Quantifying Model Uncertainty in Inverse Problems via Bayesian Deep Gradient Descent

Riccardo Barbano [1] Chen Zhang [1] Simon Arridge [1] Bangti Jin [1]

[1]University College London

Introduction

- Deep learning (DL) image reconstruction techniques have remarkable results but lack estimates of uncertainty
- This is critical in sensitive domains such as medical imaging
- There are many "types or sources" of uncertainty but the most common in medical imaging are:
  - **Epistemic** - uncertainty in the parameters (i.e., model uncertainty)
  - **Aleatoric** - stochastic variability in the data
- **Our goal**: a DL reconstruction method that allows us to account for epistemic uncertainty in medical images

Deep Unrolled Optimisation

- **Unrolled optimisation** mimics iterative methods but
  1. Executes only a finite number of iterations
  2. Computes the "updates" using DNNs
- The iterates are computed as residual updates with a feasibility projection:

$$\mathbf{x}_k = \mathrm{ReLu}(\mathbf{x}_{k-1} + \delta\mathbf{x}_{k-1})$$

- The increments $\delta\mathbf{x}_{k-1}$ are computed as

$$\delta\mathbf{x}_{k-1} = f_{\varphi_k}\left(\nabla\mathcal{D}\left(\mathbf{y}, \mathbf{A}\mathbf{x}_{k-1}\right), \mathbf{x}_{k-1}\right) =: f_{\varphi_k}(\nabla\mathcal{D}, \mathbf{x}_{k-1})$$

- $\varphi_k = (\phi_k, \theta_k)$ are (deterministic & probabilistic) parameters of the neural net (**block**) f_{φ_k}
- The entire iteration (**cascade**) consists of K sequential blocks

$$\mathbf{x}_K = (f_{\varphi_K} \circ f_{\varphi_{K-1}} \circ \cdots \circ f_{\varphi_1})(\nabla\mathcal{D}, \mathbf{x}_0) := f_{\Phi_K}(\nabla\mathcal{D}, \mathbf{x}_0), \quad \text{with } \Phi_k := (\varphi_1, \ldots, \varphi_k)$$

Bayesian Deep Gradient Descent

- Each block (network) of the cascade consists of two parts
  - Deterministic layers with weights ϕ_k
  - A final Bayesian layer with (random) weights θ_k
- For θ_k we learn the variational parameters ψ_k defining their distribution
- To estimate the posterior $p(\theta|X,Y)$ we use **variational inference** which uses an approximate, simple to compute, distribution q^*_ψ
- Moreover, we train the network **greedily**: provided previous $k-1$ blocks have been trained, in block k we use the family $\mathcal{Q}_k$

$$q_{\Psi_k}(\Theta_k) = q^*_{\Psi_{k-1}}(\Theta_{k-1}) q_{\psi_k}(\theta_k|\Theta_{k-1}), \text{ with } q_{\psi_k}(\theta_k|\Theta_{k-1}) = \prod_{d=1}^{D} \mathcal{N}(\mu_{k,d}, \sigma^2_{k,d}),$$

where $\psi_k = \{(\mu_{k,\ell}, \sigma^2_{k,d})\}_{d=1}^D$, and $q^*_{\Psi_{k-1}}$ is the distribution learnt for the previous $k-1$ blocks

- The optimal distribution is computed by minimising the negative **ELBO**

$$q^*_{\Psi_k} \in \underset{q_{\Psi_k} \in \mathcal{Q}_k}{\operatorname{argmin}} \mathcal{L}_k(q_{\Psi_k}; X, Y) := -\int q_{\Psi_k}(\Theta_k) \log p(X|Y, \Theta_k) \mathrm{d}\Theta_k + \mathrm{KL}(q_{\Psi_k}(\Theta_k) \| p(\Theta_k))$$

- The prior is set recursively as

$$p(\Theta_k) = q_{\Psi_{k-1}}(\Theta_{k-1}) p(\theta_k|\Theta_{k-1}), \text{where } p(\theta_k|\Theta_{k-1}) = \mathcal{N}(0, I).$$

- BDGD takes the likelihood as:

$$p(\mathbf{x}|\mathbf{y}, \Theta_k) = \mathcal{N}(f_{\Theta_k}(\nabla\mathcal{D}, \mathbf{x}_0), \sigma_k^2 I)$$

Estimating Predictive Uncertainty

- The optimal approximate posterior distribution is given by

$$q^*_{\Psi_K}(\Theta_K) = q^*_{\Psi_1}(\theta_1) \prod_{k=2}^{K} q^*_{\Psi_k}(\theta_k|\Theta_{k-1})$$

.

