Randomize-then-optimize, the saga continues: a sampling method for large-scale inverse problems

John Bardsley, U. Montana

joint work with: C. Fox, H. Haario, J. Kaipio, M. Howard, J. Nagy, A. Seppänen, A. Solonen

Outline

- What is an 'inverse problem'?
- Bayesian solutions of inverse problems and sampling from the posterior:
  - linear cases: deblurring, tomography,
  - nonlinear cases: nonnegativity constraints, Poisson noise, PET, EIT.
- Numerical examples.

Inverse problems as linear models

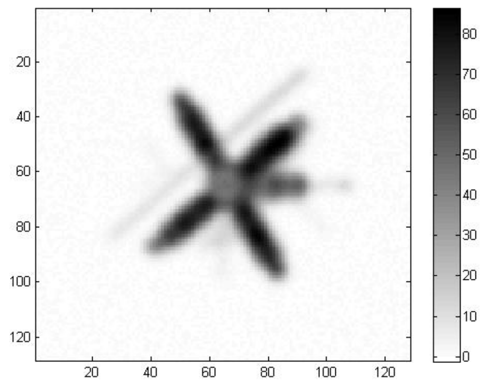
We begin by considering linear models of the form:

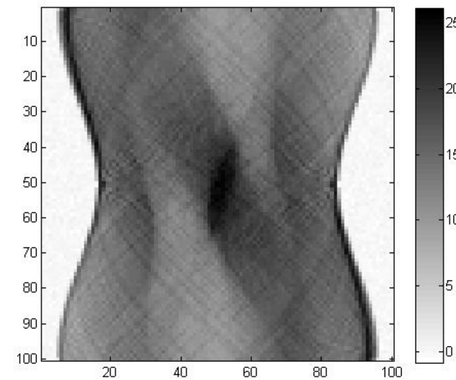
$$\mathbf{b} = \mathbf{A}\mathbf{x} + \boldsymbol{\epsilon},$$

- $\mathbf{b}$ is the $n \times 1$ data vector,
- $\mathbf{A}$ is the $n \times n$ forward map,
- $\mathbf{x}$ is the $n \times 1$ unknown,
- $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_n)$ is the $n \times 1$ iid Gaussian noise vector.

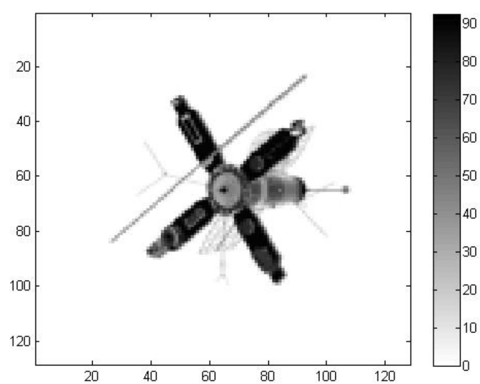
Some examples of linear problems

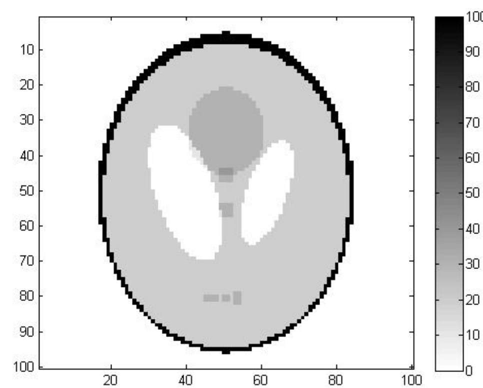
Data $\mathbf{b}$ examples:





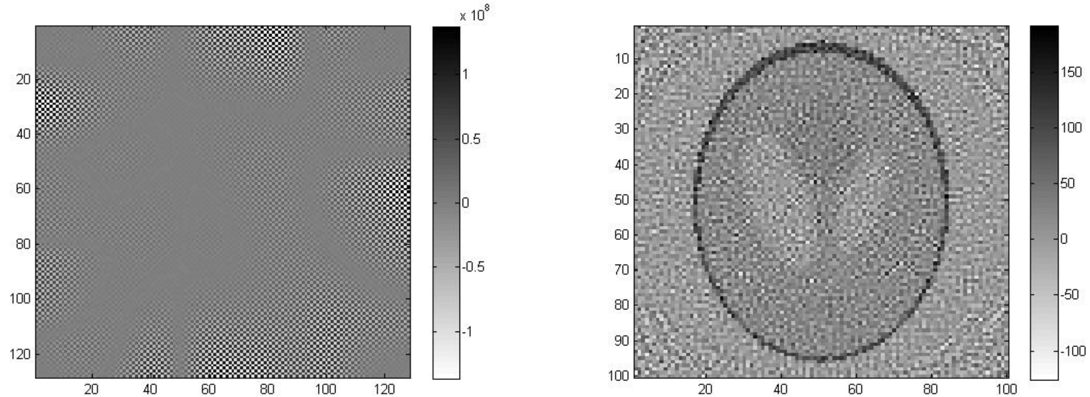
Corresponding true images $\mathbf{x}$:



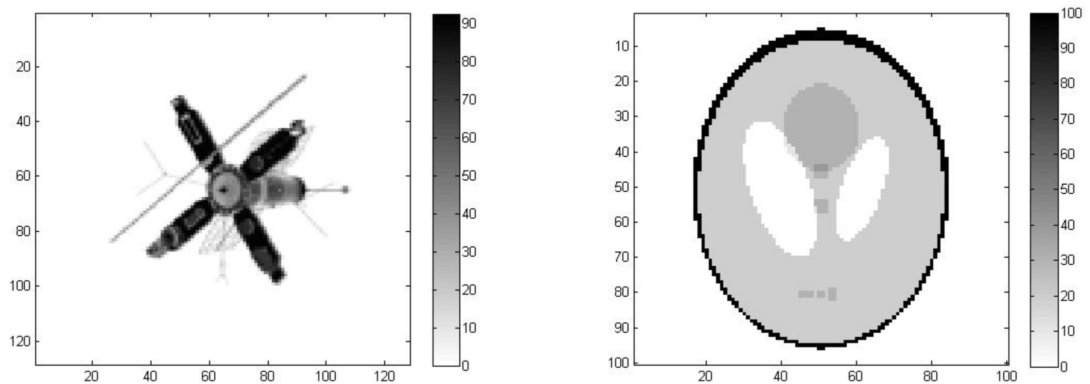


Naive Solutions

Naive solutions $\mathbf{A}^{-1}\mathbf{b}$:



Corresponding true images $\mathbf{x}$:



What characterizes an inverse problem?

Consider the continuous model (pre-discretization)

$$b = Ax,$$

where b and x are functions and A is an operator.

What characterizes an inverse problem?

Consider the continuous model (pre-discretization)

$$b = Ax,$$

where b and x are functions and A is an operator.

The singular value expansion (SVE) of A has the form

$$A(\cdot) = \sum_{i=1}^{\infty} \sigma_i u_i \langle v_i, \cdot \rangle,$$

with (u_i, v_i) the left and right singular functions, and $\sigma_i \to 0$.

What characterizes an inverse problem?

Consider the continuous model (pre-discretization)

$$b = Ax,$$

where b and x are functions and A is an operator.

The singular value expansion (SVE) of A has the form

$$A(\cdot) = \sum_{i=1}^{\infty} \sigma_i u_i \langle v_i, \cdot \rangle,$$

with (u_i, v_i) the left and right singular functions, and $\sigma_i \to 0$.

Then the SVE of A^{-1} is

$$A^{-1}(\cdot) = \sum_{i=1}^{\infty} \frac{v_i \langle u_i, \cdot \rangle}{\sigma_i},$$

which is unbounded: $\|A^{-1}\|_2^2 = \sum_{i=1}^{\infty} \left(\frac{1}{\sigma_i}\right)^2 = \infty$.

What characterizes an inverse problem?

After discretization, we have the matrix $\mathbf{A}$ with SVD

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sum_{i=1}^{n} \sigma_i \mathbf{u}_i \mathbf{v}_i^T.$$

with n large, the σ_i's clustering near 0. Hence $\|\mathbf{A}^{-1}\|_2$ is huge.

What characterizes an inverse problem?

After discretization, we have the matrix $\mathbf{A}$ with SVD

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T = \sum_{i=1}^{n} \sigma_i \mathbf{u}_i \mathbf{v}_i^T.$$

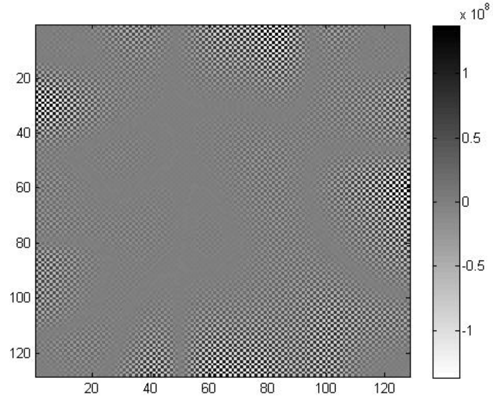
with n large, the σ_i's clustering near 0. Hence $\|\mathbf{A}^{-1}\|_2$ is huge.

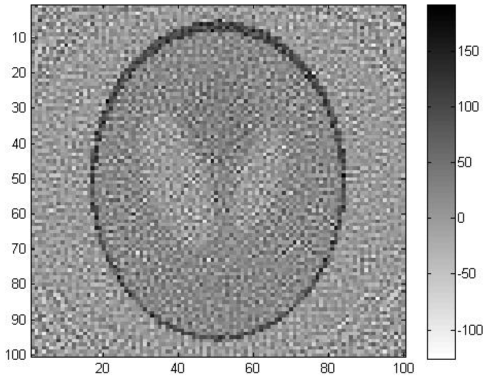
The naive solution can then be written

$$\begin{aligned} \mathbf{A}^{-1}\mathbf{b} &= \mathbf{A}^{-1}(\mathbf{A}\mathbf{x} + \boldsymbol{\epsilon}) \\ &= \mathbf{x} + \mathbf{A}^{-1}\boldsymbol{\epsilon} \\ &= \mathbf{x} + \underbrace{\sum_{i=1}^{n} \left(\frac{\mathbf{u}_i^T \boldsymbol{\epsilon}}{\sigma_i} \right) \mathbf{v}_i}_{\text{dominates}} \end{aligned}$$

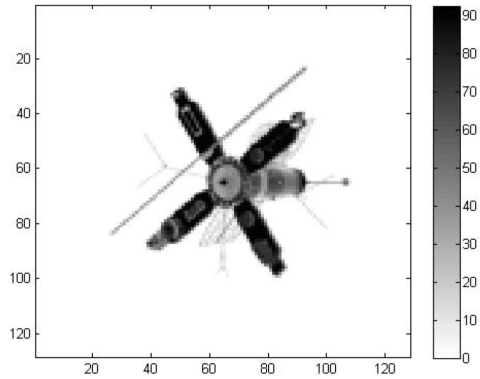
Naive Solutions

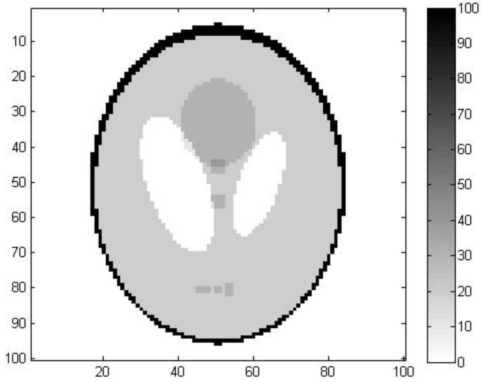
Naive solutions $\mathbf{A}^{-1}\mathbf{b} = \mathbf{x} + \sum_{i=1}^{n} \sigma_i^{-1}(\mathbf{u}_i^T \boldsymbol{\epsilon})\mathbf{v}_i$:



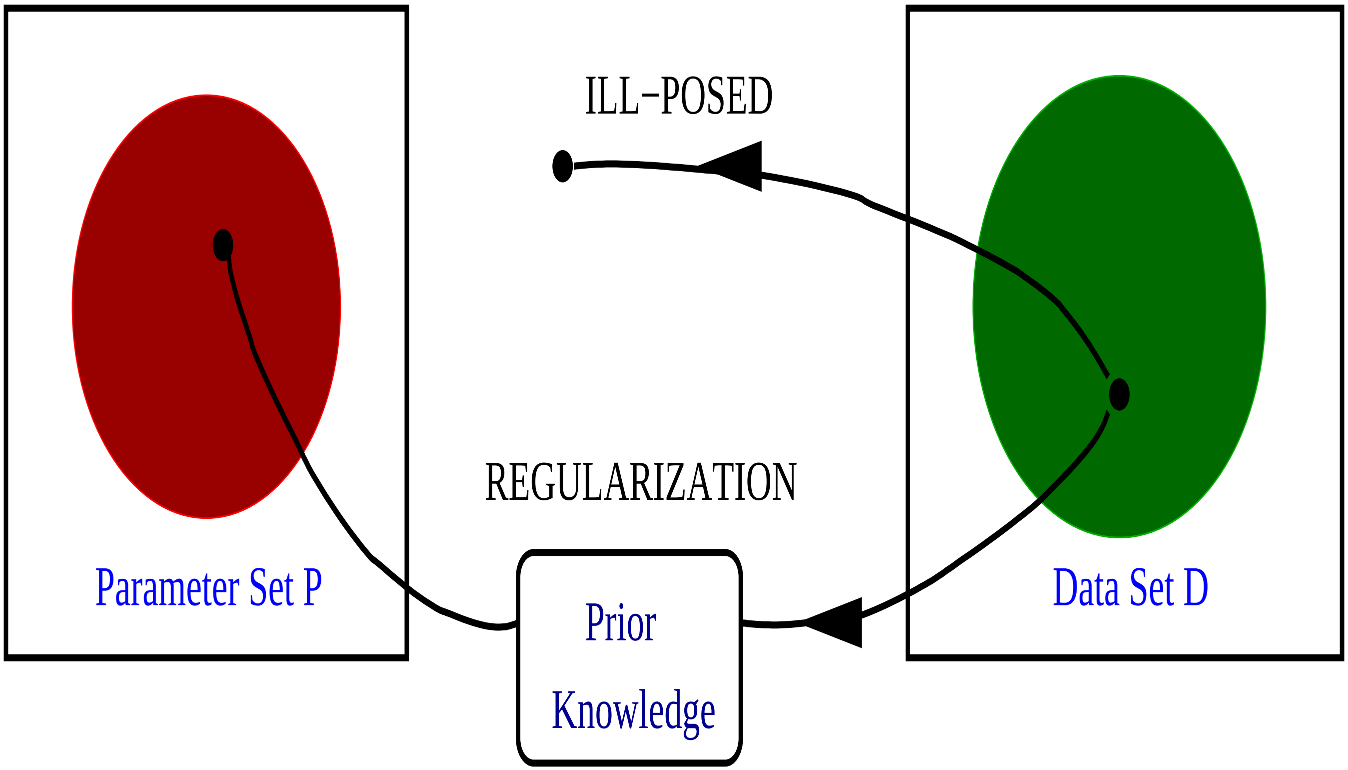


Corresponding true images $\mathbf{x}$:





The Fix: Regularization



Bayes Law and Regularization

Bayes' Law:

$$\underbrace{p(\mathbf{x}|\mathbf{b},\lambda,\delta)}_{\text{posterior}} \propto \underbrace{p(\mathbf{b}|\mathbf{x},\lambda)}_{\text{likelihood}}\,\underbrace{p(\mathbf{x}|\delta)}_{\text{prior}}\,.$$

Bayes Law and Regularization

Bayes' Law:

$$\underbrace{p(\mathbf{x}|\mathbf{b},\lambda,\delta)}_{\text{posterior}} \propto \underbrace{p(\mathbf{b}|\mathbf{x},\lambda)}_{\text{likelihood}}\,\underbrace{p(\mathbf{x}|\delta)}_{\text{prior}}\,.$$

For our statistical model, with $\lambda = 1/\sigma^2$,

$$p(\mathbf{b}|\mathbf{x},\lambda) \propto \exp\left(-\frac{\lambda}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2\right).$$

Bayes Law and Regularization

Bayes' Law:

$$\underbrace{p(\mathbf{x}|\mathbf{b},\lambda,\delta)}_{\text{posterior}} \propto \underbrace{p(\mathbf{b}|\mathbf{x},\lambda)}_{\text{likelihood}}\underbrace{p(\mathbf{x}|\delta)}_{\text{prior}}.$$

For our statistical model, with $\lambda = 1/\sigma^2$,

$$p(\mathbf{b}|\mathbf{x},\lambda) \propto \exp\left(-\frac{\lambda}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2\right).$$

And we assume that the prior has the form

$$p(\mathbf{x}|\delta) \propto \exp\left(-\frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x}\right),$$

Gaussian Markov Random field priors

The neighbor values for x_{ij} are below (in black)

$$
\begin{aligned}
\mathbf{x}_{\partial_{ij}} &= \{x_{i-1,j}, x_{i,j-1}, x_{i+1,j}, x_{i,j+1}\} \\
&= \begin{bmatrix} & x_{i,j+1} & \\ x_{i-1,j} & x_{ij} & x_{i+1,j} \\ & x_{i,j-1} & \end{bmatrix}.
\end{aligned}
$$

Gaussian Markov Random field priors

The neighbor values for x_{ij} are below (in black)

$$\mathbf{x}_{\partial_{ij}} = \{x_{i-1,j}, x_{i,j-1}, x_{i+1,j}, x_{i,j+1}\}$$

$$= \begin{bmatrix} & x_{i,j+1} & \\ x_{i-1,j} & x_{ij} & x_{i+1,j} \\ & x_{i,j-1} & \end{bmatrix}.$$

Then we assume

$$x_{i,j} | \mathbf{x}_{\partial_{i,j}} \sim \mathcal{N}\left(\bar{x}_{\partial_{i,j}}, \frac{h^2}{4\delta}\right),$$

where $\bar{x}_{ij} = \frac{1}{4}(x_{i-1,j} + x_{i,j-1} + x_{i+1,j} + x_{i,j+1})$.

Gaussian Markov Random field priors

This leads to the prior

$$p(\mathbf{x}|\delta) \propto \delta^n \exp\left(-\frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x}\right),$$

where if $r = (i, j)$ after column-stacking 2D arrays

$$[\mathbf{L}]_{rs} = \frac{1}{h^2}\begin{cases} 4 & s = r, \\ -1 & s \in \partial_r, \\ 0 & \text{otherwise.} \end{cases}$$

Gaussian Markov Random field priors

This leads to the prior

$$p(\mathbf{x}|\delta) \propto \delta^n \,\mathrm{exp}\left(-\frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x}\right),$$

where if $r = (i, j)$ after column-stacking 2D arrays

$$[\mathbf{L}]_{rs} = \frac{1}{h^2}\begin{cases} 4 & s = r, \\ -1 & s \in \partial_r, \\ 0 & \text{otherwise.} \end{cases}$$

NOTES:

1. $\mathbf{L}$ is the negative, 2D Laplacian.
2. Boundary conditions must be imposed. We have considered Dirichlet, periodic, and Neumann.

Bayes Law and Regularization

The maximizer of the posterior density is

$$\mathbf{x}_{\mathsf{MAP}} = \arg\min_{\mathbf{x}} \left\{ \frac{\lambda}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x} \right\}$$

which is the regularized solution $\mathbf{x}_\alpha$ with $\alpha = \delta/\lambda$.

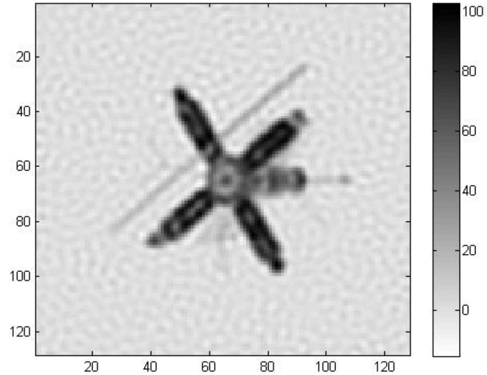
Bayes Law and Regularization

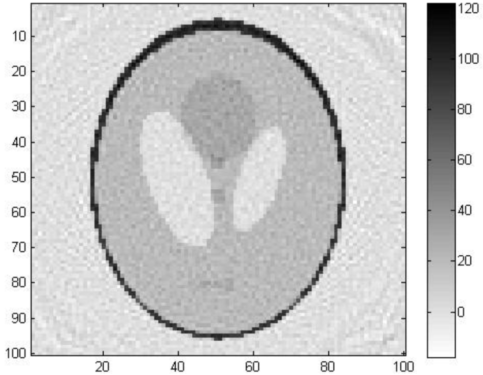
The maximizer of the posterior density is

$$\mathbf{x}_{\mathsf{MAP}} = \arg\min_{\mathbf{x}} \left\{ \frac{\lambda}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \frac{\delta}{2} \mathbf{x}^T \mathbf{L} \mathbf{x} \right\}$$

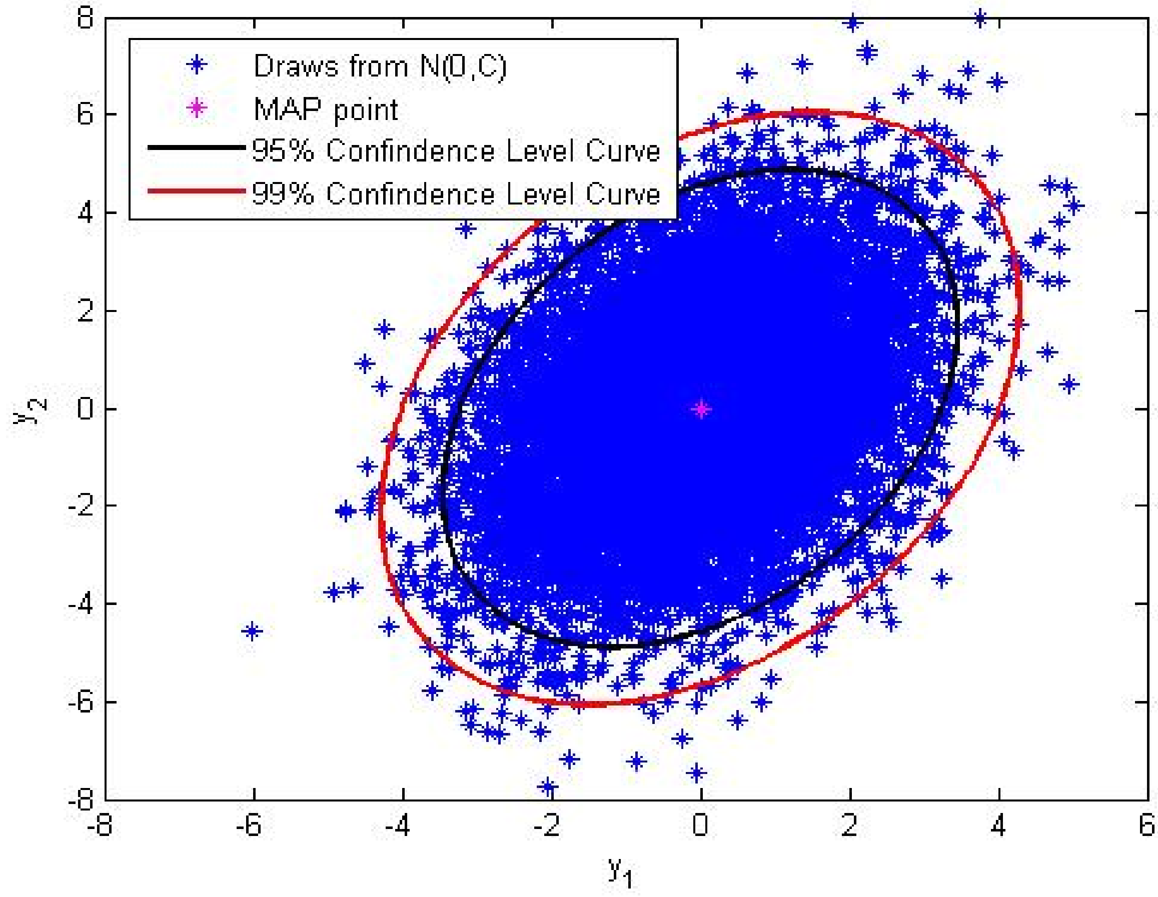
which is the regularized solution $\mathbf{x}_\alpha$ with $\alpha = \delta/\lambda$.

$\alpha = 2.5 \times 10^{-4}$ $\alpha = 1.05 \times 10^{-4}$.





Sampling vs. Computing the MAP



Bayesian Hierarchical Models for λ and δ

Uncertainty in λ and δ: $\lambda \sim p(\lambda)$ and $\delta \sim p(\delta)$. Then

$$p(\mathbf{x}, \lambda, \delta | \mathbf{b}) \propto p(\mathbf{b} | \mathbf{x}, \lambda) p(\lambda) \, p(\mathbf{x} | \delta) p(\delta),$$

is the Bayesian posterior

Bayesian Hierarchical Models for λ and δ

Uncertainty in λ and δ: $\lambda \sim p(\lambda)$ and $\delta \sim p(\delta)$. Then

$$p(\mathbf{x}, \lambda, \delta | \mathbf{b}) \propto p(\mathbf{b}|\mathbf{x}, \lambda) p(\lambda)\, p(\mathbf{x}|\delta) p(\delta),$$

is the Bayesian posterior, where

$$p(\mathbf{b}|\mathbf{x}, \lambda) \propto \lambda^{n/2} \exp\left(-\frac{\lambda}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2\right),$$

$$p(\mathbf{x}|\delta) \propto \delta^{n/2} \exp\left(-\frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x}\right).$$

Bayesian Hierarchical Models for λ and δ

Uncertainty in λ and δ: $\lambda \sim p(\lambda)$ and $\delta \sim p(\delta)$. Then

$$p(\mathbf{x},\lambda,\delta|\mathbf{b}) \propto p(\mathbf{b}|\mathbf{x},\lambda)p(\lambda)\,p(\mathbf{x}|\delta)p(\delta),$$

is the Bayesian posterior, where

$$p(\mathbf{b}|\mathbf{x},\lambda) \propto \lambda^{n/2}\exp\left(-\frac{\lambda}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2\right),$$

$$p(\mathbf{x}|\delta) \propto \delta^{n/2}\exp\left(-\frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x}\right).$$

$$\begin{aligned} p(\lambda) &\propto \lambda^{\alpha_\lambda-1}\exp(-\beta_\lambda\lambda) \\ p(\delta) &\propto \delta^{\alpha_\delta-1}\exp(-\beta_\delta\delta), \end{aligned}$$

where $\alpha_\lambda=\alpha_\delta=1$ and $\beta_\lambda=\beta_\delta=10^{-4}$, and hence

$$\text{mean}=\alpha/\beta=10^4, \quad \text{var}=\alpha/\beta^2=10^8.$$

The Full Posterior Distribution

$p(\mathbf{x}, \lambda, \delta | \mathbf{b}) \propto$ the posterior

$$\lambda^{n/2+\alpha_\lambda-1}\delta^{n/2+\alpha_\delta-1}\,\mathrm{exp}\left(-\frac{\lambda}{2}\|\mathbf{Ax}-\mathbf{b}\|^2-\frac{\delta}{2}\mathbf{x}^T\mathbf{Lx}-\beta_\lambda\lambda-\beta_\delta\delta\right).$$

The Full Posterior Distribution

$p(\mathbf{x}, \lambda, \delta | \mathbf{b}) \propto$ the posterior

$$\lambda^{n/2+\alpha_\lambda-1} \delta^{n/2+\alpha_\delta-1} \exp\left(-\frac{\lambda}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2 - \frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x} - \beta_\lambda\lambda - \beta_\delta\delta\right).$$

By conjugacy, each conditional distribution lives in the same family as the prior/hyper-prior distribution:

$$\mathbf{x}|\lambda, \delta, \mathbf{b} \sim N\left((\lambda\mathbf{A}^T\mathbf{A}+\delta\mathbf{L})^{-1}\lambda\mathbf{A}^T\mathbf{b}, (\lambda\mathbf{A}^T\mathbf{A}+\delta\mathbf{L})^{-1}\right),$$

$$\left.\begin{bmatrix} \lambda \\ \delta \end{bmatrix}\right| \mathbf{x}, \mathbf{b} \sim \Gamma\left(\begin{bmatrix} n/2+\alpha_\lambda \\ n/2+\alpha_\delta \end{bmatrix}, \begin{bmatrix} \frac{1}{2}\|\mathbf{A}\mathbf{x}^k-\mathbf{b}\|^2+\beta_\lambda \\ \frac{1}{2}\|\mathbf{L}^{1/2}\mathbf{x}^k\|^2+\beta_\delta \end{bmatrix}\right);$$

An MCMC Method for sampling from $p(\mathbf{x}, \lambda, \delta|\mathbf{b})$

A Two-Component Gibbs sampler for $p(\mathbf{x}, \delta, \lambda|\mathbf{b})$.

