Model-based Reconstruction meets Neural Networks: Non-linear Operators in BART

Moritz Blumenthal[1] Xiaoqing Wang[1,2] Guanxiong Luo[1] Zhengguo Tan[1]
H. Christian M. Holme[1,2] Jonathan Tamir[3] Michael Lustig[4] Martin Uecker[1,2]

[1]Institute for Diagnostic and Interventional Radiology, University Medical Center Göttingen, Germany
[2]German Centre for Cardiovascular Research (DZHK), Partner Site Göttingen, Germany
[3]Electrical and Computer Engineering, University of Texas, Austin, TX, USA
[4]Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, USA

BART: Software Toolbox for Computational MRI

Purposes

- rapid prototyping
- reproducible research
- clinical translation

Availability

- Linux, MacOS X, Windows
- BSD license (free for commercial use)
- `https://mrirecon.github.io/bart/`

Command line tools for MRI reconstruction

- calibration methods
  - ESPIRiT, RING, ...
- compressed sensing and parallel imaging
- calibration-less parallel imaging: NLINV and ENLIVE
- ...

BART: Software Toolbox for Computational MRI

Purposes

- rapid prototyping
- reproducible research
- clinical translation

Availability

- Linux, MacOS X, Windows
- BSD license (free for commercial use)
- `https://mrirecon.github.io/bart/`

Command line tools for MRI reconstruction

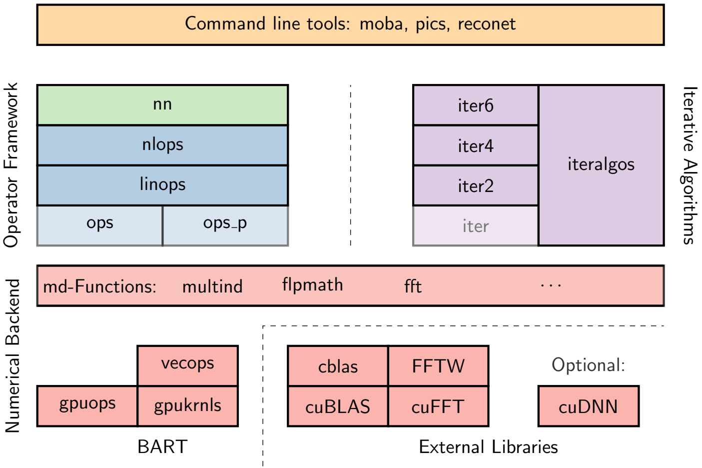
- calibration methods
  - ESPIRiT, RING, ...
- compressed sensing and parallel imaging
- calibration-less parallel imaging: NLINV and ENLIVE
- ...

BART

Outline

1. **Introduction: Non-linear Operators in BART**
   Moritz Blumenthal
2. Non-linear Operators for Model-based Reconstruction
   Xiaoqing Wang and Zhengguo Tan
3. TensorFlow-Regularizer + BART Reconstruction
   Guanxiong Luo
4. Neural Networks in BART
   Moritz Blumenthal

BART Libraries



md-Functions

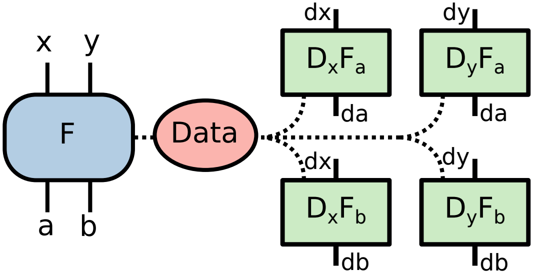
- unified interface
- transparent GPU support

Iterative Framework

- different interfaces for different optimizations (linear, non-linear, neural network)

For more information see BART Webinar #1

Non-linear Operators in BART



Example

$$F: \quad \boldsymbol{x} \mapsto z = x_1^2 + x_2^2$$

$$\mathrm{D}F\big|_{\boldsymbol{x}}: \quad \mathrm{d}\boldsymbol{x} \mapsto \mathrm{d}z = \begin{pmatrix} 2x_1 & 2x_2 \end{pmatrix} \begin{pmatrix} \mathrm{d}x_1 \\ \mathrm{d}x_2 \end{pmatrix}$$

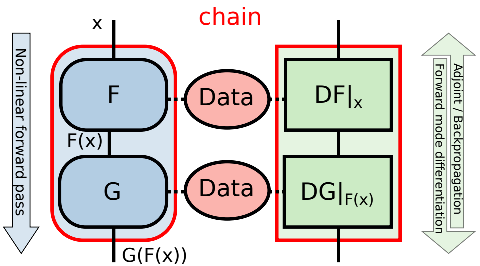
$$\mathrm{D}F^H\Big|_{\boldsymbol{x}}: \quad \mathrm{d}z \mapsto \mathrm{d}\boldsymbol{x} = \begin{pmatrix} 2\bar{x}_1 \\ 2\bar{x}_2 \end{pmatrix} \mathrm{d}z$$

