Bayesian Inversion

Bangti Jin

Course “Inverse Problems in Imaging”

Outline

Motivation

- efficient algorithms for finding a Tikhonov minimizer
- Question: How plausible is the Tikhonov minimizer ?
- $\Rightarrow$ tools for assessing the reliability of the inverse solution

Bayesian inference is one principled framework for uncertainty quantification.

starting point: Bayes' formula, i.e., for two random variables X and Y the conditional probability of X given Y is given by

$$p_{X|Y}(x|y) = \frac{p_{Y|X}(y|x)p_X(x)}{p_Y(y)},$$

- $p_{Y|X}(y|x)$: likelihood function
- $p_X(x)$: prior distribution

finite-dimensional inverse problem

$$F(X) = Y,$$

- X, Y: the unknown coefficient and the noisy data
- $F : \mathbb{R}^m \mapsto \mathbb{R}^n$: forward map
- regard the unknown X and the data Y as random variables, and encode the prior knowledge in a probability distribution.

e.g. given $X = x$, Y follows a Gaussian distribution with mean $F(x)$ and variance $\sigma^2 I$, then

$$p_{Y|X}(y|x) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{\|F(x)-y\|^2}{2\sigma^2}}.$$

the unnormalized posteriori $p(x, y)$ defined by

$$p(x, y) = p_{Y|X}(y|x)p_X(x),$$

and shall often write

$$p_{X|Y}(x|y) \propto p(x, y)$$

the posteriori $p_{X|Y}(x|y)$ up to a multiplicative constant

$p_{X|Y}(x|y)$ **holds the full information about the inverse problem**

$\Rightarrow$ **calibrating the uncertainties of the inverse solutions.**

two building blocks

- likelihood function $p_{Y|X}(y|x)$
  contains the information in the data y, or more precisely the statistics of the noise in the data y
- prior distribution $p_X(x)$
  encodes a prior knowledge available about the problem before collecting the data.

likelihood function $p_{Y|X}(y|x) \Leftarrow$ the noise statistics

- all sources of errors (e.g., these for the forward model F) are lumped into the data y.
- a careful modeling and account of all errors in the data y is essential for extracting useful information

The most popular noise model is the additive Gaussian model

$$y = y^\dagger + \xi,$$

- $\xi \in \mathbb{R}^n$ is a realization of i.i.d. Gaussian r.v. $N(0, \sigma^2)$
- ξ is independent of the true data $y^\dagger$ (and hence x) $\Rightarrow$

$$p_{Y|X}(y|x) = (2\pi\sigma^2)^{-\frac{n}{2}} e^{-\frac{1}{2\sigma^2}\|F(x)-y\|^2}.$$

The prior $p_X(x)$ encodes the prior knowledge about the sought-for solution x in a probabilistic manner.

- the prior knowledge: expert opinion, historical investigations, statistical studies and anatomical knowledge etc.
- Since inverse problems are ill-posed due to lack of information, the careful incorporation of all available prior knowledge is of utmost importance in any inversion technique
- the prior plays the role of regularization in a stochastic setting
- Hence, prior modeling stays at the heart of Bayesian model construction, and crucially affects the interpretation of the data.

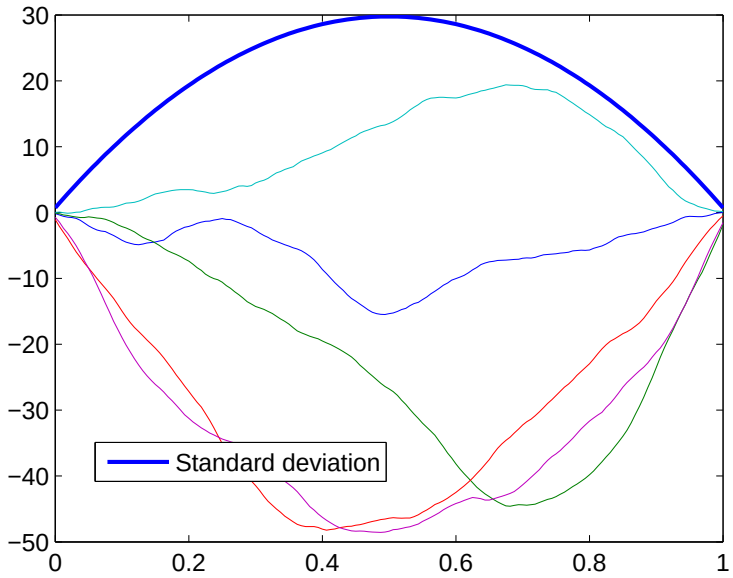
- One very versatile prior model is Markov random field