0. δ_0, and λ_0, and set $k = 0$;

1. Compute a sample

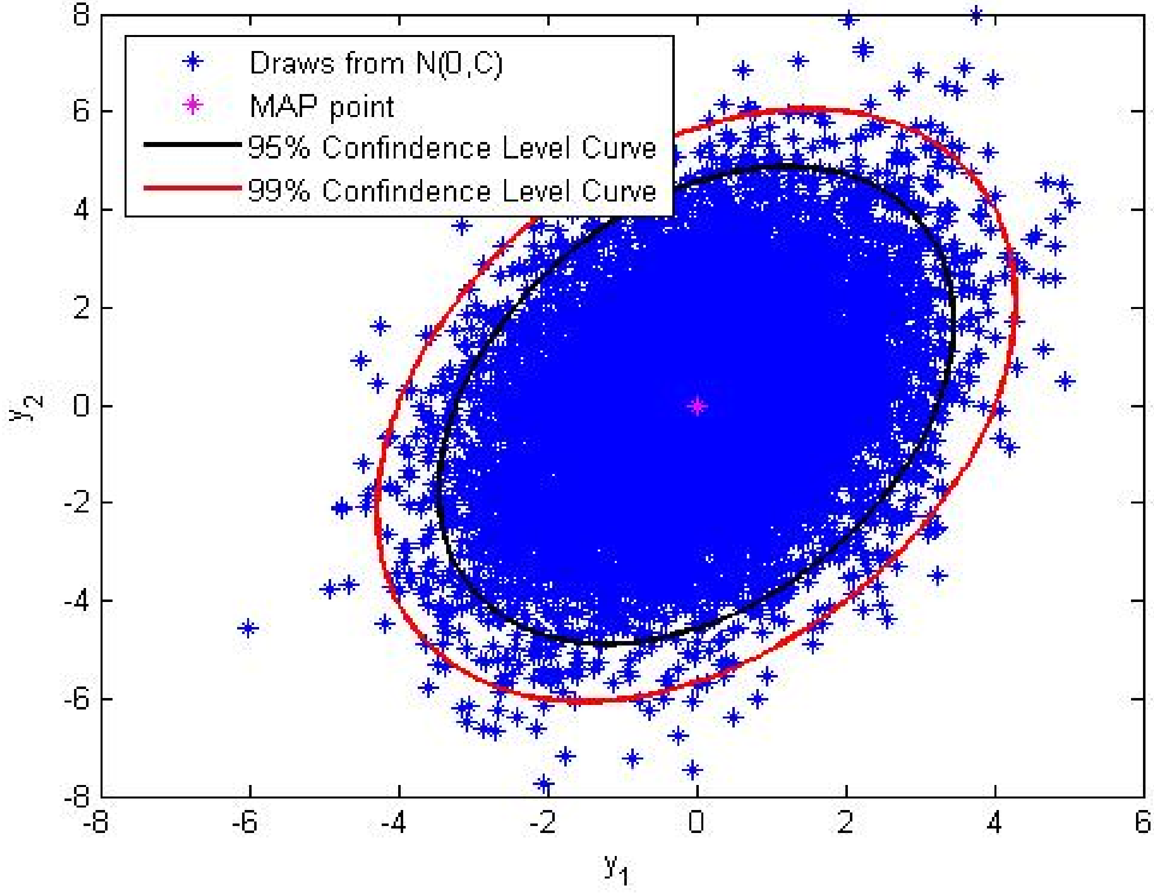
$$\mathbf{x}^{k+1} \sim N\left((\lambda_k \mathbf{A}^T\mathbf{A} + \delta_k \mathbf{L})^{-1}\lambda_k \mathbf{A}^T\mathbf{b}, (\lambda_k \mathbf{A}^T\mathbf{A} + \delta_k \mathbf{L})^{-1}\right);$$

2. Compute a sample

$$\begin{bmatrix} \lambda_{k+1} \\ \delta_{k+1} \end{bmatrix} \sim \Gamma\left(\begin{bmatrix} n/2 + \alpha_\lambda \\ n/2 + \alpha_\delta \end{bmatrix}, \begin{bmatrix} \frac{1}{2}\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|^2 + \beta_\lambda \\ \frac{1}{2}\|\mathbf{L}^{1/2}\mathbf{x}^k\|^2 + \beta_\delta \end{bmatrix}\right);$$

3. Set $k = k + 1$ and return to Step 1.

Sampling vs. Computing the MAP



The Computational Bottleneck: Step 1

The image sample, Step 1

$$\mathbf{x}^k \sim N\left((\lambda_k \mathbf{A}^T\mathbf{A} + \delta_k \mathbf{L})^{-1}\lambda_k \mathbf{A}^T\mathbf{b}, (\lambda_k \mathbf{A}^T\mathbf{A} + \delta_k \mathbf{L})^{-1}\right),$$

can be computed via

$$\begin{aligned} (\lambda_k \mathbf{A}^T\mathbf{A} + \delta_k \mathbf{L})\mathbf{x}^k &= \lambda_k \mathbf{A}^T\mathbf{b} + \mathbf{w}, \\ \mathbf{w} &\sim N(\mathbf{0}, \lambda_k \mathbf{A}^T\mathbf{A} + \delta_k \mathbf{L}), \end{aligned}$$

Notice that $\mathbf{w}$ can be computed cheaply:

$$\mathbf{w} = \sqrt{\lambda_k}\mathbf{A}^T\mathbf{v} + \sqrt{\delta_k}\mathbf{L}^{1/2}\mathbf{v}, \quad \mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n).$$

Direct Two-Component Gibbs Sampler

0. δ_0, and λ_0, and set $k = 0$;

1. First generate

$$\mathbf{w} = \sqrt{\lambda_k}\mathbf{A}^T\mathbf{v} + \sqrt{\delta_k}\mathbf{L}^{1/2}\mathbf{v}, \quad \mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n),$$

   then compute a sample

$$\mathbf{x}^{k+1} = (\lambda_k \mathbf{A}^T\mathbf{A} + \delta_k \mathbf{L})^{-1}(\lambda_k \mathbf{A}^T\mathbf{b} + \mathbf{w}).$$

2. Compute a sample

$$\begin{bmatrix} \lambda_{k+1} \\ \delta_{k+1} \end{bmatrix} \sim \Gamma\left(\begin{bmatrix} n/2 + \alpha_\lambda \\ n/2 + \alpha_\delta \end{bmatrix}, \begin{bmatrix} \frac{1}{2}\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|^2 + \beta_\lambda \\ \frac{1}{2}\|\mathbf{L}^{1/2}\mathbf{x}^k\|^2 + \beta_\delta \end{bmatrix} \right).$$

3. Set $k = k + 1$ and return to Step 1.

Assessing MCMC chain convergence

n_r chains, each of length n_s, with $\{\psi_{ij}\}$ the computed samples. Define

$$B = \frac{n_s}{n_r - 1}\sum_{j=1}^{n_r}(\overline{\psi}_{.j} - \overline{\psi}_{..})^2, \quad \overline{\psi}_{.j} = \frac{1}{n_s}\sum_{i=1}^{n_s}\psi_{ij}, \quad \overline{\psi}_{..} = \frac{1}{n_r}\sum_{j=1}^{n_r}\overline{\psi}_{.j}\,;$$

and

$$W = \frac{1}{n_r}\sum_{j=1}^{n_r} s_j^2, \quad \text{where} \quad s_j^2 = \frac{1}{n_s - 1}\sum_{i=1}^{n_s}(\psi_{ij} - \overline{\psi}_{.j})^2.$$

Then marginal posterior variance $\text{var}(\psi|\mathbf{b})$ can then be estimated by

$$\widehat{\text{var}}^+(\psi|\mathbf{b}) = \frac{n_s - 1}{n_s}W + \frac{1}{n_s}B,$$

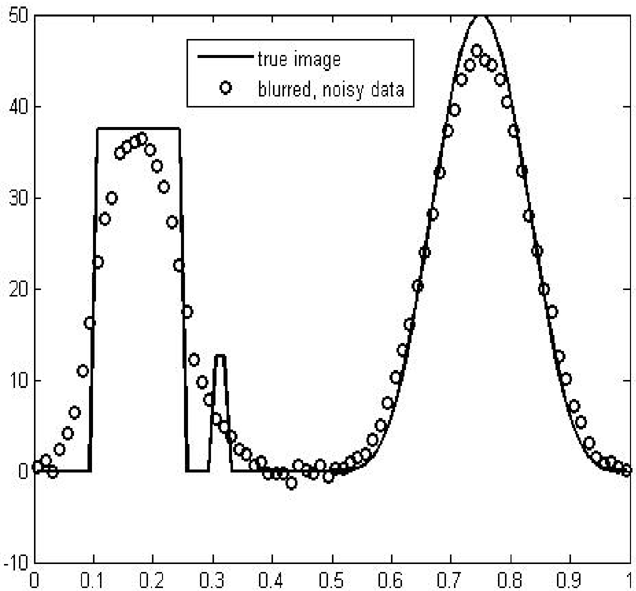
We monitor

$$\hat{R} = \sqrt{\frac{\widehat{\text{var}}^+(\psi|\mathbf{b})}{W}}, \tag{1}$$

which declines to 1 as $n_s \to \infty$.

A One-dimensional example

Sample median



Confidence Images in 1D.

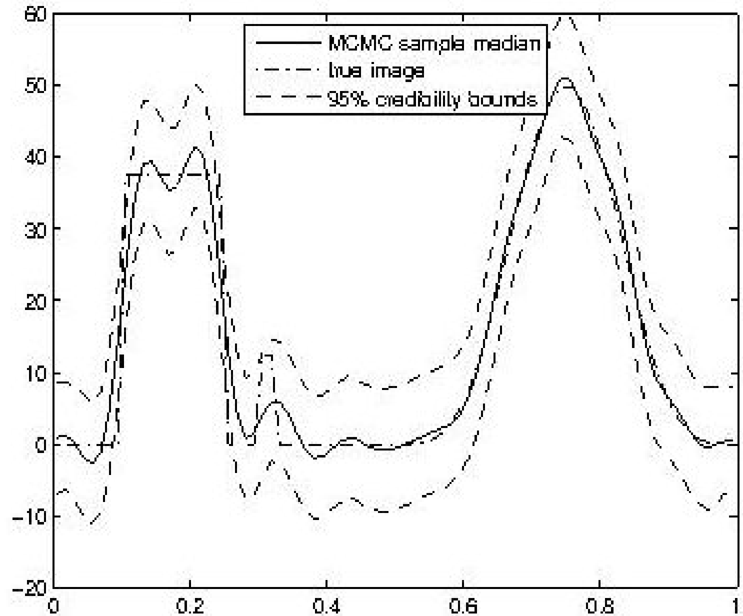


Image Deblurring: Boundary Conditions in 2D

correspond to assumptions about the values of the unknown outside of the computational domain. We consider three:

$$
\text{Periodic}: \quad \begin{array}{ccc} \mathbf{X} & \mathbf{X} & \mathbf{X} \\ \mathbf{X} & \mathbf{X} & \mathbf{X} \\ \mathbf{X} & \mathbf{X} & \mathbf{X} \end{array},
$$

$$
\text{Neumann}: \quad \begin{array}{ccc} \mathbf{X}_{vh} & \mathbf{X}_h & \mathbf{X}_{vh} \\ \mathbf{X}_v & \mathbf{X} & \mathbf{X}_v \\ \mathbf{X}_{vh} & \mathbf{X}_h & \mathbf{X}_{vh} \end{array},
$$

$$
\text{Dirichlet}: \quad \begin{array}{ccc} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{X} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}.
$$

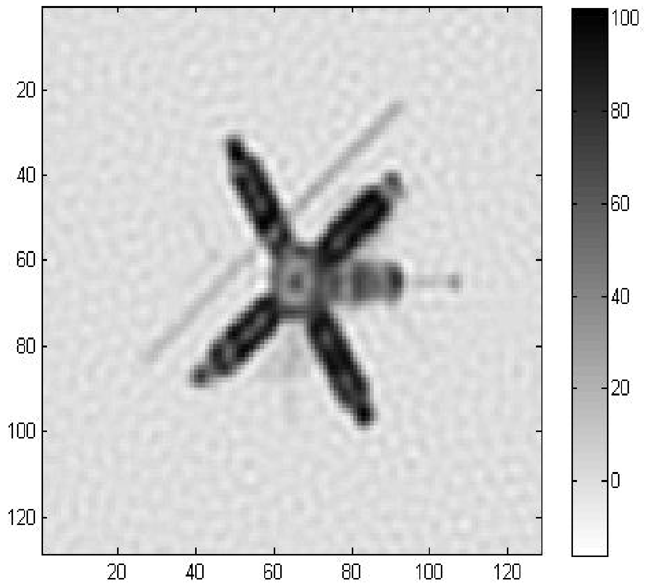
Periodic boundary conditions

In this case you can efficiently compute

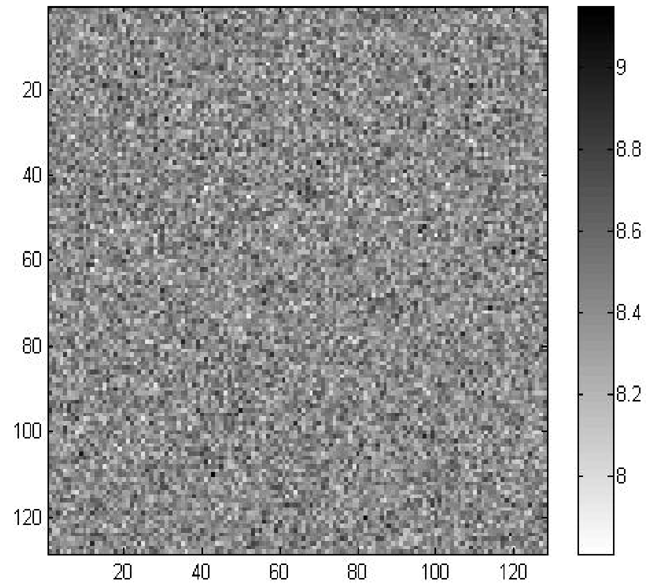
$$\mathbf{x}^k = (\lambda_k \mathbf{A}^T\mathbf{A} + \delta_k \mathbf{L})^{-1}(\lambda_k \mathbf{A}^T\mathbf{b} + \mathbf{w}).$$

Here $\mathbf{A}$ and $\mathbf{L}$ are diagonalizable by the 2d-DFT.

