathcal{A}}(\mathbf{s},\mathbf{u},\boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{y}-\mathbf{H}\mathbf{s}\|_2^2 + \lambda\sum_n |[\mathbf{u}]_n| + \boldsymbol{\alpha}^T(\mathbf{L}\mathbf{s}-\mathbf{u}) + \frac{\mu}{2}\|\mathbf{L}\mathbf{s}-\mathbf{u}\|_2^2$$

ADMM algorithm

For $k = 0, \dots, K$

Linear step

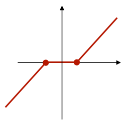
$$\mathbf{s}^{k+1} = \left(\mathbf{H}^T\mathbf{H} + \mu\mathbf{L}^T\mathbf{L}\right)^{-1}\left(\mathbf{z}_0 + \mathbf{z}^{k+1}\right)$$

with $\mathbf{z}^{k+1} = \mathbf{L}^T\left(\mu\mathbf{u}^k - \boldsymbol{\alpha}^k\right)$

$$\boldsymbol{\alpha}^{k+1} = \boldsymbol{\alpha}^k + \mu\left(\mathbf{L}\mathbf{s}^{k+1} - \mathbf{u}^k\right)$$

Proximal step = pointwise non-linearity

$$\mathbf{u}^{k+1} = \mathrm{prox}_{|\cdot|}\left(\mathbf{L}\mathbf{s}^{k+1} + \tfrac{1}{\mu}\boldsymbol{\alpha}^{k+1}; \tfrac{\lambda}{\mu}\right)$$



Identification of convolution operators

Normal matrix: $\mathbf{A} = \mathbf{H}^T\mathbf{H}$ (symmetric)

Generic linear solver: $\mathbf{s} = (\mathbf{A} + \lambda\mathbf{L}^T\mathbf{L})^{-1}\mathbf{H}^T\mathbf{y} = \mathbf{R}_\lambda \cdot \mathbf{y}$

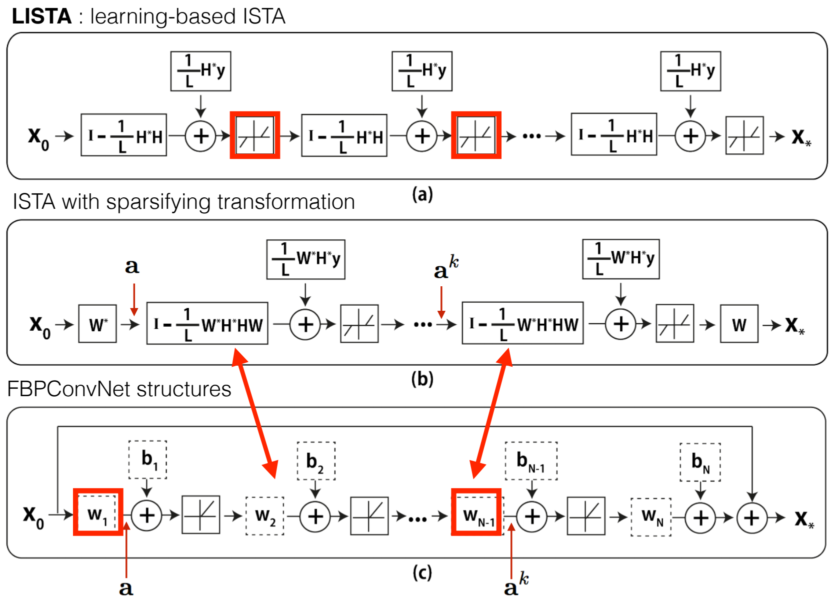
- Recognizing structured matrices
  - $\mathbf{L}$: convolution matrix $\Rightarrow$ $\mathbf{L}^T\mathbf{L}$: symmetric convolution matrix
  - $\mathbf{L}$, $\mathbf{A}$: convolution matrices $\Rightarrow$ $(\mathbf{A} + \lambda\mathbf{L}^T\mathbf{L})$: symmetric convolution matrix
  - Applicable to

    - deconvolution microscopy (Wiener filter)
    - parallel rays computer tomography (FBP)
    - MRI, including non-uniform sampling of *k*-space

- Justification for use of convolution neural nets (CNN) (see Theorem 1, Jin et al., *IEEE TIP* 2017)

Connection with deep neural networks

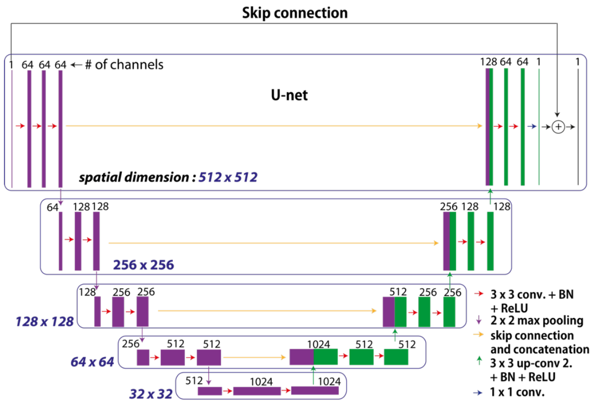
Unrolled Iterative Shrinkage Thresholding Algorithm (ISTA) (Gregor-LeCun 2010)



Recent advent of Deep ConvNets

(Jin et al. 2016; Adler-Öktem 2017; Chen et al. 2017; ...)

