ICES REPORT 12-18

May 2012

A Gentle Tutorial on Statistical Inversion using the Bayesian Paradigm

by

Tan Bui-Thanh

The Institute for Computational Engineering and Sciences
The University of Texas at Austin
Austin, Texas 78712

Reference: Tan Bui-Thanh, A Gentle Tutorial on Statistical Inversion using the Bayesian Paradigm, ICES REPORT 12-18, The Institute for Computational Engineering and Sciences, The University of Texas at Austin, May 2012.

A Gentle Tutorial on Statistical Inversion using the Bayesian Paradigm

Tan Bui-Thanh[*]

Institute for Computational Engineering and Sciences, The University of Texas at Austin

Contents

1 Introduction

In this note, we are interested in solving inverse problems using statistical techniques. Let us motivate you by considering the following particular inverse problem, namely, the deconvolution problem. Given the observation

[*]tanbui@ices.utexas.edu

signal $g(s)$, we would like to reconstruct the input signal $f(t) : [0,1] \to \mathbb{R}$, where the observation and the input obey the following relation

$$g(s_j) = \int_0^1 a(s_j,t) f(t)\, dt, \quad 0 \le j \le n. \tag{1}$$

Here, $a : [0,1] \times [0,1] \to \mathbb{R}$ is known as the blurring kernel. So, in fact we don't know the output signal completely, but at a finite number of observation points. A straightforward approach you may think of is to apply some numerical quadrature on the right side of (1), and then recover $f(t)$ at the quadrature points by inverting the resulting matrix. If you do this, you realize that the matrix is ill-conditioned, and it is not a good idea to invert it. There are techniques to go around this issue, but let us not pursue them here. Instead, we recast the deconvolution task into an optimization problem such as

$$\min_{f(t)} \sum_{j=0}^{n} \left(g(s_j) - \int_0^1 a(s_j,t) f(t)\, dt \right)^2. \tag{2}$$

The ill-conditioning nature of our inverse problem does not go away, but it manifests itself as non-convexity in the cost function (also known as the data misfit). In particular, the data misfit is not a parabola! It is the non-convexity that makes the optimization task difficult. So what is the point of recast? Well, viewing the difficulty of the inverse problem under consideration as the non-convexity of the cost function immediately suggests that we should convexify it. A simple idea to accomplish this is to add a quadratic to the cost function to make it more like a parabola, and hence making the optimization problem easier. This is essentially the idea behind the *Tikhonov regularization*, which proposes to solve the nearby problem

$$\min_{f(t)} \sum_{j=0}^{n} \left(g(s_j) - \int_0^1 a(s_j,t) f(t)\, dt \right)^2 + \frac{\kappa}{2} \left\| R^{1/2} f \right\|^2,$$

where κ is known as the regularization parameter, and $\|\cdot\|$ is some appropriate norm. Perhaps, two popular choices for $R^{1/2}$ are ∇ and Δ, the gradient and Laplace operator, respectively, and we discuss them in details in the following.

Now, in practice, we are typically not able to observe $g(s_j)$ directly but its noise-corrupted value

$$g^{obs}(s_j) = g(s_j) + e_j, \quad 0 \le j \le n,$$

where e_j, $j = 0, \ldots, n$, are some random noise. You can think of the noise as

the inaccuracy in observation/measurement devices. The question you may ask is how to incorporate this kind of randomness in the above deterministic solution methods. There are works in this direction, but let us introduce a statistical framework based on the Bayesian paradigm to you in this note. This approach is appealing since it can incorporate most, if not all, kinds of randomness in a systematic manner.

A large portion of this note follows closely the presentation of two excellent books by Somersalo *et al.* [1, 2]. The pace is necessary slow since we develop this note for readers with minimal knowledge in probability theory. The only requirement is to either be familiar with or adopt the conditional probability formula concept. This is the corner stone on which we build the rest of the theory. Clearly, the theory we present here is by no means complete since the subject is vast, and still under development.

Our presentation is in the form of dialogue, which we hope it is easier for the readers to follow. We shall give a lot of little exercises along the way to help understand the subject better. We also leave a large number of side notes, mostly in term of little questions, to keep the readers awake and make connections of different parts of the notes. On the other hand, we often discuss deeper probability concepts in the footnotes to serve as starting points for those who want to dive into the rigorous probability theory. Finally, we supply Matlab codes at the end of the note so that the readers can use them to reproduce most of the results and to start their journey into the wonderful world of Bayesian inversion.

2 Some concepts from probability theory

We begin with the definition of randomness.

