Recent Advanced in using Machine Learning for Image Reconstruction

Jonas Adler[1, 2] Ozan Öktem

[1]Department of Mathematics
KTH - Royal Institute of Technology, Stockholm, Sweden

[2]Research and Physics
Elekta, Stockholm, Sweden

Lecture overview

- Bayesian inversion
- Learn a reconstruction method (estimator)
- Task adapted reconstruction
- Posterior Sampling

Bayesian inversion

Inverse problems

Inverse problem (Functional analytic viewpoint)

Data $y \in Y$ is a single observation generated by Y-valued random variable $\mathbf{y}$ where

$$\mathbf{y} = \mathcal{A}(x^*) + \mathbf{e}.$$

Solution: A model parameter (element in X) that approximates x^*.

- $X =$ possible model parameters, $Y =$ possible data.
- Data model
  - Forward model: $\mathcal{A} \colon X \to Y$ deterministic model for data.
  - Observational noise: Random variable $\mathbf{e}$ with known distribution.
  - Data likelihood: $\mathbb{P}(y \mid x) =$ probability of data y given model parameter x.

$$\mathbb{P}(y \mid x) = \mathbb{P}\big(\mathcal{A}(x) - y\big) \quad \text{if } \mathbf{e} \sim \mathbb{P}.$$

Inverse problems

Inverse problem (Statistical viewpoint)

Data $y \in Y$ is a single observation generated by Y-valued random variable $\mathbf{y}$ where

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}.$$

Solution: A probability distribution on model parameter space X.

- $X =$ possible model parameters, $Y =$ possible data.
- Data model
  - Forward model: $\mathcal{A} \colon X \to Y$ deterministic model for data.
  - Observational noise: Random variable $\mathbf{e}$ with known distribution.
  - Data likelihood: $\mathbb{P}(y \mid x) =$ probability of data y given model parameter x.

$$\mathbb{P}(y \mid x) = \mathbb{P}\big(\mathcal{A}(x) - y\big) \quad \text{if } \mathbf{e} \sim \mathbb{P}.$$

- Randomness of $\mathbf{x}$ reflects our incomplete information.

Bayesian inversion

- Bayes theorem:

$$\mathbb{P}(x \mid y) = \frac{\mathbb{P}(y \mid x)\mathbb{P}(x)}{\mathbb{P}(y)} \quad \text{given data } y.$$

Bayesian inversion

- Bayes theorem:
$$\mathbb{P}(x \mid y) = \frac{\mathbb{P}(y \mid x)\mathbb{P}(x)}{\mathbb{P}(y)} \quad \text{given data } y.$$
- Bayesian inversion: Recover the posterior or explore it by computing estimators (point estimators, credible region estimators, extreme value probabilities, conditional covariance estimators, sampling estimators, ...)

Bayesian inversion

- Bayes theorem:

$$\mathbb{P}(x \mid y) = \frac{\mathbb{P}(y \mid x)\mathbb{P}(x)}{\mathbb{P}(y)} \quad \text{given data } y.$$

- Bayesian inversion: Recover the posterior or explore it by computing estimators (point estimators, credible region estimators, extreme value probabilities, conditional covariance estimators, sampling estimators, . . .)
- Challenges:
  - Data likelihood
  - Prior
  - Probability of data
  - Computational feasibility

Bayesian inversion

- Bayes theorem:

$$\mathbb{P}(x \mid y) = \frac{\mathbb{P}(y \mid x)\mathbb{P}(x)}{\mathbb{P}(y)} \quad \text{given data } y.$$

- Bayesian inversion: Recover the posterior or explore it by computing estimators (point estimators, credible region estimators, extreme value probabilities, conditional covariance estimators, sampling estimators, . . .)
- Challenges:
  - Data likelihood $\Longleftarrow$ physics model for generation of data
  - Prior
  - Probability of data
  - Computational feasibility

Bayesian inversion

- Bayes theorem:

$$\mathbb{P}(x \mid y) = \frac{\mathbb{P}(y \mid x)\mathbb{P}(x)}{\mathbb{P}(y)} \quad \text{given data } y.$$

- Bayesian inversion: Recover the posterior or explore it by computing estimators (point estimators, credible region estimators, extreme value probabilities, conditional covariance estimators, sampling estimators, . . .)
- Challenges:
  - Data likelihood $\Longleftarrow$ physics model for generation of data
  - Prior $\Longleftarrow$ can be handcrafted or learned
  - Probability of data
  - Computational feasibility

Bayesian inversion

- Bayes theorem:

$$\mathbb{P}(x \mid y) = \frac{\mathbb{P}(y \mid x)\mathbb{P}(x)}{\mathbb{P}(y)} \quad \text{given data } y.$$

- Bayesian inversion: Recover the posterior or explore it by computing estimators (point estimators, credible region estimators, extreme value probabilities, conditional covariance estimators, sampling estimators, ...)
- Challenges:
  - Data likelihood $\Longleftarrow$ physics model for generation of data
  - Prior $\Longleftarrow$ can be handcrafted or learned
  - Probability of data $\Longleftarrow$ major issue if this needs to be explicitly given
  - Computational feasibility

Bayesian inversion

- Bayes theorem:

$$\mathbb{P}(x \mid y) = \frac{\mathbb{P}(y \mid x)\mathbb{P}(x)}{\mathbb{P}(y)} \quad \text{given data } y.$$

- Bayesian inversion: Recover the posterior or explore it by computing estimators (point estimators, credible region estimators, extreme value probabilities, conditional covariance estimators, sampling estimators, . . .)
- Challenges:
  - Data likelihood $\Longleftarrow$ physics model for generation of data
  - Prior $\Longleftarrow$ can be handcrafted or learned
  - Probability of data $\Longleftarrow$ major issue if this needs to be explicitly given
  - Computational feasibility remains a major issue for imaging applications

Bayesian inversion

- Maximum likelihood: Maximise the data likelihood, i.e. solve

$$\arg\max_x \mathbb{P}(y \mid x) \iff \arg\min_x \Big[-\log \mathbb{P}(y \mid x)\Big] \quad \text{for given } y.$$

  + No need to specify prior or probability of data.
  + Computationally feasible for large-scale problems.
  – Not suitable for ill-posed problems (overfitting).

- MAP: Maximise the posterior, i.e. solve

$$\arg\max_x \mathbb{P}(x \mid y) \iff \arg\min_x \Big[-\log \mathbb{P}(y \mid x) - \log \mathbb{P}(x)\Big] \quad \text{for given } y.$$

  + No need to specify probability of data.
  + Suitable for ill-posed problems.
  – Need to specify prior.
  – Computationally unfeasible for large-scale problems.

Bayesian Inversion

- Conditional mean: Model parameter $\widehat{x}$ is the conditional mean, i.e.

$$\widehat{x} := \mathbb{E}[\mathbf{x} \mid \mathbf{y} = y] = \int_X x \, \mathbb{P}(x \mid y) \, \mathrm{d}x \quad \text{for given } y.$$

  + Suitable for ill-posed problems.
  – Need to specify prior.
  – Need to specify probability of data.
  – Computationally unfeasible for small- to mid-scale problems.

- Bayes estimator: Minimise expected loss w.r.t. $\ell_X : X \times X \to \mathbb{R}$, i.e.

$$\widehat{x} := \widehat{\mathcal{R}}(y) \quad \text{where} \quad \widehat{\mathcal{R}} \in \operatorname*{arg\,min}_{\mathcal{R}\colon Y \to X} \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mu}\Big[\ell_X\big(\mathcal{R}(\mathbf{y}), \mathbf{x}\big)\Big].$$

  + Suitable for ill-posed problems.
  + Suitable for supervised learning.
  + Equivalent to conditional mean when loss is squared 2-norm.
  – Need to specify joint distribution.
  – Computationally unfeasible for small- to mid-scale problems.

Bayesian inversion

- Inverse problen: $\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}$
- Data likelihood: $\mathbf{e} \sim \text{Normal}(0, \sigma^2)$
  - $\mathbb{P}(e) \propto \exp\Big(-\frac{1}{\sigma^2}\|e\|^2\Big)$
  - $\mathbb{P}(y \mid x) = \mathbb{P}\big(\mathcal{A}(x) - y\big) \implies -\log \mathbb{P}(y \mid x) \propto \frac{1}{\sigma^2}\big\|\mathcal{A}(x) - y\big\|^2$

Bayesian inversion

- Inverse problen: $\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}$
- Data likelihood: $\mathbf{e} \sim \text{Normal}(0, \sigma^2)$
    - $\mathbb{P}(e) \propto \exp\Big(-\frac{1}{\sigma^2}\|e\|^2\Big)$
    - $\mathbb{P}(y \mid x) = \mathbb{P}\big(\mathcal{A}(x) - y\big) \implies -\log \mathbb{P}(y \mid x) \propto \frac{1}{\sigma^2}\big\|\mathcal{A}(x) - y\big\|^2$
- Maximum likelihood:

$$\arg\min_x \frac{1}{\sigma^2}\big\|\mathcal{A}(x) - y\big\|^2$$

Bayesian inversion

- Inverse problen: $\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}$
- Data likelihood: $\mathbf{e} \sim \mathrm{Normal}(0, \sigma^2)$
  - $\mathbb{P}(e) \propto \exp\left(-\frac{1}{\sigma^2}\|e\|^2\right)$
  - $\mathbb{P}(y \mid x) = \mathbb{P}\big(\mathcal{A}(x) - y\big) \implies -\log \mathbb{P}(y \mid x) \propto \frac{1}{\sigma^2}\big\|\mathcal{A}(x) - y\big\|^2$
- Maximum likelihood:

$$\arg\min_x \frac{1}{\sigma^2}\big\|\mathcal{A}(x) - y\big\|^2$$

- Prior: Gibbs type of prior $\mathbb{P}(x) \propto \exp\big(-S_\theta(x)\big)$ with $S_\theta \colon X \to \mathbb{R}$ convex.
- MAP:

$$\arg\min_x \left[S_\theta(x) + \frac{1}{\sigma^2}\big\|\mathcal{A}(x) - y\big\|^2\right]$$

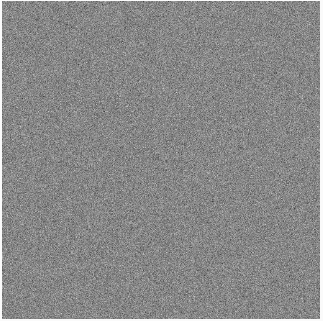
Bayesian inversion

- Probability distribution of model parameter **x**.
- Contains all other information about model parameters.
- Should not be related to the measurement.
- Assigns high probability to “natural” model parameters and low probability to unexpected model parameters.
- Design of prior distributions is the main difficulty in statistical inversion.
- Gibbs type of prior:

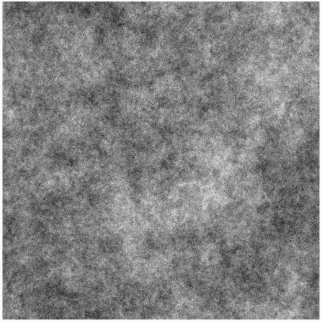
$$\mathbb{P}(x) \propto \exp\big(-S_\theta(x)\big) \quad \text{with } S_\theta \colon X \to \mathbb{R} \text{ convex.}$$

Often θ is scalar and $S_\theta(x) = \theta S(x)$ for fixed $S \colon X \to \mathbb{R}$.