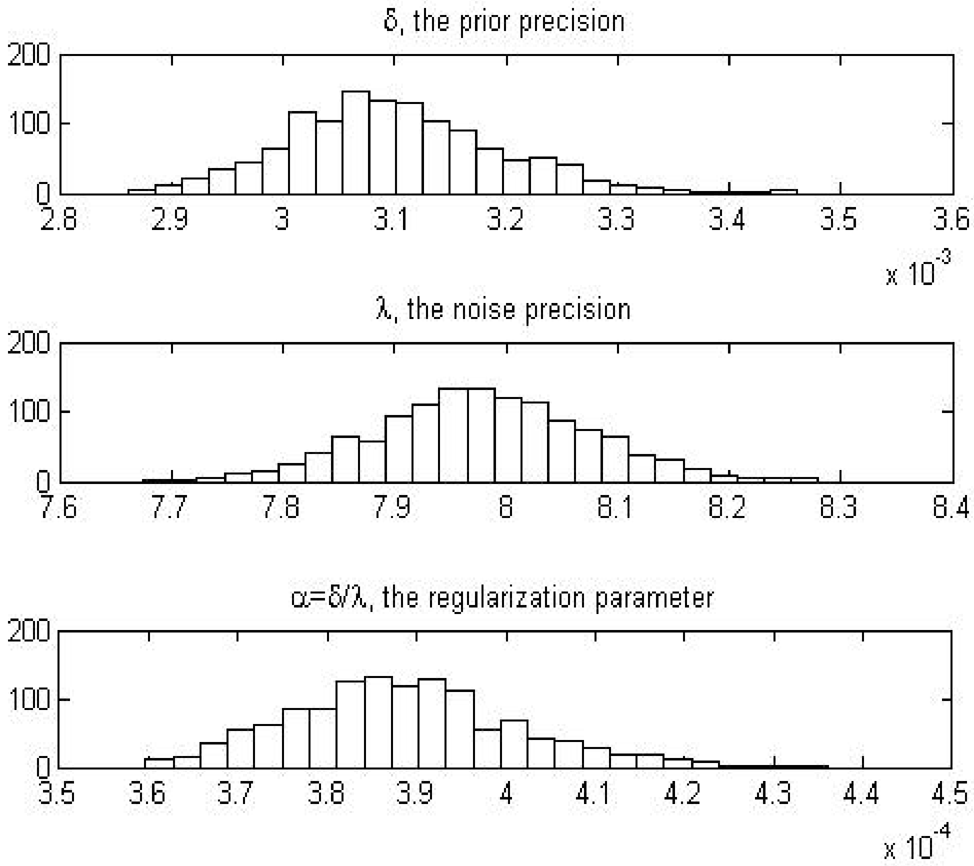
Sample mean



Pixel-wise Variance Image.



Precision & Reg. Parameter Histograms

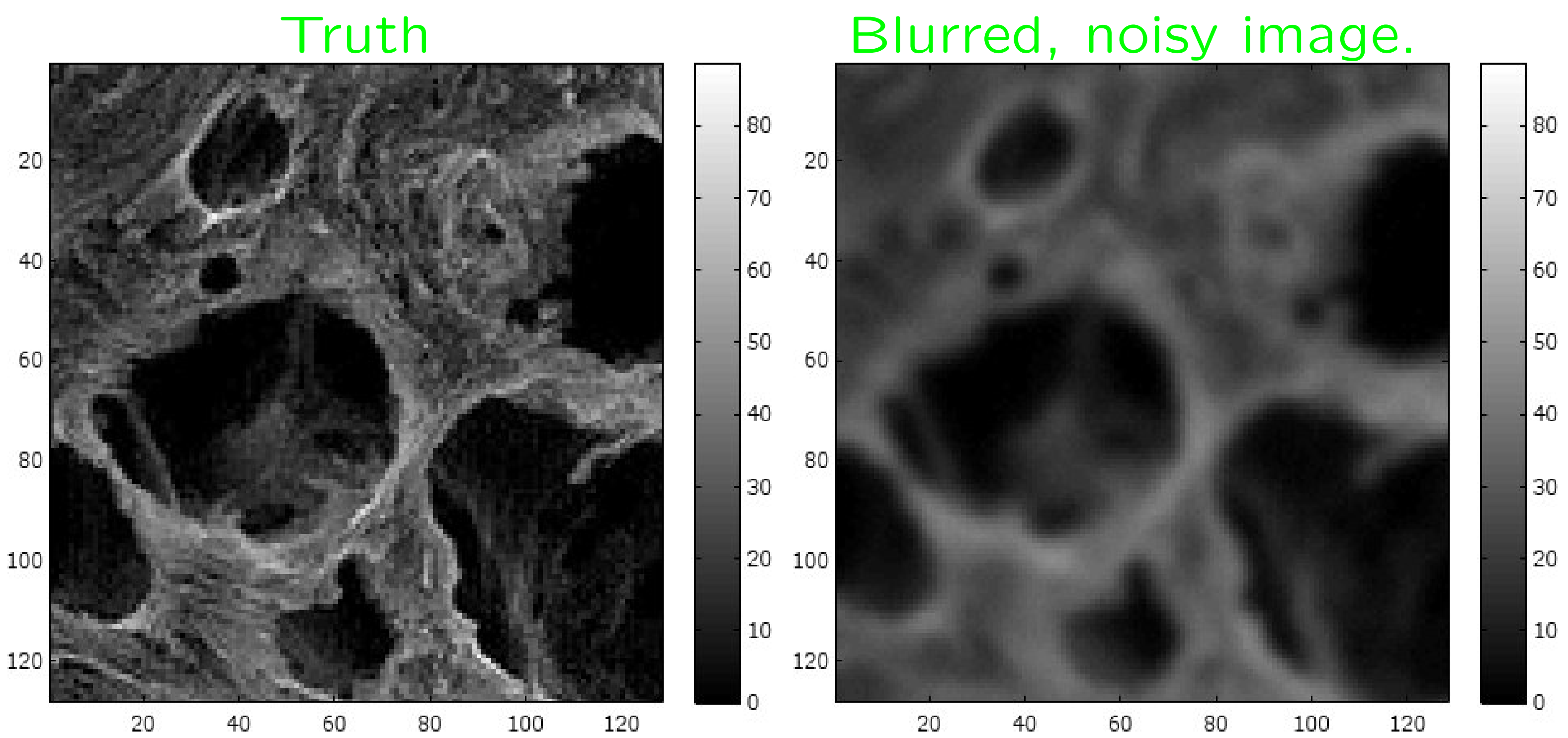


Neumann boundary conditions (w/ M. Howard & J. Nagy)

In this case you can directly solve

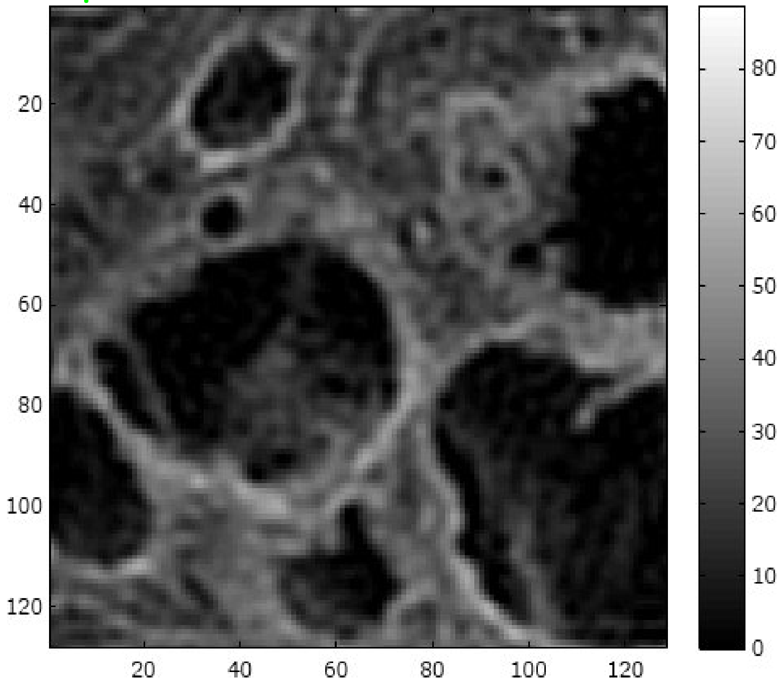
$$\mathbf{x}^k = (\lambda_k \mathbf{A}^T \mathbf{A} + \delta_k \mathbf{L})^{-1} (\lambda_k \mathbf{A}^T \mathbf{b} + \mathbf{w}).$$

Here $\mathbf{A}$ and $\mathbf{L}$ are diagonalizable by the 2d-DCT.

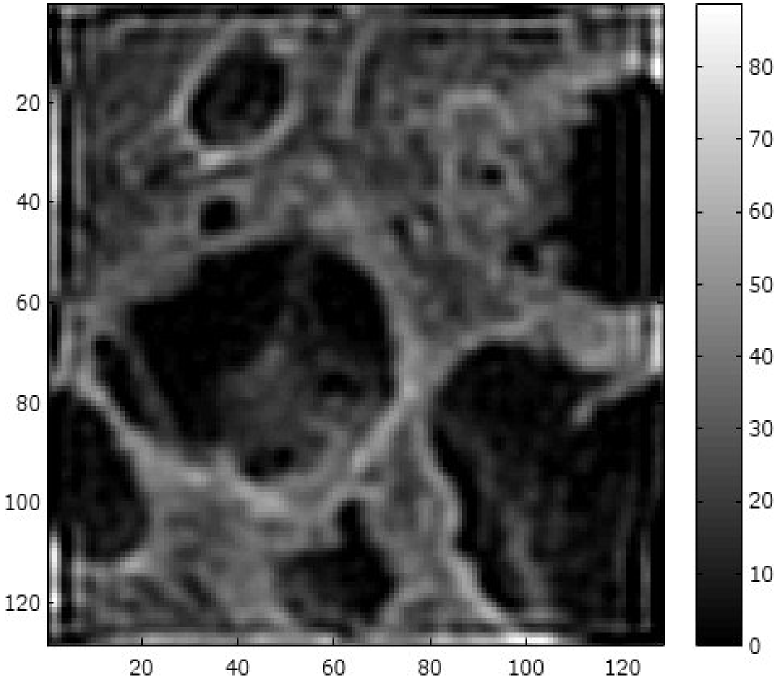


Neumann boundary conditions (w/ M. Howard & J. Nagy)

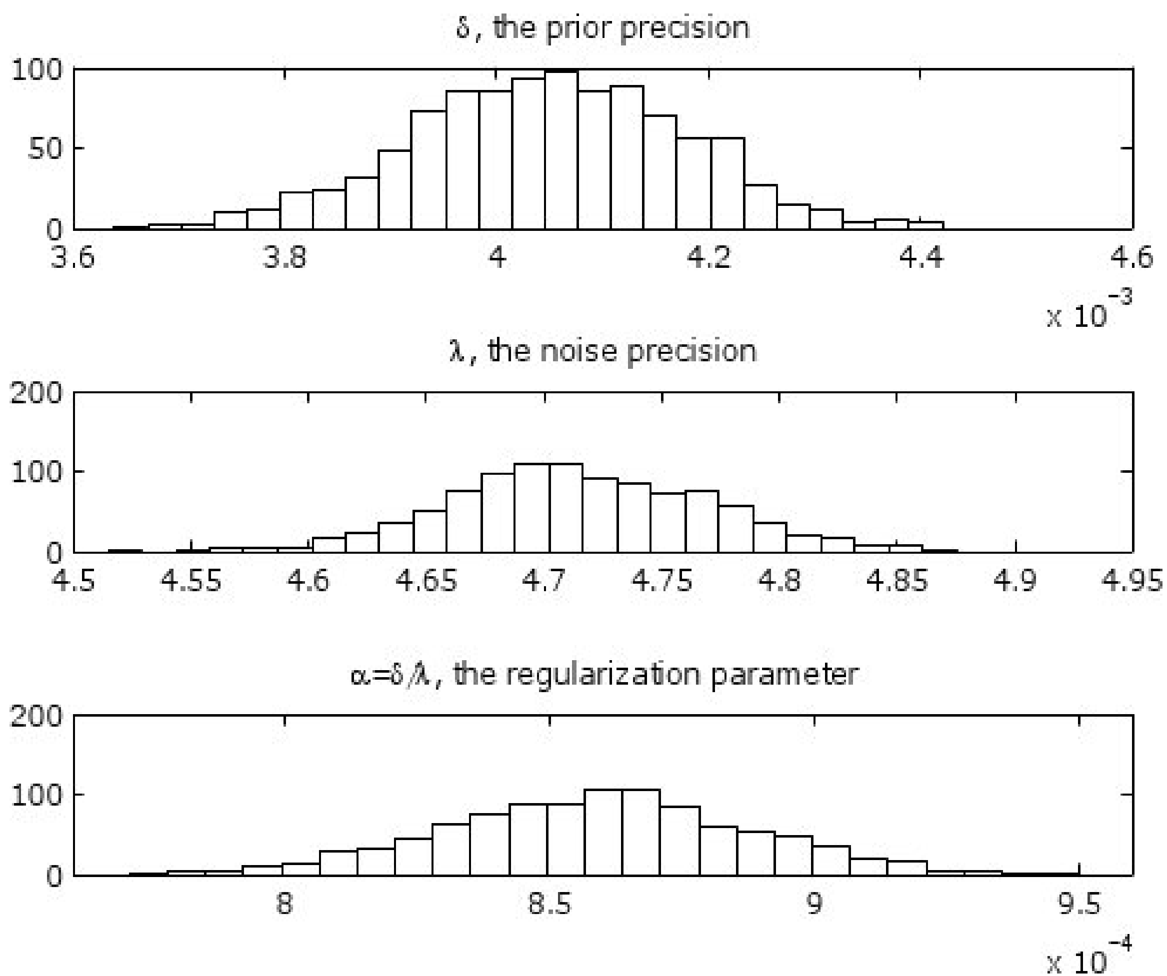
Sample Mean: Neumann BCs



Periodic BCs



Precision & Reg. Parameter Histograms



Randomize-then-Optimize

In cases where the linear system

$$(\lambda_k \mathbf{A}^T \mathbf{A} + \delta_k \mathbf{L})\mathbf{x}^k = \lambda_k \mathbf{A}^T \mathbf{b} + \mathbf{w}.$$

can't be directly solved, we restate it as an optimization problem.