$$p_X(x) \propto e^{-\lambda\psi(x)},$$

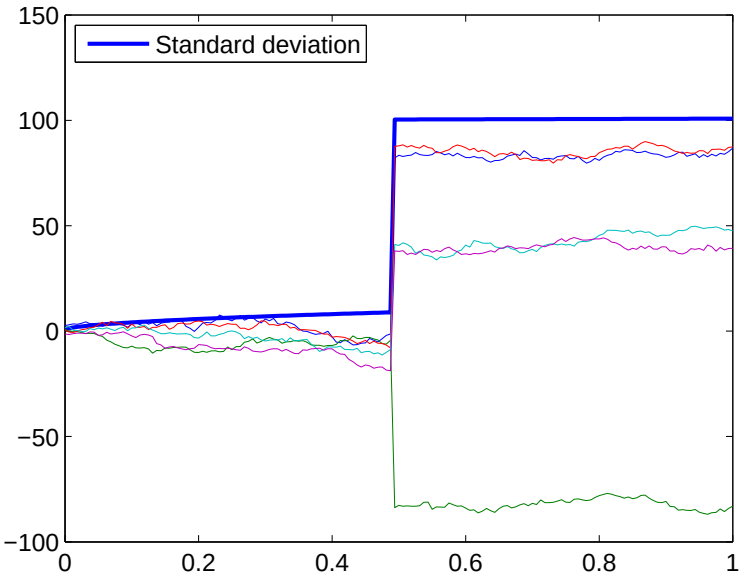
where $\psi(x)$ is a potential function dictating the interaction energy between the components of the random field x

- The scalar λ is a scale parameter, determining the strength of the local/global interactions.

  It plays the role of a regularization parameter in classical regularization theory, and hence its automated determination is very important.



(a) smoothness



(b) total variation

likelihood $p_{Y|X}(y|x)$ and the prior $p_X(x)$ may contain unknown parameters, e.g.,

$$p_{Y|X}(y|x) = p_{Y|X,\Upsilon}(y|x,\tau) \quad \text{and} \quad p_X(x) = p_{X|\Lambda}(x|\lambda)$$

- τ,λ: precision (inverse variance) and the scale parameter
- These parameters are generically known as hyperparameters
- Hierarchical Bayesian modeling provides an elegant approach to choose these parameters automatically

hierarchical Bayesian modeling

- view λ and τ as random variables with their own priors
- determine them from the data y
- convenient choice: conjugate distribution

For both λ and τ, the conjugate distribution is given by a Gamma distribution:

$$p_\Lambda(\lambda) = G(\lambda; a_0, b_0) = \frac{b_0^{a_0}}{\Gamma(a_0)} \lambda^{a_0 - 1} e^{-b_0 \lambda},$$

$$p_\Upsilon(\tau) = G(\tau; a_1, b_1) = \frac{b_1^{a_1}}{\Gamma(a_1)} \tau^{a_1 - 1} e^{-b_1 \tau}.$$

- the parameter pairs (a_0, b_0) and (a_1, b_1) determines the range of the prior knowledge on the parameters λ and τ
- noninformative prior is often adopted, which roughly amounts to setting a_0 to 1 and b_0 close to zero

posterior distribution $p_{X,\Lambda,\Upsilon|Y}(x,\lambda,\tau|y)$

$$p_{X,\Lambda,\Upsilon|Y}(x,\lambda,\tau|y) \propto p_{Y|X,\Upsilon}(y|x,\tau)p_{X|\Lambda}(x|\lambda)p_{\Lambda}(\lambda)p_{\Upsilon}(\tau).$$

Connection with Tikhonov regularization
example: Gaussian noise model + Laplace prior

$$p_{Y|X,\Upsilon}(y|x,\tau) \propto \tau^{-\frac{n}{2}} e^{-\frac{\tau}{2}\|F(x)-y\|^2},$$
$$p_{X|\Lambda}(x|\lambda) \propto \lambda^m e^{-\lambda\|x\|_1}.$$

In case of known λ and τ, a popular rule of thumb is to consider the maximum a posteriori (MAP) estimate x_{map}, i.e.,

$$\begin{aligned} x_{\text{map}} &= \arg\max_x p_{X,\Lambda,\Upsilon|Y}(x,\lambda,\tau|y) \\ &= \arg\min_x \left\{ \tfrac{\tau}{2}\|F(x)-y\|^2 + \lambda\|x\|_1 \right\}. \end{aligned}$$

the functional in the curly bracket is

$$\tfrac{1}{2}\|F(x)-y\|^2 + \lambda\tau^{-1}\|x\|_1,$$

Tikhonov regularization + sparsity constraint, with $\alpha = \lambda\tau^{-1}$.

