Hierarchical prior models for scalable Bayesian Tomography

Ali Mohammad-Djafari
Laboratoire des Signaux et Systèmes (L2S)
UMR8506 CNRS-CentraleSupélec-UNIV PARIS SUD
SUPELEC, 91192 Gif-sur-Yvette, France
http://lss.centralesupelec.fr
Email: djafari@lss.supelec.fr
http://djafari.free.fr
http://publicationslist.org/djafari

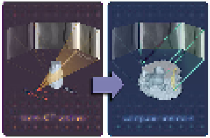
Invited talk at
WSTL, Szeged, Hungary

Contents

1. Computed Tomography or how to See inside of a body
2. Algebraic and Regularization methods
3. Basic Bayesian approach
4. Two main steps:
   - Choosing appropriate Prior model
   - Do the computational efficiently
5. Unsupervised Bayesian methods:
   JMAP, Marginalization, VBA
6. Hierarchical prior modelling
   - Sparsity enforcing through Student-t and IGSM
   - Gauss-Markov-Potts models
7. Computational tools: JMAP, Gibbs Sampling MCMC, VBA
8. Case study: Image Reconstruction with only two projections
9. Implementation issues
   - Main GPU implementation steps: Forward and Back Projections
   - Multi-Resolution implementation
10. Conclusions

Computed Tomography or how to see inside of a body

- $f(x, y)$ a section of a real 3D body $f(x, y, z)$
- $g_\phi(r)$ a line of observed radiography $g_\phi(r, z)$



- Forward model:
  Line integrals or Radon Transform

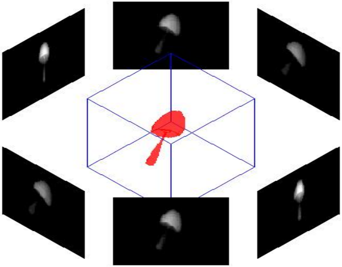
$$
\begin{aligned}
g_\phi(r) &= \int_{L_{r,\phi}} f(x, y)\, \mathrm{d}l + \epsilon_\phi(r) \\
&= \iint f(x, y)\, \delta(r - x\cos\phi - y\sin\phi)\, \mathrm{d}x\, \mathrm{d}y + \epsilon_\phi(r)
\end{aligned}
$$

- Inverse problem: Image reconstruction

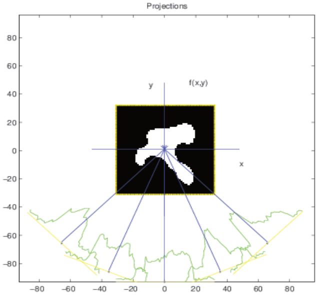
  Given the forward model $\mathcal{H}$ (Radon Transform) and
  a set of data $g_{\phi_i}(r), i = 1, \cdots, M$
  find $f(x, y)$

2D and 3D Computed Tomography

3D



2D

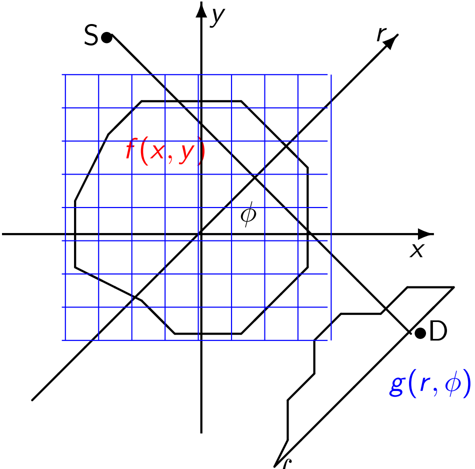


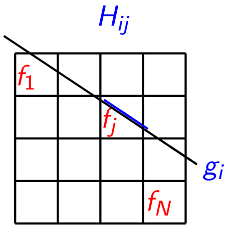
$$g_\phi(r_1, r_2) = \int_{\mathcal{L}_{r_1, r_2, \phi}} f(x, y, z) \, \mathrm{d}l \qquad g_\phi(r) = \int_{\mathcal{L}_{r, \phi}} f(x, y) \, \mathrm{d}l$$

Forward probelm: $f(x, y)$ or $f(x, y, z) \longrightarrow g_\phi(r)$ or $g_\phi(r_1, r_2)$

Inverse problem: $g_\phi(r)$ or $g_\phi(r_1, r_2) \longrightarrow f(x, y)$ or $f(x, y, z)$

Algebraic methods: Discretization





$$f(x, y) = \sum_j f_j \, b_j(x, y)$$

$$b_j(x, y) = \begin{cases} 1 & \text{if } (x, y) \in \text{ pixel } j \\ 0 & \text{else} \end{cases}$$

$$g(r, \phi) = \int_L f(x, y) \, \mathrm{d}l \qquad g_i = \sum_{j=1}^{N} H_{ij} \, f_j + \epsilon_i$$

$$\mathbf{g}_k = \mathbf{H}_k \mathbf{f} + \boldsymbol{\epsilon}_k, k = 1, \cdots, K \longrightarrow \mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$

$\mathbf{g}_k$ projection at angle ϕ_k, $\mathbf{g}$ all the projections.

Algebraic methods

$$\mathbf{g} = \begin{bmatrix} \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_K \end{bmatrix}, \mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \vdots \\ \mathbf{H}_K \end{bmatrix} \rightarrow \mathbf{g}_k = \mathbf{H}_k \mathbf{f} + \boldsymbol{\epsilon}_k \rightarrow \mathbf{g} = \sum_k \mathbf{H}_k \mathbf{f} + \boldsymbol{\epsilon}_k = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$

- $\mathbf{H}$ is huge dimensional: 2D: $10^6 \times 10^6$, 3D: $10^9 \times 10^9$.
- $\mathbf{H}\mathbf{f}$ corresponds to forward projection
- $\mathbf{H}^t\mathbf{g}$ corresponds to Back projection (BP)
- $\mathbf{H}$ may not be invertible and even not square
- $\mathbf{H}$ is, in general, ill-conditioned
- Generalized Inverse Minimum Norm Solution

$$\widehat{\mathbf{f}} = \mathbf{H}^t(\mathbf{H}\mathbf{H}^t)^{-1}\mathbf{g} = \sum_k \mathbf{H}_k^t(\mathbf{H}_k\mathbf{H}_k^t)^{-1}\mathbf{g}_k$$

  can be interpreted as the Filtered Back Projection solution
- Due to ill-posedness of the inverse problems, Generalized inversion and Least squares (LS) methods:

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \left\{ J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 \right\} \rightarrow \widehat{\mathbf{f}} = (\mathbf{H}^t\mathbf{H})^{-1}\mathbf{H}^t\mathbf{g}$$

Inversion: Deterministic methods: Data matching

- Observation model
  $\mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon \rightarrow g_i = [\mathbf{H}\mathbf{f}]_i + \epsilon_i = h_i(\mathbf{f}), \quad i = 1, \ldots, M \longrightarrow$
- Mismatch between data and output of the model $\Delta(\mathbf{g}, \mathbf{H}(\mathbf{f}))$

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{\Delta(\mathbf{g}, \mathbf{H}(\mathbf{f}))\}$$

- Examples:

– LS $\quad \Delta(\mathbf{g}, \mathbf{H}(\mathbf{f})) = \|\mathbf{g} - \mathbf{H}(\mathbf{f})\|^2 = \sum_i |g_i - h_i(\mathbf{f})|^2$

– L_p $\quad \Delta(\mathbf{g}, \mathbf{H}(\mathbf{f})) = \|\mathbf{g} - \mathbf{H}(\mathbf{f})\|^p = \sum_i |g_i - h_i(\mathbf{f})|^p, \; 1 < p < 2$

– KL $\quad \Delta(\mathbf{g}, \mathbf{H}(\mathbf{f})) = \sum_i g_i \ln \frac{g_i}{h_i(\mathbf{f})}$

- In general, does not give satisfactory results for inverse problems.