- We use Monte Carlo (MC) estimators to estimate the statistics of of the approximate predictive posterior defined as:

$$q^*(\mathbf{x}|\mathbf{y}) = \int p(\mathbf{x}|\mathbf{y}, \Theta_K) q^*_{\Psi_K}(\Theta_K) \mathrm{d}\Theta_K$$

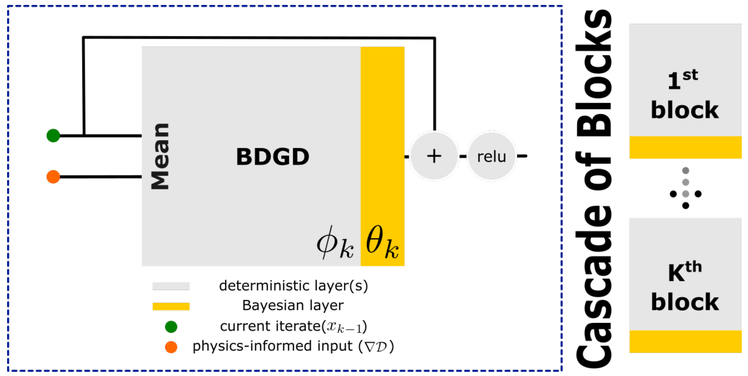
- We compute the expected image (mean) and use $T \geq 1$ Monte Carlo samples:

$$\hat{\mathbb{E}}[\mathbf{x}] := \frac{1}{T} \sum_{t=1}^{T} f_{\hat{\Theta}^t_K}(\nabla\mathcal{D}, \mathbf{x}_0)$$

- We summarise predictive uncertainty as the (entry-wise) predictive variance $\mathrm{Var}[\mathbf{x}]$ at the Kth step:

$$\mathrm{Var}[\mathbf{x}] = \mathrm{Var}_{q_{\Psi_K}(\Theta_K)}[\mathbb{E}(\mathbf{x}|\mathbf{y}, \Theta_K)] + \mathbb{E}_{q_{\Psi_K}(\Theta_K)}[\mathrm{Var}(\mathbf{x}|\mathbf{y}, \Theta_K)]$$
$$\approx \sigma_K^2 + \underbrace{\frac{1}{T}\sum_{t=1}^{T} f_{\Theta^t_K}(\nabla\mathcal{D}, \mathbf{x}_0)^2 - \left(\frac{1}{T}\sum_{t=1}^{T} f_{\Theta^t_K}(\nabla\mathcal{D}, \mathbf{x}_0)\right)^2}_{\text{epistemic}}$$

BDGD Framework Diagram



Practicalities in Training

Training

Input: # reconstruction steps K, dataset $\mathcal{D}$, initial guess $x_0^{(i)}$, batch-size M

for $k \leftarrow 1$ to K **do**
  Construct network's input:
    $\mathcal{D}_{k-1} = \{x^{(i)}_{k-1}, \nabla D(y^{(i)}, Ax^{(i)}_{k-1})\}_{i=1}^N$
  Train the k-th network $f_{\phi_k,\theta_k}(\nabla D(y^{(i)}, Ax^{(i)}_{k-1}), x^{(i)}_{k-1})$:
    `// stochastic mini-batch optimisation`
    $\psi^*_k, \phi^*_k \leftarrow \arg\min_{q \in \mathcal{Q}_k, \phi_k} \left\{ \hat{\mathcal{L}}(\phi_k, q) = -\frac{N}{M} \sum_i^M \mathbb{E}_{\hat{\Theta}_k \sim q_{\Phi_k}(\Theta_k)} \left[\log p_{\Phi_k}(x^{(i)}|y^{(i)}, \hat{\Theta}_k)\right] + \mathrm{KL}(q_{\Phi_k}(\Theta_k) \| p_{\Phi_k}(\Theta_k)) \right\}$
  `// update with` $\hat{\theta}_k \sim q^*_{\psi_k}(\theta_k|\Theta_{k-1})$
  $x^{(i)}_k \leftarrow f_{\phi_k,\hat{\theta}_k}(\nabla D(y^{(i)}, Ax^{(i)}_{k-1}), x^{(i)}_{k-1})$