A Tikhonov minimizer is an MAP estimate of some Bayesian formulation.

unknown parameters λ and $\tau \Rightarrow$ hierarchical model
conjugate prior on λ and $\tau \Rightarrow$ posterior distribution

$$p_{X,\Lambda,\Upsilon|Y}(x,\lambda,\tau|y) \propto \tau^{\frac{n}{2}+a_1-1} e^{-\frac{\tau}{2}\|F(x)-y\|^2} \cdot \lambda^{m+a_0-1} e^{-\lambda\|x\|_1} \cdot e^{-b_1\tau} \cdot e^{b_0\lambda}.$$

ways of handling the posterior distribution $p_{X,\Lambda,\Upsilon|Y}(x,\lambda,\tau|y)$

- the joint maximum a posteriori estimate $(x,\lambda,\tau)_{\text{map}}$, i.e.,

$$(x,\lambda,\tau)_{\text{map}} = \arg\min_{x,\lambda,\tau} J(x,\lambda,\tau),$$

where the functional $J(x,\lambda,\tau)$ is given by

$$J(x,\lambda,\tau) = \tfrac{\tau}{2}\|F(x)-y\|^2 + \lambda\|x\|_1 - \tilde{a}_0 \ln\lambda + b_0\lambda - \tilde{a}_1 \ln\tau + b_1\tau.$$

the augmented Tikhonov regularization for sparsity constraint

augmented Tikhonov regularization

$$J(x, \lambda, \tau) = \tfrac{\tau}{2}\|F(x) - y\|^2 + \lambda\|x\|_1 - \tilde{a}_0 \ln \lambda + b_0\lambda - \tilde{a}_1 \ln \tau + b_1\tau.$$

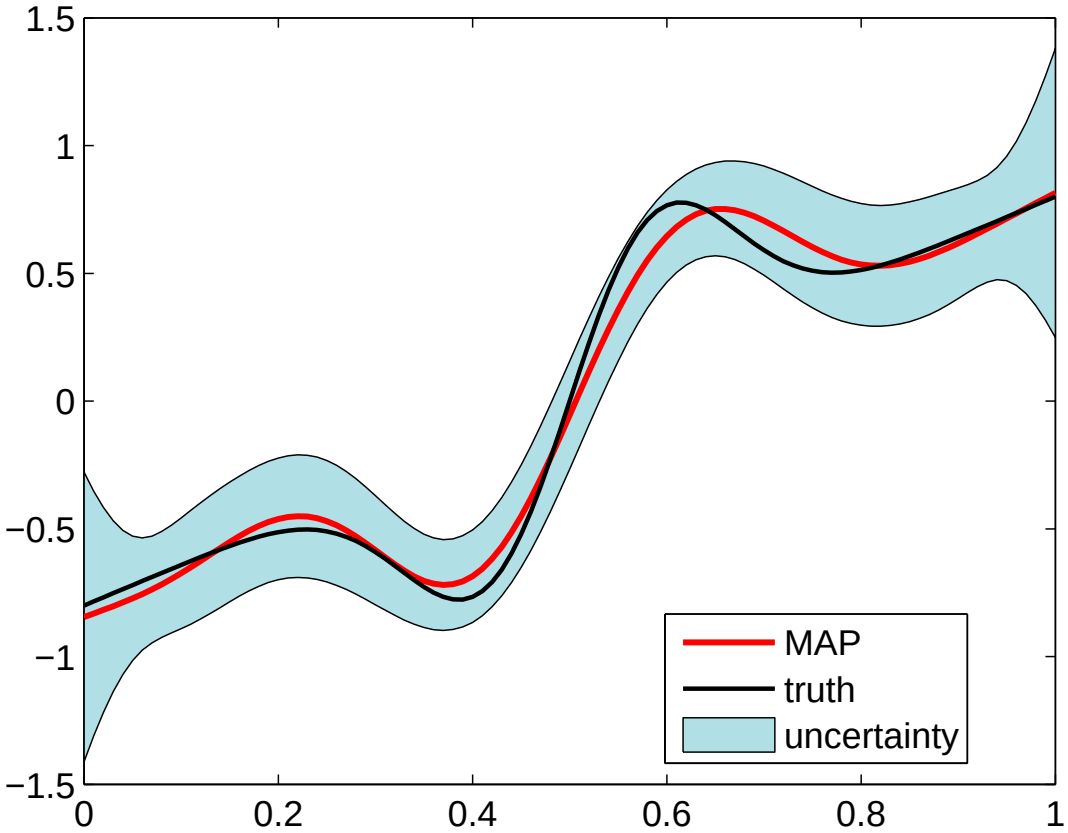
- the first two terms recover Tikhonov regularization
- the rest provides the mechanism for automatically determining the regularization parameter.
- the augmented approach does select the hyperparameters λ and τ automatically, but it remains a point estimate and ignores the statistical fluctuations around the mode.
  $\Rightarrow$ full Bayesian treatment