- CT reconstruction based on Deep ConvNets
  - Input: Sparse view FBP reconstruction
  - Training: Set of 500 high-quality full-view CT reconstructions
  - Architecture: U-Net with skip connection

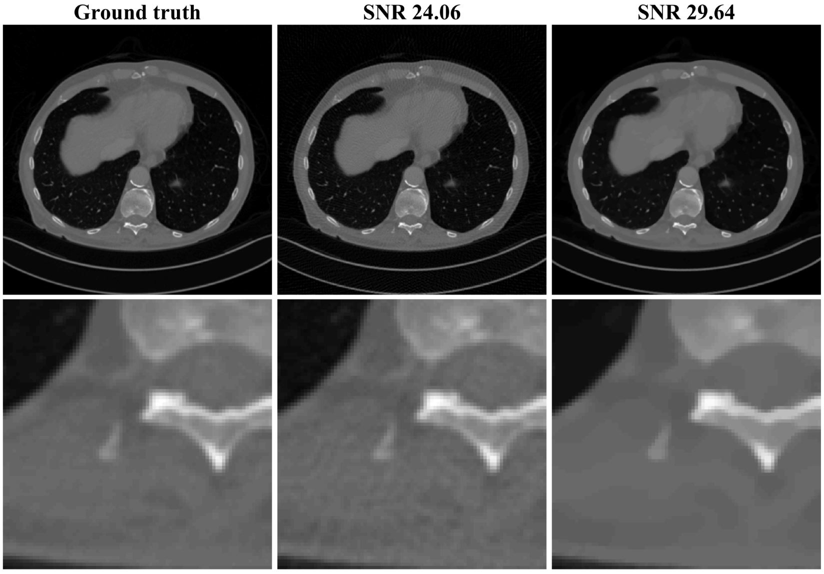


(Jin et al., IEEE TIP 2017)