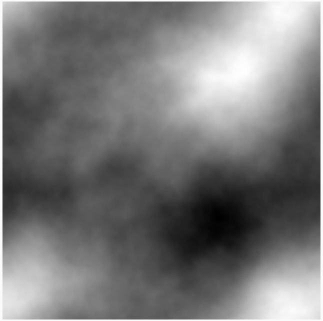
Bayesian inversion



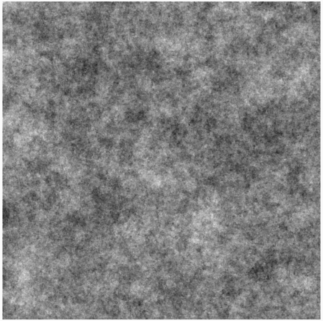
$S(x) = \|x\|_2^2$



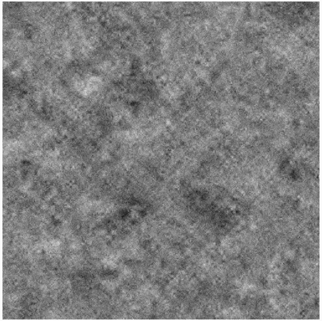
$S(x) = \|\nabla x\|_2^2$



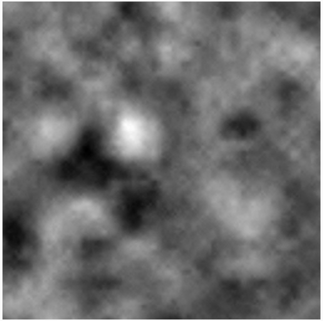
$S(x) = \|\Delta x\|_2^2$



$S(x) = \|\nabla x\|_1$

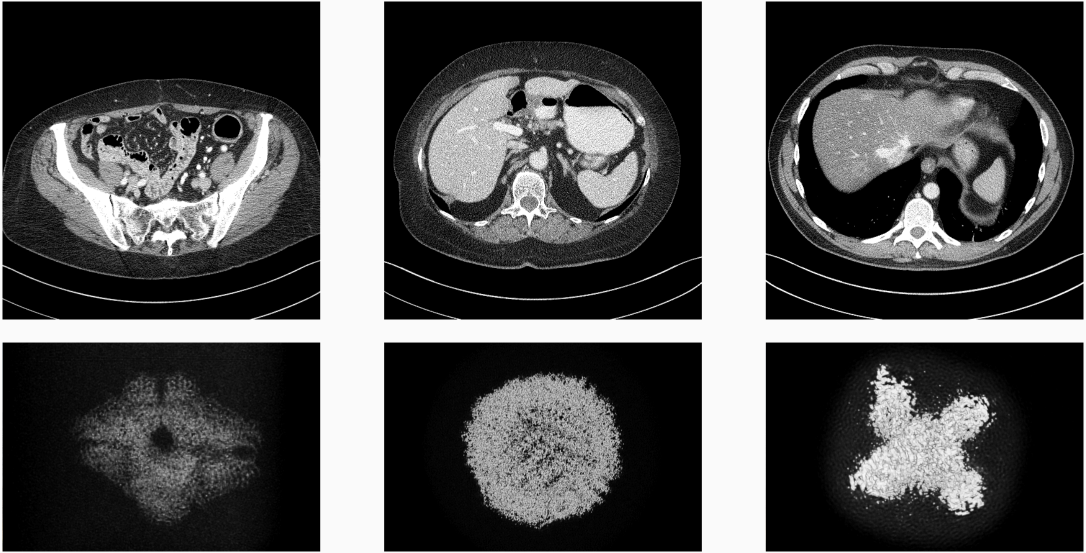


$S(x) = \|x\|_{B^1_{1,1}}$



$S(x) = \|x\|_{B^2_{1,1}}$

Examples of natural images



Bayesian inversion

- \+ Flexible and widely applicable framework for reconstruction.
- \+ Plug-and-play structure for adapting to different physics models of data generation and statistical models of noise.
- \+ Plug-and-play structure for priors.
- – Need to handcraft a prior.
- – Many estimators, like conditional mean, also require probability of data.
- – Computationally unfeasible.

Only MAP estimator currently used in imaging.

Learn an estimator

Reference(s): (Adler & Öktem, 2017, 2018b)

Learning an estimator

Inverse problem:

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}.$$

Goal: Bayes estimator of $\mathbf{x}$ given $\mathbf{y} = y$ without explicitly specifying joint probability.

Approach: Use supervised learning to compute an estimator that minimises Bayes risk:

$$\operatorname*{arg\,min}_{\mathcal{R}\colon Y \to X} \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mu}\Big[\ell_X\big(\mathcal{R}(\mathbf{y}), \mathbf{x}\big)\Big].$$

Learning an estimator

Inverse problem:

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}.$$

Goal: Bayes estimator of $\mathbf{x}$ given $\mathbf{y} = y$ without explicitly specifying joint probability.

Approach: Use supervised learning to compute an estimator that minimises Bayes risk:

$$\operatorname*{arg\,min}_{\mathcal{R}\colon Y \to X} \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mu}\Big[\ell_X\big(\mathcal{R}(\mathbf{y}), \mathbf{x}\big)\Big].$$

- Joint distribution μ of $(\mathbf{x}, \mathbf{y})$ unknown

Learning an estimator

Inverse problem:

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}.$$

Goal: Bayes estimator of $\mathbf{x}$ given $\mathbf{y} = y$ without explicitly specifying joint probability.

Approach: Use supervised learning to compute an estimator that minimises Bayes risk:

$$\operatorname*{arg\,min}_{\mathcal{R}\colon Y \to X} \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mu}\Big[\ell_X\big(\mathcal{R}(\mathbf{y}), \mathbf{x}\big)\Big].$$

- Joint distribution μ of $(\mathbf{x}, \mathbf{y})$ unknown

  $\implies$ Empirical risk minimisation using training data (x_i, y_i) generated by $(\mathbf{x}, \mathbf{y})$.

Learning an estimator

Inverse problem:

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}.$$

Goal: Bayes estimator of $\mathbf{x}$ given $\mathbf{y} = y$ without explicitly specifying joint probability.

Approach: Use supervised learning to compute an estimator that minimises Bayes risk:

$$\operatorname*{arg\,min}_{\mathcal{R}\colon Y \to X} \frac{1}{m} \sum_{i=1}^{m} \ell_X\big(\mathcal{R}(y_i), x_i\big).$$

- Joint distribution μ of $(\mathbf{x}, \mathbf{y})$ unknown

  $\implies$ Empirical risk minimisation using training data (x_i, y_i) generated by $(\mathbf{x}, \mathbf{y})$.

Learning an estimator

Inverse problem:

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}.$$

Goal: Bayes estimator of $\mathbf{x}$ given $\mathbf{y} = y$ without explicitly specifying joint probability.

Approach: Use supervised learning to compute an estimator that minimises Bayes risk:

$$\operatorname*{arg\,min}_{\mathcal{R} \colon Y \to X} \frac{1}{m} \sum_{i=1}^{m} \ell_X\big(\mathcal{R}(y_i), x_i\big).$$

- Joint distribution μ of $(\mathbf{x}, \mathbf{y})$ unknown
  $\implies$ Empirical risk minimisation using training data (x_i, y_i) generated by $(\mathbf{x}, \mathbf{y})$.
- Unfeasible to search over all reconstruction methods $\mathcal{R} \colon Y \to X$

Learning an estimator

Inverse problem:

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}.$$

Goal: Bayes estimator of $\mathbf{x}$ given $\mathbf{y} = y$ without explicitly specifying joint probability.

Approach: Use supervised learning to compute an estimator that minimises Bayes risk:

$$\arg\min_{\theta} \frac{1}{m} \sum_{i=1}^{m} \ell_X\big(\mathcal{R}_\theta(y_i), x_i\big).$$

- Joint distribution μ of $(\mathbf{x}, \mathbf{y})$ unknown
  $\implies$ Empirical risk minimisation using training data (x_i, y_i) generated by $(\mathbf{x}, \mathbf{y})$.
- Unfeasible to search over all reconstruction methods $\mathcal{R} \colon Y \to X$
  $\implies$ consider $\mathcal{R}_\theta \colon Y \to X$ parametrised by deep neural network.

Learning an estimator

Inverse problem:

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{e}.$$

Goal: Bayes estimator of $\mathbf{x}$ given $\mathbf{y} = y$ without explicitly specifying joint probability.

Approach: Use supervised learning to compute an estimator that minimises Bayes risk:

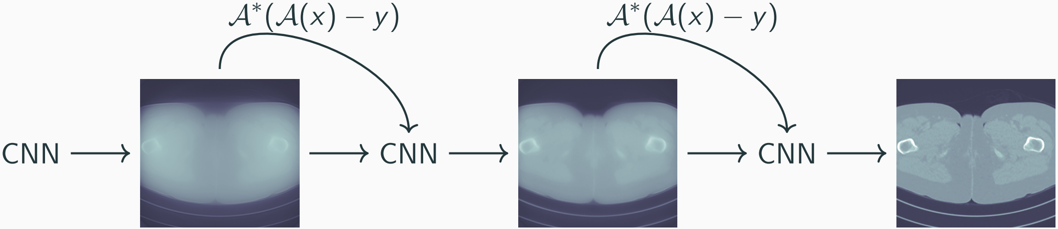
$$\arg\min_{\theta} \frac{1}{m} \sum_{i=1}^{m} \ell_X\big(\mathcal{R}_\theta(y_i), x_i\big).$$

- Joint distribution μ of $(\mathbf{x}, \mathbf{y})$ unknown
  $\implies$ Empirical risk minimisation using training data (x_i, y_i) generated by $(\mathbf{x}, \mathbf{y})$.
- Unfeasible to search over all reconstruction methods $\mathcal{R} \colon Y \to X$
  $\implies$ consider $\mathcal{R}_\theta \colon Y \to X$ parametrised by deep neural network.
- Joint distribution partially known: $\mu(x, y) = \mathbb{P}(x) \otimes \mathbb{P}(y \mid x)$
  $\implies$ use neural network architecture $\mathcal{R}_\theta \colon Y \to X$ containing the data likelihood.