Bayesian solution: $p_{X,\Lambda,\Upsilon|Y}(x,\lambda,\tau|y)$
distinct features

- $p_{X,\Lambda,\Upsilon|Y}(x,\lambda,\tau|y)$ is a probability distribution, and encompasses an ensemble of plausible solutions that are consistent with the given data y (to various extent).

$$\mu = \int x p_{X|Y}(x|y)dx,$$

$$C = \int (x-\mu)(x-\mu)^{\mathrm{t}} p_{X|Y}(x|y)dx.$$


1.5
1
0.5
0
−0.5
−1
−1.5
0
0.2
0.4
0.6
0.8
1
MAP
truth
uncertainty

distinct features

- the crucial role of proper statistical modeling in designing useful regularization formulations for practical problems.
- it provides a flexible regularization since hierarchical modeling can partially resolve the nontrivial issue of choosing an appropriate regularization parameter.

posteriori $p(x)$ lives in a very high-dimensional space $\Rightarrow$ noninformative
$\Rightarrow$ compute summarizing statistics, e.g., mean μ and covariance C

$$\mu = \int x p(x) dx \quad \text{and} \quad C = \int (x-\mu)(x-\mu)^{\mathrm{t}} p(x) dx.$$

very high-dimensional integrals, and quadrature rules are inefficient
e.g., $m = 100$, 2 points/dir $\Rightarrow 2^{100} \approx 1.27 \times 10^{30}$ points
more efficient approach

- Monte Carlo methods, especially Markov chain Monte Carlo

Monte Carlo simulation

- draw a large set of i.i.d. samples $\{x^{(i)}\}_{i=1}^{N}$ from the target distribution $p(x)$
- approximate the expectation $E_p[f]$ of any function $f : \mathbb{R}^m \to \mathbb{R}$ by the sample mean $E_N[f]$

$$E_N[f] \equiv \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)}) \to E_p[f] = \int f(x)p(x)dx \quad \text{as } N \to \infty.$$

- the Monte Carlo integration error $e_N[f]$ by

$$e_N[f] = E_p[f] - E_N[f] \approx \mathrm{Var}_p[f]^{\frac{1}{2}} N^{-1/2}\nu,$$

$\nu \sim N(0,1)$

  - the error $e_N[f]$ is $O(N^{-1/2})$
  - with a constant $\sim$ the variance of the integrand f
  - the estimate is independent of the dimensionality m

Generating a large set of i.i.d. samples from an implicit and high-dimensional joint distribution is highly nontrivial.

- nonlinear inverse problems and nongaussian models
- importance sampling

$q(x)$ is an easy-to-sample p.d.f. and close to the posteriori $p(x)$
approximate the expectation of the function f w.r.t. $p(x)$ by

$$\int f(x)p(x)dx = \int f(x)\frac{p(x)}{q(x)}q(x)dx \approx \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)})w_i,$$

where the i.i.d. samples $\{x^{(i)}\}_{i=1}^{N}$ are drawn from the auxiliary distribution $q(x)$, and the weights $w_i = \frac{p(x^{(i)})}{q(x^{(i)})}$.
The efficiency relies on the quality of the approximation $q(x)$ to the true posterior distribution $p(x)$

example: nonlinear forward model $F(x) = y$, with a Gaussian noise model and a smoothness prior, i.e.,

$$p(x) \propto e^{-\frac{\tau}{2}\|F(x)-y\|^2 - \frac{\lambda}{2}\|Lx\|^2},$$

A natural candidate model $q(x)$ is a Gaussian approximation around the mode x^*. One approach is to linearize the forward model $F(x)$ around the mode x^*:

$$F(x) = F(x^*) + F'(x^*)(x - x^*) + \text{h.o.t.},$$

which gives the following Gaussian approximation

$$q(x) \propto e^{-\frac{\tau}{2}\|F'(x^*)(x-x^*)-(y-F(x^*))\|^2 - \frac{\lambda}{2}\|Lx\|^2}.$$