Regularization theory

Inverse problems = Ill posed problems

$\longrightarrow$ Need for prior information

Functional space (Tikhonov): $g = \mathcal{H}(f) + \epsilon$

$$J(f) = ||g - \mathcal{H}(f)||_2^2 + \lambda||\mathcal{D}f||_2^2$$

Finite dimensional space (Philips & Towmey): $\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$

$$J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\|\mathbf{f}\|^2$$

- Minimum norme LS (MNLS): $J(\mathbf{f}) = ||\mathbf{g} - \mathbf{H}(\mathbf{f})||^2 + \lambda||\mathbf{f}||^2$
- Classical regularization: $J(\mathbf{f}) = ||\mathbf{g} - \mathbf{H}(\mathbf{f})||^2 + \lambda||\mathbf{D}\mathbf{f}||^2$
- More general regularization:

$$J(\mathbf{f}) = \mathcal{Q}(\mathbf{g} - \mathbf{H}(\mathbf{f})) + \lambda\Omega(\mathbf{D}\mathbf{f})$$

or

$$J(\mathbf{f}) = \Delta_1(\mathbf{g}, \mathbf{H}(\mathbf{f})) + \lambda\Delta_2(\mathbf{D}\mathbf{f}, \mathbf{f}_0)$$

Limitations:

- Errors are considered implicitly white and Gaussian
- Limited prior information on the solution
- Lack of tools for the determination of the hyperparameters

Inversion: Probabilistic methods

Taking account of errors and uncertainties $\longrightarrow$ Probability theory

- Maximum Likelihood (ML)
- Minimum Inaccuracy (MI)
- Probability Distribution Matching (PDM)
- Maximum Entropy (ME) and Information Theory (IT)
- Bayesian Inference (BAYES)

Advantages:

- Explicit account of the errors and noise
- A large class of priors via explicit or implicit modelling
- A coherent approach to combine information content of the data and priors

Limitations:

- Practical implementation and cost of calculation

Bayesian estimation approach

$$\mathcal{M}: \qquad \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon$$

- Observation model $\mathcal{M}$ + Hypothesis on the noise $\epsilon \longrightarrow$
  $$p(\mathbf{g}|\mathbf{f};\mathcal{M}) = p_{\epsilon}(\mathbf{g} - \mathbf{H}\mathbf{f})$$
- A priori information $\qquad p(\mathbf{f}|\mathcal{M})$
- Bayes : $$p(\mathbf{f}|\mathbf{g};\mathcal{M}) = \frac{p(\mathbf{g}|\mathbf{f};\mathcal{M})\, p(\mathbf{f}|\mathcal{M})}{p(\mathbf{g}|\mathcal{M})}$$

Link with regularization :

- Maximum A Posteriori (MAP) :
  $$\begin{aligned} \widehat{\mathbf{f}} &= \arg\max_{\mathbf{f}} \{p(\mathbf{f}|\mathbf{g})\} = \arg\max_{\mathbf{f}} \{p(\mathbf{g}|\mathbf{f})\ p(\mathbf{f})\} \\ &= \arg\min_{\mathbf{f}} \{J(\mathbf{f}) = -\ln p(\mathbf{g}|\mathbf{f}) - \ln p(\mathbf{f})\} \end{aligned}$$
- Regularization:
  $$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{J(\mathbf{f}) = Q(\mathbf{g}, \mathbf{H}\mathbf{f}) + \lambda\Omega(\mathbf{f})\}$$

  with $Q(\mathbf{g}, \mathbf{H}\mathbf{f}) = -\ln p(\mathbf{g}|\mathbf{f})$ and $\lambda\Omega(\mathbf{f}) = -\ln p(\mathbf{f})$

Case of linear models and Gaussian priors

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$

- Prior knowledge on the noise:

$$\boldsymbol{\epsilon} \sim \mathcal{N}(0, v_\epsilon{}^2\mathbf{I}) \rightarrow p(\mathbf{g}|\mathbf{f}) \propto \exp\left[-\frac{1}{2v_\epsilon{}^2}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2\right]$$

- Prior knowledge on $\mathbf{f}$:

$$\mathbf{f} \sim \mathcal{N}(0, v_f^2(\mathbf{D}'\mathbf{D})^{-1}) \rightarrow p(\mathbf{f}) \propto \exp\left[-\frac{1}{2v_f^2}\|\mathbf{D}\mathbf{f}\|^2\right]$$

- A posteriori:

$$p(\mathbf{f}|\mathbf{g}) \propto \exp\left[-\frac{1}{2v_\epsilon{}^2}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 - \frac{1}{2v_f^2}\|\mathbf{D}\mathbf{f}\|^2\right]$$

- MAP : $\quad \widehat{\mathbf{f}} = \arg\max_{\mathbf{f}} \{p(\mathbf{f}|\mathbf{g})\} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\}$

  with $\qquad J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\|\mathbf{D}\mathbf{f}\|^2, \qquad \lambda = \frac{v_\epsilon{}^2}{v_f^2}$

- Advantage : characterization of the solution

$$p(\mathbf{f}|\mathbf{g}) = \mathcal{N}(\widehat{\mathbf{f}}, \widehat{\boldsymbol{\Sigma}}) \quad \text{with}$$

$$\widehat{\mathbf{f}} = \left(\mathbf{H}'\mathbf{H} + \lambda\mathbf{D}'\mathbf{D}\right)^{-1}\mathbf{H}'\mathbf{g}, \quad \widehat{\boldsymbol{\Sigma}} = v_\epsilon\left(\mathbf{H}'\mathbf{H} + \lambda\mathbf{D}'\mathbf{D}\right)^{-1}$$

MAP estimation with other priors:

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\} \quad \text{with} \quad J(\mathbf{f}) = \frac{1}{v_\epsilon}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \Omega(\mathbf{f})$$

Separable priors:

- Gaussian:
  $p(f_j) \propto \exp\left[-\alpha|f_j|^2\right] \longrightarrow \Omega(\mathbf{f}) = \alpha \sum_j |f_j|^2 = \|\mathbf{f}\|_2^2$
- Generalized Gaussian:
  $p(f_j) \propto \exp\left[-\alpha|f_j|^p\right], \; 1 < p < 2 \rightarrow \Omega(\mathbf{f}) = \alpha \sum_j |f_j|^p = \|\mathbf{f}\|_p^p$
- Gamma: $p(f_j) \propto f_j^\alpha \exp\left[-\beta f_j\right] \longrightarrow \Omega(\mathbf{f}) = \alpha \sum_j \ln f_j + \beta f_j$
- Beta:
  $p(f_j) \propto f_j^\alpha (1 - f_j)^\beta \longrightarrow \Omega(\mathbf{f}) = \alpha \sum_j \ln f_j + \beta \sum_j \ln(1 - f_j)$

Markovian models:

$$p(f_j|\mathbf{f}) \propto \exp\left[-\alpha \sum_{i \in N_j} \phi(f_j, f_i)\right] \longrightarrow \quad \Omega(\mathbf{f}) = \alpha \sum_j \sum_{i \in N_j} \phi(f_j, f_i),$$

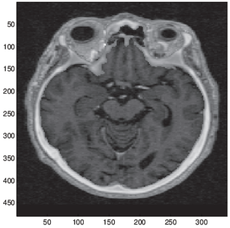
Main advantages of the Bayesian approach

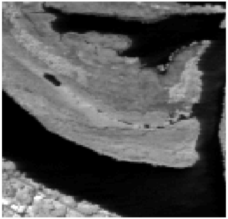
- MAP $=$ Regularization
- Posterior mean ? Marginal MAP ?
- More information in the posterior law than only its mode or its mean
- Tools for estimating hyper parameters
- Tools for model selection
- More specific and specialized priors, particularly through the hidden variables and hierarchical models
- More computational tools:
  - Expectation-Maximization for computing the maximum likelihood parameters
  - MCMC for posterior exploration
  - Variational Bayes for analytical computation of the posterior marginals
  - ...

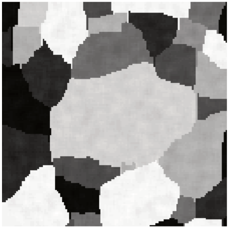
Two main steps in the Bayesian approach

- Prior modeling
  - Separable:
    Gaussian, Gamma,
    Sparsity enforcing: Generalized Gaussian, mixture of Gaussians, mixture of Gammas, ...
  - Markovian:
    Gauss-Markov, GGM, ...
  - Markovian with hidden variables
    (contours, region labels)
- Choice of the estimator and computational aspects
  - MAP, Posterior mean, Marginal MAP
  - MAP needs optimization algorithms
  - Posterior mean needs integration methods
  - Marginal MAP and Hyperparameter estimation need integration and optimization
  - Approximations:
    - Gaussian approximation (Laplace)
    - Numerical exploration MCMC
    - Variational Bayes (Separable approximation)

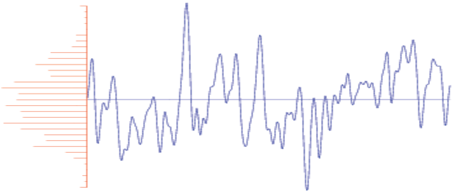
Which images I am looking for?



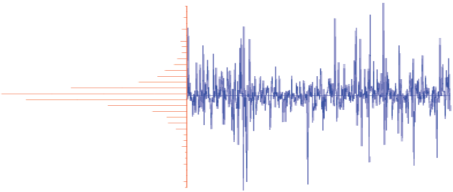




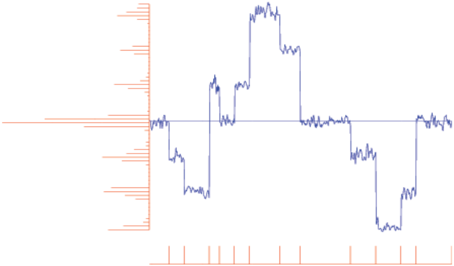
Which image I am looking for?