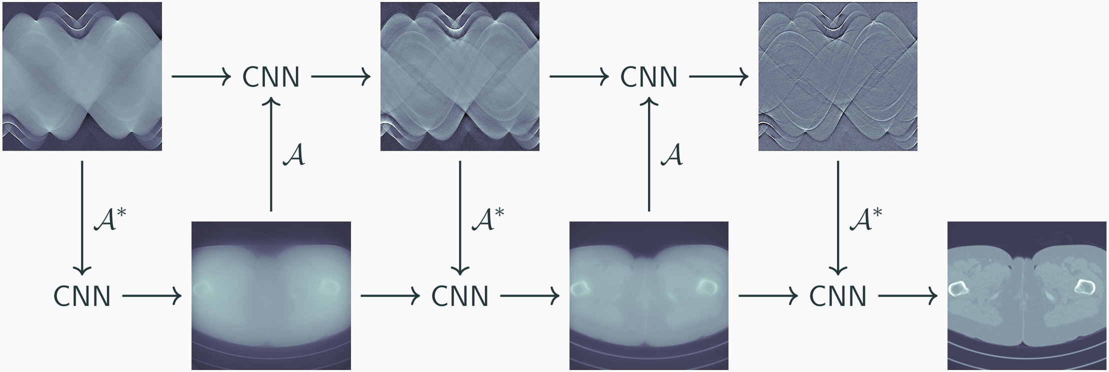
Learned iterative reconstruction

- Use deep learning to compute Bayes estimator
- Encode data likelihood into neural network architecture, prior implicitly given by supervised training data
- Can be adapted to a specific task

Learned iterative reconstruction



Learned iterative reconstruction



Learned iterative reconstruction

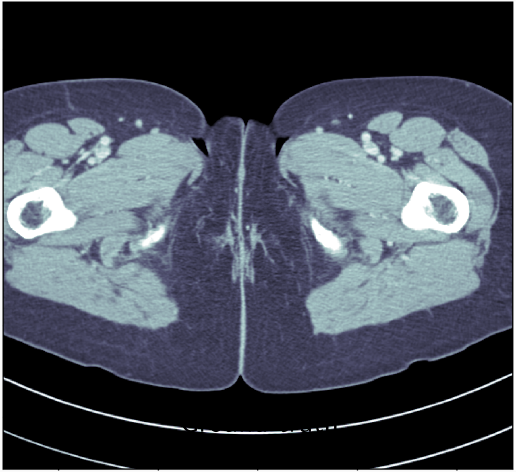
- How to parametrise the learned operators Γ_{θ^d} and Λ_{θ^p}?
- Use recent advancements in deep learning and experience from variational regularisation:
  - Translation invariance (convolutional networks)
  - Pointwise non-linearities
  - Perturbations of the identity (residual networks)
- Example settings for 2D tomography:
  - 3 layer residual network with PReLU nonlinearities and 3×3 convolutions
  - Unroll with $N = 10$ ‘iterations’ $\implies$ 60 layers.
  - Deep learning network has ‘only’ has 264 960 parameters.

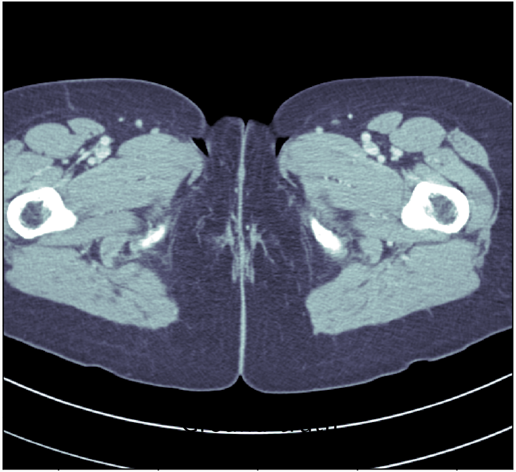
Learned iterative reconstruction

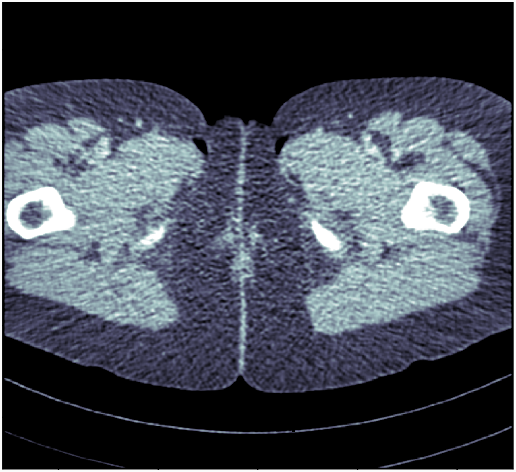
Inverse problem: Recover attenuation coefficient from tomographic data (sinogram)

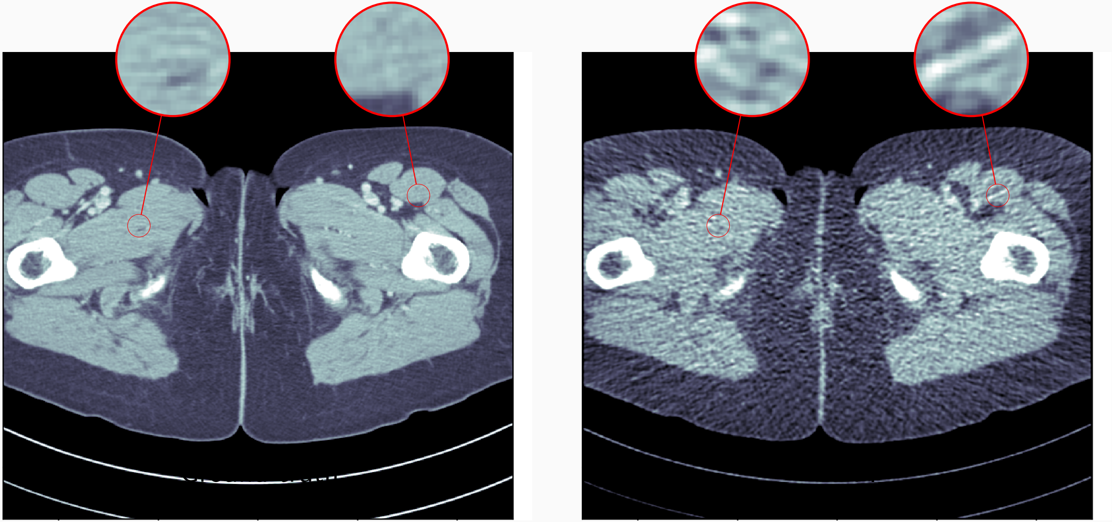
$$y = \mathcal{A}(x) + e$$

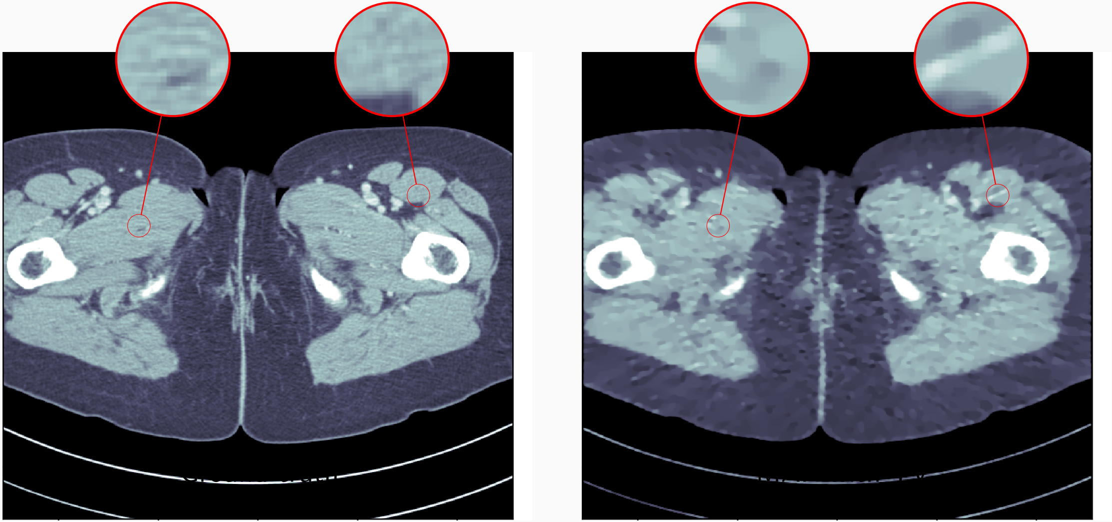
- Forward operator: 2D ray transform
- Geometry: Fan beam, 1000 lines/angle, 1000 angles
- Noise: Poisson noise, 10^4 incident photons/detector element
- Image: 512×512 pixel
- Training data: About 2 000 pairs (x_i, y_i) from 9 patients

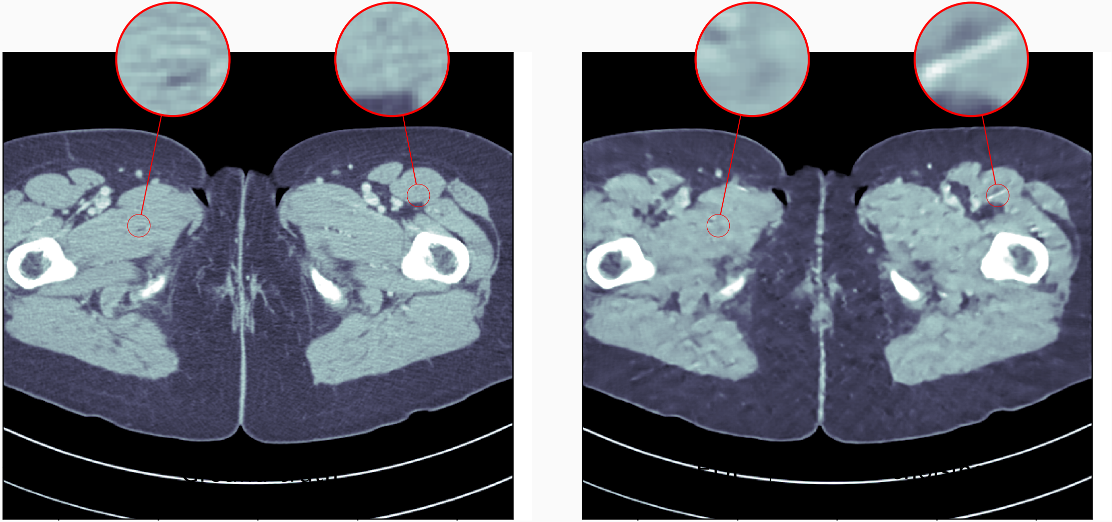


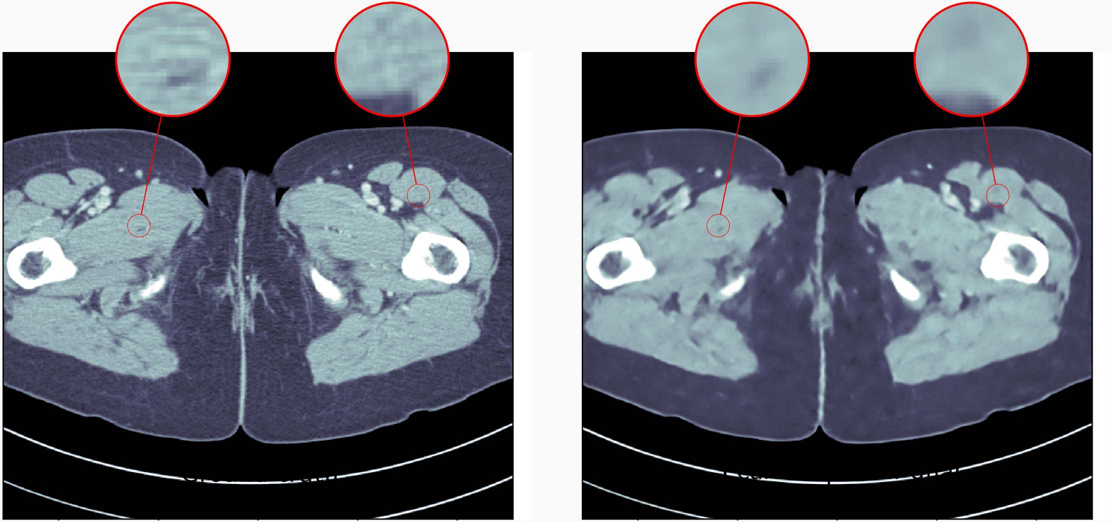












Learned iterative reconstruction

Quantitative comparison (SSIM = structural similarity index, 1 = perfect match)