A more refined approach: the full Hessian

$$\begin{aligned}\|F(x) - y\|^2 \approx \|F(x^*) - y\|^2 &+ 2\langle F'(x^*)^*(F(x^*) - y), x - x^*\rangle \\ &+ \langle F'(x^*)(x - x^*), F'(x^*)(x - x^*)\rangle \\ &+ \langle F''(x^*)(F(x^*) - y)(x - x^*), x - x^*\rangle.\end{aligned}$$

Markov chain Monte Carlo: general-purposed approach for exploring posteriori $p(x)$

- basic idea: given a target distribution $p(x)$, construct an aperiodic and irreducible Markov chain such that its stationary distribution is $p(x)$.
- By running the chain for sufficiently long, simulated values from the chain can be regarded as dependent samples from the target distribution $p(x)$, and used for computing summarizing statistics.
- Metropolis: simulating energy levels of atoms in a crystalline structure
- Hastings: statistical problems

The Metropolis-Hastings algorithm is the most basic MCMC method

1: Initialize $x^{(0)}$ and set N;
2: **for** $i = 0 : N$ **do**
3: sample $u \sim U(0, 1)$;
4: sample $x^{(*)} \sim q(x^{(i)}, x^{(*)})$
5: **if** $u < \alpha(x^{(i)}, x^{(*)})$, c.f., (**??**) **then**
6: $x^{(i+1)} = x^{(*)}$;
7: **else**
8: $x^{(i+1)} = x^{(i)}$;
9: **end if**
10: **end for**

- the uniform distribution $U(0, 1)$
- $p(x)$: the target distribution
- $q(x, x')$ is an easy-to-sample proposal distribution

Having generated a new state x' from the distribution $q(x, x')$, we then accept this point as the new state of the chain with probability $\alpha(x, x')$ given by

$$\alpha(x, x') = \min\left\{1, \frac{p(x')q(x', x)}{p(x)q(x, x')}\right\}.$$

However, if we reject x', then the chain remains in the current state x.

- $p(x)$ enters the algorithm only through α via the ratio $p(x')/p(x)$, so a knowledge of the distribution only up to a multiplicative constant is sufficient for implementation
- if q is symmetric, i.e., $q(x, x') = q(x', x)$, $\alpha(x, x')$ reduces to

$$\alpha(x, x') = \min\left\{1, \frac{p(x')}{p(x)}\right\}.$$

The Metropolis-Hastings algorithm guarantees that the Markov chain converges to the target distribution $p(x)$ for any reasonable proposal distribution $q(x)$. There are many possible choices for the proposal

random walker sampler

- If $q(x, x') = f(x' - x)$ for p.d.f. f, then $x^{(*)} = x^{(i)} + \xi$, $\xi \sim f$
- Markov chain is driven by a random walk
- f: uniform, multivariate normal or t-distribution
- With i.i.d. Gaussian distribution $N(0, \sigma^2)$, $x_j^{(*)} = x_j^{(i)} + \xi$, with $\xi \sim N(\xi; 0, \sigma^2)$. The variance σ^2 of the proposal distribution f controls the size of the random walks, and should be carefully tuned to improve the MCMC convergence and estimation efficiency.

it is necessary to tune σ^2 carefully to achieve good mixing. Heuristically, the optimal acceptance ratio should be around 0.25 for some model problems.

independent sampler $q(x, x') = q(x')$

- the acceptance probability $\alpha(x, x')$

$$\alpha(x, x') = \min\{1, w(x')/w(x)\},$$

$w(x) = p(x)/q(x)$ is the importance weight function.