X-ray computer tomography data

Dose reduction by 7: 143 views

Ground truth	FBP SNR 24.06	TV SNR 29.64



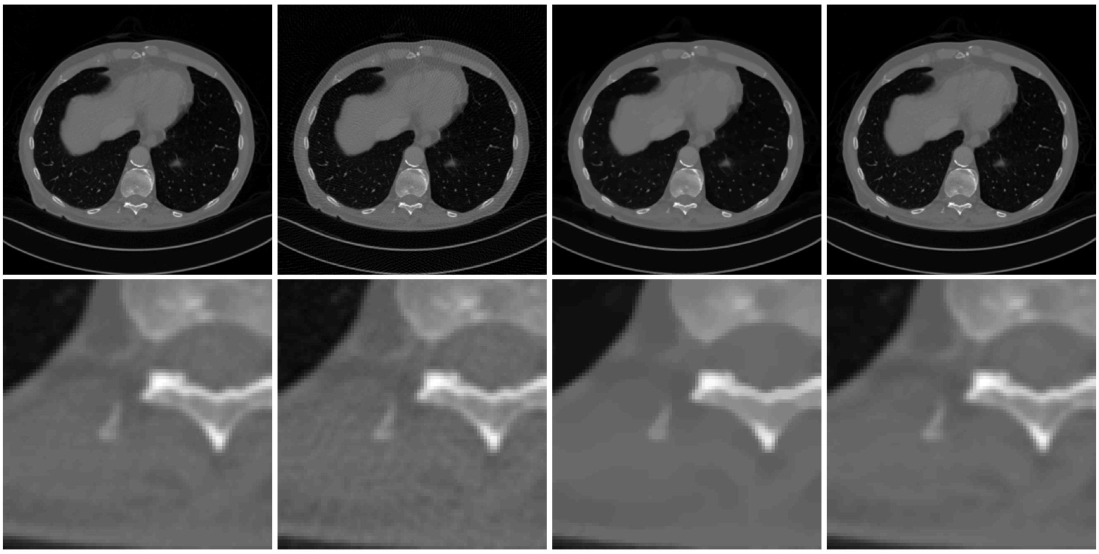
Reconstructed from from 1000 views

MAYO CLINIC

X-ray computer tomography data

Dose reduction by 7: 143 views

Ground truth	FBP SNR 24.06	TV SNR 29.64	FBPConvNet SNR 35.38



Reconstructed from
from 1000 views

MAYO CLINIC

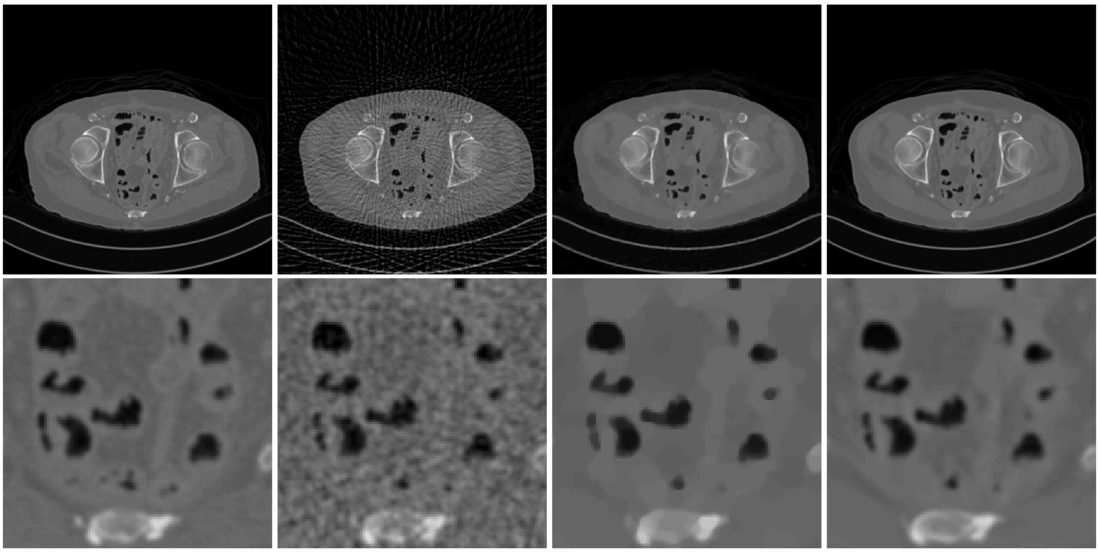
(Jin et al, *IEEE Trans. Im Proc.*, 2017)

IEEE
2019 Best Paper Award

X-ray computer tomography data

Dose reduction by 20: 50 views

Ground truth	FBP SNR 13.43	TV SNR 24.89	FBPConvNet SNR 28.53



Reconstructed from
from 1000 views

MAYO CLINIC

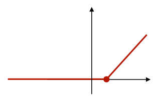
(Jin-McCann-Froustey-Unser, *IEEE Trans. Im Proc.*, 2017)

OUTLINE

- Introduction ✔
- Continuous-domain theory of sparsity ✔
- From compressed sensing to deep neural networks ✔
- **Deep neural networks vs. deep splines**
  - Background
  - Continuous piecewise linear (CPWL) functions / splines
  - New representer theorem for deep neural networks

Deep neural networks and splines

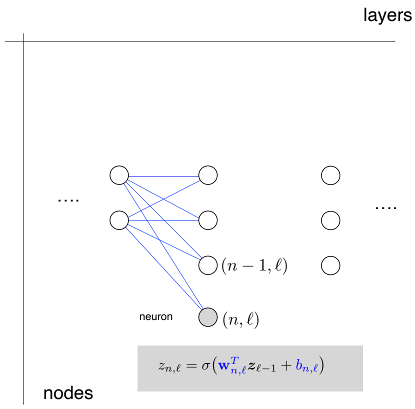
$\mathrm{ReLU}(x;b) = (x-b)_+$



- Preferred choice of activation function: ReLU
  - ReLU works nicely with dropout / ℓ_1-regularization (Glorot *ICAIS* 2011)
  - Networks with hidden ReLU are easier to train
  - State-of-the-art performance (LeCun-Bengio-Hinton *Nature* 2015)
- Deep nets as Continuous PieceWise-Linear maps
  - ReLU $\Rightarrow$ CPWL (Montufar *NIPS* 2014)
  - CPWL $\Rightarrow$ Deep ReLU network (Strang *SIAM News 2018*)
- Deep ReLU nets = hierarchical splines
  - ReLU is a piecewise-linear spline (Poggio-Rosasco 2015)

Feedforward deep neural network

- Layers: $\ell = 1, \ldots, L$
- Deep structure descriptor: $(N_0, N_1, \cdots, N_L)$
- Neuron or node index: $(n, \ell), \;\; n = 1, \cdots, N_\ell$
- Activation function: $\sigma : \mathbb{R} \to \mathbb{R}$ (ReLU)