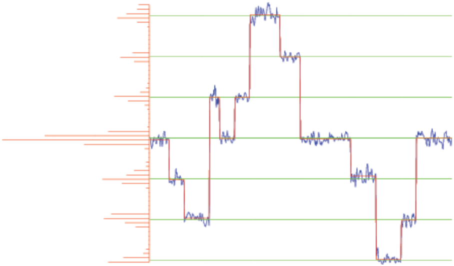
Gauss-Markov



Generalized GM



Piecewize Gaussian



Mixture of GM

Different prior models for signals and images: Separable

- Simple Gaussian, Gamma, Generalized Gaussian

$$p(\mathbf{f}) \propto \exp\left[-\alpha \sum_j \phi(f_j)\right]$$

- Simple Markovian models: Gauss-Markov, Generalized Gauss-Markov

$$p(\mathbf{f}) \propto \exp\left[-\alpha \sum_j \sum_{j \in \mathcal{N}(i)} \phi(f_j - f_i)\right]$$

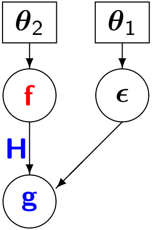
- Hierarchical models with hidden variables: Bernouilli-Gaussian, Gaussian-Gamma

$$p(\mathbf{f}|\mathbf{z}) \propto \exp\left[\sum_j \phi(f_j|z_j)\right] \text{ and } p(\mathbf{z}) \propto \exp\left[\sum_j \phi(z_j)\right]$$

with different choices for $\phi(f_j|z_j)$ and $p\phi(z_j)$

Bayesian inference as Hierarchical models

Simple case:



$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$

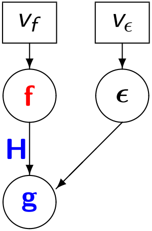
$$p(\mathbf{f}|\mathbf{g}, \boldsymbol{\theta}) \propto p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta}_1)\, p(\mathbf{f}|\boldsymbol{\theta}_2)$$

– Objective: Infer $\mathbf{f}$

– MAP: $\widehat{\mathbf{f}} = \arg\max_{\mathbf{f}} \{p(\mathbf{f}|\mathbf{g}, \boldsymbol{\theta})\}$

– Posterior Mean (PM): $\widehat{\mathbf{f}} = \int \mathbf{f}\, p(\mathbf{f}|\mathbf{g}, \boldsymbol{\theta})\, \mathrm{d}\mathbf{f}$

Example: Caussian case:



$$\begin{cases} p(\mathbf{g}|\mathbf{f}, \theta_1) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, \theta_1 \mathbf{I}) \\ p(\mathbf{f}|\theta_2) = \mathcal{N}(\mathbf{g}|0, \theta_2 \mathbf{I}) \end{cases} \rightarrow p(\mathbf{f}|\mathbf{g}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{f}|\widehat{\mathbf{f}}, \widehat{\boldsymbol{\Sigma}})$$

– MAP: $\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\}$ with

$$J(\mathbf{f}) = \frac{1}{v_\epsilon}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \frac{1}{v_f}\|\mathbf{f}\|^2$$

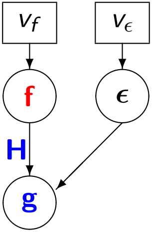
– Posterior Mean (PM)=MAP:

$$\begin{cases} \widehat{\mathbf{f}} = (\mathbf{H}^t\mathbf{H} + \lambda \mathbf{I})^{-1}\mathbf{H}^t\mathbf{g} \\ \widehat{\boldsymbol{\Sigma}} = (\mathbf{H}^t\mathbf{H} + \lambda \mathbf{I})^{-1} \end{cases} \quad \text{with } \lambda = \frac{v_\epsilon}{v_f}.$$

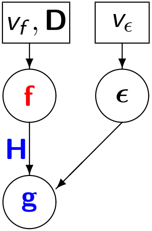
Gaussian model: Simple separable and Markovian

$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$

Separable Gaussian



Gauss-Markov



$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$

$$\begin{cases} p(\mathbf{g}|\mathbf{f}, \theta_1) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|v_f) = \mathcal{N}(\mathbf{g}|0, \theta_2 \mathbf{I}) \end{cases} \rightarrow p(\mathbf{f}|\mathbf{g}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{f}|\widehat{\mathbf{f}}, \widehat{\boldsymbol{\Sigma}})$$

– MAP: $\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\}$ with

$$J(\mathbf{f}) = \frac{1}{v_\epsilon} \|\mathbf{g} - |\mathbf{H}\mathbf{f}\|^2 + \frac{1}{v_f} \|\mathbf{f}\|^2$$

– Posterior Mean (PM)=MAP:

$$\begin{cases} \widehat{\mathbf{f}} = (\mathbf{H}^t\mathbf{H} + \lambda \mathbf{I}]^{-1}\mathbf{H}^t\mathbf{g} \\ \widehat{\boldsymbol{\Sigma}} = v_\epsilon (\mathbf{H}^t\mathbf{H} + \lambda \mathbf{I}]^{-1} \end{cases} \quad \text{with } \lambda = \frac{v_\epsilon}{v_f}.$$

Markovian case:

$$p(\mathbf{f}|v_f, \mathbf{D}) = \mathcal{N}(\mathbf{g}|0, v_f(\mathbf{D}\mathbf{D}^t)^{-1})$$

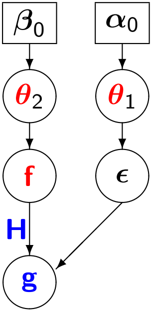
– MAP: $J(\mathbf{f}) = \frac{1}{v_\epsilon} \|\mathbf{g} - |\mathbf{H}\mathbf{f}\|^2 + \frac{1}{v_f} \|\mathbf{D}\mathbf{f}\|^2$

– Posterior Mean (PM)=MAP:

$$\begin{cases} \widehat{\mathbf{f}} = (\mathbf{H}^t\mathbf{H} + \lambda \mathbf{D}^t\mathbf{D}]^{-1}\mathbf{H}^t\mathbf{g} \\ \widehat{\boldsymbol{\Sigma}} = v_\epsilon (\mathbf{H}^t\mathbf{H} + \lambda \mathbf{D}^t\mathbf{D}]^{-1} \end{cases} \quad \text{with } \lambda = \frac{v_e}{v_f}.$$

Bayesian inference (Unsupervised case)

Unsupervised case: Hyper parameter estimation



$$p(\mathbf{f}, \theta | \mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \theta_1)\, p(\mathbf{f}|\theta_2)\, p(\theta)$$

– Objective: Infer $(\mathbf{f}, \theta)$

JMAP: $(\widehat{\mathbf{f}}, \widehat{\theta}) = \arg\max_{(\mathbf{f},\theta)} \{p(\mathbf{f}, \theta|\mathbf{g})\}$

– Marginalization 1:

$$p(\mathbf{f}|\mathbf{g}) = \int p(\mathbf{f}, \theta|\mathbf{g})\, d\theta$$

– Marginalization 2:

$p(\theta|\mathbf{g}) = \int p(\mathbf{f}, \theta|\mathbf{g})\, d\mathbf{f}$ followed by:

$$\widehat{\theta} = \arg\max_{\theta} \{p(\theta|\mathbf{g})\} \rightarrow \widehat{\mathbf{f}} = \arg\max_{\mathbf{f}} \left\{p(\mathbf{f}|\mathbf{g}, \widehat{\theta})\right\}$$

– MCMC Gibbs sampling:

$\mathbf{f} \sim p(\mathbf{f}|\theta, \mathbf{g}) \rightarrow \theta \sim p(\theta|\mathbf{f}, \mathbf{g})$ until convergence

Use samples generated to compute mean and variances

– VBA: Approximate $p(\mathbf{f}, \theta|\mathbf{g})$ by $q_1(\mathbf{f})\, q_2(\theta)$

Use $q_1(\mathbf{f})$ to infer $\mathbf{f}$ and $q_2(\theta)$ to infer θ

Variational Bayesian Approximation

- Approximate $p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g})$ by $q(\mathbf{f},\boldsymbol{\theta}|\mathbf{g}) = q_1(\mathbf{f}|\mathbf{g})\, q_2(\boldsymbol{\theta}|\mathbf{g})$ and then continue computations.
- Criterion $\mathrm{KL}(q(\mathbf{f},\boldsymbol{\theta}|\mathbf{g}) : p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g}))$

$$\mathrm{KL}(q:p) = \iint q \ln q/p = \iint q_1 q_2 \ln \frac{q_1 q_2}{p} =$$
$$\int q_1 \ln q_1 + \int q_2 \ln q_2 - \iint q \ln p = -H(q_1) - H(q_2) - < \ln p >_q$$