- There are many different ways to generate the independent proposal distribution $q(x)$, e.g., Gaussian approximations from the linearized forward model, coarse-scale/reduced-order representation

- the first samples are poor approximations as samples from $p(x)$
- discards these initial samples (burning-in period)
- assess the convergence of the MCMC chains

Brooks and Gelman statistics 1998: Suppose we have L Markov chains, each of N samples, with the ith sample from the jth chain denoted by $x_j^{(i)}$. Then we compute

$$\widehat{V} = \frac{N-1}{N} W + \left(1 + \frac{1}{L}\right) \frac{B}{N},$$

$$W = \frac{1}{L(N-1)} \sum_{j=1}^{L} \sum_{i=1}^{N} (x_j^{(i)} - \bar{x}_j)(x_j^{(i)} - \bar{x}_j)^{\mathrm{t}},$$

$$\frac{B}{N} = \frac{1}{L-1} \sum_{j=1}^{L} (\bar{x}_j - \bar{x})(\bar{x}_j - \bar{x})^{\mathrm{t}},$$

which represent respectively the within and between-sequence [illegible] distance

- If the state space is high dimensional, it is rather difficult to update the entire vector x in one single step since the acceptance probability $\alpha(x, x')$ is often very small.
- to update a part of the components of x each time and to implement an updating cycle inside each step
- block Gauss-Seidel iteration in numerical linear algebra
- The extreme case is the Gibbs sampler Geman-Geman, 1984

which updates a single component each time.
suppose we want to update the ith component x_i of x, then we choose the full conditional as the proposal distribution $q(x, x')$, i.e.,

$$q(x, x') = \begin{cases} p(x_i'|x_{-i}) & x'_{-i} = x_{-i}, \\ 0 & \text{otherwise}, \end{cases}$$

where x_{-i} denotes $(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_m)^{\mathrm{t}}$.

With this proposal, the acceptance probability $\alpha(x, x')$ is given by

$$\begin{aligned}\alpha(x, x') &= \frac{p(x')q(x', x)}{p(x)q(x, x')} = \frac{p(x')/p(x_i'|x_{-i})}{p(x)/p(x_i|x'_{-i})} \\ &= \frac{p(x')/p(x_i'|x'_{-i})}{p(x)/p(x_i|x_{-i})} = \frac{p(x'_{-i})}{p(x_{-i})} = 1,\end{aligned}$$

these proposals are automatically accepted.

Gibbs algorithm

1: Initialize $x^{(0)}$ and set N.
2: **for** $i = 0 : N$ **do**
3: sample $x_1^{(i+1)} \sim p(x_1 | x_2^{(i)}, x_3^{(i)}, \ldots, x_m^{(i)})$,
4: sample $x_2^{(i+1)} \sim p(x_2 | x_1^{(i+1)}, x_3^{(i)}, \ldots, x_m^{(i)})$,
5: $\vdots$
6: sample $x_m^{(i+1)} \sim p(x_m | x_1^{(i+1)}, x_2^{(i+1)}, \ldots, x_{m-1}^{(i+1)})$,
7: **end for**

example: Gibbs sampler for Gaussian noise + smoothness prior $p(\lambda) \propto \lambda^{a_0-1} e^{-b_0\lambda}$ on the scale parameter λ, i.e., posteriori

$$p(x,\lambda) \propto e^{-\frac{\tau}{2}\|Ax-y\|^2} \cdot \lambda^{\frac{m}{2}} e^{-\frac{\lambda}{2}x^t W x} \lambda^{a_0-1} e^{-b_0\lambda},$$

where the matrix W encodes the local interaction structure
full conditional $p(x_i|x_{-i},\lambda)$

$$p(x_i|x_{-i},\lambda) \sim N(\mu_i,\sigma_i^2), \quad \mu_i = \frac{b_i}{2a_i}, \quad \sigma_i = \frac{1}{\sqrt{a_i}},$$

with a_i and b_i given by

$$a_i = \tau \sum_{j=1}^{n} A_{ji}^2 + \lambda W_{ii} \quad \text{and} \quad b_i = 2\tau \sum_{j=1}^{n} \mu_j A_{ji} - \lambda \mu_p,$$

and $\mu_j = y_j - \sum_{k\neq i} A_{jk}x_k$ and $\mu_p = \sum_{j\neq i} W_{ji}x_j + \sum_{k\neq i} W_{ik}x_k$. Lastly, we deduce the full conditional for λ:

$$p(\lambda|x) \sim G\left(\lambda; \tfrac{m}{2} + a_0, \tfrac{1}{2}x^t W x + \beta_0\right).$$