Randomize-then-Optimize

In cases where the linear system

$$(\lambda_k \mathbf{A}^T\mathbf{A} + \delta_k \mathbf{L})\mathbf{x}^k = \lambda_k \mathbf{A}^T\mathbf{b} + \mathbf{w}.$$

can't be directly solved, we restate it as an optimization problem.

1. Randomize: generate new 'data'

$$\widehat{\mathbf{b}} \sim \mathcal{N}(\mathbf{b}, \lambda_k^{-1}\mathbf{I}_n) \quad \text{and} \quad \widehat{\mathbf{c}} \sim \mathcal{N}(\mathbf{0}, \delta_k^{-1}\mathbf{L}^\dagger).$$

   where '$\dagger$' denotes pseudo-inverse.

2. Optimize: solve

$$\mathbf{x}^k = \arg\min_{\mathbf{x}} \frac{1}{2} \left\| \begin{bmatrix} \lambda_k^{1/2}(\mathbf{A}\mathbf{x} - \widehat{\mathbf{b}}) \\ \delta_k^{1/2}\mathbf{L}^{1/2}(\mathbf{x} - \widehat{\mathbf{c}}) \end{bmatrix} \right\|_2^2.$$

Two Component Gibbs sampler using RTO

0. δ_0, and λ_0, and set $k = 0$;

1. First generate

$$\widehat{\mathbf{b}} \sim \mathcal{N}(\mathbf{b}, \lambda_k^{-1}\mathbf{I}_n) \quad \text{and} \quad \widehat{\mathbf{c}} \sim \mathcal{N}(\mathbf{0}, \delta_k^{-1}\mathbf{L}^{\dagger}).$$

then compute

$$\mathbf{x}^k = \arg\min_{\mathbf{x}} \frac{1}{2} \left\| \begin{bmatrix} \lambda_k^{1/2}(\mathbf{A}\mathbf{x} - \widehat{\mathbf{b}}) \\ \delta_k^{1/2}\mathbf{L}^{1/2}(\mathbf{x} - \widehat{\mathbf{c}}) \end{bmatrix} \right\|_2^2.$$

2. Compute a sample

$$\begin{bmatrix} \lambda_{k+1} \\ \delta_{k+1} \end{bmatrix} \sim \Gamma\left(\begin{bmatrix} n/2 + \alpha_\lambda \\ n/2 + \alpha_\delta \end{bmatrix}, \begin{bmatrix} \frac{1}{2}\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|^2 + \beta_\lambda \\ \frac{1}{2}\|\mathbf{L}^{1/2}\mathbf{x}^k\|^2 + \beta_\delta \end{bmatrix} \right).$$

3. Set $k = k + 1$ and return to Step 1.

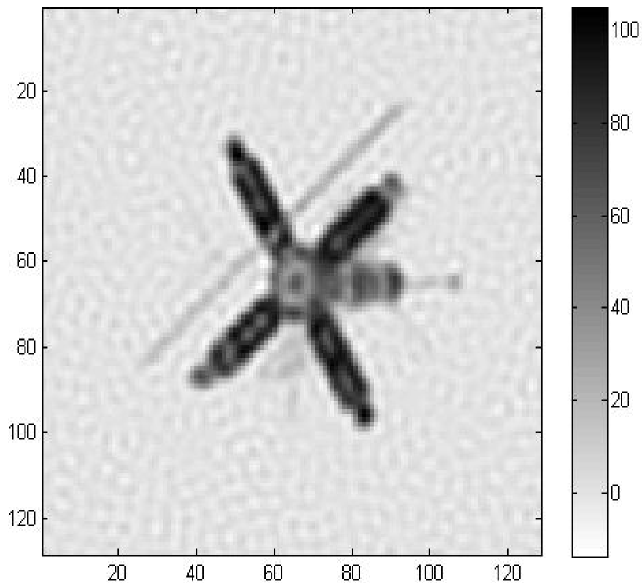
Deblurring with Dirichlet boundary conditions

In this case you must solve

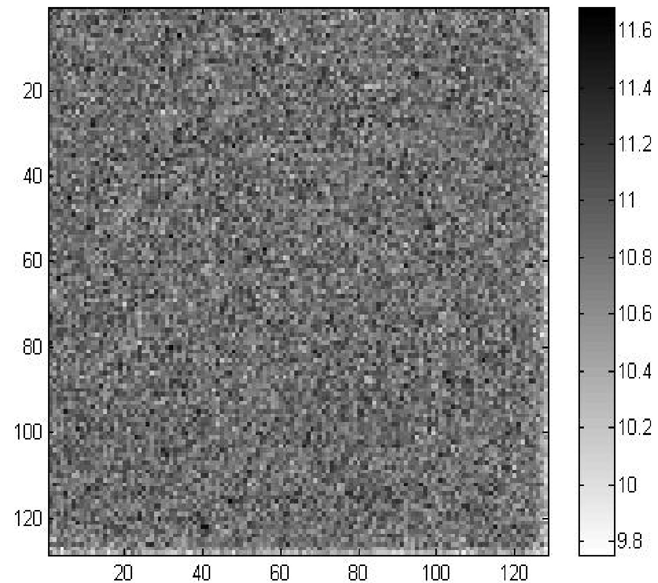
$$\mathbf{x}^k = \arg\min_{\mathbf{x}} \left\{ \frac{\lambda_k}{2} \|\mathbf{A}\mathbf{x} - \hat{\mathbf{b}}\|_2^2 + \frac{\delta_k}{2} \|\mathbf{L}^{1/2}(\mathbf{x} - \hat{\mathbf{c}})\|_2^2 \right\}.$$

We use a circulant preconditioned CG algorithm.

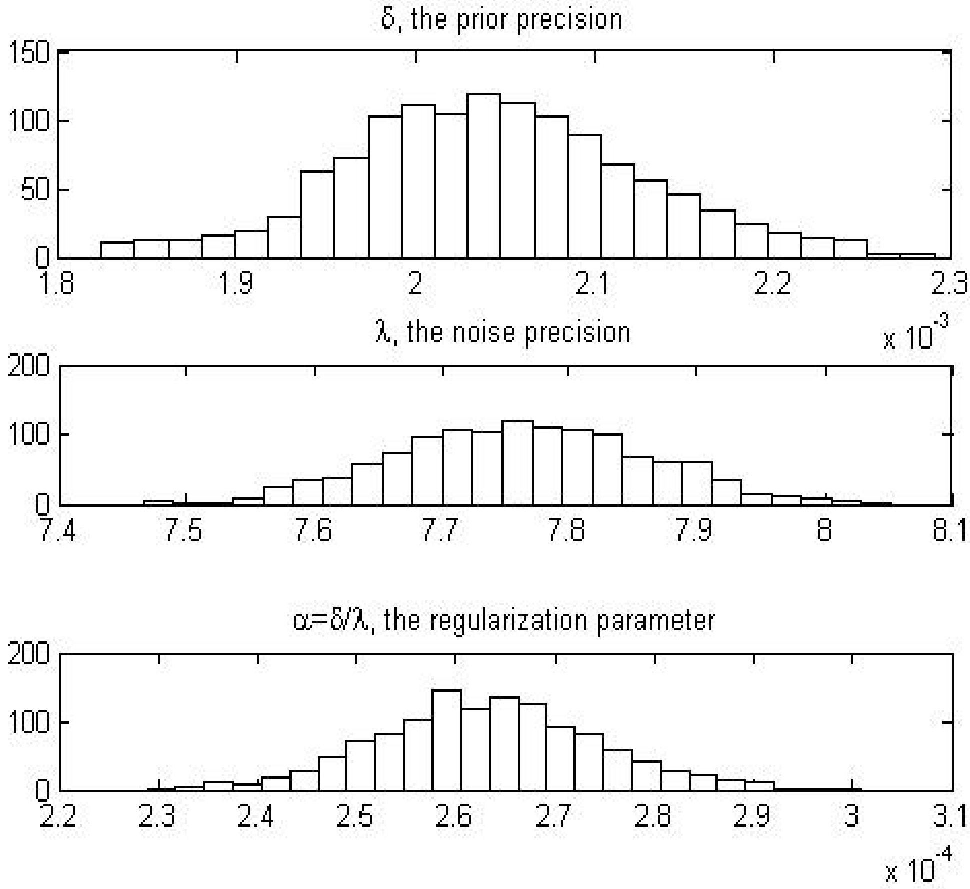
Sample median



Pixel-wise standard deviation.



Precision & Reg. Parameter Histograms



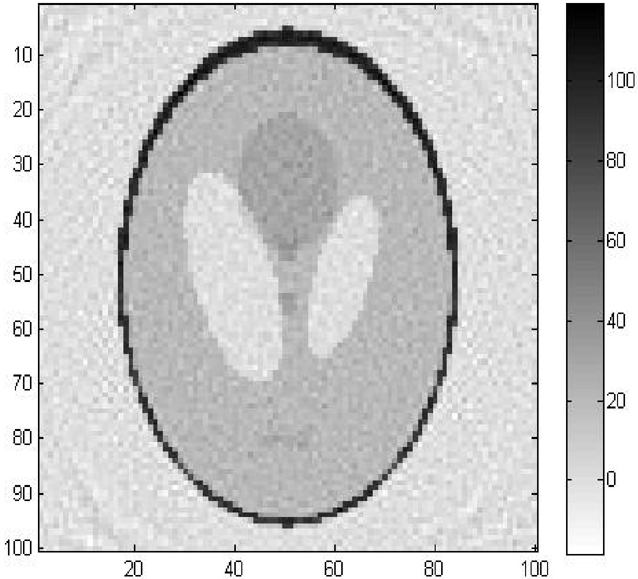
Computed Tomography

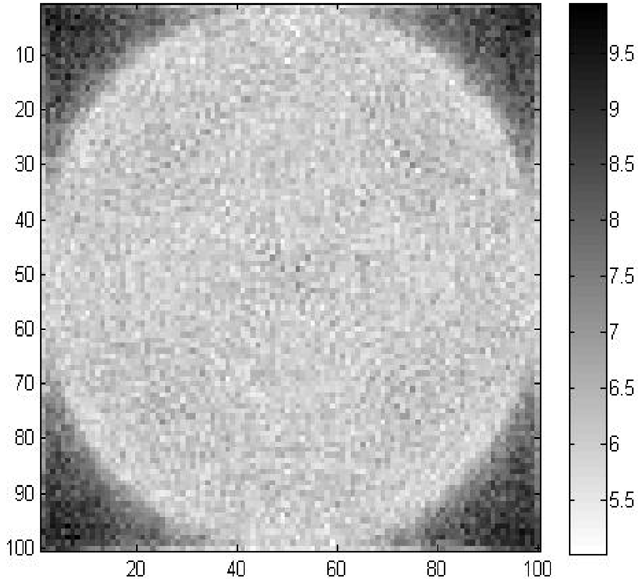
In this case you must solve

$$\mathbf{x}^k = \arg\min_{\mathbf{x}} \left\{ \frac{\lambda_k}{2} \|\mathbf{A}\mathbf{x} - \widehat{\mathbf{b}}\|_2^2 + \frac{\delta_k}{2} \|\mathbf{L}^{1/2}(\mathbf{x} - \widehat{\mathbf{c}})\|_2^2 \right\}.$$

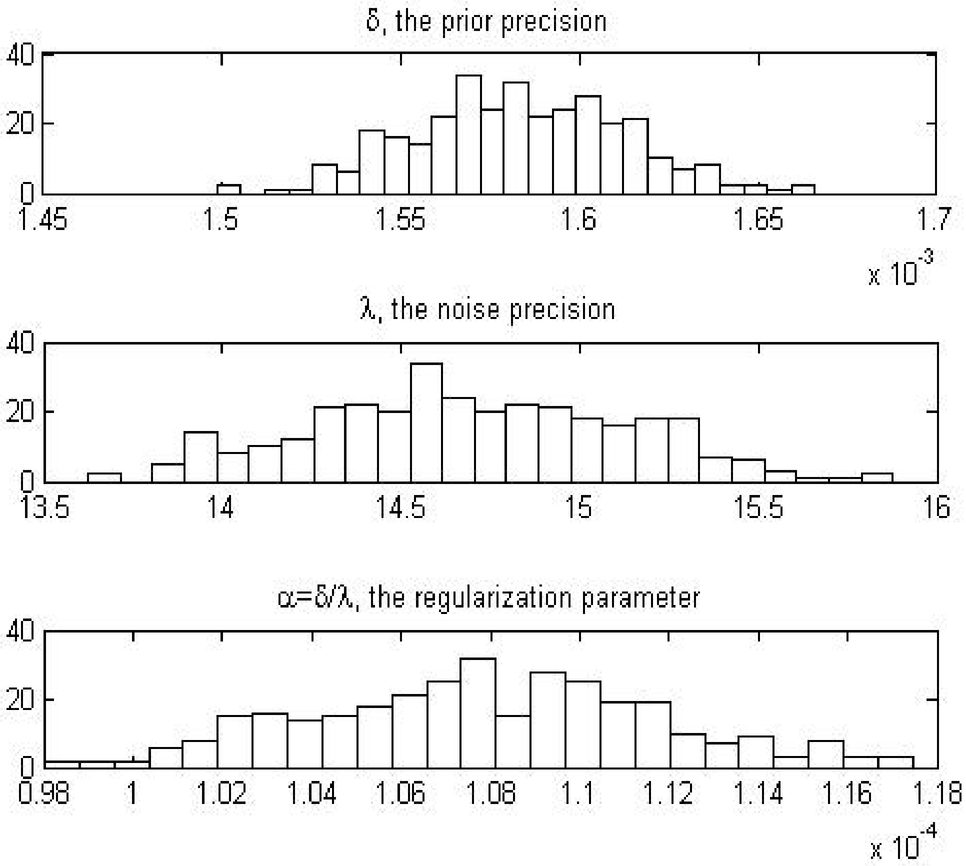
Pretending we have accurate solutions yields:

Sample median Pixel-wise Variance Image.