- Alternate optimization algorithm $q_1 \longrightarrow q_2 \longrightarrow q_1 \longrightarrow q_2, \cdots$

$$\begin{cases} q_1(\mathbf{f}) & \propto \exp\left[\langle \ln p(\mathbf{g},\mathbf{f},\boldsymbol{\theta};\mathcal{M})\rangle_{q_2(\boldsymbol{\theta})}\right] \\ q_2(\boldsymbol{\theta}) & \propto \exp\left[\langle \ln p(\mathbf{g},\mathbf{f},\boldsymbol{\theta};\mathcal{M})\rangle_{q_1(\mathbf{f})}\right] \end{cases}$$

$$\boxed{p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g})} \longrightarrow \boxed{\text{Variational Bayesian Approximation}} \begin{array}{l} \longrightarrow \widehat{q}_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}} \\ \longrightarrow \widehat{q}_2(\boldsymbol{\theta}) \longrightarrow \widehat{\boldsymbol{\theta}} \end{array}$$

JMAP, Marginalization, VBA

- JMAP:

$$\boxed{\begin{array}{c} p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g}) \\ \text{optimization} \end{array}} \begin{array}{l} \longrightarrow \widehat{\mathbf{f}} \\ \longrightarrow \widehat{\boldsymbol{\theta}} \end{array}$$

- Marginalization

$$\boxed{p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g})} \longrightarrow \boxed{p(\boldsymbol{\theta}|\mathbf{g})} \longrightarrow \widehat{\boldsymbol{\theta}} \longrightarrow \boxed{p(\mathbf{f}|\widehat{\boldsymbol{\theta}},\mathbf{g})} \longrightarrow \widehat{\mathbf{f}}$$

Joint Posterior — Marginalize over $\mathbf{f}$

- Variational Bayesian Approximation

$$\boxed{p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g})} \longrightarrow \boxed{\begin{array}{c} \text{Variational} \\ \text{Bayesian} \\ \text{Approximation} \end{array}} \begin{array}{l} \longrightarrow q_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}} \\ \longrightarrow q_2(\boldsymbol{\theta}) \longrightarrow \widehat{\boldsymbol{\theta}} \end{array}$$

Variational Bayesian Approximation

- Approximate $p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g})$ by $q(\mathbf{f}, \boldsymbol{\theta}) = q_1(\mathbf{f})\, q_2(\boldsymbol{\theta})$ and then use them for any inferences on $\mathbf{f}$ and $\boldsymbol{\theta}$ respectively.
- Criterion $\mathrm{KL}(q(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) : p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}))$
  $$\mathrm{KL}(q : p) = \int\int q \ln \frac{q}{p} = \int\int q_1 q_2 \ln \frac{q_1 q_2}{p}$$
- Iterative algorithm $q_1 \longrightarrow q_2 \longrightarrow q_1 \longrightarrow q_2, \cdots$

$$\left\{ \begin{array}{ll} \widehat{q}_1(\mathbf{f}) & \propto \exp\left[\langle \ln p(\mathbf{g}, \mathbf{f}, \boldsymbol{\theta}; \mathcal{M}) \rangle_{\widehat{q}_2(\boldsymbol{\theta})}\right] \\ \widehat{q}_2(\boldsymbol{\theta}) & \propto \exp\left[\langle \ln p(\mathbf{g}, \mathbf{f}, \boldsymbol{\theta}; \mathcal{M}) \rangle_{\widehat{q}_1(\mathbf{f})}\right] \end{array} \right.$$

$p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) \longrightarrow$ **Variational Bayesian Approximation** $\longrightarrow q_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}}$, $\longrightarrow q_2(\boldsymbol{\theta}) \longrightarrow \widehat{\boldsymbol{\theta}}$

BVA: Choice of family of laws q_1 and q_2

- Case 1 : $\longrightarrow$ Joint MAP

$$\left\{\begin{array}{ll}\widehat{q}_1(\mathbf{f}|\widetilde{\mathbf{f}}) & = \delta(\mathbf{f}-\widetilde{\mathbf{f}}) \\ \widehat{q}_2(\boldsymbol{\theta}|\widetilde{\boldsymbol{\theta}}) & = \delta(\boldsymbol{\theta}-\widetilde{\boldsymbol{\theta}})\end{array}\right. \longrightarrow \left\{\begin{array}{l}\widetilde{\mathbf{f}} = \arg\max_{\mathbf{f}} \left\{p(\mathbf{f},\widetilde{\boldsymbol{\theta}}|\mathbf{g};\mathcal{M})\right\} \\ \widetilde{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \left\{p(\widetilde{\mathbf{f}},\boldsymbol{\theta}|\mathbf{g};\mathcal{M})\right\}\end{array}\right.$$

- Case 2 : $\longrightarrow$ EM

$$\left\{\begin{array}{ll}\widehat{q}_1(\mathbf{f}) & \propto p(\mathbf{f}|\boldsymbol{\theta},\mathbf{g}) \\ \widehat{q}_2(\boldsymbol{\theta}|\widetilde{\boldsymbol{\theta}}) & = \delta(\boldsymbol{\theta}-\widetilde{\boldsymbol{\theta}})\end{array}\right. \longrightarrow \left\{\begin{array}{ll}Q(\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}}) & = \langle \ln p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g};\mathcal{M})\rangle_{q_1(\mathbf{f}|\widetilde{\boldsymbol{\theta}})} \\ \widetilde{\boldsymbol{\theta}} & = \arg\max_{\boldsymbol{\theta}} \left\{Q(\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}})\right\}\end{array}\right.$$

- Appropriate choice for inverse problems with Exponential families and Conjugate priors

$$\left\{\begin{array}{ll}\widehat{q}_1(\mathbf{f}) & \propto p(\mathbf{f}|\widetilde{\boldsymbol{\theta}},\mathbf{g};\mathcal{M}) \\ \widehat{q}_2(\boldsymbol{\theta}) & \propto p(\boldsymbol{\theta}|\widetilde{\mathbf{f}},\mathbf{g};\mathcal{M})\end{array}\right. \longrightarrow \left\{\begin{array}{l}\text{Accounts for the uncertainties of} \\ \widehat{\boldsymbol{\theta}} \text{ for } \widehat{\mathbf{f}} \text{ and vice versa.}\end{array}\right.$$

JMAP, EM and VBA

JMAP Alternate optimization Algorithm:

$$\boldsymbol{\theta}^{(0)} \longrightarrow \widetilde{\boldsymbol{\theta}} \longrightarrow \boxed{\widetilde{\mathbf{f}} = \arg\max_{\mathbf{f}} \left\{ p(\mathbf{f}, \widetilde{\boldsymbol{\theta}} | \mathbf{g}) \right\}} \longrightarrow \widetilde{\mathbf{f}} \longrightarrow \widehat{\mathbf{f}}$$

$$\uparrow \qquad\qquad\qquad\qquad\qquad\qquad \downarrow$$

$$\widehat{\boldsymbol{\theta}} \longleftarrow \widetilde{\boldsymbol{\theta}} \longleftarrow \boxed{\widetilde{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \left\{ p(\widetilde{\mathbf{f}}, \boldsymbol{\theta} | \mathbf{g}) \right\}} \longleftarrow \widetilde{\mathbf{f}}$$

EM:

$$\boldsymbol{\theta}^{(0)} \longrightarrow \widetilde{\boldsymbol{\theta}} \longrightarrow \boxed{q_1(\mathbf{f}) = p(\mathbf{f} | \widetilde{\boldsymbol{\theta}}, \mathbf{g})} \longrightarrow q_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}}$$

$$\uparrow \qquad\qquad\qquad\qquad\qquad\qquad \downarrow$$

$$\widehat{\boldsymbol{\theta}} \longleftarrow \widetilde{\boldsymbol{\theta}} \longleftarrow \boxed{\begin{array}{c} Q(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) = \langle \ln p(\mathbf{f}, \boldsymbol{\theta} | \mathbf{g}) \rangle_{q_1(\mathbf{f})} \\ \widetilde{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \left\{ Q(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) \right\} \end{array}} \longleftarrow q_1(\mathbf{f})$$

VBA:

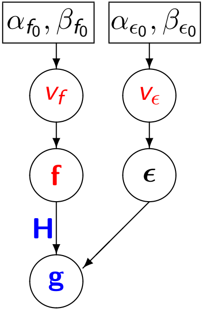
$$\boldsymbol{\theta}^{(0)} \longrightarrow q_2(\boldsymbol{\theta}) \longrightarrow \boxed{q_1(\mathbf{f}) \propto \exp\left[\langle \ln p(\mathbf{f}, \boldsymbol{\theta} | \mathbf{g}) \rangle_{q_2(\boldsymbol{\theta})} \right]} \longrightarrow q_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}}$$

$$\uparrow \qquad\qquad\qquad\qquad\qquad\qquad \downarrow$$

$$\widehat{\boldsymbol{\theta}} \longleftarrow q_2(\boldsymbol{\theta}) \longleftarrow \boxed{q_2(\boldsymbol{\theta}) \propto \exp\left[\langle \ln p(\mathbf{f}, \boldsymbol{\theta} | \mathbf{g}) \rangle_{q_1(\mathbf{f})} \right]} \longleftarrow q_1(\mathbf{f})$$