2.1 definition. An even is *deterministic* if its outcome is completely predictable.

2.2 definition. A *random event* is the complement of a deterministic event, that is, its outcome is not fully predictable.

2.3 example. If today is Wednesday, then "tomorrow is Thursday" is deterministic, but whether it rains tomorrow is not fully predictable.

As a result, randomness means lack of information and it is the direct consequence of our ignorance. To express our belief[1] on random events, we use probability; probability of uncertain events is always less than 1, an event that surely happens has probability 1, and an event that never happens has

[1] Different person has different belief which leads to different solution of the Bayesian inference problem. Specifically, one's belief is based on his known information (expressed in terms of σ-algebra) and "weights" on each information (expressed in terms of probability measure). That is, people working with different probability spaces have different solutions.

o probability. In particular, to reflect the subjective nature, we call it *subjective probability* or *Bayesian probability* since it represents belief, and hence depending upon one's experience/knowledge to decide what is reasonable to believe.

2.4 example. Let us consider the event of tossing a coin. Clearly, this is a random event since we don't know whether head or tail will appear. Nevertheless, we believe that out of n tossing times, $n/2$ times is head and $n/2$ times is tail.[2] We express this belief in terms of probability as: the (subjective) probability of getting a head is $\frac{1}{2}$ and the (subjective) probability of getting a tail is $\frac{1}{2}$.

We define $(\Omega, \mathcal{F}, \mathbb{P})$ as a *probability space*. One typically call Ω the *sample space*, $\mathcal{F}$ a σ-algebra containing all events $A \subset \Omega$, and $\mathbb{P}$ a probability measure defined on $\mathcal{F}$. We can think of an event A as information and the probability that A happens, i.e. $\mathbb{P}[A]$, is the weight assigned to that information. We require that

$$0 \leq \mathbb{P}[A] \overset{\text{def}}{=} \int_A d\omega \leq 1, \quad \mathbb{P}[\emptyset] = 0, \quad \mathbb{P}[\Omega] = 1.$$

2.5 example. Back to the tossing coin example, we trivially have $\Omega = \{head, tail\}$, $\mathcal{F} = \{\emptyset, \{head\}, \{tail\}, \Omega\}$. The weights are $\mathbb{P}[\emptyset] = 0$, $\mathbb{P}[\{tail\}] = \mathbb{P}[\{head\}] = \frac{1}{2}$, and $\mathbb{P}[\{head, tail\}] = 1$.

Two events A and B are independent[3] if

$$\mathbb{P}[A \cap B] = \mathbb{P}[A] \times \mathbb{P}[B].$$

One of the central ideas in Bayesian probability is the *conditional probability*[4]. The conditional probability of A on/given B is defined as[5]

$$\mathbb{P}[A|B] = \frac{\mathbb{P}[A \cap B]}{\mathbb{P}[B]}, \tag{3}$$

This is the corner stone formula to build most of results in this note, make sure that you feel comfortable with it.

which can also be rephrased as the probability that A happens *provided* B has already happened.

2.6 example. Assume that we want to roll a dice. Denote B as the event of getting of face bigger than 4, and A the event of getting face 6. Using (3) we

[2] One can believe that out of n tossing times, $n/3$ times is head and $2n/3$ times is tail if he uses an *unfair* coin.

[3] Probability theory is often believed to be a part of measure theory, but independence is where it departs from the measure theory umbrella.

[4] A more general and rigorous tool is conditional expectation, a particular of which is conditional probability.

[5] This was initially introduced by Kolmogorov, a father of modern probability theory.

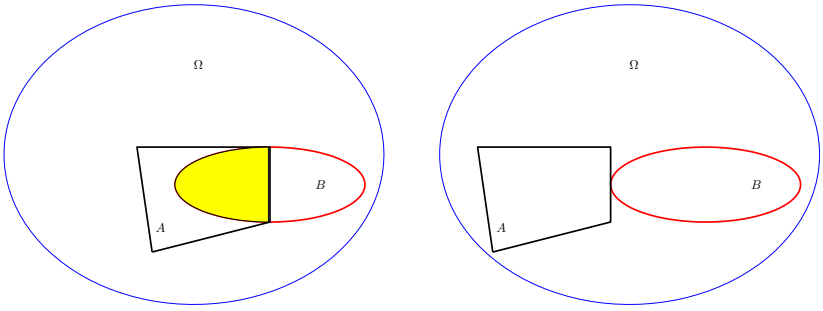
have

$$\mathbb{P}[A|B] = \frac{1/6}{1/3} = 1/2.$$

We can solve the problem using a more elementary argument. B happens when we either get face 5 or face 6. The probability of getting face 6 when B has already happened is clearly $\frac{1}{2}$.

