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 \( \eta \)
- (-) 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

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 ⇒ 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 $\ell$ 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
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
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:

Stability:
  ▶ [Antun et al., 2020, Colbrook et al., 2021, Genzel et al., 2020, Gottschling et al., 2020]
Helmholtz equation $Lu = (-\Delta - \frac{\omega^2}{c(x)^2})u = 0$ for the total field $u(x)$

- $\omega$ frequency, $c(x)$ unknown sound speed
- $c_0$ background velocity. Equv. 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 $s \in S^1$, 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 $r \in S^1$. 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 (\( \eta \) 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 \textit{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

- $\omega = 60$. $\Omega$: unit box discretized with $80 \times 80$ samples.
- $d(r, s): (r, s)$ uniformly on $S^1 \times S^1$, $80 \times 80$ discretization.
- $\eta(x)$: Gaussian mixture.
- 12.5K $(\eta, d)$ pairs for training and testing.
- Relative error: inverse map $\approx 1\%$; forward map $\approx 4\%$.
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: $d(r, s) \Rightarrow \eta(x)$
Consider first the forward map $A : \eta(x) \Rightarrow d(r, s)$

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

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

This is a 1d convolution in $s/\theta$ with $h$ and $\rho$ 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
  \[ d(s, h) \Rightarrow 1d-CNN \Rightarrow BCR-Net \Rightarrow \eta(\theta, \rho) \]

- Train all parameters end-to-end
Results

- $d(r,s)$: $(r,s)$ uniformly on the boundary of $\Omega$
- $\eta(x)$: a group of ellipsis inclusions
- $20K (\eta, d)$ 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


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


