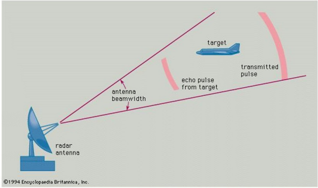
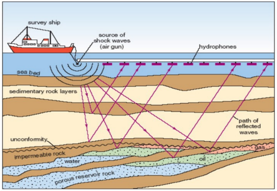
Solving Inverse Problems with Deep Learning

Lexing Ying
Department of Mathematics
Stanford University

Work with Yuwei Fan, Yuehaw Khoo, and Cindy Orozco Bohorquez
Conference on Fast Direct Solvers, Purdue, 10/24/2021

Inverse problem





Inverse problem: discover internal structure from boundary measurements

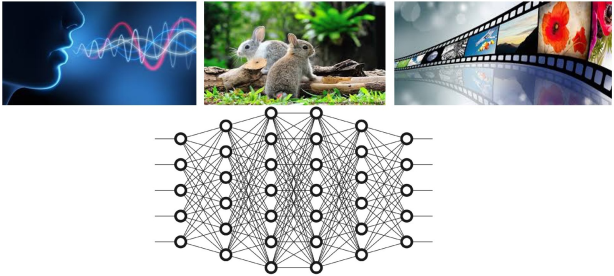
- Farfield imaging
- Seismic imaging
- Electrical impedance tomography
- Traveltime tomography
- Optical tomography

Goal: inverse map

- From boundary measurement d to internal parameter η
- (-) Often ill-conditioned
- (-) A high dimensional nonlinear function

[Image credit: Google]

Deep learning



Key ingredients

- Neural network: A flexible representation for high-dim functions
- Automatic feature extraction
- Efficient optimization: stochastic gradient (SGD), backpropagation, ...
- Software and hardware support

[Image credit: Google]

Deep learning for inverse problems

Goal: representing the inverse map with a DNN

Challenges

- Limited data for inverse problems
- Regression instead of classification

Plan: a seamless integration of physics and data

- Use math/physics to design new DNN modules
- Use math/physics to assemble the DNN from these modules
- Train weights end-to-end using limited data

Applications

- Farfield imaging
- Traveltime tomography
- Seismic imaging
- Electrical impedance tomography
- Optical tomography

Outline

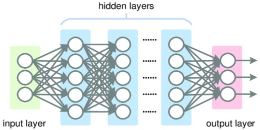
New modules

Applications

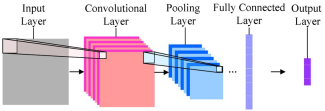
New modules

Mathematical structures and DNN modules

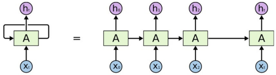
- Dense operator: fully-connected
- Translation-inv local operator: CNN
- Markov chain: RNN
- ODE/time-stepping/semi-group: ResNet



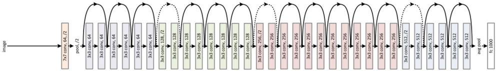
FCN



CNN



RNN



ResNet

We will consider operators used a lot in inverse problems

- (A) Pseudo-differential operators (e.g. normal op) ???
- (B) Fourier integral operators (e.g. forward wave op) ???

(A) Pseudo-differential operators (PDOs)

$$(Af)(x) = \int K(x,y)f(y)dy = \int a(x,\xi)e^{2\pi i x\cdot\xi}\hat{f}(\xi)d\xi$$

PDOs: generalization of diff and conv ops

- Green's fns, normal ops
- Keep the location of singularity

How to build a DNN module for PDOs?

- Sparse approximation for PDO $\Rightarrow$ linear DNN
- Generalize to nonlinear DNN

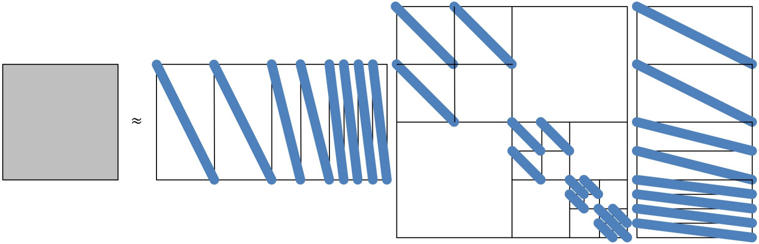
Sparse approximation

- Viewed as a matrix, off-diagonal blocks are smooth and low-rank
- Wavelet transform and non-standard form
- (Fast multipole, hierarchical matrices)
  - Leads to multiscale neural network (MNN), joint work with Yuwei Fan, Jordi Feliu-Faba, Lin Lin, Leo Zepeda-Nunez