Method	PSNR (dB)	SSIM	Runtime (ms)	Parameters
FBP	33.65	0.829	423	1
MAP with TV	37.48	0.946	64 371	1
FBP + U-Net	41.92	0.941	463	10^7
Learned primal-dual	44.11	0.969	620	$2.4 \cdot 10^5$

Comments

- Improved reconstruction quality against state-of-the-art (2 years ago)
- No need to manually set ‘obscure’ parameters
- Execution time allows for clinical implementation
- Very modest requirements on amount of supervised training data

Task adapted reconstruction

Joint work with Sebastian Lunz, Olivier Verdier, and Carola Schönlieb

Reference(s): (Adler et al., 2018a, 2018b)

Task adapted reconstruction

- Medical imaging is not done for fun, we want to solve a task!
    - Images typically summarised, either by an expert or by using specific descriptors, in an analysis step.
    - Summaries used as input for decision making.

Task adapted reconstruction

- Medical imaging is not done for fun, we want to solve a task!
    - Images typically summarised, either by an expert or by using specific descriptors, in an analysis step.
    - Summaries used as input for decision making.
- Task adapted reconstruction: Methods that jointly perform reconstruction and task (segmentation, classification, radiomics, . . .).

Task Adapted Reconstruction

- Bayes estimator: $\mathcal{R}_\theta \colon Y \to X$ where θ minimises Bayes risk:

$$L(\theta) := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mu}\Big[\ell_{\mathsf{X}}(\mathcal{R}_\theta(\mathbf{y}), \mathbf{x})\Big] \quad \text{given loss } \ell_{\mathsf{X}} \colon X \times X \to \mathbb{R}.$$

Task Adapted Reconstruction

- Bayes estimator: $\mathcal{R}_\theta \colon Y \to X$ where θ minimises Bayes risk:

$$L(\theta) := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mu}\Big[\ell_X(\mathcal{R}_\theta(\mathbf{y}), \mathbf{x})\Big] \quad \text{given loss } \ell_X \colon X \times X \to \mathbb{R}.$$

- Traditional losses: $\|x_1 - x_2\|_2^2$ or $\|x_1 - x_2\|_1$

Task Adapted Reconstruction

- Bayes estimator: $\mathcal{R}_\theta \colon Y \to X$ where θ minimises Bayes risk:

$$L(\theta) := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mu}\Big[\ell_X(\mathcal{R}_\theta(\mathbf{y}), \mathbf{x})\Big] \quad \text{given loss } \ell_X \colon X \times X \to \mathbb{R}.$$

- Traditional losses: $\|x_1 - x_2\|_2^2$ or $\|x_1 - x_2\|_1$
- Fancy losses
  - Adversarial: Uses a neural network to quantify if the result is "good".
  - Perceptual: Quantifies image similarity by comparing features extracted using a previously trained neural network.

Task Adapted Reconstruction

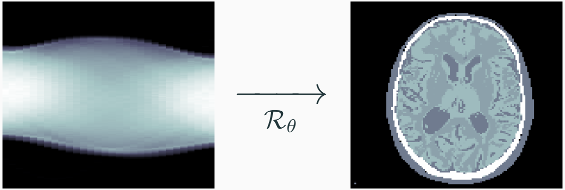
- Bayes estimator: $\mathcal{R}_\theta \colon Y \to X$ where θ minimises Bayes risk:

$$L(\theta) := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mu}\Big[\ell_X(\mathcal{R}_\theta(\mathbf{y}), \mathbf{x})\Big] \quad \text{given loss } \ell_X \colon X \times X \to \mathbb{R}.$$

- Traditional losses: $\|x_1 - x_2\|_2^2$ or $\|x_1 - x_2\|_1$
- Fancy losses
  - Adversarial: Uses a neural network to quantify if the result is "good".
  - Perceptual: Quantifies image similarity by comparing features extracted using a previously trained neural network.
- Adapt loss to the task.

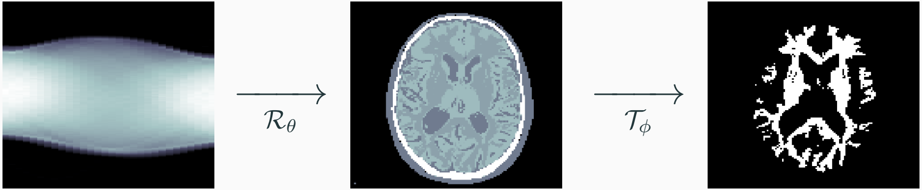
Task Adapted Reconstruction

- We can learn to go from data to reconstruction



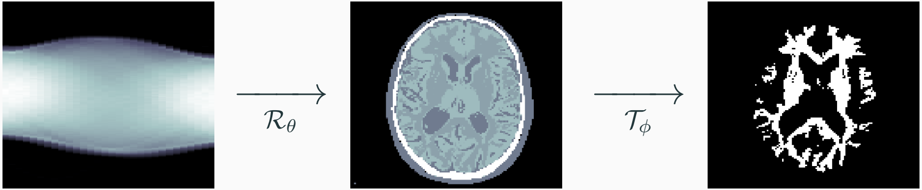
Task Adapted Reconstruction

- We can learn to go from data to reconstruction
- Combine with learned task operator



Task Adapted Reconstruction

- We can learn to go from data to reconstruction
- Combine with learned task operator
- End-to-end differentiable training!



Task Adapted Reconstruction

- Sequential training: First train a reconstruction, then train the task

$$L_{\text{rec}}(\theta) = \mathbb{E}_{\mathbf{x},\mathbf{y}}\Big[\ell_{\mathsf{X}}\big(\mathcal{R}_\theta(\mathbf{y}), \mathbf{x}\big)\Big]$$

$$L_{\text{task}}(\phi) = \mathbb{E}_{\mathbf{y},\mathbf{d}}\Big[\ell_{\mathsf{D}}\big(\mathcal{T}_\phi \circ \mathcal{R}_{\theta^*}(\mathbf{y}), \mathbf{d}\big)\Big] \quad \text{where } \theta^* \text{ minimises } \theta \mapsto L_{\text{rec}}(\theta).$$

Task Adapted Reconstruction

- Sequential training: First train a reconstruction, then train the task

$$L_{\mathrm{rec}}(\theta) = \mathbb{E}_{\mathbf{x},\mathbf{y}}\Big[\ell_{\mathrm{X}}\big(\mathcal{R}_\theta(\mathbf{y}), \mathbf{x}\big)\Big]$$

$$L_{\mathrm{task}}(\phi) = \mathbb{E}_{\mathbf{y},\mathbf{d}}\Big[\ell_{\mathrm{D}}\big(\mathcal{T}_\phi \circ \mathcal{R}_{\theta^*}(\mathbf{y}), \mathbf{d}\big)\Big] \quad \text{where } \theta^* \text{ minimises } \theta \mapsto L_{\mathrm{rec}}(\theta).$$

- End-to-end training: Straight from data to task

$$L(\theta, \phi) = \mathbb{E}_{\mathbf{y},\mathbf{d}}\Big[\ell_{\mathrm{D}}\big(\mathcal{T}_\phi \circ \mathcal{R}_\theta(\mathbf{y}), \mathbf{d}\big)\Big].$$

Task Adapted Reconstruction

- Sequential training: First train a reconstruction, then train the task

$$L_{\text{rec}}(\theta) = \mathbb{E}_{\mathbf{x},\mathbf{y}}\Big[\ell_X\big(\mathcal{R}_\theta(\mathbf{y}), \mathbf{x}\big)\Big]$$

$$L_{\text{task}}(\phi) = \mathbb{E}_{\mathbf{y},\mathbf{d}}\Big[\ell_D\big(\mathcal{T}_\phi \circ \mathcal{R}_{\theta^*}(\mathbf{y}), \mathbf{d}\big)\Big] \quad \text{where } \theta^* \text{ minimises } \theta \mapsto L_{\text{rec}}(\theta).$$

- End-to-end training: Straight from data to task

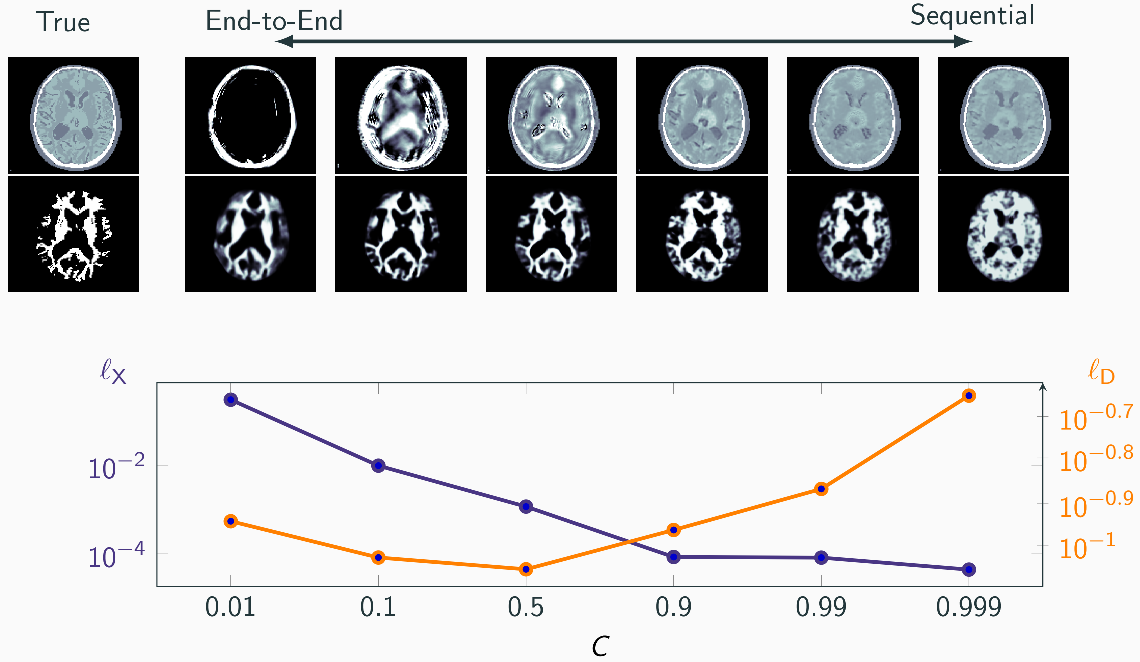
$$L(\theta, \phi) = \mathbb{E}_{\mathbf{y},\mathbf{d}}\Big[\ell_D\big(\mathcal{T}_\phi \circ \mathcal{R}_\theta(\mathbf{y}), \mathbf{d}\big)\Big].$$