The conditional probability can also be understood as the probability when the sample space is restricted to B.

2.7 exercise. Determine $\mathbb{P}[A|B]$ in Figure 1.



(a) $\mathbb{P}[A|B] =$ (b) $\mathbb{P}[A|B] =$

Figure 1: Demonstration of conditional probability.

2.8 exercise. Show that the following Bayes formula for conditional probability holds

$$\mathbb{P}[A|B] = \frac{\mathbb{P}[B|A]\mathbb{P}[A]}{\mathbb{P}[B]}. \tag{4}$$

By inspection, if A and B are mutually independent, we have

$$\mathbb{P}[A|B] = \mathbb{P}[A], \quad \mathbb{P}[B|A] = \mathbb{P}[B].$$

The probability space $(\Omega, \mathcal{F}, \mathbb{P})$ is an abstract object which is useful for theoretical developments, but far from practical considerations. In practice, it is usually circumvented by probability densities over the *state space*, which are easier to handle and have certain physical meanings. We shall come back to this point in a moment.

2.9 definition. The state space S is the set containing all the possible outcomes.

In this note, the state space S (and also T) is the standard Euclidean space $\mathbb{R}^n$, where n is the dimension. We are in position to introduce the key player, the *random variable*.

2.10 definition. A random variable M is a map[6] from the sample space Ω to the state space S

$$M : \Omega \ni \omega \mapsto M(\omega) \in S.$$

We call $M(\omega)$ a random variable since we are uncertain about its outcome. In other words, we admit our ignorance about M by calling it a random variable. This ignorance is in turn a direct consequence of the uncertainty in the outcome of elementary event ω.

The usual convention is to use lower case letter $m = M(\omega)$ as an arbitrary realization of the (upper case letter) random variable M, and we utilize this convention throughout the note.

2.11 definition. The *probability distribution* (or distribution or law for short) of a random variable M is defined as

$$\mu_M(A) = \mathbb{P}\left[M^{-1}(A)\right] = \mathbb{P}\left[\{M \in A\}\right], \quad \forall A \in S, \tag{5}$$

where we have used the popular notation[7]

$$M^{-1}(A) \overset{\text{def}}{=} \{M \in A\} \overset{\text{def}}{=} \{\omega \in \Omega : M(\omega) \in A\}.$$

From the definition, we can see that the distribution is a probability measure[8] on S. In other words, the random variable M induces a probability measure, defined as μ_M, on the state space S. The key property of the induced probability measure μ_M is the following. The probability for an event A in the state space to happen, denoted as $\mu_M(A)$, is defined as the probability for an event $B = M^{-1}(A)$ in the sample space to happen (see Figure 2 for an illustration). The distribution and the *probability density*[9] π_X of M obey the

[6] Measurable map is the rigorous definition, but we avoid technicalities here since it involves operations on σ-algebra.

[7] Rigorously, A must be a measurable subset of S.

[8] In fact, it is the push-forward measure by the random variable M.

[9] Here, the density is understood with respect to the Lebesgue measure on $S = \mathbb{R}^n$. Rigorously, π_M is the Radon-Nikodym derivative of μ_M with respect to the Lebesgue measure. As a result, π_M should be understood as equivalent class on $L^1(S)$.

following relation

$$\mu_M(A) \overset{\text{def}}{=} \int_A \pi_M(m)\, dm \overset{\text{def}}{=} \int_{\{M\in A\}} d\omega, \quad \forall A \subset S \tag{6}$$

where the second equality of the definition is from (5). The meaning of random variable $M(\omega)$ can now be seen in Figure 2. It maps the event $B \in \Omega$ into the set $A = M(B)$ in the state space such that the area under the density function $\pi_M(m)$ and above A is exactly the probability that B happens. *Do you see this?*

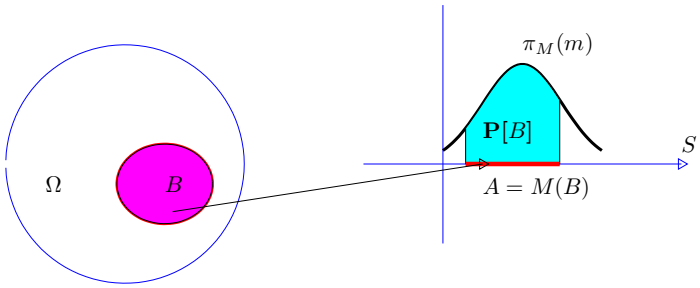


Figure 2: Demonstration of random variable: $\mu_M(A) = \mathbb{P}[B]$.