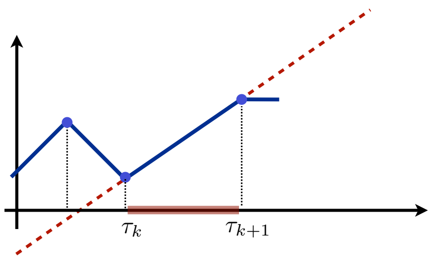


- Linear step: $\mathbb{R}^{N_{\ell-1}} \to \mathbb{R}^{N_\ell}$
  $\boldsymbol{f}_\ell : \boldsymbol{x} \mapsto \boldsymbol{f}_\ell(\boldsymbol{x}) = \mathbf{W}_\ell \boldsymbol{x} + \mathbf{b}_\ell$
- Nonlinear step: $\mathbb{R}^{N_\ell} \to \mathbb{R}^{N_\ell}$
  $\boldsymbol{\sigma}_\ell : \boldsymbol{x} \mapsto \boldsymbol{\sigma}_\ell(\boldsymbol{x}) = \big(\sigma(x_1), \ldots, \sigma(x_{N_\ell})\big)$

Learned

$$\mathbf{f}_{\text{deep}}(\boldsymbol{x}) = (\boldsymbol{\sigma}_L \circ \boldsymbol{f}_L \circ \boldsymbol{\sigma}_{L-1} \circ \cdots \circ \boldsymbol{\sigma}_2 \circ \boldsymbol{f}_2 \circ \boldsymbol{\sigma}_1 \circ \boldsymbol{f}_1)(\boldsymbol{x})$$

Continuous-PieceWise Linear (CPWL) functions



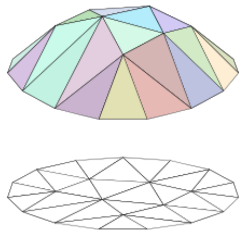
- 1D: Non-uniform spline de degree 1

Partition: $\mathbb{R} = \bigcup_{k=0}^{K} P_k$ with $P_k = [\tau_k, \tau_{k+1})$, $\tau_0 = -\infty < \tau_1 < \cdots < \tau_K < \tau_{K+1} = +\infty$.

The function $f_{\text{spline}} : \mathbb{R} \to \mathbb{R}$ is a piecewise-linear spline with knots $\tau_1, \ldots, \tau_K$ if

- (i) : f_{spline} is continuous $\mathbb{R} \to \mathbb{R}$
- (ii) : for $x \in P_k$: $f_{\text{spline}}(x) = f_k(x) \triangleq a_k x + b_k$ with $(a_k, b_k) \in \mathbb{R}^2$, $k = 0, \ldots, K$
- $f_{\text{spline}}(x) = \tilde{b}_0 + \tilde{b}_1 x + \sum_{k=1}^{K} \tilde{a}_k (x - \tau_k)_+$ with $\tilde{b}_0, \tilde{b}_1 \in \mathbb{R}$, $(\tilde{a}_k) \in \mathbb{R}^K$.

CPWL functions in high dimensions



Multidimensional generalization

Partition of domain into a finite number of non-overlapping **convex polytopes**; i.e.,

$\mathbb{R}^N = \bigcup_{k=1}^{K} P_k$ with $\mu(P_{k_1} \cap P_{k_2}) = 0$ for all $k_1 \neq k_2$

The function $f_{\mathrm{CPWL}} : \mathbb{R}^N \to \mathbb{R}$ is **continuous piecewise-linear** with partition $P_1, \dots, P_K$

- (i) : f_{CPWL} is continuous $\mathbb{R}^N \to \mathbb{R}$
- (ii) : for $\boldsymbol{x} \in P_k$: $f_{\mathrm{CPWL}}(\boldsymbol{x}) = f_k(\boldsymbol{x}) \triangleq \mathbf{a}_k^T \boldsymbol{x} + b_k$ with $\mathbf{a}_k \in \mathbb{R}^N, b_k \in \mathbb{R}$, $k = 1, \dots, K$

The vector-valued function $\mathbf{f}_{\mathrm{CPWL}} = (f_1, \dots, f_M) : \mathbb{R}^N \to \mathbb{R}^M$ is a CPWL
if each component function $f_m : \mathbb{R}^N \to \mathbb{R}$ is CPWL.

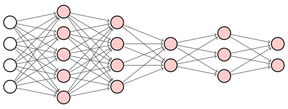
Algebra of CPWL functions

- any linear combination of (vector-valued) CPWL functions $\mathbb{R}^N \to \mathbb{R}^{N'}$ is CPWL, and,
- the composition $\mathbf{f}_2 \circ \mathbf{f}_1$ of any two CPWL functions with compatible domain and range—i.e., $\mathbf{f}_2 : \mathbb{R}^{N_1} \to \mathbb{R}^{N_2}$ and $\mathbf{f}_1 : \mathbb{R}^{N_0} \to \mathbb{R}^{N_1}$—is CPWL $\mathbb{R}^{N_0} \to \mathbb{R}^{N_2}$.