PDO: Wavelets

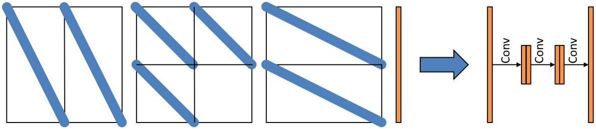
Given an $n \times n$ PDO A, the non-standard form (Beylkin, Coifman, Rokhlin)

- Represents in the redundant wavelet/scaling function frame



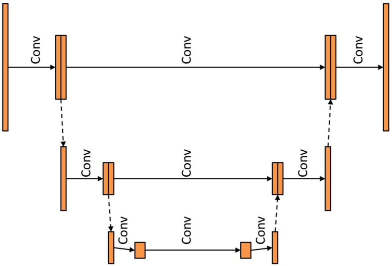
- Keeps only $O(n)$ significant coefficients
- MatVec implemented with (FWT, sparse multiplication, IWT)

Represent MatVec at each scale ℓ as a DNN

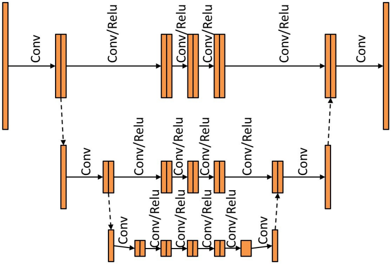


PDO: BCR-Net [Fan, Orozco, Y.]

Represent the whole MatVec as a DNN (dotted line = copying the scaling coeffs)



Generalize via inserting ReLUs (to support nonlinearity)



(B) Fourier integral operators (FIOs)

$$(Af)(x) = \int a(x,\xi)e^{2\pi i \Phi(x,\xi)} f(\xi) d\xi$$

FIOs: generalization of Fourier transform

- Wave propagation in variable media
- Moves singularities in a well-defined way (Hamiltonian flow)

How to build a DNN module for FIOs?

- Sparse approximate for FIO $\Rightarrow$ linear DNN
- Generalize to nonlinear DNN

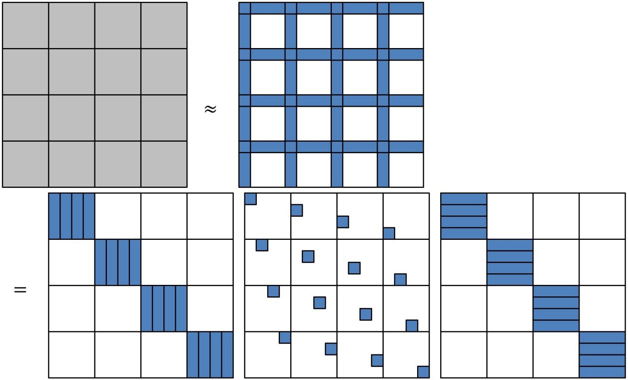
Sparse approximation

- Viewed as a matrix, "square-root-sized" blocks are numerically low-rank
- Butterfly factorization

FIO: Butterfly factorization

Given an $n \times n$ Fourier integral operator A

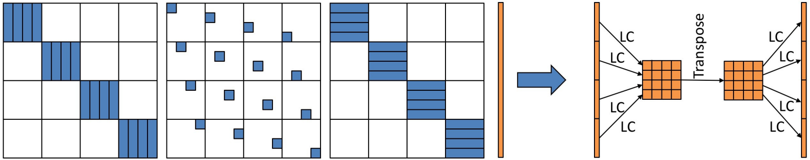
- Partition hierarchically into $\sqrt{n} \times \sqrt{n}$ blocks
- Apply a low-rank approximation to each block



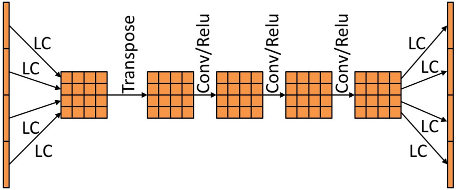
(a hierarchical butterfly factorization also exists)

FIO: SwitchNet [Khoo, Y.]