- Task adapted training: Anything in between

$$L(\theta, \phi) = \mathbb{E}_{\mathbf{x},\mathbf{y},\mathbf{d}}\Big[C\,\ell_X\big(\mathcal{R}_\theta(\mathbf{y}), \mathbf{x}\big) + (1 - C)\,\ell_D\big(\mathcal{T}_\phi \circ \mathcal{R}_\theta(\mathbf{y}), \mathbf{d}\big)\Big].$$

Task Adapted Reconstruction

- 7 CT brain scans
  - Segmented semi-manually
  - Simulated low-dose data
- Task: Segment white matter given CT sinogram
- Reconstruction operator: $\mathcal{R}_\theta \colon Y \to X$ (learned primal-dual)
- Task operator: $\mathcal{T}_\phi \colon X \to D$ (U-Net)



True
End-to-End
Sequential
ℓ_X
ℓ_D
10^{-2}
10^{-4}
$10^{-0.7}$
$10^{-0.8}$
$10^{-0.9}$
10^{-1}
0.01
0.1
0.5
0.9
0.99
0.999
C

Task Adapted Reconstruction

Segmentation only an example, can perform reconstruction jointly with any task given by a trainable differentiable neural network.

- Semantic segmentation (Thoma, 2016; Guo et al., 2018).
- Caption generation (Karpathy & Fei-Fei, 2017; Li et al., 2018).
- Image translation (Wolterink et al., 2017).
- Object recognition (Sermanet et al., 2013; He et al., 2016; Farabet et al., 2013).
- Non-rigid image registration (Ghosal & Ray, 2017; Dalca et al., 2018).

...

Deep posterior sampling

Generative models for uncertainty quantification in inverse problems

Reference(s): (Adler & Öktem, 2018a)

Conditional generative models

Input: Supervised training data (x_i, y_i) generated by $(\mathbf{x}, \mathbf{y})$.

Goal: Sample from unknown posterior $\mathbb{P}(x \mid y)$.

Approach: Learn how to sample from posterior by solving

$$\min_{\theta} \mathbb{E}_{\mathbf{y} \sim \mathbb{P}_{data}} \Big[\mathcal{W}\big(\mathrm{G}_\theta(\mathbf{y}), \mathbb{P}(\mathbf{x} \mid \mathbf{y}) \big) \Big].$$

- For each $y \in Y$, $\mathrm{G}_\theta(y)$ is a probability distribution on model parameters in X.
- $\mathcal{W}$ is the Wasserstein 1-distance, measures how close $\mathrm{G}_\theta(y)$ is to the posterior.

Conditional generative models

Input: Supervised training data (x_i, y_i) generated by $(\mathbf{x}, \mathbf{y})$.

Goal: Sample from unknown posterior $\mathbb{P}(x \mid y)$.

Approach: Learn how to sample from posterior by solving

$$\min_{\theta} \mathbb{E}_{\mathbf{y} \sim \mathbb{P}_{data}} \Big[\mathcal{W}\big(\mathrm{G}_\theta(\mathbf{y}), \mathbb{P}(\mathbf{x} \mid \mathbf{y}) \big) \Big].$$

- For each $y \in Y$, $\mathrm{G}_\theta(y)$ is a probability distribution on model parameters in X.
- $\mathcal{W}$ is the Wasserstein 1-distance, measures how close $\mathrm{G}_\theta(y)$ is to the posterior.
- Unfeasible: Not possible to evaluate expectation (probability of data $\mathbb{P}$ and posterior $\mathbb{P}$ are both unknown).

Conditional generative models

Input: Supervised training data (x_i, y_i) generated by $(\mathbf{x}, \mathbf{y})$.

Goal: Sample from unknown posterior $\mathbb{P}(x \mid y)$.

Approach: Learn how to sample from posterior by solving

$$\min_{\theta} \mathbb{E}_{\mathbf{y} \sim \mathbb{P}_{data}}\Big[\mathcal{W}\big(\mathrm{G}_\theta(\mathbf{y}), \mathbb{P}(\mathbf{x} \mid \mathbf{y})\big)\Big].$$

- For each $y \in Y$, $\mathrm{G}_\theta(y)$ is a probability distribution on model parameters in X.
- $\mathcal{W}$ is the Wasserstein 1-distance, measures how close $\mathrm{G}_\theta(y)$ is to the posterior.
- Unfeasible: Not possible to evaluate expectation (probability of data $\mathbb{P}$ and posterior $\mathbb{P}$ are both unknown).

  $\Longrightarrow$ Re-write using the Kantorovich-Rubinstein dual characterization of $\mathcal{W}$.

Conditional generative models

Input: Supervised training data (x_i, y_i) generated by $(\mathbf{x}, \mathbf{y})$.

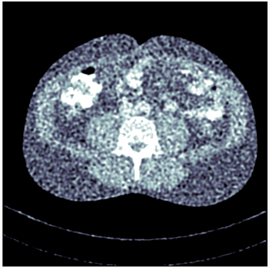
Goal: Sample from unknown posterior $\mathbb{P}(x \mid y)$.

Approach: Learn how to sample from posterior by solving

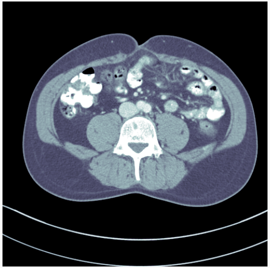
$$\min_{\theta} \left\{ \max_{\phi} \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mu} \left[\mathrm{D}_{\phi}(\mathbf{x}, \mathbf{y}) - \mathbb{E}_{\mathbf{z}} \left[\mathrm{D}_{\phi}(\mathrm{G}_{\theta}(\mathbf{z}, \mathbf{y}), \mathbf{y}) \right] \right] \right\}.$$

- Unfeasible: Not possible to evaluate expectation (probability of data $\mathbb{P}$ and posterior $\mathbb{P}$ are both unknown).
  $\implies$ Re-write using the Kantorovich-Rubinstein dual characterization of $\mathcal{W}$.
- Generator (and discriminator) trained against supervised training data
  $\implies$ trained generator can be used to sample from posterior by sampling $\mathbf{z}$.

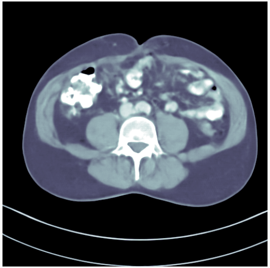
One of the images is the ground truth (phantom), can you figure out which one?



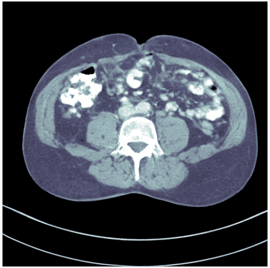
1



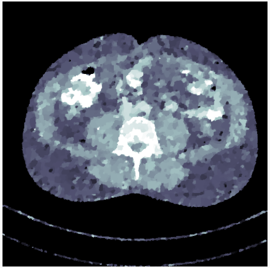
2



3

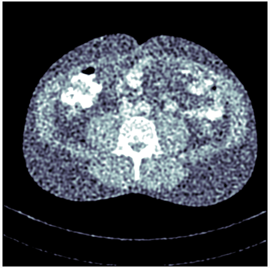


4

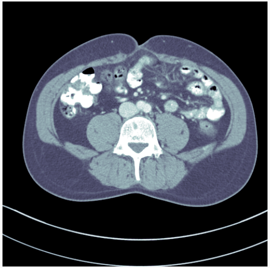


5

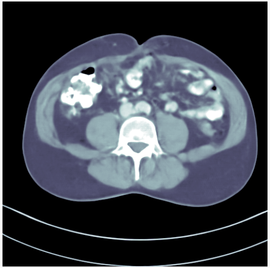
One of the images is the ground truth (phantom), can you figure out which one?



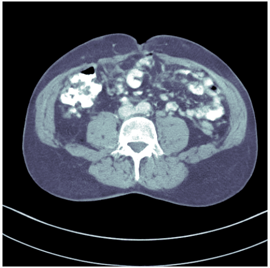
FBP



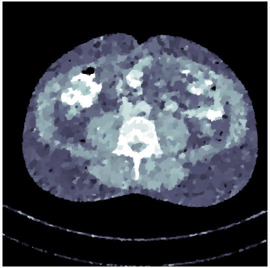
2



Conditional mean

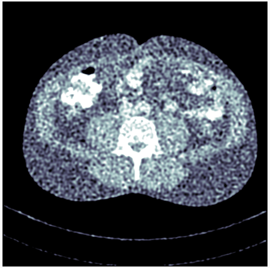


4

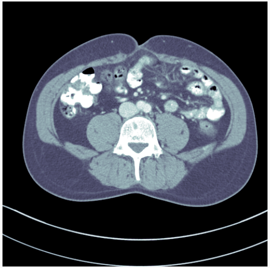


Total variation

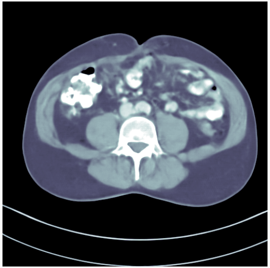
One of the images is the ground truth (phantom), can you figure out which one?