  Sketch of proof: The continuity property is preserved through composition. The composition of two affine transforms is an affine transform, including the scenari where the domain is partitioned.

- The max (resp. min) pooling of two (or more) CPWL functions is CPWL.

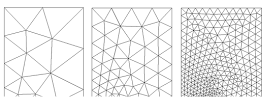
Implication for deep ReLU neural networks



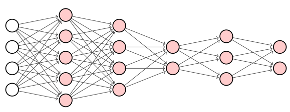
$$\mathbf{f}_{\text{deep}}(\boldsymbol{x}) = (\boldsymbol{\sigma}_L \circ \boldsymbol{f}_L \circ \boldsymbol{\sigma}_{L-1} \circ \cdots \circ \boldsymbol{\sigma}_2 \circ \boldsymbol{f}_2 \circ \boldsymbol{\sigma}_1 \circ \boldsymbol{f}_1)(\boldsymbol{x})$$

- Each scalar neuron activation, $\sigma_{n,\ell}(x) = \mathrm{ReLU}(x)$, is CPWL.
- Each layer function $\boldsymbol{\sigma}_\ell \circ \boldsymbol{f}_\ell(\boldsymbol{x}) = (\mathbf{W}_\ell \boldsymbol{x} + \mathbf{b}_\ell)_+$ is CPWL
- The whole feedforward network $\mathbf{f}_{\text{deep}} : \mathbb{R}^{N_0} \to \mathbb{R}^{N_L}$ is CPWL
- This holds true as well for deep architectures that involve Max pooling for dimension reduction
- The CPWL also remains valid for more complicated neuronal responses as long as they are CPWL; that is, **linear splines**.

CPWL functions: further properties



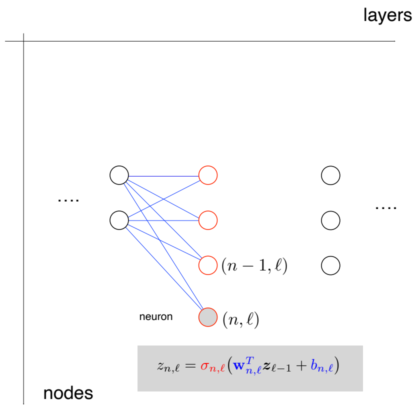
- The CPWL model has universal approximation properties (as one increases the number of regions)
- Any CPWL function $\mathbb{R}^N \to \mathbb{R}$ can be implement via a deep ReLU network with no more than $\log_2(N+1)+1$ layers



(Arora ICLR 2018)

Refinement: free-form activation functions

- Layers: $\ell = 1, \dots, L$
- Deep structure descriptor: $(N_0, N_1, \cdots, N_L)$
- Neuron or node index: $(n, \ell), \ \ n = 1, \cdots, N_\ell$
- Activation function: $\sigma : \mathbb{R} \to \mathbb{R}$ (ReLU)
- Linear step: $\mathbb{R}^{N_{\ell-1}} \to \mathbb{R}^{N_\ell}$
  $\boldsymbol{f}_\ell : \boldsymbol{x} \mapsto \boldsymbol{f}_\ell(\boldsymbol{x}) = \mathbf{W}_\ell \boldsymbol{x} + \mathbf{b}_\ell$
- Nonlinear step: $\mathbb{R}^{N_\ell} \to \mathbb{R}^{N_\ell}$
  $\boldsymbol{\sigma}_\ell : \boldsymbol{x} \mapsto \boldsymbol{\sigma}_\ell(\boldsymbol{x}) = \big(\sigma_{n,\ell}(x_1), \dots, \sigma_{N_\ell,\ell}(x_{N_\ell})\big)$



$$\mathbf{f}_{\text{deep}}(\boldsymbol{x}) = (\boldsymbol{\sigma}_L \circ \boldsymbol{f}_L \circ \boldsymbol{\sigma}_{L-1} \circ \cdots \circ \boldsymbol{\sigma}_2 \circ \boldsymbol{f}_2 \circ \boldsymbol{\sigma}_1 \circ \boldsymbol{f}_1)(\boldsymbol{x})$$

Joint learning / training ?