Unsupervised Gaussian model

Unsupervised Gaussian case: $\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}|0, v_\epsilon \mathbf{I})$



$$\begin{cases} p(\mathbf{g}|\mathbf{f}, \mathbf{v}_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|\mathbf{v}_f) = \mathcal{N}(\mathbf{g}|0, v_f \mathbf{I}) \end{cases}$$

$$\begin{cases} p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(v_f) = \mathcal{IG}(v_f|\alpha_{f_0}, \beta_{f_0}) \end{cases}$$

$p(\mathbf{f}, v_\epsilon, v_f|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, v_\epsilon)\, p(\mathbf{f}|v_f)\, p(v_\epsilon)\, p(v_f)$

– JMAP: $\quad (\widehat{\mathbf{f}}, \hat{v}_\epsilon, \widehat{v}_f) = \arg\max_{(\mathbf{f}, v_\epsilon, v_f)} \{p(\mathbf{f}, v_\epsilon, v_f|\mathbf{g})\}$

Alternate optimization:

$$\begin{cases} \widehat{\mathbf{f}} = \left(\mathbf{H}'\mathbf{H} + \widehat{\lambda}\mathbf{I}\right)^{-1} \mathbf{H}'\mathbf{g} \quad \text{with } \widehat{\lambda} = \frac{\hat{v}_\epsilon}{\widehat{v}_f} \\ \hat{v}_\epsilon = \frac{\widetilde{\beta}_\epsilon}{\widetilde{\alpha}_\epsilon}, \widetilde{\beta}_\epsilon = \beta_{\epsilon_0} + \|\mathbf{g} - \mathbf{H}\widehat{\mathbf{f}}\|^2, \widetilde{\alpha}_\epsilon = \alpha_{\epsilon_0} + M/2 \\ \widehat{v}_f = \frac{\widetilde{\beta}_f}{\widetilde{\alpha}_f}, \widetilde{\beta}_f = \beta_{f_0} + \|\widehat{\mathbf{f}}\|^2, \widetilde{\alpha}_f = \alpha_{f_0} + N/2 \end{cases}$$

– VBA: Approximate $p(\mathbf{f}, v_\epsilon, v_f|\mathbf{g})$ by $q_1(\mathbf{f})\, q_2(v_\epsilon)\, q_3(v_f)$

Alternate optimization:

$$\begin{cases} q_1(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\widetilde{\boldsymbol{\mu}}, \widetilde{\boldsymbol{\Sigma}}_f) \\ q_2(v_\epsilon) = \mathcal{IG}(v_\epsilon|\widetilde{\alpha}_\epsilon, \widetilde{\beta}_\epsilon) \\ q_3(v_f) = \mathcal{IG}(v_\epsilon|\widetilde{\alpha}_f, \widetilde{\beta}_f) \end{cases}$$

Unsupervised Gaussian model

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}|0, v_\epsilon \mathbf{I})$$

$$\begin{cases} p(\mathbf{g}|\mathbf{f}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|v_f) = \mathcal{N}(\mathbf{g}|0, v_f \mathbf{I}) \end{cases}, \begin{cases} p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(v_f) = \mathcal{IG}(v_f|\alpha_{f_0}, \beta_{f_0}) \end{cases}$$

$$p(\mathbf{f}, v_\epsilon, v_f|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, v_\epsilon)\, p(\mathbf{f}|v_f)\, p(v_\epsilon)\, p(v_f)$$

- JMAP: $(\widehat{\mathbf{f}}, \hat{v}_\epsilon, \widehat{v}_f) = \arg\max_{(\mathbf{f}, v_\epsilon, v_f)} \{p(\mathbf{f}, v_\epsilon, v_f|\mathbf{g})\}$

$$\begin{cases} \widehat{\mathbf{f}} = \left(\mathbf{H}'\mathbf{H} + \widehat{\lambda}\mathbf{I}\right)^{-1} \mathbf{H}'\mathbf{g} \quad \text{with } \widehat{\lambda} = \frac{\hat{v}_\epsilon}{\widehat{v}_f} \\ \hat{v}_\epsilon = \frac{\widetilde{\beta}_\epsilon}{\widetilde{\alpha}_\epsilon}, \quad \widetilde{\beta}_\epsilon = \beta_{\epsilon_0} + \|\mathbf{g} - \mathbf{H}\widehat{\mathbf{f}}\|^2, \quad \widetilde{\alpha}_\epsilon = \alpha_{\epsilon_0} + M/2 \\ \widehat{v}_f = \frac{\widetilde{\beta}_f}{\widetilde{\alpha}_f}, \quad \widetilde{\beta}_f = \beta_{f_0} + \|\widehat{\mathbf{f}}\|^2, \quad \widetilde{\alpha}_f = \alpha_{f_0} + N/2 \end{cases}$$

- VBA: Approximate $p(\mathbf{f}, v_\epsilon, v_f|\mathbf{g})$ by $q_1(\mathbf{f})\, q_2(v_\epsilon)\, q_3(v_f)$

$$\begin{cases} q_1(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\widetilde{\boldsymbol{\mu}}, \widetilde{\boldsymbol{\Sigma}}_f) \\ q_2(v_\epsilon) = \mathcal{IG}(v_\epsilon|\widetilde{\alpha}_\epsilon, \widetilde{\beta}_\epsilon) \\ q_3(v_f) = \mathcal{IG}(v_\epsilon|\widetilde{\alpha}_f, \widetilde{\beta}_f) \end{cases} \begin{cases} \widetilde{\boldsymbol{\mu}} = \left(\mathbf{H}'\mathbf{H} + \widetilde{\lambda}\mathbf{I}\right)^{-1} \mathbf{H}'\mathbf{g} \quad \text{with } \widetilde{\lambda} = \frac{\hat{v}_\epsilon}{\widehat{v}_f} \\ \widetilde{\boldsymbol{\Sigma}} = \hat{v}_\epsilon \left(\mathbf{H}'\mathbf{H} + \widetilde{\lambda}\mathbf{I}\right)^{-1} \\ \hat{v}_\epsilon = \frac{\widetilde{\beta}_\epsilon}{\widetilde{\alpha}_\epsilon}, \ \widetilde{\beta}_\epsilon = \beta_{\epsilon_0} + < \|\mathbf{g} - \mathbf{H}\widehat{\mathbf{f}}\|^2 >, \ \widetilde{\alpha}_\epsilon = \alpha_{\epsilon_0} + \frac{M}{2} \\ \widehat{v}_f = \frac{\widetilde{\beta}_f}{\widetilde{\alpha}_f}, \ \widetilde{\beta}_f = \beta_{f_0} + < \|\widehat{\mathbf{f}}\|^2 >, \ \widetilde{\alpha}_f = \alpha_{f_0} + \frac{N}{2} \end{cases}$$

$$< \|\mathbf{g} - \mathbf{H}\widehat{\mathbf{f}}\|^2 > = \|\mathbf{g} - \mathbf{H}\widetilde{\boldsymbol{\mu}}\|^2 + \mathbf{H}\operatorname{diag}\left[\widetilde{\boldsymbol{\Sigma}}\right]\mathbf{H}^t, < \|\widehat{\mathbf{f}}\|^2 > = \|\widetilde{\boldsymbol{\mu}}\|^2 + \operatorname{diag}\left[\widetilde{\boldsymbol{\Sigma}}\right]$$

Sparsity enforcing models

- Student-t model

$$\mathcal{S}t(f|\nu) \propto \exp\left[-\frac{\nu+1}{2}\log\left(1+f^2/\nu\right)\right]$$

- Infinite Gausian Scaled Mixture (IGSM) equivalence

$$\mathcal{S}t(f|\nu) \propto \int_0^\infty \mathcal{N}(f|,0,1/z)\,\mathcal{G}(z|\alpha,\beta)\,\mathrm{d}z, \quad \text{with } \alpha=\beta=\nu/2$$

$$\left\{\begin{array}{ll} p(\mathbf{f}|\mathbf{z}) & = \prod_j p(f_j|z_j) = \prod_j \mathcal{N}(f_j|0,1/z_j) \propto \exp\left[-\frac{1}{2}\sum_j z_j f_j^2\right] \\ p(\mathbf{z}|\alpha,\beta) & = \prod_j \mathcal{G}(z_j|\alpha,\beta) \propto \prod_j z_j^{(\alpha-1)} \exp\left[-\beta z_j\right] \\ & \propto \exp\left[\sum_j (\alpha-1)\ln z_j - \beta z_j\right] \\ p(\mathbf{f},\mathbf{z}|\alpha,\beta) & \propto \exp\left[-\frac{1}{2}\sum_j z_j f_j^2 + (\alpha-1)\ln z_j - \beta z_j\right] \end{array}\right.$$