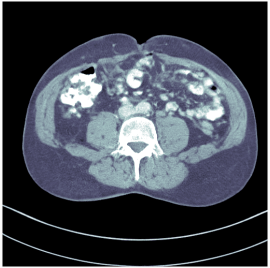
FBP



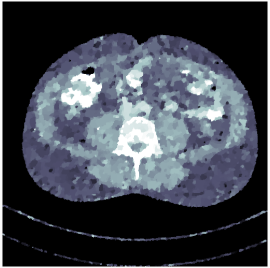
Phantom



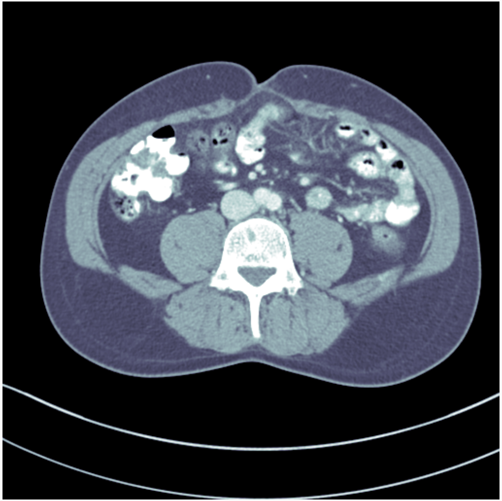
Conditional mean



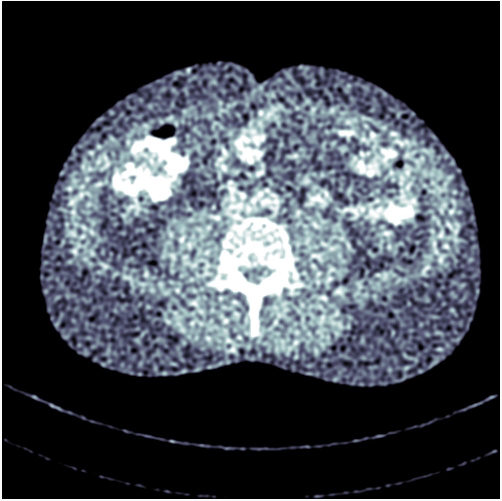
Deep posterior sample



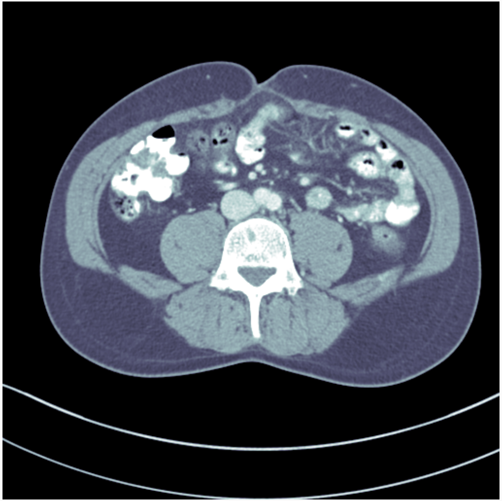
Total variation



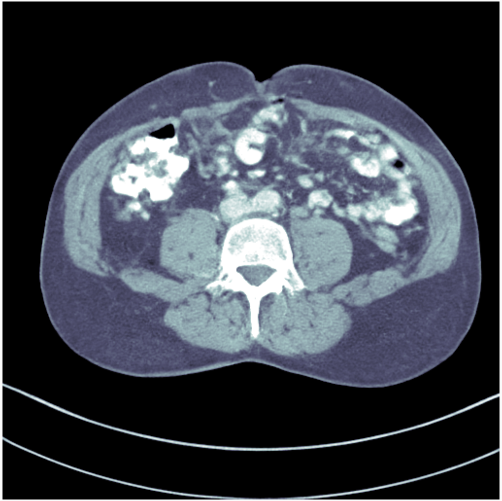
Phantom



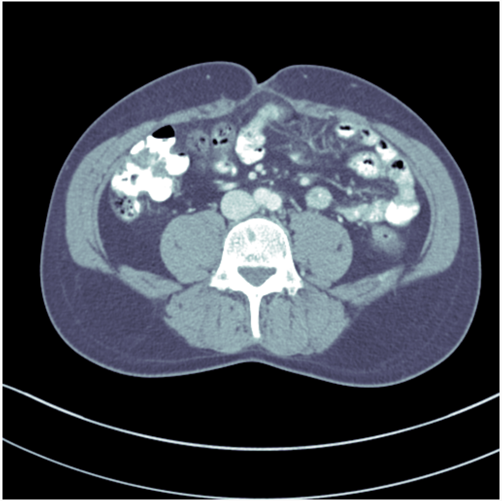
FBP reconstruction



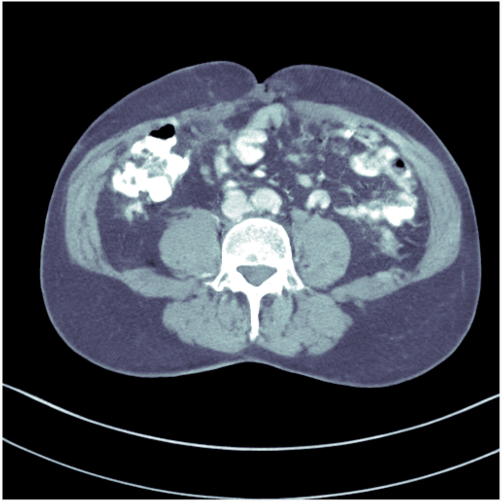
Phantom



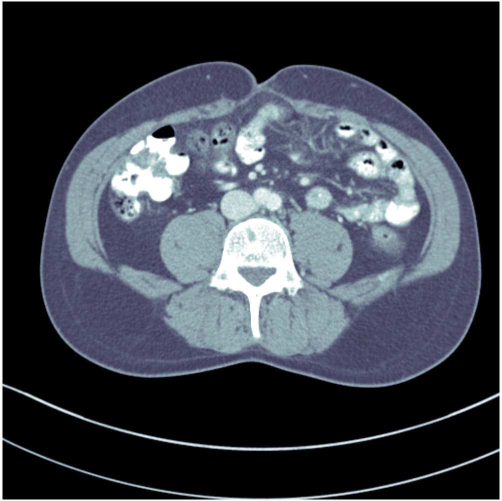
Posterior sample 1



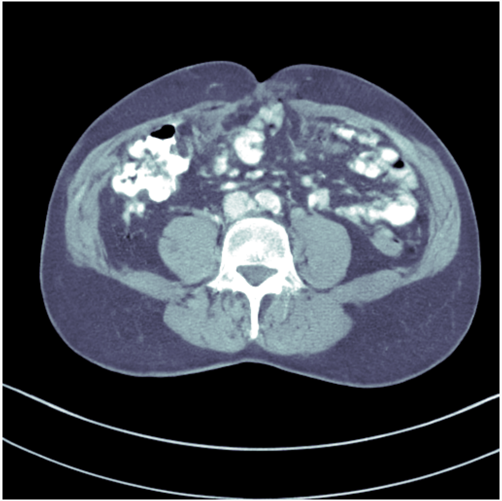
Phantom



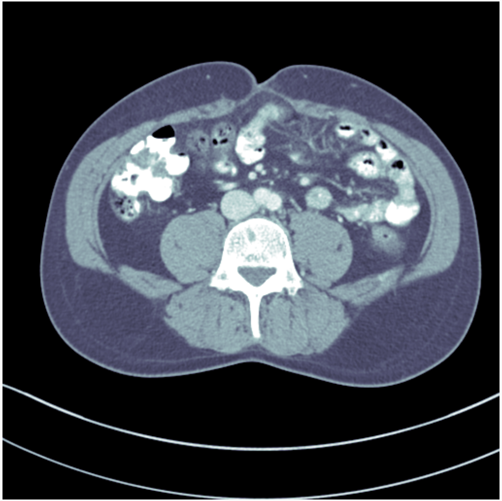
Posterior sample 2



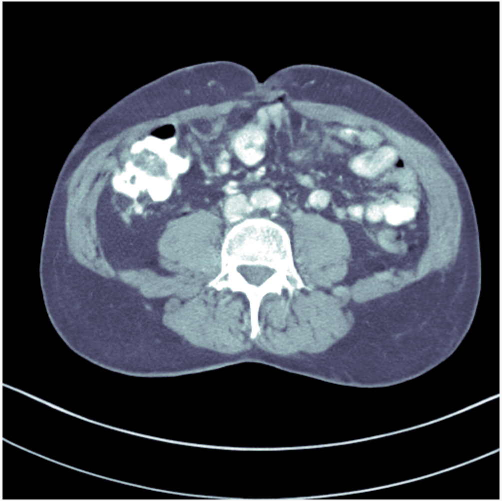
Phantom



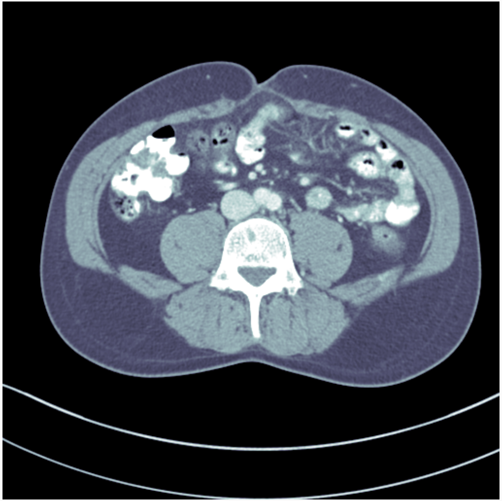
Posterior sample 3



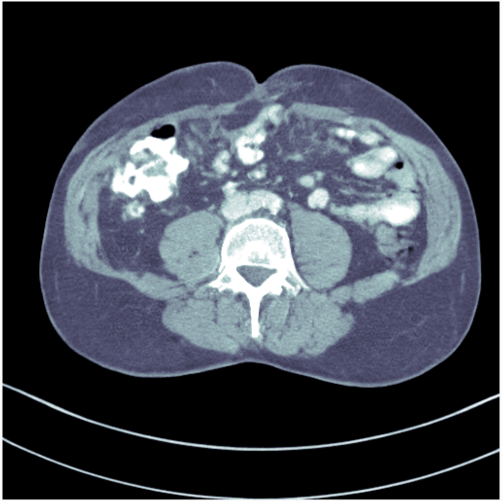