Constraining activation functions

- Regularization functional
  - Should not penalize simple solutions (e.g., identity or linear scaling)
  - Should impose diffentiability (for DNN to be trainable via backpropagation)
  - Should favor simplest CPWL solutions; i.e., with “sparse 2nd derivatives”
- Second total-variation of $\sigma : \mathbb{R} \to \mathbb{R}$

$$\mathrm{TV}^{(2)}(\sigma) \triangleq \|\mathrm{D}^2 \sigma\|_{\mathcal{M}} = \sup_{\varphi \in \mathcal{S}(\mathbb{R}) : \|\varphi\|_\infty \le 1} \langle \mathrm{D}^2 \sigma, \varphi \rangle$$

- Native space for $\big(\mathcal{M}(\mathbb{R}), \mathrm{D}^2\big)$

$$\mathrm{BV}^{(2)}(\mathbb{R}) = \{f : \mathbb{R} \to \mathbb{R} : \|\mathrm{D}^2 f\|_{\mathcal{M}} < \infty\}$$

Representer theorem for deep neural networks

Theorem ($\mathrm{TV}^{(2)}$-optimality of deep spline networks) (U. JMLR 2019)

- neural network $\mathbf{f}: \mathbb{R}^{N_0} \to \mathbb{R}^{N_L}$ with **deep structure** $(N_0, N_1, \dots, N_L)$
  $\boldsymbol{x} \mapsto \mathbf{f}(\boldsymbol{x}) = (\boldsymbol{\sigma}_L \circ \boldsymbol{\ell}_L \circ \boldsymbol{\sigma}_{L-1} \circ \cdots \circ \boldsymbol{\ell}_2 \circ \boldsymbol{\sigma}_1 \circ \boldsymbol{\ell}_1)(\boldsymbol{x})$
- **normalized** linear transformations $\boldsymbol{\ell}_\ell : \mathbb{R}^{N_{\ell-1}} \to \mathbb{R}^{N_\ell}, \boldsymbol{x} \mapsto \mathbf{U}_\ell \boldsymbol{x}$ with weights $\mathbf{U}_\ell = [\mathbf{u}_{1,\ell} \ \cdots \ \mathbf{u}_{N_\ell,\ell}]^T \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$ such that $\|\mathbf{u}_{n,\ell}\| = 1$
- **free-form** activations $\boldsymbol{\sigma}_\ell = (\sigma_{1,\ell}, \dots, \sigma_{N_\ell,\ell}) : \mathbb{R}^{N_\ell} \to \mathbb{R}^{N_\ell}$ with $\sigma_{1,\ell}, \dots, \sigma_{N_\ell,\ell} \in \mathrm{BV}^{(2)}(\mathbb{R})$

Given a series data points $(\boldsymbol{x}_m, \boldsymbol{y}_m)$ $m = 1, \dots, M$, we then define the training problem

$$\arg \min_{(\mathbf{U}_\ell), (\sigma_{n,\ell} \in \mathrm{BV}^{(2)}(\mathbb{R}))} \left(\sum_{m=1}^{M} E(\boldsymbol{y}_m, \mathbf{f}(\boldsymbol{x}_m)) + \mu \sum_{\ell=1}^{N} R_\ell(\mathbf{U}_\ell) + \lambda \sum_{\ell=1}^{L} \sum_{n=1}^{N_\ell} \mathrm{TV}^{(2)}(\sigma_{n,\ell}) \right) \quad (1)$$

- $E : \mathbb{R}^{N_L} \times \mathbb{R}^{N_L} \to \mathbb{R}^+$: arbitrary convex error function
- $R_\ell : \mathbb{R}^{N_\ell \times N_{\ell-1}} \to \mathbb{R}^+$: convex cost

If solution of (1) exists, then it is achieved by a **deep spline network** with activations of the form

$$\sigma_{n,\ell}(x) = b_{1,n,\ell} + b_{2,n,\ell} x + \sum_{k=1}^{K_{n,\ell}} a_{k,n,\ell}(x - \tau_{k,n,\ell})_+,$$

with adaptive parameters $K_{n,\ell} \le M - 2$, $\tau_{1,n,\ell}, \dots, \tau_{K_{n,\ell},n,\ell} \in \mathbb{R}$, and $b_{1,n,\ell}, b_{2,n,\ell}, a_{1,n,\ell}, \dots, a_{K_{n,\ell},n,\ell} \in \mathbb{R}$.

Outcome of representer theorem

Each neuron (fixed index (n, ℓ)) is characterized by

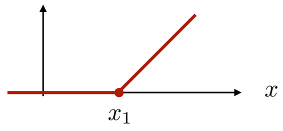
- its number $0 \le K_{n,\ell}$ of knots (ideally, much smaller than M);
- the location $\{\tau_k = \tau_{k,n,\ell}\}_{k=1}^{K_{n,\ell}}$ of these knots (ReLU biases);
- the expansion coefficients $\mathbf{b}_{n,\ell} = (b_{1,n,\ell}, b_{2,n,\ell}) \in \mathbb{R}^2$,
  $\boldsymbol{a}_{n,\ell} = (a_{1,n,\ell}, \dots, a_{K,n,\ell}) \in \mathbb{R}^K$.

These parameters (including the number of knots) are **data-dependent** and adjusted automatically during training.