Represent the whole MatVec as a DNN



Generalize via inserting ReLUs (to support nonlinearity)



Outline

New modules

Applications

Applications

Consider several inverse problems

- (A) Farfield imaging
- (B) Traveltime tomography

For each one, the workflow is

- 1. Design DNN for the inverse map
  Motivation: perturbation theory, filtered backprojection
  Hypothesis: DNN figures out where to do perturbation
  Components: BCR-Net, SwitchNet, conv layers
- 2. Train the weights end-to-end from limited data
- 3. Use DNN for prediction

Related work on DL for inverse problem

Reviews:

- [Arridge et al., 2019, Kong et al., 2019, McCann and Unser, 2019, Ravishankar et al., 2019, Wiecha et al., 2021]

Microlocal analysis:

- [Andrade-Loarca et al., 2021, Bubba et al., 2021]

Linear inverse problem:

- [Hand and Voroninski, 2018, Ongie et al., 2020]

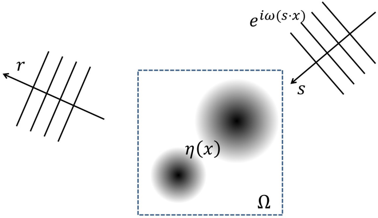
Unrolling approach:

- [Adler and Öktem, 2017, Cheng et al., 2019, de Hoop et al., 2019, Gilton et al., 2019, Hauptmann et al., 2020, Monga et al., 2021, Putzky and Welling, 2019, Zhang and Dong, 2020]

Stability:

- [Antun et al., 2020, Colbrook et al., 2021, Genzel et al., 2020, Gottschling et al., 2020]

(A) Farfield imaging [Khoo, Y.]



Helmholtz equation $Lu = (-\Delta - \frac{\omega^2}{c(x)^2})u = 0$ for the total field $u(x)$

- ω frequency, $c(x)$ unknown sound speed
- c_0 background velocity. Eqv. unknown $\eta(x) = \frac{\omega^2}{c(x)^2} - \frac{\omega^2}{c_0^2}$
- $u(x) = u_s(x) + e^{i\omega s\cdot x}$
  incoming wave $e^{i\omega s\cdot x}$ for $s \in \mathbb{S}^1$, scattered field $u_s(x)$
- Far field pattern $u_s^\infty(r) \equiv \lim_{\rho\to\infty} u_s(\rho \cdot r)\sqrt{\rho}e^{-i\omega\rho}$ at dir $r \in \mathbb{S}^1$.
  Define $d(r, s) = u_s^\infty(r)$
- Inverse problem: $d(r, s) \Rightarrow \eta(x)$

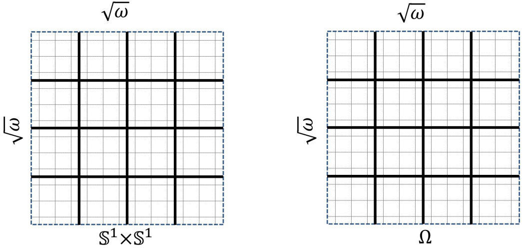
Analysis

Consider first the forward map $A : \eta(x) \Rightarrow d(r, s)$

- The linear regime (η small) as a motivation

$$d(r, s) \approx \int_{x\in\Omega} e^{i\omega(s-r)\cdot x}\eta(x)dx$$

- A is an FIO from $x \in \Omega$ to $(r, s) \in \mathbb{S}^1 \times \mathbb{S}^1$ (nonuniform FT)



- Use the SwitchNet module to represent A (and its adjoint A^*)

DNN Architecture

Inverse map $d(r, s) \Rightarrow \eta(x)$

- Consider the filtered backprojection (linear regime)

$$\eta \approx (A^* A + \epsilon I)^{-1} A^* d$$

- A^* is also an FIO operator (SwitchNet)
- $(A^* A + \epsilon I)^{-1}$ is a PDO (BCR-Net)

DNN

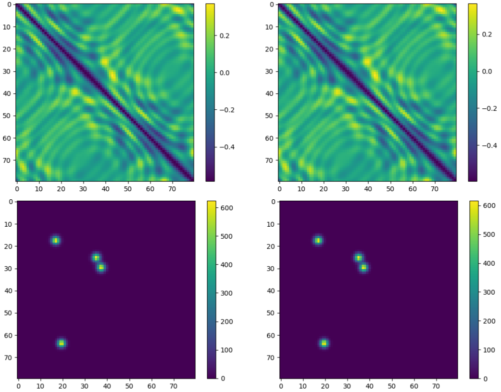
- Use the same architecture of filtered backprojection but include nonlinearities

$$d(r, s) \Rightarrow \text{SwitchNet} \Rightarrow \text{BCR-Net} \Rightarrow \eta(x)$$

- Train all parameters end-to-end

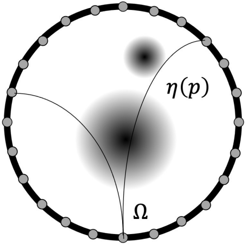
Results

- $\omega = 60$. Ω: unit box discretized with 80×80 samples.
- $d(r,s)$: (r,s) uniformly on $\mathbb{S}^1 \times \mathbb{S}^1$, 80×80 discretization.
- $\eta(x)$: Gaussian mixture.
- 12.5K (η, d) pairs for training and testing.
- Relative error: inverse map $\approx 1\%$; forward map $\approx 4\%$.