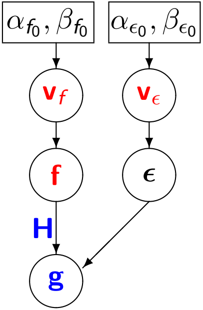
Non stationary noise and Integrated Gaussian Scaled Mixture (IGSM) model

– Non stationary noise:

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}, \;\; \epsilon_i \sim \mathcal{N}(\epsilon_i|0, v_{\epsilon_i}) \rightarrow \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}|0, \mathbf{V}_\epsilon = \mathrm{diag}\,[v_{\epsilon 1}, \cdots, v_{\epsilon M}])$$

– Student-t prior model and its equivalent IGSM :

$$f_j | v_{f_j} \sim \mathcal{N}(f_j|0, v_{f_j}) \text{ and } v_{f_j} \sim \mathcal{IG}(v_{f_j}|\alpha_{f_0}, \beta_{f_0}) \rightarrow f_j \sim \mathcal{S}t(f_j|\alpha_{f_0}, \beta_{f_0})$$



$$\begin{cases} p(\mathbf{g}|\mathbf{f}, \mathbf{v}_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, \mathbf{V}_\epsilon), & \mathbf{V}_\epsilon = \mathrm{diag}\,[\mathbf{v}_\epsilon] \\ p(\mathbf{f}|\mathbf{v}_f) = \mathcal{N}(\mathbf{g}|0, \mathbf{V}_f), & \mathbf{V}_f = \mathrm{diag}\,[\mathbf{v}_f] \end{cases}$$

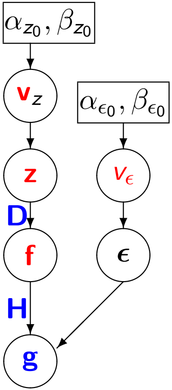
$$\begin{cases} p(\mathbf{v}_\epsilon) = \prod_i \mathcal{IG}(v_{\epsilon_i}|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(\mathbf{v}_f) = \prod_i \mathcal{IG}(v_{f_j}|\alpha_{f_0}, \beta_{f_0}) \end{cases}$$

$$p(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \mathbf{v}_\epsilon)\, p(\mathbf{f}|\mathbf{v}_f)\, p(\mathbf{v}_\epsilon)\, p(\mathbf{v}_f)$$

– JMAP: $(\widehat{\mathbf{f}}, \hat{\mathbf{v}}_\epsilon, \hat{\mathbf{v}}_f) = \arg\max_{(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f)} \{p(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f|\mathbf{g})\}$

– VBA: Approximate $p(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f|\mathbf{g})$
by $q_1(\mathbf{f})\, q_2(\mathbf{v}_\epsilon)\, q_3(\mathbf{v}_f)$

Sparse model in a Transform domain 1



$\mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon, \quad \mathbf{f} = \mathbf{D}\mathbf{z}, \quad \mathbf{z}$ sparse

$$\begin{cases} p(\mathbf{g}|\mathbf{z}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{D}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{z}|\mathbf{v}_z) = \mathcal{N}(\mathbf{z}|0, \mathbf{V}_z), \qquad \mathbf{V}_z = \operatorname{diag}[\mathbf{v}_z] \end{cases}$$

$$\begin{cases} p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(\mathbf{v}_z) = \prod_i \mathcal{IG}(v_{z_j}|\alpha_{z_0}, \beta_{z_0}) \end{cases}$$

$$p(\mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{z}, v_\epsilon)\, p(\mathbf{z}|\mathbf{v}_z)\, p(v_\epsilon)\, p(\mathbf{v}_z)\, p(v_\xi)$$

– JMAP:

$$(\widehat{\mathbf{z}}, \hat{v}_\epsilon, \widehat{\mathbf{v}}_z) = \underset{(\mathbf{z}, v_\epsilon, \mathbf{v}_z)}{\arg\max} \left\{ p(\mathbf{z}, v_\epsilon, \mathbf{v}_z|\mathbf{g}) \right\}$$

Alternate optimization:

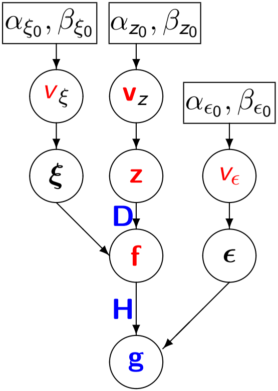
$$\begin{cases} \widehat{\mathbf{z}} = \arg\min_{\mathbf{z}} \{J(\mathbf{z})\} \text{ with:} \\ J(\mathbf{z}) = \frac{1}{2\hat{v}_\epsilon}\|\mathbf{g} - \mathbf{H}\mathbf{D}\mathbf{z}\|^2 + \|\mathbf{V}_z{}^{-1/2}\mathbf{z}\|^2 \\ \widehat{v}_{z_j} = \frac{\beta_{z_0} + \widehat{z}_j^2}{\alpha_{z_0} + 1/2} \\ \hat{v}_\epsilon = \frac{\beta_{\epsilon_0} + \|\mathbf{g} - \mathbf{H}\mathbf{D}\widehat{\mathbf{z}}\|^2}{\alpha_{\epsilon_0} + M/2} \end{cases}$$

– VBA: Approximate

$p(\mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g})$ by $q_1(\mathbf{z})\, q_2(v_\epsilon)\, q_3(\mathbf{v}_z)$

Alternate optimization.

Sparse model in a Transform domain 2



$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}, \quad \mathbf{f} = \mathbf{D}\mathbf{z} + \boldsymbol{\xi}, \quad \mathbf{z} \text{ sparse}$$

$$\begin{cases} p(\mathbf{g}|\mathbf{f}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|\mathbf{z}) = \mathcal{N}(\mathbf{f}|\mathbf{D}\mathbf{z}, v_\xi \mathbf{I}), \\ p(\mathbf{z}|\mathbf{v}_z) = \mathcal{N}(\mathbf{z}|0, \mathbf{V}_z), \qquad \mathbf{V}_z = \operatorname{diag}[\mathbf{v}_z] \end{cases}$$

$$\begin{cases} p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(\mathbf{v}_z) = \prod_i \mathcal{IG}(v_{z_j}|\alpha_{z_0}, \beta_{z_0}) \\ p(v_\xi) = \mathcal{IG}(v_\xi|\alpha_{\xi_0}, \beta_{\xi_0}) \end{cases}$$

$$p(\mathbf{f}, \mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \mathbf{v}_\epsilon)\, p(\mathbf{f}|\mathbf{z}_f)\, p(\mathbf{z}|\mathbf{v}_z)\, p(v_\epsilon)\, p(\mathbf{v}_z)\, p(v_\xi)$$

– JMAP:

$$(\widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \hat{v}_\epsilon, \widehat{\mathbf{v}}_z, \widehat{v}_\xi) = \underset{(\mathbf{f}, \mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi)}{\arg\max} \{p(\mathbf{f}, \mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g})\}$$

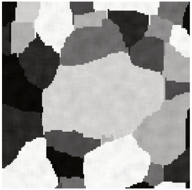
Alternate optimization.

– VBA: Approximate

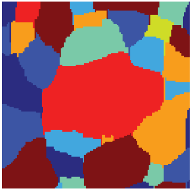
$p(\mathbf{f}, \mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g})$ by $q_1(\mathbf{f})\, q_2(\mathbf{z})\, q_3(v_\epsilon)\, q_4(\mathbf{v}_z)\, q_5(v_\xi)$

Alternate optimization.

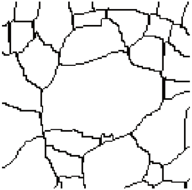
Gauss-Markov-Potts prior models for images



$f(\mathbf{r})$



$z(\mathbf{r})$



$c(\mathbf{r}) = 1 - \delta(z(\mathbf{r}) - z(\mathbf{r}'))$

$$p(f(\mathbf{r})|z(\mathbf{r}) = k, m_k, v_k) = \mathcal{N}(m_k, v_k)$$

$$p(f(\mathbf{r})) = \sum_k P(z(\mathbf{r}) = k)\,\mathcal{N}(m_k, v_k) \quad \text{Mixture of Gaussians}$$

- Separable iid hidden variables: $p(\mathbf{z}) = \prod_{\mathbf{r}} p(z(\mathbf{r}))$
- Markovian hidden variables: $p(\mathbf{z})$ Potts-Markov:

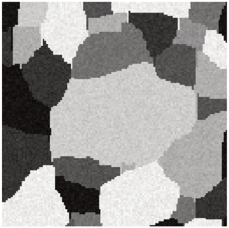
$$p(z(\mathbf{r})|z(\mathbf{r}'), \mathbf{r}' \in \mathcal{V}(\mathbf{r})) \propto \exp\left[\gamma \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right]$$

$$p(\mathbf{z}) \propto \exp\left[\gamma \sum_{\mathbf{r} \in \mathcal{R}} \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right]$$