$$F: \quad \mathbb{C}^{N_1+\cdots+N_I} \to \mathbb{C}^{M_0+\cdots+M_O}$$

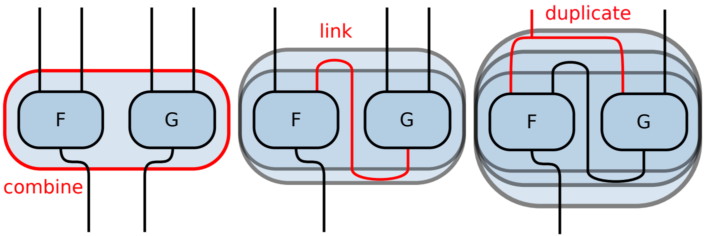
$$\mathrm{D}_iF_o: \quad \mathbb{C}^{N_i} \to \mathbb{C}^{M_o}$$

$$\mathrm{D}_iF_o{}^H: \quad \mathbb{C}^{M_o} \to \mathbb{C}^{N_i}$$

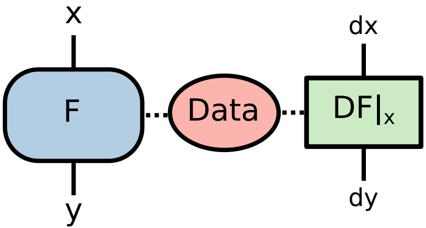
Non-linear Operators — Chaining



- Automatic differentiation
  - ⇒ Chained derivatives are constructed automatically
- Forward mode (linop)
- Reverse mode (adjoint linop)



Non-linear Operators for Model-based Reconstruction



$$\mathrm{D}F|_{\boldsymbol{x}_n} : \mathrm{d}\boldsymbol{x} \mapsto \mathrm{d}\boldsymbol{y} = \left(\frac{\partial F}{\partial \boldsymbol{x}} \Big|_{\boldsymbol{x}_n} \right) \mathrm{d}\boldsymbol{x}$$

$$\mathrm{D}F^H \Big|_{\boldsymbol{x}_n} : \mathrm{d}\boldsymbol{y} \mapsto \mathrm{d}\boldsymbol{x} = \left(\frac{\partial F}{\partial \boldsymbol{x}} \Big|_{\boldsymbol{x}_n} \right)^H \mathrm{d}\boldsymbol{y}$$

Gauß-Newton-Method

Optimize:

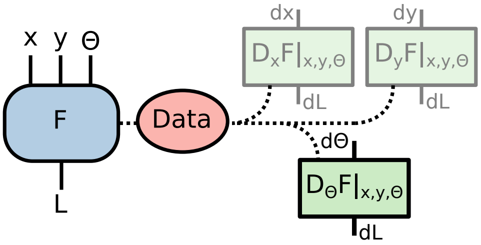
$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \|F(\boldsymbol{x}) - \boldsymbol{y}\|^2$$

Newton-Step

Solve for update $\mathrm{d}\boldsymbol{x}$:

$$(\mathrm{D}F|_{\boldsymbol{x}_n})^H (\mathrm{D}F|_{\boldsymbol{x}_n}) \, \mathrm{d}\boldsymbol{x} = (\mathrm{D}F|_{\boldsymbol{x}_n})^H (\boldsymbol{y} - F(\boldsymbol{x}_n))$$

Non-linear Operators for Neural Networks



$$F: \quad \boldsymbol{x} \mapsto z = x_1^2 + x_2^2$$

$$\mathrm{D}F\big|_{\boldsymbol{x}}: \quad \mathrm{d}\boldsymbol{x} \mapsto \mathrm{d}z = \begin{pmatrix} 2x_1 & 2x_2 \end{pmatrix} \begin{pmatrix} \mathrm{d}x_1 \\ \mathrm{d}x_2 \end{pmatrix}$$

$$\mathrm{D}F^H\Big|_{\boldsymbol{x}}: \quad \mathrm{d}z \mapsto \mathrm{d}\boldsymbol{x} = \begin{pmatrix} 2\bar{x}_1 \\ 2\bar{x}_2 \end{pmatrix} \mathrm{d}z$$

Loss Operator

$$F: \mathbb{C}^{N_1+N_2+N_3} \to \mathbb{R}$$
$$[\boldsymbol{x}, \boldsymbol{y}, \Theta] \mapsto L(\boldsymbol{y}, Net(\boldsymbol{x}, \Theta))$$

Gradient

$$\nabla_\Theta L = \left(\mathrm{D}_\Theta F^H{}_{|[\boldsymbol{x},\boldsymbol{y},\Theta]}\right) 1$$

Outline

1. Introduction: Non-linear Operators in BART
   Moritz Blumenthal
2. **Non-linear Operators for Model-based Reconstruction**
   Xiaoqing Wang and Zhengguo Tan
3. TensorFlow-Regularizer + BART Reconstruction
   Guanxiong Luo
4. Neural Networks in BART
   Moritz Blumenthal