Output: approximate posterior at each reconstruction step

Inference

Input: # reconstruction steps K, observation y, initial guess x_0, trained parameters (Φ_K, Ψ_K), # samples T

for $t \leftarrow 1$ to T **do**
  `//with` $\hat{\Theta}^{(t)}_K \sim q^*_{\Phi_K}(\Theta_K)$
  Sample $x^{(t)}_K = f_{\Phi_K,\hat{\Theta}^{(t)}_K}(\nabla D(y, Ax_0), x_0)$

Evaluate $\hat{\mathbb{E}}[x]$ and $\widehat{\mathrm{Var}}[x]$ with $\{x^{(t)}_K\}_{t=1}^T$
Output: $\hat{\mathbb{E}}[x]$ and $\widehat{\mathrm{Var}}[x]$

Results

Sparse view CT, with respect to the number of directions (dirs), and limited angle CT, with respect to the available range of angles. We report the mean PSNR over the ellipses dataset as well as the PSNR for the Shepp-Logan.

Table 1:Sparse View CT (30 dirs)

Methods	Ellipses Phantoms	SL Phantom
FBP	25.5264	18.4667
TV	35.1587	37.2162
LPD	**44.5122 ± 0.4911**	44.0472 ± 0.4187
DGD	43.2577 ± 0.4183	44.6913 ± 0.6644
BDGD - MFVI	**44.6642 ± 0.4637**	**47.2946 ± 0.5778**
BDGD - MCDO	43.2126 ± 0.1285	45.1725 ± 0.4461

Table 2:Limited View CT $[0, 2\pi/3)$

Methods	Ellipses Phantoms	SL Phantom
FBP	18.5958	17.1085
TV	32.9134	29.2113
LPD [1]	40.7578 ± 0.3050	33.8427 ± 1.2380
DGD [2]	42.6994 ± 0.4243	42.8905 ± 0.5883
BDGD - MFVI	**44.0297 ± 0.4698**	**45.5140 ± 0.8261**
BDGD - MCDO	41.5367 ± 0.3884	41.4397 ± 0.6299

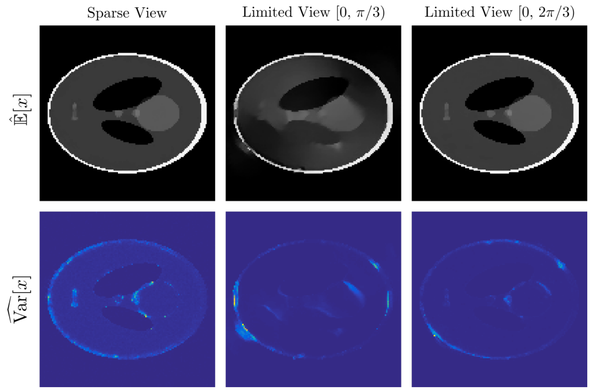


Fig. 1:Mean estimates and epistemic uncertainty maps by BDGD-MFVI for different geometries: (*Left*) sparse view with 30 directions, (*Centre*) limited view $[0, \pi/3)$, (*Right*) limited view $[0, 2\pi/3)$.

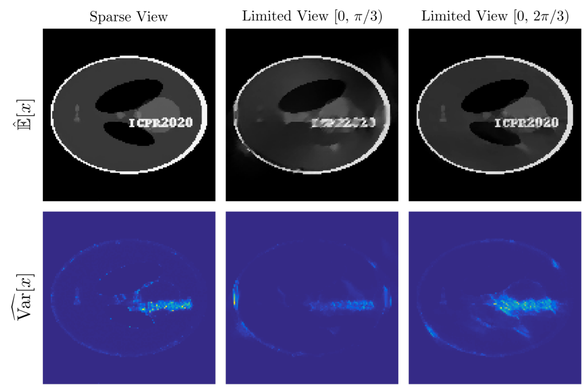


Fig. 2:Out-of-distribution reconstruction for different geometries by BDGD-MFVI: (*Left*) sparse view with 30 directions, (*Centre*) limited view $[0, \pi/3)$, (*Right*) limited view $[0, 2\pi/3)$.

Bibliography

[1] Jonas Adler and Ozan Öktem.
Learned primal-dual reconstruction.
IEEE Trans. Med. Imag., 37(6):1322--1332, 2018.

[2] Andreas Hauptmann, Felix Lucka, Marta Betcke, Nam Huynh, Jonas Adler, Ben Cox, Paul Beard, Sebastien Ourselin, and Simon Arridge.
Model-based learning for accelerated, limited-view 3-d photoacoustic tomography.
IEEE Trans. Med. Imag., 37(6):1382--1393, 2018.