- Link with ℓ_1 minimization techniques

$$\mathrm{TV}^{(2)}\{\sigma_{n,\ell}\} = \sum_{k=1}^{K_{n,\ell}} |a_{k,n,\ell}| = \|\mathbf{a}_{n,\ell}\|_1$$

Deep spline networks: Discussion



- Global optimality achieved with **spline activations**

- Justification of popular schemes / Backward compatibility

  - Standard ReLU networks $(K_{n,\ell} = 1,\ \boldsymbol{b}_{n,\ell} = \boldsymbol{0})$ (Glorot *ICAIS* 2011) (LeCun-Bengio-Hinton *Nature* 2015)

  - Linear regression: $\lambda \to \infty \Rightarrow K_{n,\ell} = 0$

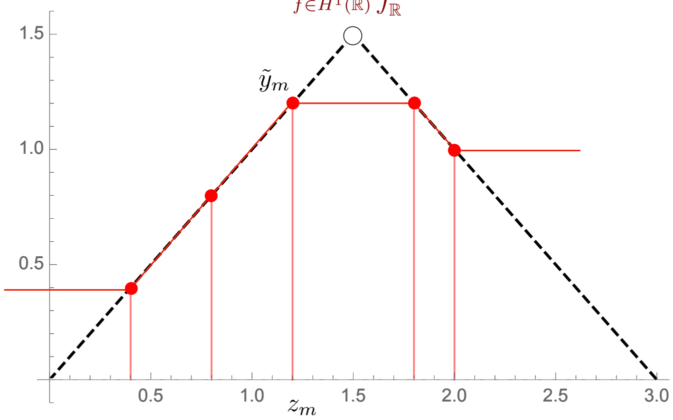
  - State-of-the-art Parametric ReLU networks $(K_{n,\ell} = 1)$ (He et al. CVPR 2015)
    1 ReLU + linear term (per neuron)

  - Adaptive-piecewise linear (APL) networks $(K_{n,\ell} = 5$ or $7,\ \boldsymbol{b}_{n,\ell} = \boldsymbol{0})$ (Agostinelli et al. 2015)

Comparison of linear interpolators

$$\arg \min_{f \in H^1(\mathbb{R})} \int_{\mathbb{R}} |\mathrm{D}f(x)|^2 \mathrm{d}x \quad \text{s.t.} \quad f(x_m) = y_m,\ m = 1, \dots, M$$

(de Boor 1966)



$$\arg \min_{f \in \mathrm{BV}^{(2)}(\mathbb{R})} \|\mathrm{D}^2 f\|_{\mathcal{M}} \quad \text{s.t.} \quad f(x_m) = y_m,\ m = 1, \dots, M$$

(U. JMLR 2019; Lemma 2)

Deep spline networks (Cont'd)

- Key features
  - Direct control of complexity (number of knots): adjustment of λ
  - Ability to suppress unnecessary layers

- Generalizations
  - Broad family of cost functionals
  - Cases where a subset of network components is fixed
  - Generalized forms of regularization: $\psi\big(\mathrm{TV}^{(2)}(\sigma_{n,\ell})\big)$

- Challenges
  - Adaptive knots: more difficult optimization problem

    $\Rightarrow$ In need for more powerful training algorithms
  - Optimal allocation of knots
    ℓ_1-minimization with knot deletion mechanism (even for single layer)
  - Finding the tradeoff: more complex activations vs. deeper architectures

CONCLUSION: The return of the spline

- Continuous-domain formulation of compressed sensing
  - gTV regularization $\Rightarrow$ global optimizer is a ***L*-spline**
  - Sparsifying effect: few **adaptive** knots
  - Discretization consistent with standard paradigm: minimization

- Foundations of machine learning
  - Traditional kernel methods are closely related to splines
    (with one knot/kernel per data point)
  - Deep ReLU neural nets are **high-dimensional** piecewise-linear **splines**
  - **Free-form** activations with TV-regularization $\Rightarrow$ ***Deep splines***

- Favorable properties of splines
  - **Simplicity** (e.g., piecewise polynomial)
  - (higher-order) **continuity**: the difficult part in high dimensions
  - **Adaptivity/sparsity**: the fewest possible pieces = Occam's razor
  - **Efficiency**: B-spline calculus

ACKNOWLEDGMENTS

Many thanks to (former) members of
EPFL's Biomedical Imaging Group

- Dr. Julien Fageot
- Prof. John Paul Ward
- Dr. Mike McCann
- Dr. Kyong Jin
- Harshit Gupta
- Dr. Ha Nguyen
- Dr. Emrah Bostan
- Prof. Ulugbek Kamilov
- Prof. Matthieu Guerquin-Kern

  ....

and collaborators ...

- Prof. Demetri Psaltis
- Prof. Marco Stampanoni
- Prof. Carlos-Oscar Sorzano
- ....