We deduce the change of variable formula

$$\mu_M(dm) = \pi(m)\, dm = d\omega.$$

We will occasionally simply write $\pi(m)$ instead of $\pi_M(m)$ if there is no ambiguity.

2.12 remark. In theory, we introduce the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ in order to compute the probability of a subset[10] A in the state space S, and this is essentially the meaning of (5). However, once we know the probability density function $\pi_M(m)$, we can operate directly on the state space S without the need for referring back to probability space $(\Omega, \mathcal{F}, \mathbb{P})$, as shown in definition (6). This is the key observation, a consequence of which is that we simply ig-

[10] Again, it needs to be measurable.

nore the underlying probability space in practice, since we don't need them in computation of probability in the state space. However, to intuitively understand the source of randomness, we need to go back to the probability space where the outcome of all events, except Ω, is uncertain. As a result, the pair $(S, \pi_M(m))$ contains complete information describing our ignorance about the outcome of random variable M. To the rest of this note, we shall work directly on the state space.

2.13 remark. At this point, you may wonder what is the point of introducing the abstract probability space $(\Omega, \mathcal{F}, \mathbb{P})$ to make life more complicated? Well, its introduction is two fold. First, as discussed above, the probability space not only shows the origin of randomness but also provides the probability measure $\mathbb{P}$ for the computation of the randomness; it is also used to define random variables and furnishes a decent understanding about them. Second, the concepts of distribution and density in (5) and (6), which are introduced for random variable M, a map from Ω to S, are valid for maps[11] from an arbitrary space V to another space W. Here, W plays the role of S, and V the role of Ω on which we have a probability measure. For example, later in Section 3, we introduce the parameter-to-observable map $h(m) : S \to \mathbb{R}^r$, then S plays the role of Ω and $\mathbb{R}^r$ of S in (5) and (6).

2.14 definition. The *expectation* or the *mean* of a random variable M is the quantity

$$\mathbb{E}[M] \overset{\text{def}}{=} \int_S m\pi(m)\, dm \overset{\text{def}}{=} \int_\Omega M(\omega)\, d\omega = \overline{m}, \tag{7}$$

and the *variance* is

$$\mathbb{V}ar[M] \overset{\text{def}}{=} \mathbb{E}\left[(M-\overline{m})^2\right] \overset{\text{def}}{=} \int_S (m-\overline{m})^2 \pi(m)\, dm \overset{\text{def}}{=} \int_\Omega (M(\omega)-\overline{m})^2\, d\omega.$$

As we will see, the Bayes formula for probability densities is about the joint density of two or more random variables. So let us define the joint distribution and joint density of two random variables here.

2.15 definition. Denote μ_{MY} and π_{MY} as the joint distribution and density, respectively, of two random variables M with values in S and Y with values in T defined on the same probability space, then the joint distribution function and the joint probability density, in the light of (6), satisfy

$$\mu_{MY}(\{M \in A\}, \{Y \in B\}) \overset{\text{def}}{=} \int_{A\times B} \pi_{MY}(m,y)\, dmdy, \quad \forall A \times B \subset S \times T, \tag{8}$$

[11] Again, they must be measurable.

where the notation $A \times B \subset S \times T$ simply means that $A \in S$ and $B \in T$.

We say that M and Y are independent if

$$\mu_{MY}\left(\{M \in A\}, \{Y \in B\}\right) = \mu_M(A)\,\mu_Y(B), \quad \forall A \times B \subset S \times T,$$

or if

$$\pi_{MY}(m,y) = \pi_M(m)\,\pi_Y(y).$$

2.16 definition. The *marginal* density of M is the probability density of M when Y may take on any value, i.e.,

$$\pi_M(m) = \int_T \pi_{MY}(m,y)\; dy.$$

Similarly, the marginal density of Y is the density of Y regardless of M, namely,

$$\pi_Y(y) = \int_S \pi_{MY}(m,y)\; dm.$$

Before deriving the Bayes formula, we define conditional density $\pi(m|y)$ in the same spirit as (6) as

$$\mu_{M|y}\left(\{M \in A\}|y\right) = \int_A \pi(m|y)\; dm.$$

Let us prove the following important result.