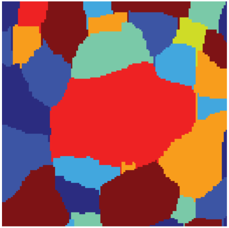
Four different cases

To each pixel of the image is associated 2 variables $f(\mathbf{r})$ and $z(\mathbf{r})$

- $\mathbf{f}|\mathbf{z}$ Gaussian iid, $\mathbf{z}$ iid :
  Mixture of Gaussians
- $\mathbf{f}|\mathbf{z}$ Gauss-Markov, $\mathbf{z}$ iid :
  Mixture of Gauss-Markov
- $\mathbf{f}|\mathbf{z}$ Gaussian iid, $\mathbf{z}$ Potts-Markov :
  Mixture of Independent Gaussians
  (MIG with Hidden Potts)
- $\mathbf{f}|\mathbf{z}$ Markov, $\mathbf{z}$ Potts-Markov :
  Mixture of Gauss-Markov
  (MGM with hidden Potts)

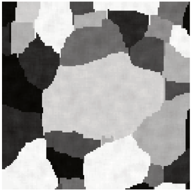


$f(\mathbf{r})$

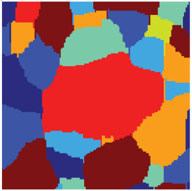


$z(\mathbf{r})$

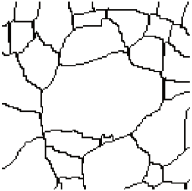
Gauss-Markov-Potts prior models for images



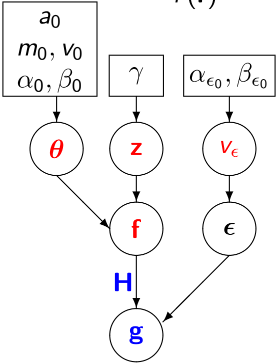
$f(\mathbf{r})$



$z(\mathbf{r})$



$c(\mathbf{r}) = 1 - \delta(z(\mathbf{r}) - z(\mathbf{r}'))$



$$\begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon \\ p(\mathbf{g}|\mathbf{f}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \end{cases}$$

$$\begin{cases} p(f(\mathbf{r})|z(\mathbf{r}) = k, m_k, v_k) = \mathcal{N}(f(\mathbf{r})|m_k, v_k) \\ p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}) = \sum_k \prod_{\mathbf{r} \in \mathcal{R}_k} a_k \mathcal{N}(f(\mathbf{r})|m_k, v_k), \\ \qquad \boldsymbol{\theta} = \{(a_k, m_k, v_k), k = 1, \cdots, K\} \\ p(\boldsymbol{\theta}) = \mathcal{D}(\mathbf{a}|a_0)\mathcal{N}(\mathbf{a}|m_0, v0)\mathcal{IG}(\mathbf{v}|\alpha_0, \beta_0) \\ p(\mathbf{z}|\gamma) \propto \exp\left[\gamma \sum_{\mathbf{r}} \sum_{\mathbf{r}' \in \mathcal{N}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right] \quad \text{Potts MRF} \end{cases}$$

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, v_\epsilon)\, p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta})\, p(\mathbf{z}|\gamma)$$

MCMC: Gibbs Sampling

VBA: Alternate optimization.

Bayesian Computation and Algorithms

- Joint posterior probability law of all the unknowns $\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}$

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta}_1)\; p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}_2)\; p(\mathbf{z}|\boldsymbol{\theta}_3)\; p(\boldsymbol{\theta})$$

- Often, the expression of $p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g})$ is complex.
- Its optimization (for Joint MAP) or
  its marginalization or integration (for Marginal MAP or PM)
  is not easy
- Two main techniques:
  - MCMC:
    Needs the expressions of the conditionals
    $p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}, \mathbf{g})$, $p(\mathbf{z}|\mathbf{f}, \boldsymbol{\theta}, \mathbf{g})$, and $p(\boldsymbol{\theta}|\mathbf{f}, \mathbf{z}, \mathbf{g})$
  - VBA: Approximate $p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g})$ by a separable one

$$q(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) = q_1(\mathbf{f})\; q_2(\mathbf{z})\; q_3(\boldsymbol{\theta})$$

    and do any computations with these separable ones.

MCMC based algorithm

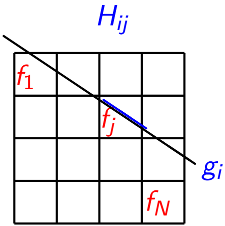
$$p(\mathbf{f},\mathbf{z},\boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f},\mathbf{z},\boldsymbol{\theta}_1)\, p(\mathbf{f}|\mathbf{z},\boldsymbol{\theta}_2)\, p(\mathbf{z}|\boldsymbol{\theta}_3)\, p(\boldsymbol{\theta})$$

General Gibbs sampling scheme:

$$\widehat{\mathbf{f}} \sim p(\mathbf{f}|\widehat{\mathbf{z}},\widehat{\boldsymbol{\theta}},\mathbf{g}) \longrightarrow \widehat{\mathbf{z}} \sim p(\mathbf{z}|\widehat{\mathbf{f}},\widehat{\boldsymbol{\theta}},\mathbf{g}) \longrightarrow \widehat{\boldsymbol{\theta}} \sim (\boldsymbol{\theta}|\widehat{\mathbf{f}},\widehat{\mathbf{z}},\mathbf{g})$$

- Generate samples $\mathbf{f}$ using $p(\mathbf{f}|\widehat{\mathbf{z}},\widehat{\boldsymbol{\theta}},\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f},\boldsymbol{\theta})\, p(\mathbf{f}|\widehat{\mathbf{z}},\widehat{\boldsymbol{\theta}})$
  When Gaussian, can be done via optimization of a quadratic criterion.
- Generate samples $\mathbf{z}$ using $p(\mathbf{z}|\widehat{\mathbf{f}},\widehat{\boldsymbol{\theta}},\mathbf{g}) \propto p(\mathbf{g}|\widehat{\mathbf{f}},\widehat{\mathbf{z}},\widehat{\boldsymbol{\theta}})\, p(\mathbf{z})$
  Often needs sampling (hidden discrete variable)
- Generate samples $\boldsymbol{\theta}$ using
  $p(\boldsymbol{\theta}|\widehat{\mathbf{f}},\widehat{\mathbf{z}},\mathbf{g}) \propto p(\mathbf{g}|\widehat{\mathbf{f}},\sigma_\epsilon^2\mathbf{I})\, p(\widehat{\mathbf{f}}|\widehat{\mathbf{z}},(m_k,v_k))\, p(\boldsymbol{\theta})$
  Use of Conjugate priors $\longrightarrow$ analytical expressions.
- After convergence use samples to compute means and variances.

Computed Tomography with only two projections



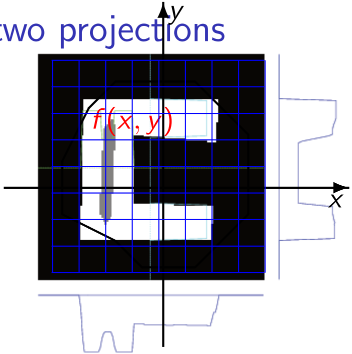
$$g(r,\phi) = \int_L f(x,y)\,\mathrm{d}l$$

$$f(x,y) = \sum_j f_j\, b_j(x,y)$$

$$b_j(x,y) = \begin{cases} 1 & \text{if } (x,y) \in \text{ pixel } j \\ 0 & \text{else} \end{cases}$$

$$g_i = \sum_{j=1}^{N} H_{ij}\, f_j + \epsilon_i$$

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$



Case study: Reconstruction from 2 projections

$g_1(x) = \int f(x,y)\,\mathrm{d}y,$

$g_2(y) = \int f(x,y)\,\mathrm{d}x$

Very ill-posed inverse problem

$f(x,y) = g_1(x)\, g_2(y)\, \Omega(x,y)$

$\Omega(x,y)$ is a Copula:

$\int \Omega(x,y)\,\mathrm{d}x = 1$

$\int \Omega(x,y)\,\mathrm{d}y = 1$

Simple example

1	3	4
2	4	6
3	7	

?	?	4
?	?	6
3	7	

f_1	f_3	g_3
f_2	f_4	g_4
g_1	g_2	

1	-1	0
-1	1	0
0	0	

-1	1	0
1	-1	0
0	0	

$$\begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}$$

f_1	f_4	f_7	g_4
f_2	f_5	f_8	g_5
f_3	f_6	f_9	g_6
g_1	g_2	g_3	

- $\mathbf{H}\mathbf{f} = \mathbf{g} \longrightarrow \widehat{\mathbf{f}} = \mathbf{H}^{-1}\mathbf{g}$ if $\mathbf{H}$ invertible.
- $\mathbf{H}$ is rank deficient: $\text{rank}(\mathbf{H}) = 3$
- Problem has infinite number of solutions.
- How to find all those solutions ?
- Which one is the good one? Needs prior information.
- To find an unique solution, one needs either more data or prior information.