Phantom



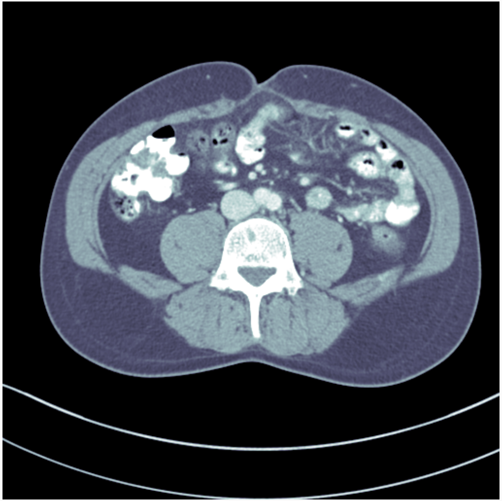
Posterior sample 4



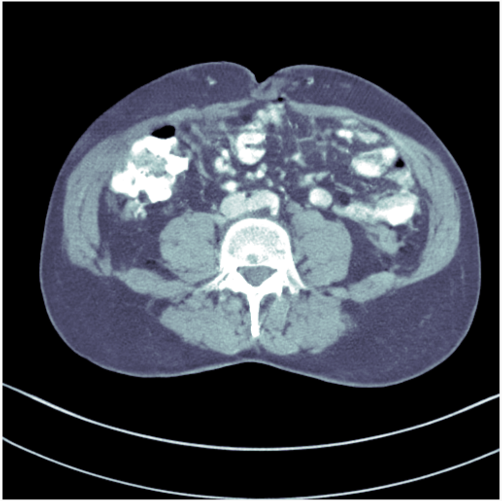
Phantom



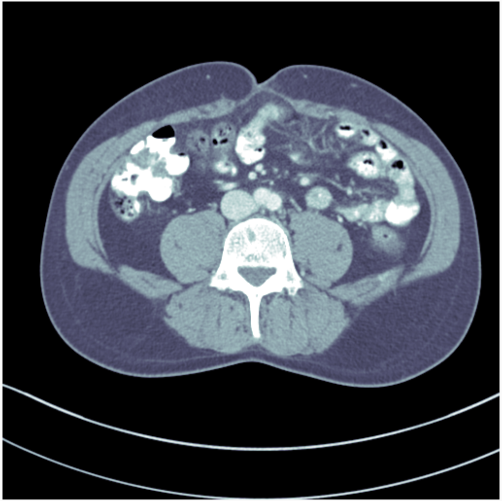
Posterior sample 5



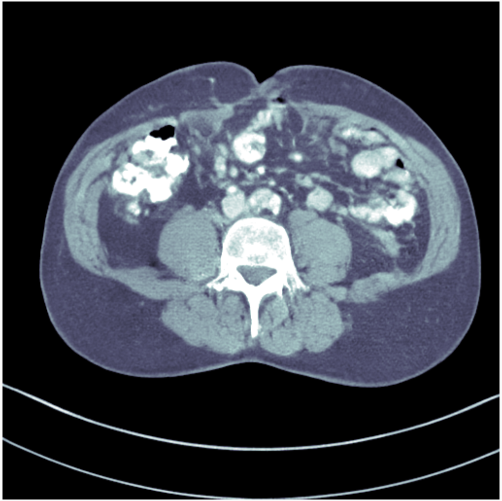
Phantom



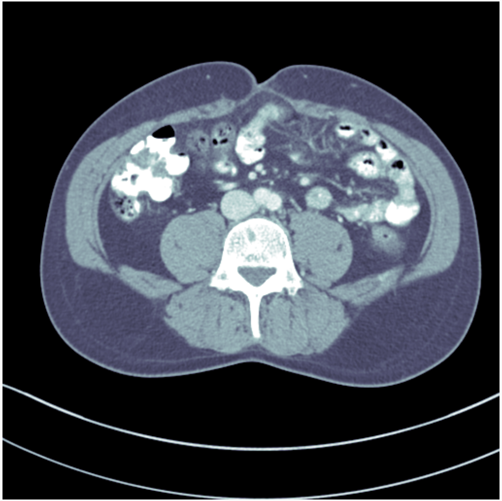
Posterior sample 6



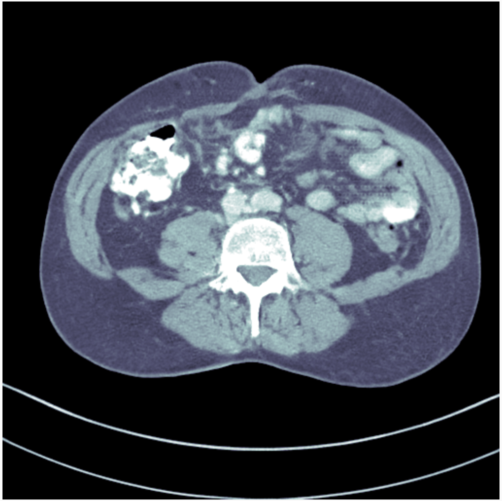
Phantom



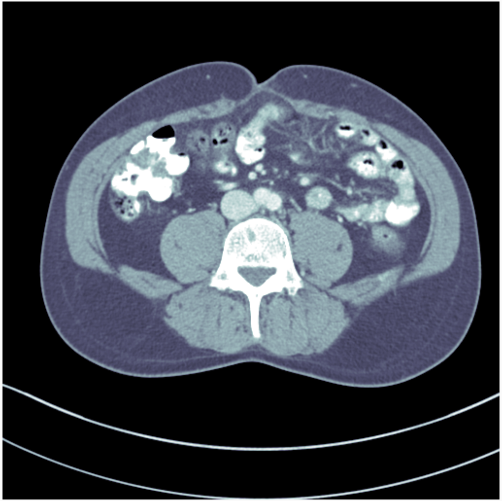
Posterior sample 7



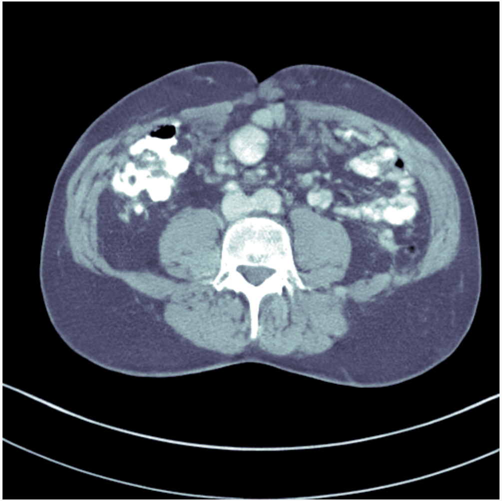
Phantom



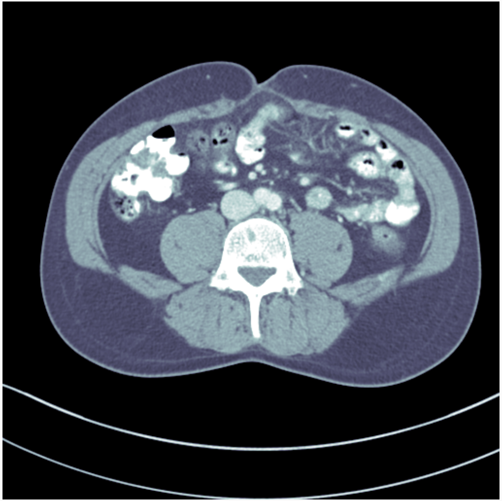
Posterior sample 8



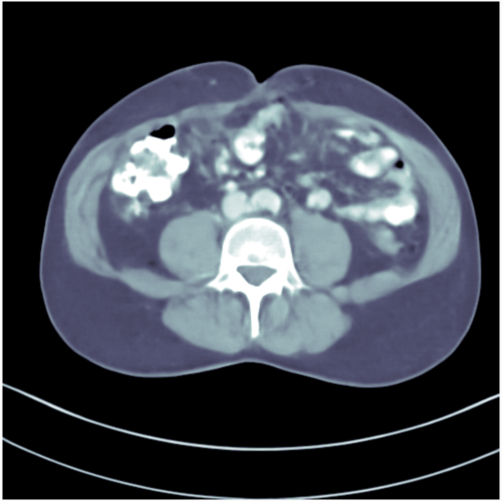
Phantom



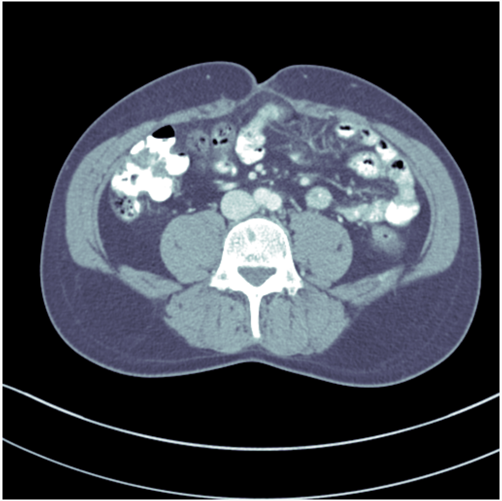
Posterior sample 9



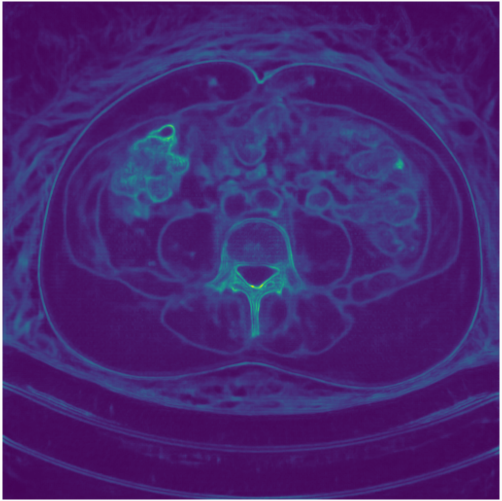
Phantom



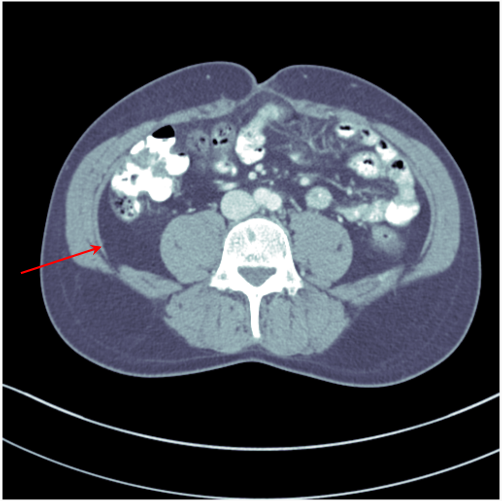
Conditional mean (1000 samples)