Precision & Reg. Parameter Histograms



Nonnegativity Constrained MCMC Method

with Colin Fox

0. δ_0, and λ_0, and set $k = 0$;

1. First generate

$$\hat{\mathbf{b}} \sim \mathcal{N}(\mathbf{b}, \lambda_k^{-1}\mathbf{I}_n) \quad \text{and} \quad \hat{\mathbf{c}} \sim \mathcal{N}(\mathbf{0}, \delta_k^{-1}\mathbf{L}^{\dagger}).$$

then compute

$$\mathbf{x}^k = \arg\min_{\mathbf{x}\geq\mathbf{0}} \frac{1}{2}\left\| \begin{bmatrix} \lambda_k^{1/2}(\mathbf{A}\mathbf{x} - \hat{\mathbf{b}}) \\ \delta_k^{1/2}\mathbf{L}^{1/2}(\mathbf{x} - \hat{\mathbf{c}}) \end{bmatrix} \right\|_2^2 .$$

2. Compute a sample

$$\begin{bmatrix} \lambda_{k+1} \\ \delta_{k+1} \end{bmatrix} \sim \Gamma\left(\begin{bmatrix} n/2 + \alpha_\lambda \\ n_p/2 + \alpha_\delta \end{bmatrix}, \begin{bmatrix} \frac{1}{2}\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|^2 + \beta_\lambda \\ \frac{1}{2}\|\mathbf{L}^{1/2}\mathbf{x}^k\|^2 + \beta_\delta \end{bmatrix} \right).$$

3. Set $k = k + 1$ and return to Step 1.

Nonnegativity Constrained RTO

Generate

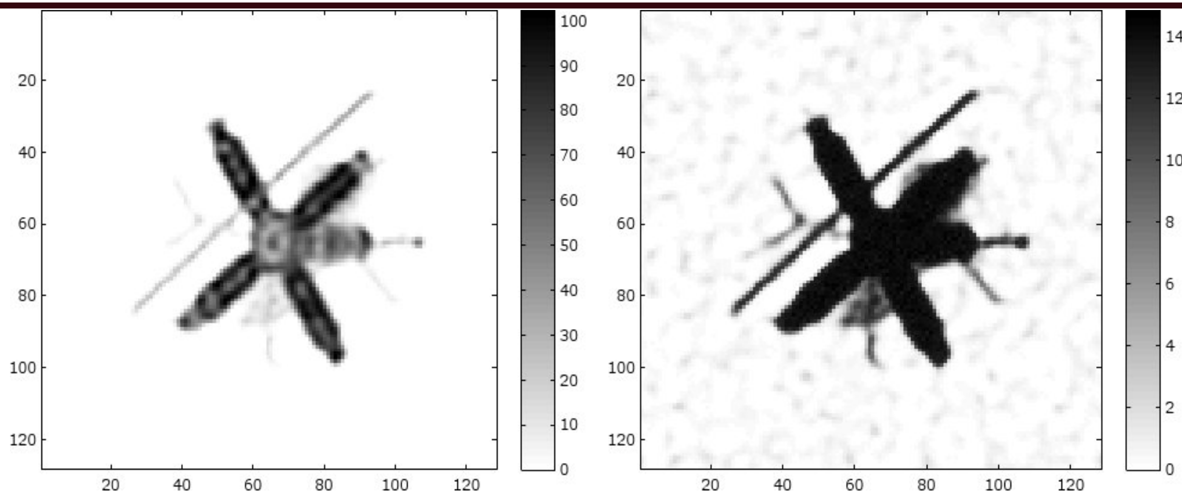
$$\widehat{\mathbf{b}} \sim \mathcal{N}(\mathbf{b}, \lambda_k^{-1}\mathbf{I}_n) \quad \text{and} \quad \widehat{\mathbf{c}} \sim \mathcal{N}(\mathbf{0}, \delta_k^{-1}\mathbf{L}^\dagger).$$

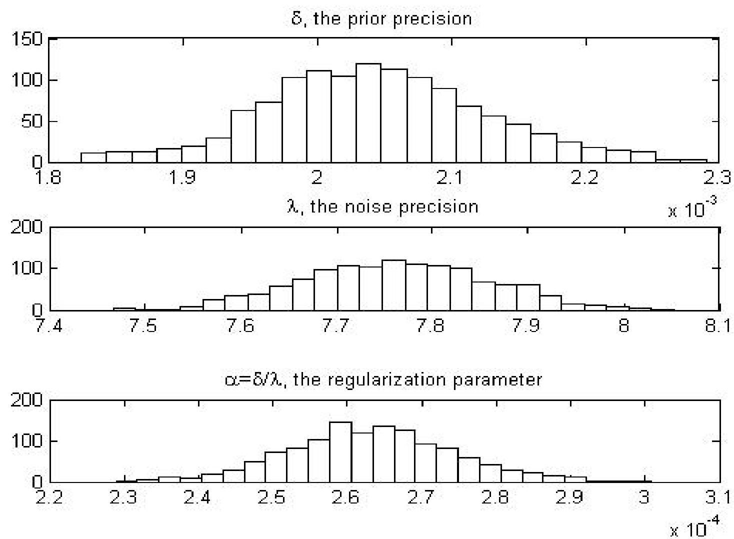
then compute

$$\mathbf{x}^k = \arg\min_{\mathbf{x}\geq\mathbf{0}} \frac{1}{2} \left\| \begin{bmatrix} \lambda_k^{1/2}(\mathbf{A}\mathbf{x} - \widehat{\mathbf{b}}) \\ \delta_k^{1/2}\mathbf{L}^{1/2}(\mathbf{x} - \widehat{\mathbf{c}}) \end{bmatrix} \right\|_2^2 ?$$

Question: What is $p(\mathbf{x}^k)$?

Nonnegativity Constraints: Deblur (w/ C. Fox)





Inverse Problems with Poisson data

In this case the data model has the form

$$\mathbf{b} = \text{Poisson}(\mathbf{A}\mathbf{x} + \mathbf{g}),$$

- $\mathbf{b}$ is the $m \times 1$ data vector,
- $\mathbf{A}$ is an $m \times n$ ill-condition matrix,
- $\mathbf{x}$ is the $n \times 1$ unknown,
- $\mathbf{g}$ is the $m \times 1$ known background.

The Full Posterior Distribution

Then

$$p(\mathbf{b}|\mathbf{x}) \propto \exp\left(-\sum_{i=1}^{n}([\mathbf{A}\mathbf{x}]_i + \beta_i) - b_i \ln([\mathbf{A}\mathbf{x}]_i + \beta_i)\right).$$

If we assume, as above, Gaussian prior and Gamma hyper-prior, we obtain

$$p(\mathbf{x}, \delta|\mathbf{b}) \propto \text{the posterior}$$

$$\delta^{n/2+\alpha-1} \exp\left(-\sum_{i=1}^{n}([\mathbf{A}\mathbf{x}]_i + \beta_i) - b_i \ln([\mathbf{A}\mathbf{x}]_i + \beta_i) - \frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x} - \beta\delta\right).$$

A Two-Component Gibbs Sampler for Poisson Data

Sample cyclically from $p(\mathbf{x}|\mathbf{b},\delta)$ and $p(\delta|\mathbf{b},\mathbf{x})$

0. δ_0, and λ_0, and set $k = 0$.
1. Compute a sample $\mathbf{x}_{k+1}$ from

$$p(\mathbf{x}|\delta_k,\mathbf{b}) \propto \exp\left(-\sum_{i=1}^{n}([\mathbf{Ax}]_i+\beta_i) - b_i \ln([\mathbf{Ax}]_i+\beta_i) - \frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x}\right).$$

2. Compute a sample δ_{k+1} from

$$p(\delta|\mathbf{x}_{k+1},\mathbf{b}) \sim \Gamma\left(n/2+\alpha, \frac{1}{2}(\mathbf{x}^k)^T\mathbf{L}\mathbf{x}^k+\beta\right).$$

3. Set $k = k + 1$ and return to Step 1.

Randomize-then-Optimize for Step 1

1. first randomize the 'data',

$$\hat{\mathbf{b}} \sim \text{Poiss}(\mathbf{b}) \text{ and } \hat{\mathbf{c}} \sim N(\mathbf{0}, \delta_k^{-1}\mathbf{I}),$$

2. then optimize to obtain a sample,

$$\mathbf{x}^k = \arg\min_{\mathbf{x} \geq \mathbf{0}} \left\{ \sum_{i=1}^{n} \{[\mathbf{A}\mathbf{x}]_i + g_i - \hat{b}_i \ln([\mathbf{A}\mathbf{x}]_i + g_i)\} + \frac{\delta_k}{2} \|\mathbf{L}^{1/2}(\mathbf{x} - \hat{\mathbf{c}})\|^2 \right\}.$$

Question: Is the density $p(\mathbf{x}^k)$ defined by RTO close to

$$p(\mathbf{x}|\delta_k, \mathbf{b}) \propto \exp\left(-\sum_{i=1}^{n} ([\mathbf{A}\mathbf{x}]_i + \beta_i) - b_i \ln([\mathbf{A}\mathbf{x}]_i + \beta_i) - \frac{\delta}{2}\mathbf{x}^T\mathbf{L}\mathbf{x} \right)?$$

A Two-Component Gibbs Sampler for Poisson Data

with Haario and Solonen

0. δ_0, and λ_0, and set $k = 0$;
1. First generate

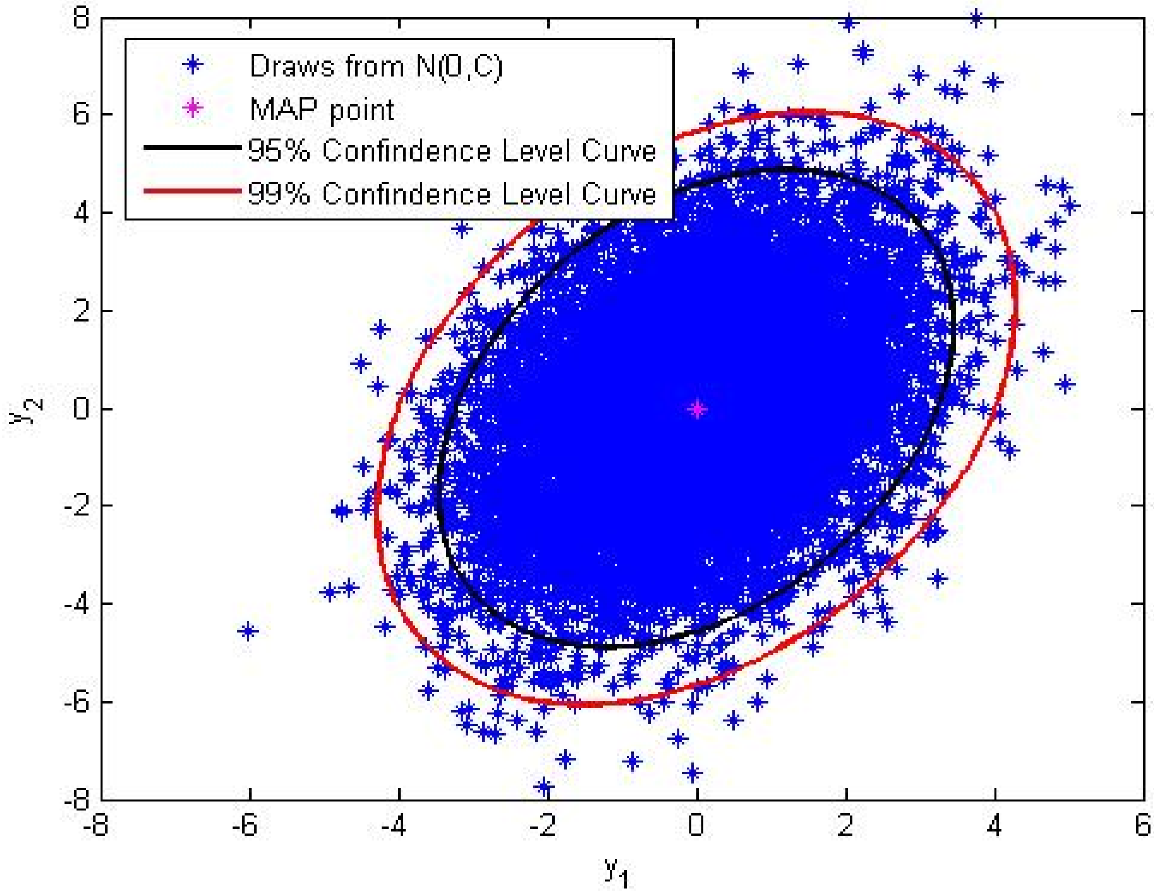
$$\hat{\mathbf{b}} \sim \text{Poiss}(\mathbf{b}) \text{ and } \hat{\mathbf{c}} \sim N(\mathbf{0}, \delta_k^{-1}\mathbf{I}),$$

   then compute

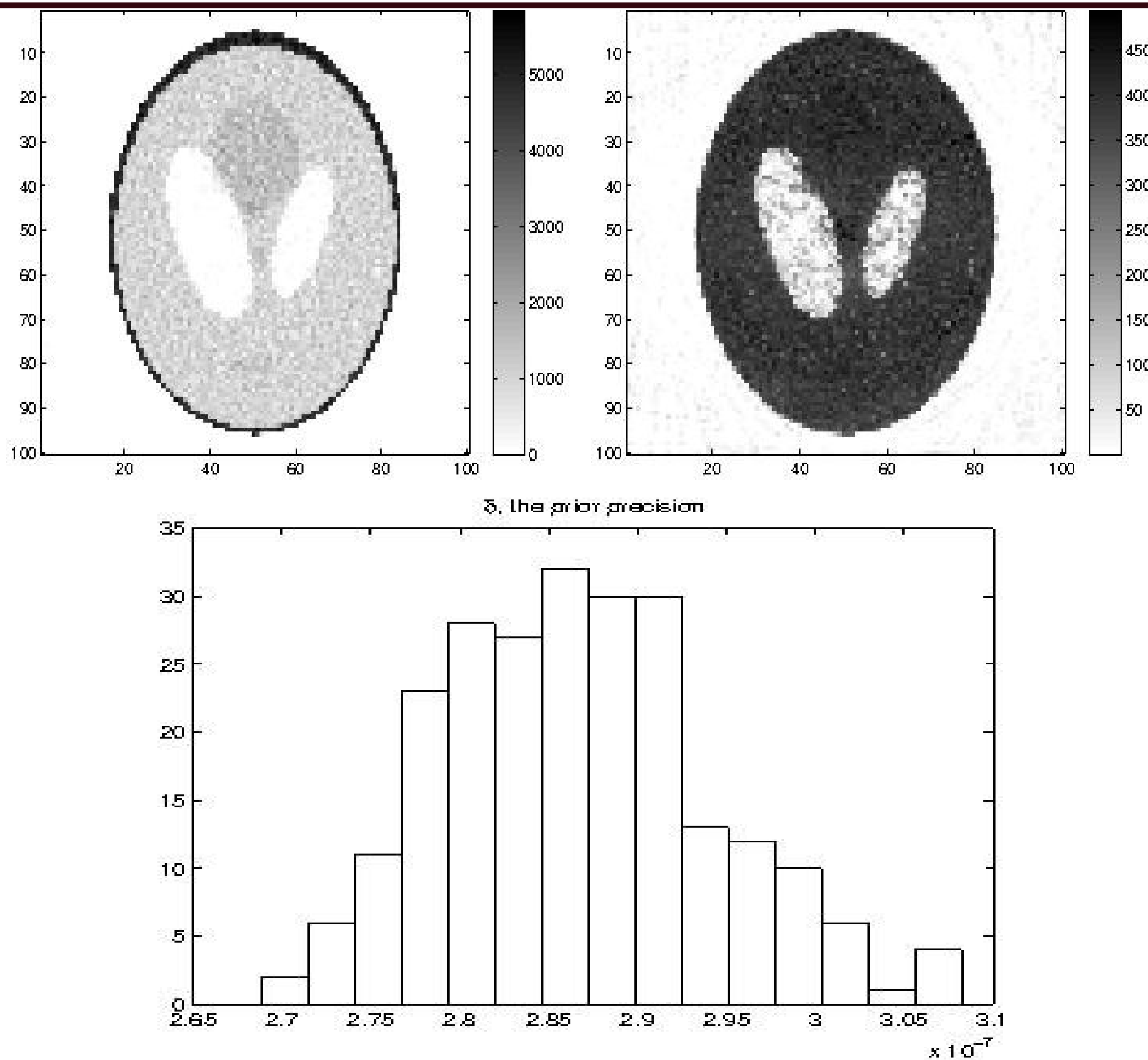
$$\mathbf{x}^k = \arg\min_{\mathbf{x}\geq\mathbf{0}} \left\{ \sum_{i=1}^{n} \{[\mathbf{A}\mathbf{x}]_i + g_i - \hat{b}_i \ln([\mathbf{A}\mathbf{x}]_i + g_i)\} + \frac{\delta_k}{2}\|\mathbf{L}^{1/2}(\mathbf{x} - \hat{\mathbf{c}})\|^2 \right\}.$$

2. $\delta_{k+1} \sim \Gamma\left(n_p/2 + \alpha, \frac{1}{2}(\mathbf{x}^k)^T\mathbf{L}\mathbf{x}^k + \beta\right)$;
3. Set $k = k + 1$ and return to Step 1.