2.17 theorem. *The conditional density of M given Y is given by*

$$\boxed{\pi(m|y) = \tfrac{\pi(m,y)}{\pi(y)}}.$$

Proof. From the definition of conditional probability (3), we have

$$\begin{aligned}
&\mu_{M|Y}\left(\{m \in A\}|y\right) = \mathbb{P}\left[\{M \in A\}|Y = y\right] && \text{(definition (5))}\\
&= \lim_{\Delta y \to 0} \frac{\mathbb{P}\left[\{M \in A\}, y \le y' \le y + \Delta y\right]}{\mathbb{P}\left[y \le y' \le y + \Delta y\right]} && \text{(definition (3))}\\
&= \lim_{\Delta y \to 0} \frac{\int_A \pi(m,y)\; dm \Delta y}{\pi(y)\,\Delta y} && \text{(definitions (6), (8))}\\
&= \int_A \frac{\pi(m,y)}{\pi(y)}\, dm,
\end{aligned}$$

which ends the proof. □

By symmetry, we have

$$\pi(m, y) = \pi(m|y)\,\pi(y) = \pi(y|m)\,\pi(m),$$

from which the well-known Bayes formula follows

$$\pi(m|y) = \frac{\pi(y|m)\pi(m)}{\pi(y)}. \tag{9}$$

2.18 exercise. Prove directly the Bayes formula for conditional density (9) using the Bayes formula for conditional probability (4).

2.19 definition (Likelihood). We call $\pi(y|m)$ the likelihood. It is the probability density of y given $M = m$.

2.20 definition (Prior). We call $\pi(m)$ the prior. It is the probability density of M regardless of Y. The prior encodes, in the Bayesian framework, all information before any observations/data are made.

2.21 definition (Posterior). The density $\pi(m|y)$ is called the *posterior*, the distribution of parameter m given the measurement y, and it is the solution of the Bayesian inverse problem under consideration.

Can you write the conditional mean using $d\omega$?

2.22 definition. The *conditional mean* is defined as

$$\mathbb{E}[M|y] = \int_S m\pi(m|y)\, dm.$$

2.23 exercise. Show that

$$\mathbb{E}[M] = \int_T \mathbb{E}[M|y]\,\pi(y)\, dy.$$

3 Construction of likelihood

In this section, we present a popular approach to construct the likelihood. We begin with the additive noise case. The ideal deterministic model is given by

$$y = h(m),$$

where $y \in \mathbb{R}^r$. But due to random additive noise E, we have the following statistical model instead

$$Y^{obs} = h(M) + E, \tag{10}$$

where Y^{obs} is the actual observation rather than $Y = f(M)$. Since the noise comes from external sources, in this note, it is assumed to be independent of M. In the likelihood modeling, we pretend to have realization(s) of M and the task is to construct the distribution of Y^{obs}. From (10), one can see that the randomness in Y^{obs} is the randomness in E shifted by an amount $h(m)$, see Figure 3, and hence $\pi_{Y^{obs}|m}\left(y^{obs}|m\right) = \pi_E\left(y^{obs} - h(m)\right)$. More rigorously,

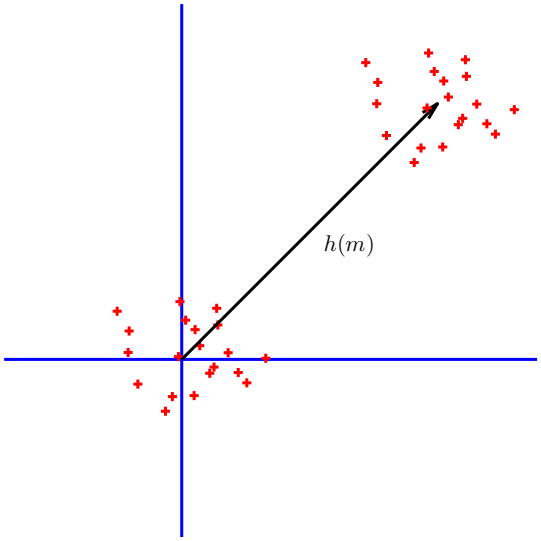


Figure 3: Likelihood model with additive noise.