DNN results ground truth

(B) Traveltime tomography [Fan, Y.]



Eikonal equation $|\nabla u(x)| = \frac{1}{c(x)} = 1 + \eta(x)$ for traveltime $u(x)$

- $\eta(x)$: unknown slowness deviation
- Let $u^s(\cdot)$ be the solution with condition $u^s(s) \equiv 0$ for boundary point s
- Record solution $u^s(r)$ at boundary point r.
  Define $d(r, s) \equiv u^s(r)$
- Inverse problem: $d(r, s) \Rightarrow \eta(x)$

Analysis

Consider first the forward map $A : \eta(x) \Rightarrow d(r, s)$

- The linear regime (η small) as a motivation
  d in the source/offset coordinates with $(r, s) \equiv (s + h, s)$
  η in the polar coordinates (ρ, θ)

$$d(s, h) \approx \int_0^1 \int_0^{2\pi} \kappa(h, \rho, s - \theta)\eta(\theta, \rho) \mathrm{d}\rho \mathrm{d}\theta$$

  This is a 1d convolution in s/θ with h and ρ as channels
- Use CNN or BCR-Net for A (and its adjoint A^*)

DNN architecture

Inverse map $d(r,s) \Rightarrow \eta(x)$

- Consider the filtered backprojection (linear regime)

$$\eta \approx (A^*A + \epsilon I)^{-1} A^* d$$

- A^* is 1d convolution (1d CNN or BCR-Net)
- $(A^*A + \epsilon I)^{-1}$ is a PDO (BCR-Net)

DNN

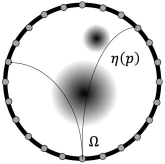
- Use the same architecture of filtered backprojection but include nonlinearities

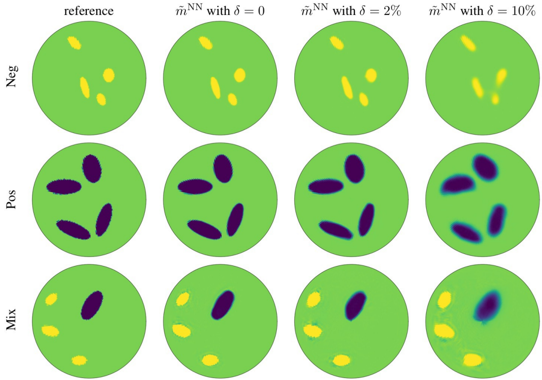
$$d(s,h) \Rightarrow \text{1d-CNN} \Rightarrow \text{BCR-Net} \Rightarrow \eta(\theta,\rho)$$

- Train all parameters end-to-end

Results

- $d(r,s)$: (r,s) uniformly on the boundary of Ω
- $\eta(x)$: a group of ellipsis inclusions
- 20K (η, d) pairs used for training and testing
- Results with different noise added





Summary

New modules for incorporating math/physics into DNN design

- BCR-Net from PDOs
- Switch-Net from FIOs

Applications from inverse problems

- Farfield imaging
- Traveltime tomography

Take-away message

- Fast algorithm decides the structure
- Data decides the weights
- Modularized construction
- End-to-end training

Thank you

Research supported by NSF and DOE/SciDAC

References

- Yuehaw Khoo and Lexing Ying. SwitchNet: a neural network model for forward and inverse scattering problems. SIAM J. Sci. Comput., 41(5), 2019.
- Yuwei Fan, Cindy Orozco Bohorquez, and Lexing Ying. BCR-Net: a neural network based on the nonstandard wavelet form. Journal of Computational Physics, 384, 2019.
- Yuwei Fan and Lexing Ying, Solving electrical impedance tomography with deep learning. Journal of Computational Physics, 404, 2020.
- Yuwei Fan and Lexing Ying, Solving traveltime tomography with deep learning. arXiv:1911.11636.
- Yuwei Fan and Lexing Ying, Solving optical tomography with deep learning. arXiv:1910.04756.
- Yuwei Fan and Lexing Ying, Solving inverse wave scattering with deep learning. arXiv:1911.13202.