Application in CT: Reconstruction from 2 projections

$\mathbf{g}\vert\mathbf{f}$	$\mathbf{f}\vert\mathbf{z}$	$\mathbf{z}$	$\mathbf{c}$
$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$	iid Gaussian	iid	$q(\mathbf{r}) \in \{0, 1\}$
$\mathbf{g}\vert\mathbf{f} \sim \mathcal{N}(\mathbf{H}\mathbf{f}, \sigma_\epsilon^2 \mathbf{I})$	or	or	$1 - \delta(z(\mathbf{r}) - z(\mathbf{r}'))$
Gaussian	Gauss-Markov	Potts	binary

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta} | \mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta}_1)\, p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}_2)\, p(\mathbf{z}|\boldsymbol{\theta}_3)\, p(\boldsymbol{\theta})$$

Proposed algorithms

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta}_1)\, p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}_2)\, p(\mathbf{z}|\boldsymbol{\theta}_3)\, p(\boldsymbol{\theta})$$

- MCMC based general scheme:

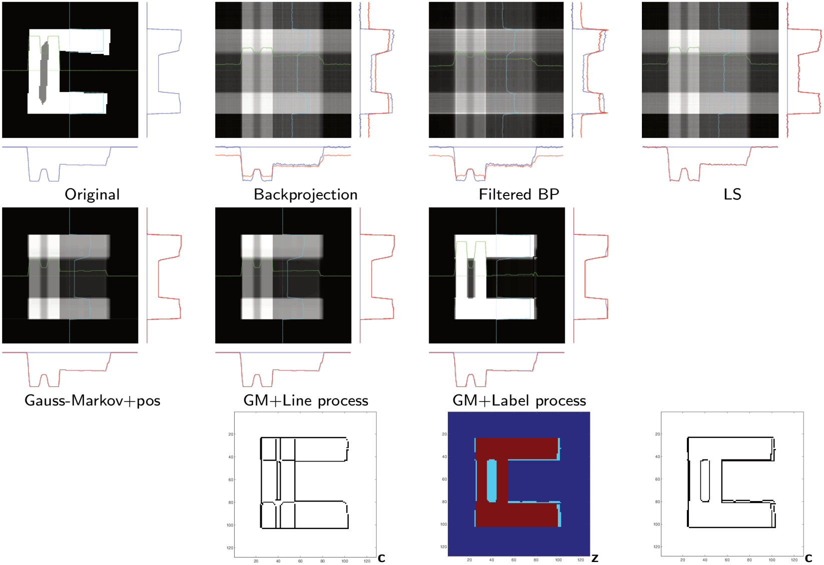
$$\widehat{\mathbf{f}} \sim p(\mathbf{f}|\widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \longrightarrow \widehat{\mathbf{z}} \sim p(\mathbf{z}|\widehat{\mathbf{f}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \longrightarrow \widehat{\boldsymbol{\theta}} \sim (\boldsymbol{\theta}|\widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \mathbf{g})$$

Iterative algorithme:

- Estimate $\mathbf{f}$ using $p(\mathbf{f}|\widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta})\, p(\mathbf{f}|\widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}})$
  Needs optimization of a quadratic criterion.
- Estimate $\mathbf{z}$ using $p(\mathbf{z}|\widehat{\mathbf{f}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \propto p(\mathbf{g}|\widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}})\, p(\mathbf{z})$
  Needs sampling of a Potts Markov field.
- Estimate $\boldsymbol{\theta}$ using
  $p(\boldsymbol{\theta}|\widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \mathbf{g}) \propto p(\mathbf{g}|\widehat{\mathbf{f}}, \sigma_\epsilon^2 \mathbf{I})\, p(\widehat{\mathbf{f}}|\widehat{\mathbf{z}}, (m_k, v_k))\, p(\boldsymbol{\theta})$
  Conjugate priors $\longrightarrow$ analytical expressions.

- Variational Bayesian Approximation
  - Approximate $p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g})$ by $q_1(\mathbf{f})\, q_2(\mathbf{z})\, q_3(\boldsymbol{\theta})$

Results



Original
Backprojection
Filtered BP
LS
Gauss-Markov+pos
GM+Line process
GM+Label process
c
z
c

Implementation issues

- In almost all the algorithms, the step of computation of $\widehat{\mathbf{f}}$ needs an optimization algorithm.
- The criterion to optimize is often in the form of

$$J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda \|\mathbf{D}\mathbf{f}\|^2$$

- Very often, we use the gradient based algorithms which need to compute

$$\nabla J(\mathbf{f}) = -2\mathbf{H}^t(\mathbf{g} - \mathbf{H}\mathbf{f}) + 2\lambda \mathbf{D}^t\mathbf{D}\mathbf{f}$$

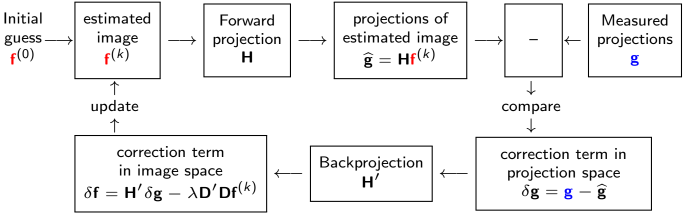
- So, for the simplest case, in each step, we have

$$\widehat{\mathbf{f}}^{(k+1)} = \widehat{\mathbf{f}}^{(k)} + \alpha^{(k)} \left[\mathbf{H}^t(\mathbf{g} - \mathbf{H}\widehat{\mathbf{f}}^{(k)}) + 2\lambda \mathbf{D}^t\mathbf{D}\widehat{\mathbf{f}}^{(k)}\right]$$

Gradient based algorithms

$$\widehat{\mathbf{f}}^{(k+1)} = \widehat{\mathbf{f}}^{(k)} + \alpha \left[\mathbf{H}' \left(\mathbf{g} - \mathbf{H}\widehat{\mathbf{f}}^{(k)} \right) - \lambda \mathbf{D}'\mathbf{D}\widehat{\mathbf{f}}^{(k)} \right]$$

1. Compute $\widehat{\mathbf{g}} = \mathbf{H}\widehat{\mathbf{f}}$ (Forward projection)
2. Compute $\delta\mathbf{g} = \mathbf{g} - \widehat{\mathbf{g}}$ (Error or residual)
3. Compute $\delta\mathbf{f}_1 = \mathbf{H}'\delta\mathbf{g}$ (Backprojection of error)
4. Compute $\delta\mathbf{f}_2 = -\mathbf{D}'\mathbf{D}\widehat{\mathbf{f}}$ (Correction due to regularization)
5. Update $\widehat{\mathbf{f}}^{(k+1)} = \widehat{\mathbf{f}}^{(k)} + [\delta\mathbf{f}_1 + \delta\mathbf{f}_2]$



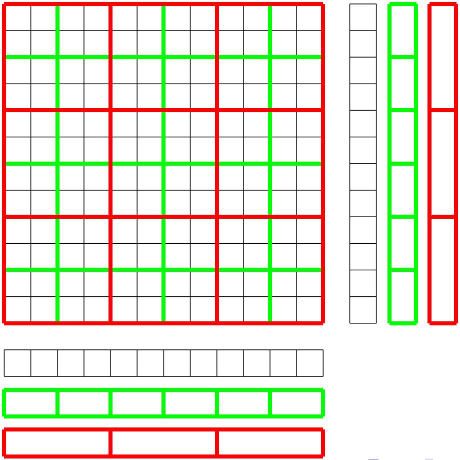
- Steps 1 and 3 need great computational cost and have been implemented on GPU.

Multi-Resolution Implementation

Sacle 1: black $\mathbf{g}^{(1)} = \mathbf{H}^{(1)}\mathbf{f}^{(1)}$ $(N \times N)$

Sacle 2: green $\mathbf{g}^{(2)} = \mathbf{H}^{(2)}\mathbf{f}^{(2)}$ $(N/2 \times N/2)$

Sacle 3: red $\mathbf{g}^{(3)} = \mathbf{H}^{(3)}\mathbf{f}^{(3)}$ $(N/4 \times N/4)$



Conclusions

- Computed Tomography is an Inverse problem
- Analytical methods have many limitations
- Algebraic methods push further these limitations
- Deterministic Regularization methods push still further the limitations of ill-conditioning.
- Probabilistic and in particular the Bayesian approach has many potentials
- Hierarchical prior model with hidden variables are very powerful tools for Bayesian approach to inverse problems.
- Gauss-Markov-Potts models for images incorporating hidden regions and contours
- Main Bayesian computation tools: JMAP, MCMC and VBA
- Application in different imaging system (X ray CT, Microwaves, PET, Ultrasound, Optical Diffusion Tomography (ODT), Acoustic source localization,...)
- Current Projects: Efficient implementation in 2D and 3D cases