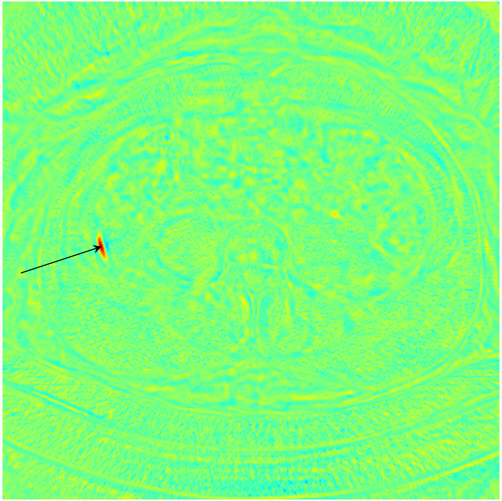
Phantom



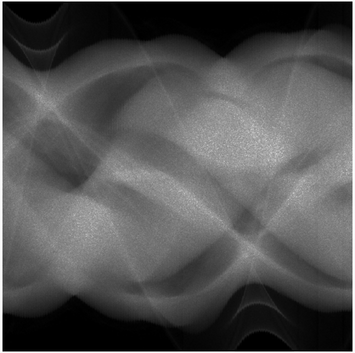
Standard deviation



Phantom



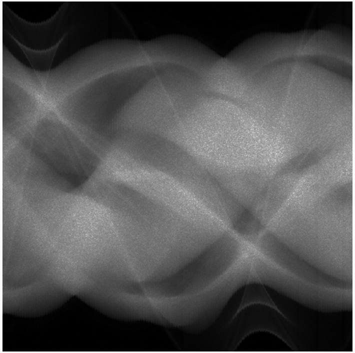
Correlation w.r.t. single point



Data

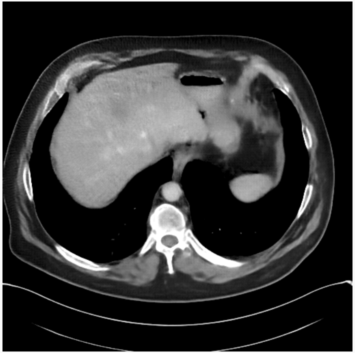
FBP

- Case: Patient with suspected metastasis to the liver.
- Data: Clinical helical 3D CT data, 2% of a normal dose (ultra low-dose).
- Liver lesion: $\triangle$ = difference in average contrast between ROI and liver.
- Hypothesis test: Based on 1 000 samples, the ROI contains a lesion at 95%



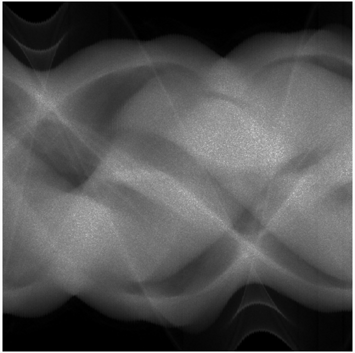
Data

FBP



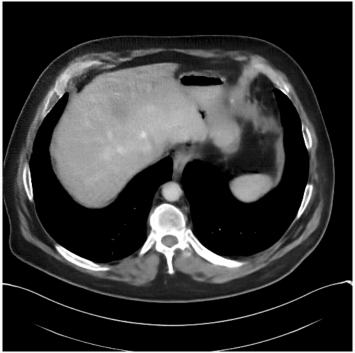
Posterior mean

- Case: Patient with suspected metastasis to the liver.
- Data: Clinical helical 3D CT data, 2% of a normal dose (ultra low-dose).
- Liver lesion: $\triangle =$ difference in average contrast between ROI and liver.
- Hypothesis test: Based on 1 000 samples, the ROI contains a lesion at 95%



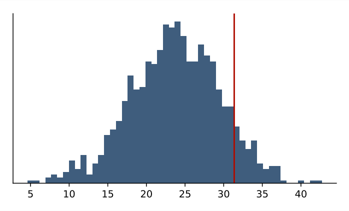
Data

FBP

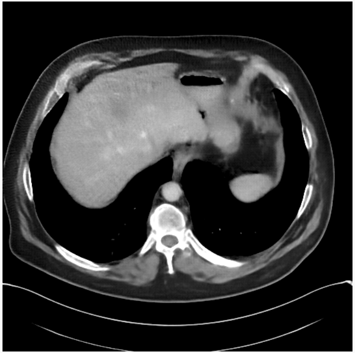


Posterior mean

- Case: Patient with suspected metastasis to the liver.
- Data: Clinical helical 3D CT data, 2% of a normal dose (ultra low-dose).
- Liver lesion: $\triangle =$ difference in average contrast between ROI and liver.
- Hypothesis test: Based on 1 000 samples, the ROI contains a lesion at 95%

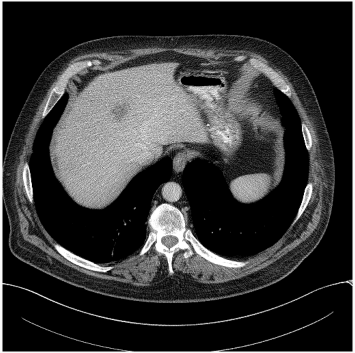


Histogram of $\triangle$

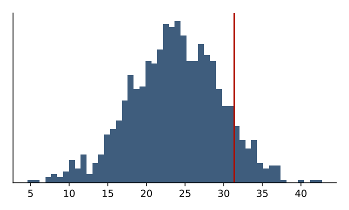


Posterior mean with ROI

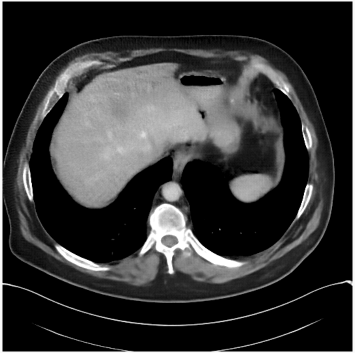
- Case: Patient with suspected metastasis to the liver.
- Data: Clinical helical 3D CT data, 2% of a normal dose (ultra low-dose).
- Liver lesion: $\triangle =$ difference in average contrast between ROI and liver.
- Hypothesis test: Based on 1 000 samples, the ROI contains a lesion at 95%



Normal dose image



Histogram of $\triangle$



Posterior mean with ROI

- Case: Patient with suspected metastasis to the liver.
- Data: Clinical helical 3D CT data, 2% of a normal dose (ultra low-dose).
- Liver lesion: $\triangle =$ difference in average contrast between ROI and liver.
- Hypothesis test: Based on 1 000 samples, the ROI contains a lesion at 95%

Deep posterior sampling

- Images: 512×512 pixel slices.
- Generating a sample from the posterior takes about 10 ms on a ‘gaming’ PC.
- Training the generator and discriminator used 2000 pairs (x_i, y_i) from 9 patients.
- Generator and discriminator over-parametrised, about 10^8 parameters

DLIP: Deep Learning in Inverse Problems

Overview

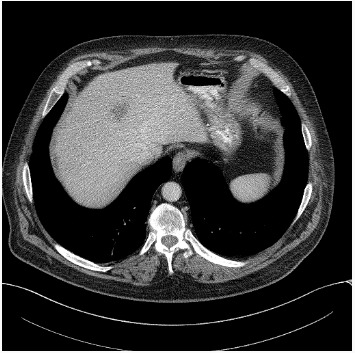
Deep Learning and Inverse Problems 2019 (DLIP2019) is a one week workshop for researchers and practitioners working on deep learning techniques for inverse problems. Our objective is to enable open discussions on both practical and theoretical aspects, and give researchers time to discuss these problems in depth.

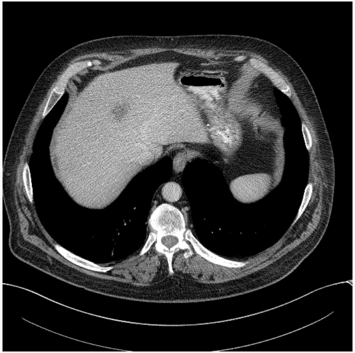
The workshop will feature some invited talks, but we hope that most attendants will also contribute with their own knowledge.

Post doctoral fellowship

Postdoc in Deep Learning based Reconstruction for Spectral-CT

- Theory and methods for machine learning in image reconstruction in spectral-CT.
- Opportunity to work with photon counting spectral-CT in a clinical setting.
- Pursued jointly with MedTechLabs and the Medical Imaging group at KTH led by Mats Danielsson.
- Contact: Ozan Öktem (e-mail: ozan@kth.se) for more information.



Thank you for your attention!

References

References

Adler, J., Lunz, S., Verdier, O., Schönlieb, C.-B., & Öktem, O. (2018a). Task adapted reconstruction for inverse problems. *ArXiv e-print, cs.CV*(1809.00948).

Adler, J., Lunz, S., Verdier, O., Schönlieb, C.-B., & Öktem, O. (2018b). Task adapted reconstruction for inverse problems. In *NIPS 2018 meets medical imaging: Workshop within the 32nd annual conference on neural information processing systems (NIPS 2018).* (Also available as arXiv e-print: https://arxiv.org/abs/1809.00948)

Adler, J., & Öktem, O. (2017). Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, *33*(12), 124007 (24pp). (Special issue: 100 Years of the Radon transform)

Adler, J., & Öktem, O. (2018a). Deep Bayesian inversion: Computational uncertainty quantification for large scale inverse problems. *ArXiv e-print*, *stat.ML*(1811.05910).

Adler, J., & Öktem, O. (2018b). Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging*, *37*(6), 1322–1332.

Dalca, A. V., Balakrishnan, G., Guttag, J., & Sabuncu, M. R. (2018). Unsupervised learning for fast probabilistic diffeomorphic registration. In F. A. F., J. A. Schnabel, C. Davatzikos, L. C. Alberola, & G. Fichtinger (Eds.), *21st international conference on medical image computing and computer assisted intervention MICCAI 2018* (Vol. 11070, pp. 729–738). Springer Verlag.

Farabet, C., Couprie, C., Najman, L., & LeCun, Y. (2013). Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, *35*(8), 1915–1929.

Ghosal, S., & Ray, N. (2017). Deep deformable registration: Enhancing accuracy by fully convolutional neural net. *Pattern Recognition Letters*, *94*, 81–86.

Guo, Y., Liu, Y., Georgiou, T., & Lew, M. S. (2018). A review of semantic segmentation using deep neural networks. *International Journal of Multimedia Information Retrieval*, *7*(2), 87–93.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 770–778).

Karpathy, A., & Fei-Fei, L. (2017). Deep visual-semantic alignments for generating image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(4), 664–676.

Li, C. Y., Liang, X., Hu, Z., & Xing, E. P. (2018). Hybrid retrieval-generation reinforced agent for medical image report generation. *ArXiv e-print*, *cs.CV*(1805.08298).

Romano, Y., Isidoro, J., & Milanfar, P. (2017). RAISR: Rapid and accurate image super resolution. *IEEE Transactions on Computational Imaging*, *3*(1), 110–125.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & LeCun, Y. (2013). OverFeat: Integrated recognition, localization and detection using convolutional networks. *ArXiv e-print*, *cs.CV*(1312.6229).

Syu, N.-S., Chen, Y.-S., & Chuang, Y.-Y. (2018). Learning deep convolutional networks for demosaicing. *ArXiv e-print*, *cs.CV*(1802.03769).

Thoma, M. (2016). A survey of semantic segmentation. *ArXiv e-print, cs.CV*(1602.06541).

Wolterink, J. M., Dinkla, A. M., Savenije, M. H. F., Seevinck, P. R., van den Berg, C. A. T., & Išgum, I. (2017). Deep MR to CT synthesis using unpaired data. In S. Tsaftaris, A. Gooya, A. Frangi, & J. Prince (Eds.), *Simulation and synthesis in medical imaging. SASHIMI 2017* (Vol. 10557, pp. 14–23).

Xie, J., Xu, L., & Chen, E. (2012). Image denoising and inpainting with deep neural networks. In *Proceedings of the 25th international conference on neural information processing systems (nips 2012)* (pp. 341–349).