Adler, J. and Öktem, O. (2017).
Solving ill-posed inverse problems using iterative deep neural networks.
Inverse Problems, 33(12):124007.

Andrade-Loarca, H., Kutyniok, G., Öktem, O., and Petersen, P. (2021).
Deep microlocal reconstruction for limited-angle tomography.
arXiv preprint arXiv:2108.05732.

Antun, V., Renna, F., Poon, C., Adcock, B., and Hansen, A. C. (2020).
On instabilities of deep learning in image reconstruction and the potential costs of AI.
Proceedings of the National Academy of Sciences, 117(48):30088–30095.

Arridge, S., Maass, P., Öktem, O., and Schönlieb, C.-B. (2019).
Solving inverse problems using data-driven models.
Acta Numerica, 28:1–174.

Bubba, T. A., Galinier, M., Lassas, M., Prato, M., Ratti, L., and Siltanen, S. (2021).
Deep neural networks for inverse problems with pseudodifferential operators: An application to limited-angle tomography.
SIAM Journal on Imaging Sciences, 14(2):470–505.

Cheng, J., Wang, H., Zhu, Y., Liu, Q., Zhang, Q., Su, T., Chen, J., Ge, Y., Hu, Z., Liu, X., et al. (2019).

Model-based deep medical imaging: the roadmap of generalizing iterative reconstruction model using deep learning.
arXiv preprint arXiv:1906.08143.

Colbrook, M. J., Antun, V., and Hansen, A. C. (2021).
Can stable and accurate neural networks be computed?–on the barriers of deep learning and smale's 18th problem.
arXiv preprint arXiv:2101.08286.

de Hoop, M. V., Lassas, M., and Wong, C. A. (2019).
Deep learning architectures for nonlinear operator functions and nonlinear inverse problems.
arXiv preprint arXiv:1912.11090.

Genzel, M., Macdonald, J., and März, M. (2020).
Solving inverse problems with deep neural networks–robustness included?
arXiv preprint arXiv:2011.04268.

Gilton, D., Ongie, G., and Willett, R. (2019).
Neumann networks for linear inverse problems in imaging.
IEEE Transactions on Computational Imaging, 6:328–343.

Gottschling, N. M., Antun, V., Adcock, B., and Hansen, A. C. (2020).
The troublesome kernel: why deep learning for inverse problems is typically unstable.
arXiv preprint arXiv:2001.01258.

Hand, P. and Voroninski, V. (2018).
Global guarantees for enforcing deep generative priors by empirical risk.
In *Conference On Learning Theory*, pages 970–978. PMLR.

Hauptmann, A., Adler, J., Arridge, S., and Öktem, O. (2020).
Multi-scale learned iterative reconstruction.
IEEE transactions on computational imaging, 6:843–856.

Kong, Q., Trugman, D. T., Ross, Z. E., Bianco, M. J., Meade, B. J., and Gerstoft, P. (2019).
Machine learning in seismology: Turning data into insights.
Seismological Research Letters, 90(1):3–14.

McCann, M. T. and Unser, M. (2019).
Biomedical image reconstruction: From the foundations to deep neural networks.
arXiv preprint arXiv:1901.03565.

Monga, V., Li, Y., and Eldar, Y. C. (2021).
Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing.
IEEE Signal Processing Magazine, 38(2):18–44.

Ongie, G., Jalal, A., Metzler, C. A., Baraniuk, R. G., Dimakis, A. G., and Willett, R. (2020).
Deep learning techniques for inverse problems in imaging.

IEEE Journal on Selected Areas in Information Theory, 1(1):39–56.

Putzky, P. and Welling, M. (2019).
Invert to learn to invert.
Advances in Neural Information Processing Systems, 32:446–456.

Ravishankar, S., Ye, J. C., and Fessler, J. A. (2019).
Image reconstruction: From sparsity to data-adaptive methods and machine learning.
Proceedings of the IEEE, 108(1):86–109.

Wiecha, P. R., Arbouet, A., Girard, C., and Muskens, O. L. (2021).
Deep learning in nano-photonics: inverse design and beyond.
Photonics Research, 9(5):B182–B200.

Zhang, H.-M. and Dong, B. (2020).
A review on deep learning in medical image reconstruction.
Journal of the Operations Research Society of China, pages 1–30.