References

- Optimality of splines
  - M. Unser, J. Fageot, J.P. Ward, "Splines Are Universal Solutions of Linear Inverse Problems with Generalized-TV Regularization," *SIAM Review*, vol. 59, No. 4, pp. 769-793, 2017.
- Image reconstruction with sparsity constraints (CS)
  - M. Guerquin-Kern, M. Häberlin, K.P. Pruessmann, M. Unser, "A Fast Wavelet-Based Reconstruction Method for Magnetic Resonance Imaging," *IEEE Transactions on Medical Imaging*, vol. 30, no. 9, pp. 1649-1660, 2011.
  - E. Bostan, U.S. Kamilov, M. Nilchian, M. Unser, "Sparse Stochastic Processes and Discretization of Linear Inverse Problems," *IEEE Trans. Image Processing*, vol. 22, no. 7, pp. 2699-2710, 2013.
- Deep neural networks
  - K.H. Jin, M.T. McCann, E. Froustey, M. Unser, "Deep Convolutional Neural Network for Inverse Problems in Imaging," *IEEE Trans. Image Processing*, vol. 26, no. 9, pp. 4509-4522, Sep. 2017.
  - H. Gupta, K.H. Jin, H.Q. Nguyen, M.T. McCann, M. Unser, "CNN-Based Projected Gradient Descent for Consistent CT Image Reconstruction," *IEEE Trans. Medical Imaging*, vol. 37, no. 6, pp. 1440-1453, 2018.
  - M. Unser, "A representer theorem for deep neural networks," *J. Machine Learning Research*, vol. 20, pp. 1-30, Jul. 2019.

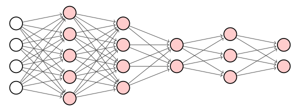
- Preprints and demos: **http://bigwww.epfl.ch/**

Sketch of proof

$$\min_{(\mathbf{U}_\ell),(\sigma_{n,\ell}\in \mathrm{BV}^{(2)}(\mathbb{R}))} \left(\sum_{m=1}^{M} E\big(\boldsymbol{y}_m, \mathbf{f}(\boldsymbol{x}_m)\big) + \mu \sum_{\ell=1}^{N} R_\ell(\mathbf{U}_\ell) + \lambda \sum_{\ell=1}^{L} \sum_{n=1}^{N_\ell} \mathrm{TV}^{(2)}(\sigma_{n,\ell}) \right)$$

Optimal solution $\tilde{\mathbf{f}} = \tilde{\boldsymbol{\sigma}}_L \circ \tilde{\boldsymbol{\ell}}_L \circ \tilde{\boldsymbol{\sigma}}_{L-1} \circ \cdots \circ \tilde{\boldsymbol{\ell}}_2 \circ \tilde{\boldsymbol{\sigma}}_1 \circ \tilde{\boldsymbol{\ell}}_1$ with optimized weights $\tilde{\mathbf{U}}_\ell$ and neuronal activations $\tilde{\sigma}_{n,\ell}$.

Apply "optimal" network $\tilde{\mathbf{f}}$ to each data point $\boldsymbol{x}_m$:



- Initialization (input): $\tilde{\boldsymbol{y}}_{m,0} = \boldsymbol{x}_m$.
- For $\ell = 1, \dots, L$

$$\boldsymbol{z}_{m,\ell} = (z_{1,m,\ell}, \dots, z_{N_\ell,m,\ell}) = \tilde{\mathbf{U}}_\ell \tilde{\boldsymbol{y}}_{m,\ell-1}$$
$$\tilde{\boldsymbol{y}}_{m,\ell} = (\tilde{y}_{1,m,\ell}, \dots, \tilde{y}_{N_\ell,m,\ell}) \in \mathbb{R}^{N_\ell}$$

with $\tilde{y}_{n,m,\ell} = \tilde{\sigma}_{n,\ell}(z_{n,m,\ell}) \quad n = 1, \dots, N_\ell.$ $\qquad \Rightarrow \quad \tilde{\mathbf{f}}(\boldsymbol{x}_m) = \tilde{\boldsymbol{y}}_{m,L}$

This fixes two terms of minimal criterion: $\sum_{m=1}^{M} E\big(\boldsymbol{y}_m, \tilde{\boldsymbol{y}}_{m,L}\big)$ and $\sum_{\ell=1}^{L} R_\ell(\tilde{\mathbf{U}}_\ell)$.

$\tilde{\mathbf{f}}$ achieves global optimum

$$\Leftrightarrow \quad \tilde{\sigma}_{n,\ell} = \arg \min_{f \in \mathrm{BV}^{(2)}(\mathbb{R})} \|\mathrm{D}^2 f\|_{\mathcal{M}} \quad \text{s.t.} \quad f(z_{n,m,\ell}) = \tilde{y}_{n,m,\ell}, \;\; m = 1, \dots, M$$