assume that both Y^{obs} and E are random variables on a same probability space, we have

$$\int_A \pi_{Y^{obs}|m}\left(y^{obs}|m\right)\, dy^{obs} \overset{\text{def}}{=} \mu_{Y^{obs}|m}(A) \overset{(5)}{=} \mu_E(A - h(m))$$
$$= \int_{A-h(m)} \pi_E(e)\, de \overset{\text{change of variable}}{=} \int_A \pi_E\left(y^{obs} - h(m)\right) dy, \quad \forall A \subset S,$$

which implies

$$\pi_{Y^{obs}|m}\left(y^{obs}|m\right) = \pi_E\left(y^{obs} - h(m)\right).$$

We next consider multiplicative noise. The statistical model in this case

reads

$$Y^{obs} = Eh(M). \tag{11}$$

3.1 exercise. Show that the likelihood for multiplicative noise model (11) has the following form

$$\pi_{Y^{obs}|m}\left(y^{obs}|m\right) = \frac{\pi_E\left(y^{obs}/h(m)\right)}{h(m)}, \quad h(m) \neq 0.$$

3.2 exercise. Can you generalize the result for the noise model $e = g\left(y^{obs}, h(x)\right)$?

4 Construction of Prior(S)

As discussed previously, the prior belief depends on a person's knowledge and experience. In order to obtain a good prior, one sometimes needs to perform some expert elicitation. Nevertheless, there is no universal rule and one has to be careful in constructing a prior. In fact, prior construction is a subject of current research, and it is problem-dependent. For concreteness, let us consider the following one dimensional deblurring (deconvolution) problem

$$g(s_j) = \int_0^1 a(s_j, t) f(t)\, dt + e(s_j), \quad 0 \leq j \leq n,$$

where $a(s,t) = \frac{1}{\sqrt{2\pi\beta^2}} \exp(-\frac{1}{2\beta^2}(t-s)^2)$ is a given kernel, and $s_j = j/n, j = 0, \ldots, n$ the mesh points. Our task is to reconstruct $f(t) : [0,1] \to \mathbb{R}$ from the noisy observations $g(s_j)$, $j = 0, \ldots, n$. To cast the function reconstruction problem, which is in infinite dimensional space, into a reconstruction problem in $\mathbb{R}^n$, we discretize $f(t)$ on the same mesh and use simple rectangle method for the integral. Let us define $Y^{obs} = [g(s_0), \ldots, g(s_n)]^T$, $M = (f(s_0), \ldots, f(s_n))^T$, and $\mathcal{A}_{i,j} = a(s_i, s_j)/n$, then the discrete deconvolution problem reads

$$Y^{obs} = \mathcal{A}M + E.$$

Here, we assume $E \sim \mathcal{N}(0, \sigma^2 I)$, where I is the identity matrix in $\mathbb{R}^{(n+1)\times(n+1)}$. Since Section 3 suggests the likelihood of the form

$$\pi\left(y^{obs}|m\right) = \mathcal{N}\left(y^{obs} - \mathcal{A}m, \sigma^2 I\right),$$

the Bayesian solution to our inverse problem is, by virtue of the Bayes formula (9), given by

$$\pi_{\text{post}}\left(m|y^{obs}\right) \propto \mathcal{N}\left(y^{obs} - \mathcal{A}m, \sigma^2 I\right) \times \pi_{\text{prior}}(m), \tag{12}$$

where we have ignored the denominator $\pi\left(y^{obs}\right)$ since it does not depend on the parameter of interest m. We now start our prior elicitation.

4.1 *Smooth priors*

In this section, we believe that the unknown function $f(t)$ is smooth, which can be translated into, among other possibilities, the following simplest requirement on the pointwise values $f(s_i)$, and hence m_i,

$$m_i = \frac{1}{2}\left(m_{i-1} + m_{i+1}\right), \tag{13}$$

that is, the value of $f(s)$ at a point is more or less the same of its neighbor. But, this is by no means the correct behavior of the unknown function $f(s)$. We therefore admit some uncertainty in our belief (13) by adding an *innovative* term W_j such that

$$M_i = \frac{1}{2}\left(M_{i-1} + M_{i+1}\right) + W_j,$$

where $W \sim \mathcal{N}\left(0, \gamma^2 I\right)$. The standard deviation γ determines how much the reconstructed function $f(t)$ departs from the smoothness model (13). In terms of matrices, we obtain

$$LM = W,$$

where L is given by

$$L = \frac{1}{2}\begin{bmatrix} -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \end{bmatrix} \in \mathbb{R}^{(n-1)\times(n+1)},$$

which is the second order finite difference matrix approximating the Laplacian Δf. Indeed,

$$\Delta f(s_j) \approx n^2 \left(LM\right)_j. \tag{14}$$

The prior distribution is therefore given by (using the technique in Section 3)

$$\pi_{\text{pre}} \propto \exp\left(-\frac{1}{\gamma^2}\|LM\|^2\right). \tag{15}$$