Sampling vs. Computing the MAP



Positron Emission Tomography



Nonlinear models

We begin by considering linear models of the form:

$$\mathbf{b} = \mathbf{A}(\mathbf{x}) + \boldsymbol{\epsilon},$$

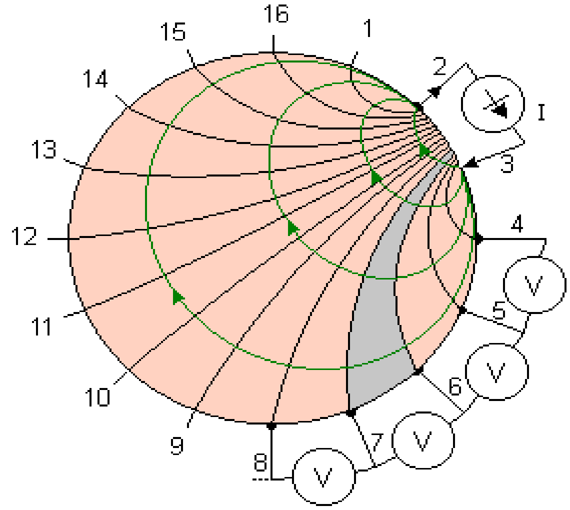
- $\mathbf{b}$ is the $m \times 1$ data vector,
- $\mathbf{A} : \mathbb{R}^n \to \mathbb{R}^m$ is the *nonlinear* forward map,
- $\mathbf{x}$ is the $n \times 1$ unknown,
- $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_n)$ is the $n \times 1$ iid Gaussian noise vector.

EIT Model

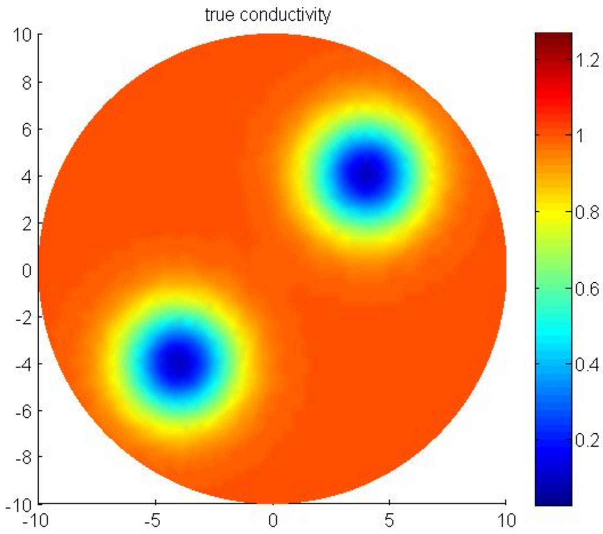
Let u be voltage, σ electrical conductivity:

$$\begin{aligned} \nabla \cdot (\sigma \nabla u) &= 0, \quad && \Omega \\ \text{BCs}, & && \partial\Omega \end{aligned}$$

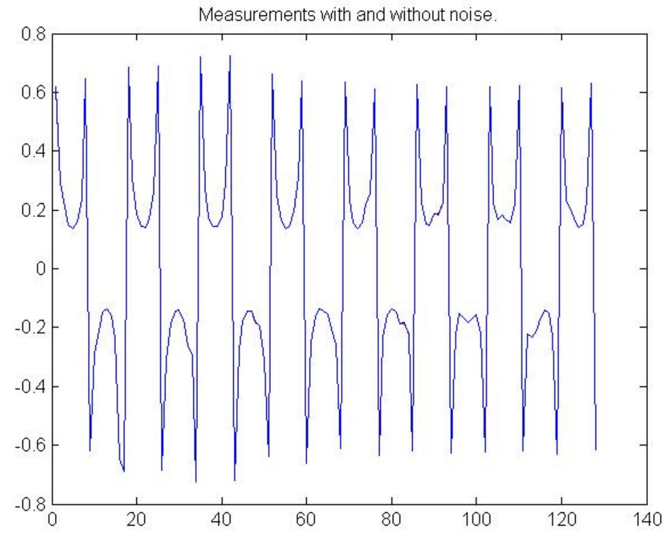
Inverse Problem: given inputs and measurements of u at the boundary, determine the conductivity σ in the interior.



EIT data



true conductivity
10
8
6
4
2
0
-2
-4
-6
-8
-10
-10
-5
0
5
10
1.2
1
0.8
0.6
0.4
0.2



Measurements with and without noise.
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
0
20
40
60
80
100
120
140

RTO in the nonlinear case

w/ Haario, Kaipio, Seppanen, Solonen

1. Randomize: generate new 'data'

$$\widehat{\mathbf{b}} \sim \mathcal{N}(\mathbf{b}, \lambda_k^{-1}\mathbf{I}_n) \quad \text{and} \quad \widehat{\mathbf{c}} \sim \mathcal{N}(\mathbf{0}, \delta_k^{-1}\mathbf{L}^{\dagger}).$$

2. Optimize: solve

$$\mathbf{x}^k = \arg\min_{\mathbf{x}} \frac{1}{2} \left\| \begin{bmatrix} \lambda_k^{1/2}(\mathbf{A}(\mathbf{x}) - \widehat{\mathbf{b}}) \\ \delta_k^{1/2}\mathbf{L}^{1/2}(\mathbf{x} - \widehat{\mathbf{c}}) \end{bmatrix} \right\|_2^2 .$$

RTO in the nonlinear case

w/ Haario, Kaipio, Seppanen, Solonen

1. Randomize: generate new 'data'

$$\widehat{\mathbf{b}} \sim \mathcal{N}(\mathbf{b}, \lambda_k^{-1}\mathbf{I}_n) \quad \text{and} \quad \widehat{\mathbf{c}} \sim \mathcal{N}(\mathbf{0}, \delta_k^{-1}\mathbf{L}^{\dagger}).$$

2. Optimize: solve

$$\mathbf{x}^k = \arg\min_{\mathbf{x}} \frac{1}{2} \left\| \begin{bmatrix} \lambda_k^{1/2}(\mathbf{A}(\mathbf{x}) - \widehat{\mathbf{b}}) \\ \delta_k^{1/2}\mathbf{L}^{1/2}(\mathbf{x} - \widehat{\mathbf{c}}) \end{bmatrix} \right\|_2^2.$$

Question: Is the density $p(\mathbf{x}^k)$ defined by RTO close to

$$p(\mathbf{x}|\mathbf{b}, \lambda_k, \delta_k) \propto \exp\left(-\frac{1}{2} \left\| \begin{bmatrix} \lambda_k^{1/2}(\mathbf{A}(\mathbf{x}) - \widehat{\mathbf{b}}) \\ \delta_k^{1/2}\mathbf{L}^{1/2}(\mathbf{x} - \widehat{\mathbf{c}}) \end{bmatrix} \right\|_2^2 \right)$$

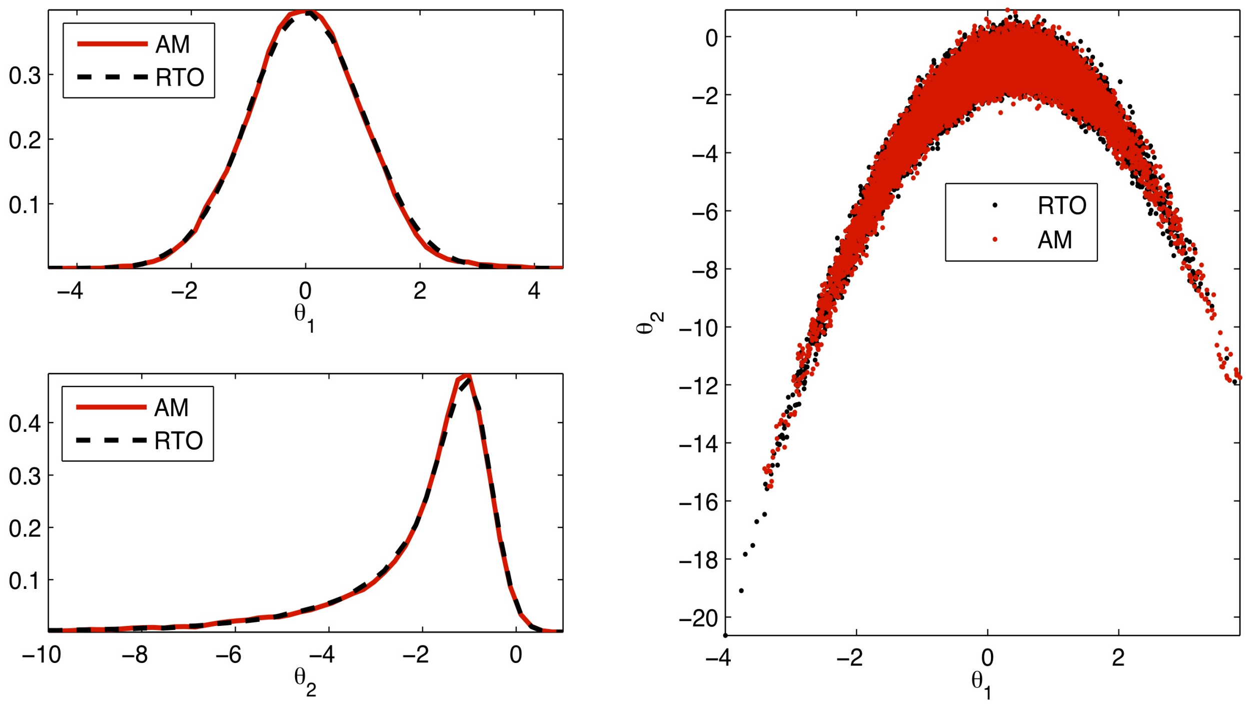
as in the linear case?

Numerical Comparison of AM and RTO

by Antti Solonen

Test 1: Let $f^{-1}(\mathbf{x}) = [x_1/a, ax_2 + ab(x_1^2 + a^2)]$ and define

$$\pi(\mathbf{x}) \propto \exp\left((f^{-1}(\mathbf{x}) - \mathbf{v})^T \boldsymbol{\Sigma}^{-1} (f^{-1}(\mathbf{x}) - \mathbf{v})\right),$$



Numerical Comparison of AM and RTO

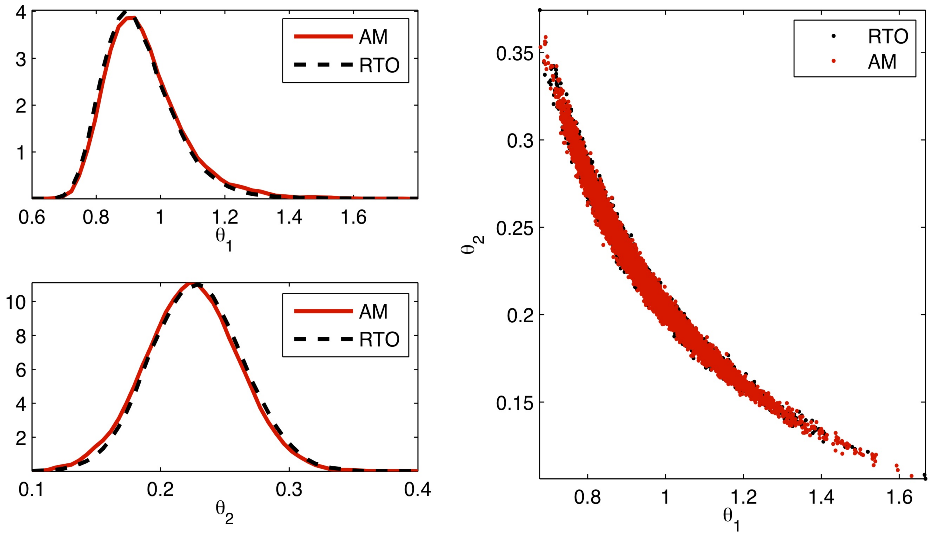
by Antti Solonen

Test 2: use RTO to sample

$$(x_1, x_2) = \arg \min_{(x_1,x_2)} \sum_{i=1}^{T} (b_i - x_1(1 - \exp(-x_2 t_i)))^2,$$

where $(b_1, \ldots, b_n)$ and $(t_1, \ldots, t_n)$ are measured data and

$$b_i = x_1(1 - \exp(-x_2 t_i)) + \epsilon_i, \quad i = 1, \ldots, T.$$



Two-Component Gibbs Sampler, Nonlinear Case

with Seppänen, Solonen, Haario, and Kaipio

0. δ_0, and λ_0, and set $k = 0$;