But L has rank of $n-1$, and hence π_{pre} is a degenerate Gaussian density in $\mathbb{R}^{n+1}$. The reason is that we have not specified the smoothness of $f(s)$ at the boundary points. In other words, we have not specified any boundary conditions for the Laplacian $\Delta f(s)$. This is a crucial point in prior elicitation via differential operators. One needs to make sure that the operator is positive definite by incorporating some well-posed boundary conditions. Throughout the lecture notes, unless otherwise stated, $\|\cdot\|$ denotes the usual Euclidean norm.[12]

Let us first consider the case with zero Dirichlet boundary condition, that is, we believe that $f(s)$ is smooth and (close to) zero at the boundaries, then

$$\begin{aligned} M_0 &= \frac{1}{2}(M_{-1} + M_1) + W_0 = \frac{1}{2}M_1 + W_0, \quad W_0 \sim \mathcal{N}\left(0, \gamma^2\right) \\ M_n &= \frac{1}{2}(M_{n-1} + M_{n+1}) + W_n = \frac{1}{2}M_{n-1} + W_n, \quad W_n \sim \mathcal{N}\left(0, \gamma^2\right). \end{aligned}$$

Note that we have extended $f(s)$ by zero outside the domain $[0,1]$ since we "know" that it is smooth. Consequently, we have $L_D M = W$ with

$$L_D = \frac{1}{2}\begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}, \tag{16}$$

which is the second order finite difference matrix corresponding to zero Dirichlet boundary conditions. The prior density in this case reads

$$\pi^D_{\text{prior}}(m) \propto \exp\left(-\frac{1}{2\gamma^2}\|L_D m\|^2\right). \tag{17}$$

It is instructive to draw some random realizations from π^D_{prior} (we are ahead of ourselves here since sampling will be discussed in Section 7), and we show five of them in Figure 4 together with the prior standard deviation curve. As can be seen, all the draws are almost zero at the boundary and the prior variance (uncertainty) is close to zero as well. This is not surprising since our prior belief says so. How do we compute the standard deviation curve?

Why are they not exactly zero?

[12] The ℓ^2-norm if you wish

Well, it is straightforward. We first compute the pointwise variance as

$$\mathbb{V}ar\left[M_j\right] \stackrel{\text{def}}{=} \mathbb{E}\left[M_j^2\right] = e_j^T \left(\int_{\mathbb{R}^{n+1}} m^2 \pi_{\text{prior}}^D \, dm \right) e_j \stackrel{\text{def}}{=} \gamma^2 e_j^T \left(L_D^T L_D \right)^{-1} e_j,$$

where e_j is the jth canonical basis vector in $\mathbb{R}^{n+1}$, and we have used the fact that the prior is Gaussian in the last equality. So we in fact plot the square root of the diagonal of $\gamma^2 \left(L_D^T L_D \right)^{-1}$, the covariance matrix, as the standard deviation curve. One can see that the uncertainty is largest in the middle of the domain since it is farthest from the constrained boundary. The points closer to the boundaries have smaller variance, that is, they are more correlated to the "known" boundary data, and hence less uncertain.

Do we really have the complete continuous curve?

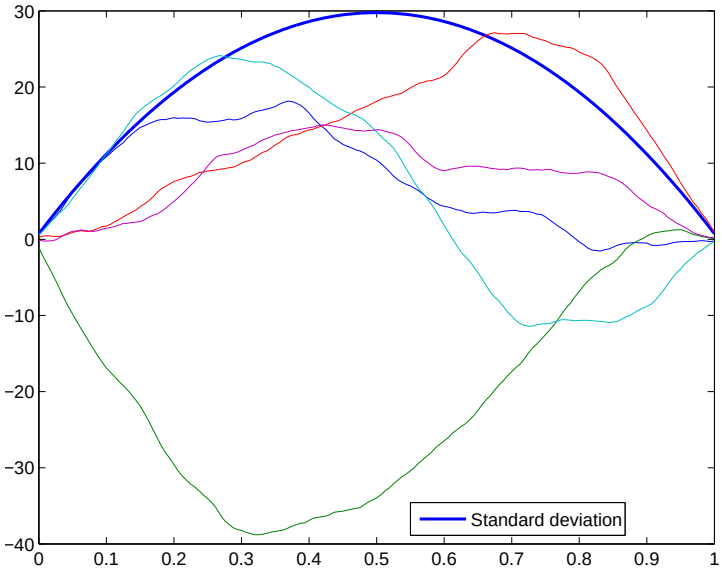


Figure 4: Prior random draws from π_{prior}^D together with the standard deviation curve.