1. First generate

$$\hat{\mathbf{b}} \sim \mathcal{N}(\mathbf{b}, \lambda_k^{-1}\mathbf{I}_n) \quad \text{and} \quad \hat{\mathbf{c}} \sim \mathcal{N}(\mathbf{0}, \delta_k^{-1}\mathbf{L}^{\dagger}).$$

then compute

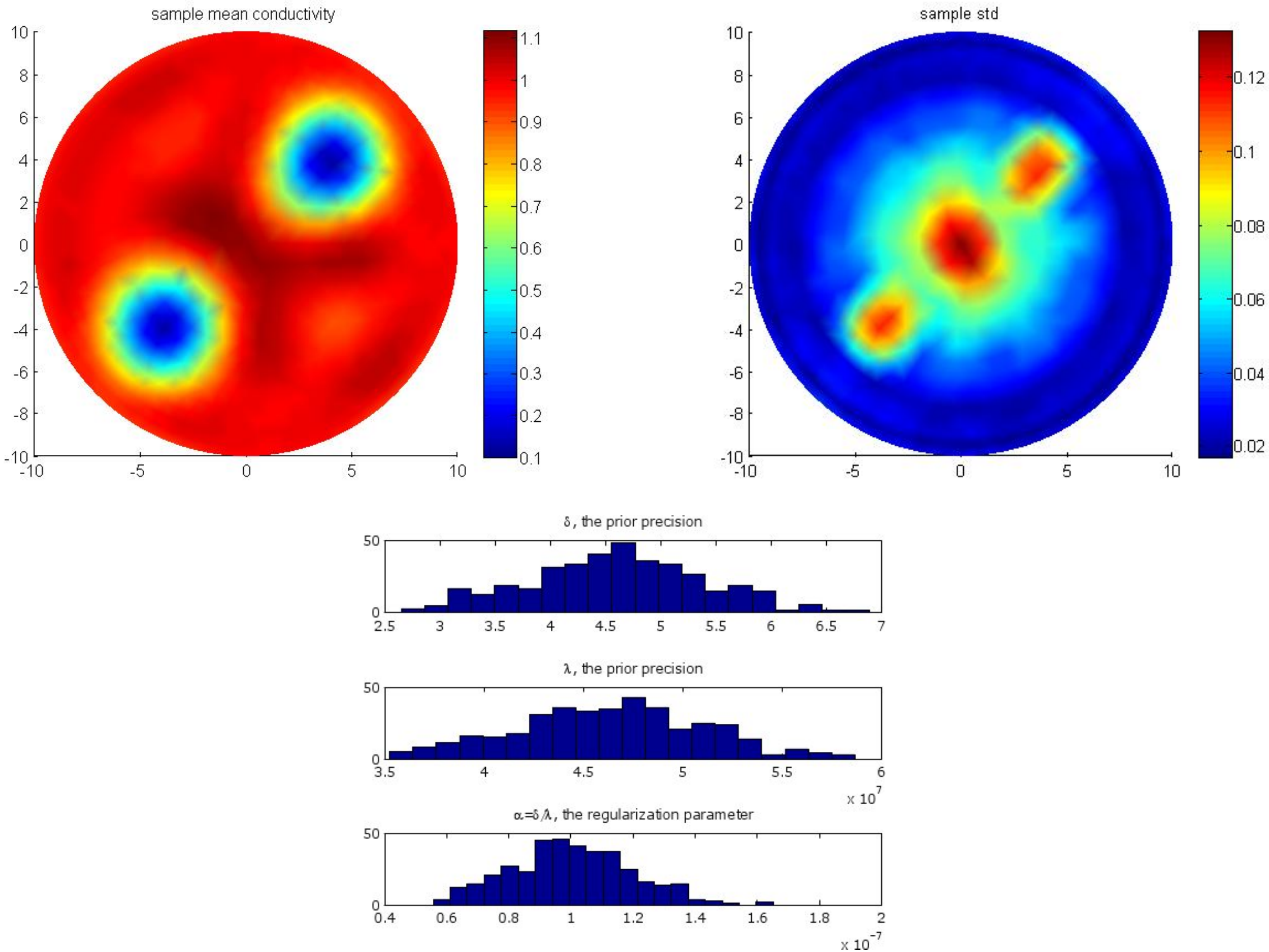
$$\mathbf{x}^k = \arg\min_{\mathbf{x}} \frac{1}{2} \left\| \begin{bmatrix} \lambda_k^{1/2}(\mathbf{A}(\mathbf{x}) - \hat{\mathbf{b}}) \\ \delta_k^{1/2}\mathbf{L}^{1/2}(\mathbf{x} - \hat{\mathbf{c}}) \end{bmatrix} \right\|_2^2 .$$

2. Compute a sample

$$\begin{bmatrix} \lambda_{k+1} \\ \delta_{k+1} \end{bmatrix} \sim \Gamma\left(\begin{bmatrix} n/2 + \alpha_\lambda \\ n/2 + \alpha_\delta \end{bmatrix}, \begin{bmatrix} \frac{1}{2}\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|^2 + \beta_\lambda \\ \frac{1}{2}\|\mathbf{L}^{1/2}\mathbf{x}^k\|^2 + \beta_\delta \end{bmatrix} \right).$$

3. Set $k = k + 1$ and return to Step 1.

Sample mean and standard deviation images



Nonlinear RTO Proof, Scalar Case

Let b be fixed 'data' from the model

$$b = a(x) + v, \quad v \sim \mathcal{N}(0, \sigma^2).$$

Let $\widehat{v}$ be a fixed realization from v and define

$$x_{\widehat{v}} = \arg\min_x \left\{ f(x) = (a(x) - (b + \widehat{v}))^2 \right\}.$$

Nonlinear RTO Proof, Scalar Case

Let b be fixed 'data' from the model

$$b = a(x) + v, \quad v \sim \mathcal{N}(0, \sigma^2).$$

Let $\widehat{v}$ be a fixed realization from v and define

$$x_{\widehat{v}} = \arg\min_x \left\{ f(x) = (a(x) - (b + \widehat{v}))^2 \right\}.$$

Note/Question: We want to sample from

$$p(x|b) \propto \exp\left(-\frac{(a(x) - (b + \widehat{v}))^2}{2\sigma^2} \right).$$

Are we doing this in RTO?

Nonlinear RTO Proof, Scalar Case

First order optimality: We know $f'(x_{\widehat{v}}) = 0$, and hence

$$a'(x_{\widehat{v}})(a(x_{\widehat{v}}) - (b + \widehat{v})) = 0.$$

Assuming $a'(x_{\widehat{v}}) \neq 0$, then

$$\widehat{v} = a(x_{\widehat{v}}) - b.$$

Nonlinear RTO Proof, Scalar Case

First order optimality: We know $f'(x_{\widehat{v}}) = 0$, and hence

$$a'(x_{\widehat{v}})(a(x_{\widehat{v}}) - (b + \widehat{v})) = 0.$$

Assuming $a'(x_{\widehat{v}}) \neq 0$, then

$$\widehat{v} = a(x_{\widehat{v}}) - b.$$

Change of variables: we expand a about $x_{\widehat{v}}$ to obtain

$$a(x) - b = \underbrace{a(x_{\widehat{v}}) - b}_{=\widehat{v}} + a'(x_{\widehat{v}})(x - x_{\widehat{v}}) + \mathcal{O}((x - x_{\widehat{v}})^2),$$

which motivates the change of variables

$$v = \underbrace{a(x_{\widehat{v}}) - b}_{=\widehat{v}} + a'(x_{\widehat{v}})(x - x_{\widehat{v}}).$$

Nonlinear RTO Proof, Scalar Case

Then, if

$$p(v) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{v^2}{2\sigma^2}\right),$$

the change of variables

$$v = \underbrace{a(x_{\widehat{v}}) - b}_{=\widehat{v}} + a'(x_{\widehat{v}})(x - x_{\widehat{v}}).$$

yields

$$p(x) = \frac{|a'(x_{\widehat{v}})|}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(a(x_{\widehat{v}}) - b + a'(x_{\widehat{v}})(x - x_{\widehat{v}}))^2}{2\sigma^2}\right).$$

Nonlinear RTO Proof, Scalar Case

Then, if

$$p(v) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{v^2}{2\sigma^2}\right),$$

the change of variables

$$v = \underbrace{a(x_{\widehat{v}}) - b}_{=\widehat{v}} + a'(x_{\widehat{v}})(x - x_{\widehat{v}}).$$

yields

$$p(x) = \frac{|a'(x_{\widehat{v}})|}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(a(x_{\widehat{v}}) - b + a'(x_{\widehat{v}})(x - x_{\widehat{v}}))^2}{2\sigma^2}\right).$$

Finally, the fact that $v = \widehat{v} \iff x = x_{\widehat{v}}$ yields

$$p(x_{\widehat{v}}) = \frac{|a'(x_{\widehat{v}})|}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(a(x_{\widehat{v}}) - b)^2}{2\sigma^2}\right).$$

Nonlinear RTO Proof, Scalar Case

The implication of this result is that RTO samples satisfy

$$p(x) \propto |a'(x)| p(x|b).$$

To obtain samples from $p(x|b)$, use importance sampling.

Nonlinear RTO Proof, Scalar Case

The implication of this result is that RTO samples satisfy

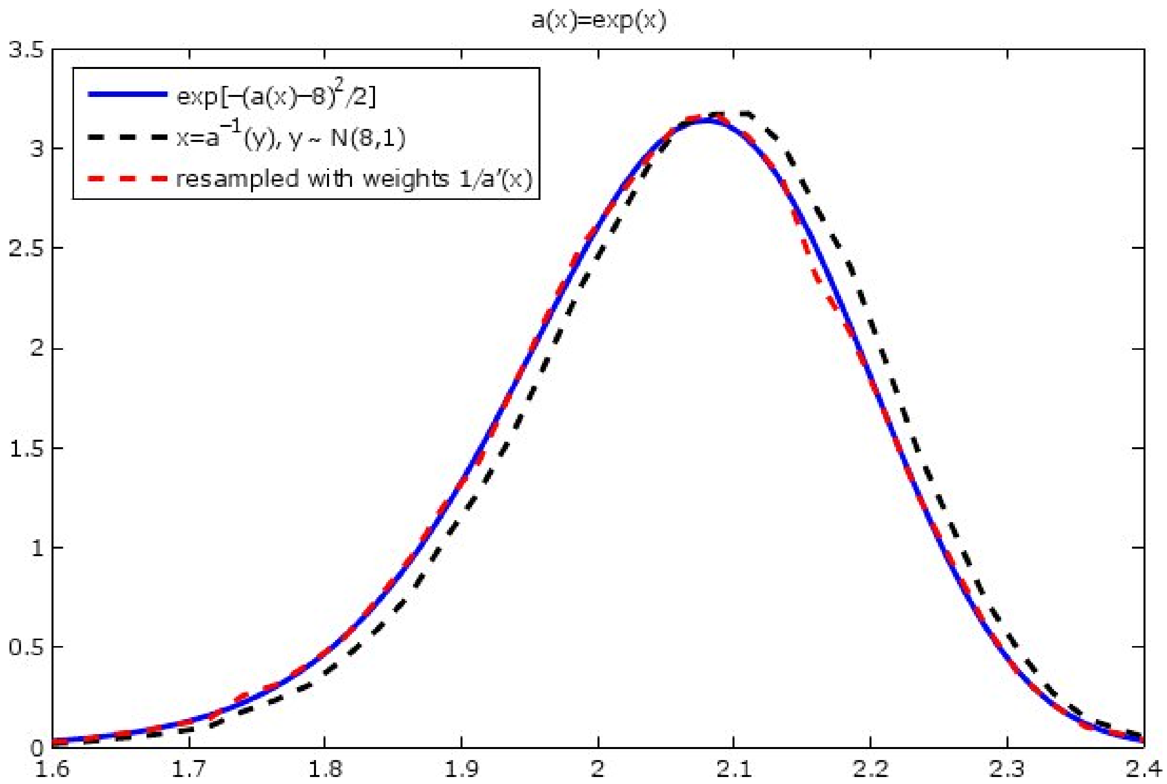
$$p(x) \propto |a'(x)| p(x|b).$$

To obtain samples from $p(x|b)$, use importance sampling.

Example, computing $E(x|b)$: suppose $x^i \sim p(x)$, $i = 1, \ldots, k$.

$$\begin{aligned} E(x|b) &= \int x\, p(x|b)\, dx \\ &= \int x\, (p(x|b)/p(x)) p(x)\, dx \\ &\approx \left(\sum_{i=1}^{k} w^i\right)^{-1} \sum_{i=1}^{k} x^i w^i, \qquad w^i = |a'(x^i)|^{-1}. \end{aligned}$$

1D Demo by Antti Solonen



Nonlinear RTO Proof, Vector Case

Provided the above approach is valid, it can be extended to the vector case to obtain

$$p(\mathbf{x}) \propto \sqrt{|J(\mathbf{x})^T J(\mathbf{x})|}\,p(\mathbf{x}|\mathbf{b}).$$

where $J(\mathbf{x})$ is the Jacobian of $A(\cdot)$ at $\mathbf{x}$.

Nonlinear RTO Proof, Vector Case

Provided the above approach is valid, it can be extended to the vector case to obtain

$$p(\mathbf{x}) \propto \sqrt{|J(\mathbf{x})^T J(\mathbf{x})|}\, p(\mathbf{x}|\mathbf{b}).$$

where $J(\mathbf{x})$ is the Jacobian of $A(\cdot)$ at $\mathbf{x}$.

Example, computing $E(\mathbf{x}|\mathbf{b})$: suppose $\mathbf{x}^i \sim p(\mathbf{x})$, $i = 1, \ldots, k$.

$$\begin{aligned} E(\mathbf{x}|\mathbf{b}) &= \int \mathbf{x}\, p(\mathbf{x}|\mathbf{b})\, d\mathbf{x} \\ &= \int \mathbf{x}\, (p(\mathbf{x}|\mathbf{b})/p(\mathbf{x}))p(\mathbf{x})\, d\mathbf{x} \\ &\approx \left(\sum_{i=1}^{k} w^i\right)^{-1} \sum_{i=1}^{k} \mathbf{x}^i w^i, \qquad w^i = |J(\mathbf{x}^i)^T J(\mathbf{x}^i)|^{-1/2}. \end{aligned}$$

Summary

1. Randomize-then-Optimize yields high quality samples for large-scale inverse problems.
2. In nonlinear cases (nonnegativity constraints, Poisson noise, nonlinear models) the theory for RTO has not been developed.
3. Preliminary results indicate that RTO can be used within an importance sampling framework.