Now, you may ask why $f(s)$ must be zero at the boundary, and you are right! There is no reason to believe that must be the case. However, we don't know the exact values of $f(s)$ at the boundary either, even though we believe that we may have non-zero Dirichlet boundary condition. If this is the case, we have to admit our ignorance and let the data from the likelihood correct us in the posterior. To be consistent with the Bayesian philosophy, if we do not know anything about boundary conditions, let them be, for convenience,

Gaussian random variables such as

$$M_0 \sim \mathcal{N}\left(0, \frac{\gamma^2}{\delta_0^2}\right), \quad M_n \sim \mathcal{N}\left(0, \frac{\gamma^2}{\delta_n^2}\right).$$

Hence, the prior can now be written as

$$\pi^R_{\text{prior}}(m) \propto \exp\left(-\frac{1}{2\gamma^2}\|L_R m\|^2\right), \tag{18}$$

where

$$L_R = \frac{1}{2}\begin{bmatrix} 2\delta_0 & 0 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & 0 & 2\delta_n \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}.$$

A question that immediately arises is how to determine δ_0 and δ_n. Since the boundary values are now independent random variables, we are less certain about them compared to the previous case. But to which uncertain level we want them to be? Well, let's make every values equally uncertain, meaning we have the same ignorance about the values at these points, that is, we would like to have the same variances everywhere. To approximately accomplish this, we require

$$\mathbb{V}ar[M_0] = \frac{\gamma^2}{\delta_0^2} = \mathbb{V}ar[M_n] = \frac{\gamma^2}{\delta_n^2} = \mathbb{V}ar\left[M_{[n/2]}\right] = \gamma^2 e_{[n/2]}^T \left(L_R^T L_R\right)^{-1} e_{[n/2]},$$

where $[n/2]$ denotes the largest integer smaller than $n/2$. It follows that

$$\delta_0^2 = \delta_n^2 = \frac{1}{e_{[n/2]}^T \left(L_R^T L_R\right)^{-1} e_{[n/2]}}.$$

However, this requires to solve a nonlinear equation for $\delta_0 = \delta_n$, since L_R depends on them. To simplify the computation, we use the following approximation

Is it sensible to do so?

$$\delta_0^2 = \delta_n^2 = \frac{1}{e_{[n/2]}^T \left(L_D^T L_D\right)^{-1} e_{[n/2]}}.$$

Again, we draw five random realizations from π^R_{prior} and put them together with the standard deviation curve in Figure 5. As can be observed, the

uncertainty is more or less the same at every point and prior realizations are no longer constrained to have zero boundary conditions.

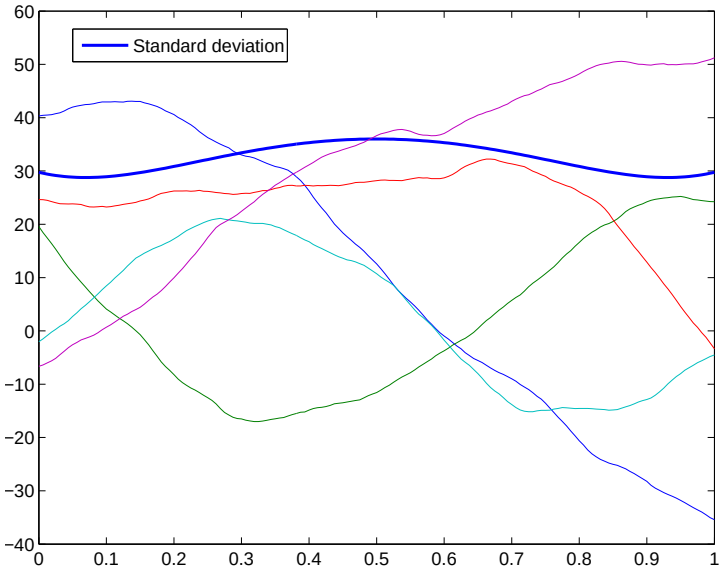


Figure 5: Prior random draws from π^R_{prior} together with the standard deviation curve.

4.1 exercise. Consider the following general scheme

$$m_i = \lambda_i m_{i-1} + (1-\lambda_i) m_{i+1} + E_i, \quad 0 \le \lambda_i \le 1.$$

Convince yourself that by choosing a particular set of λ_i, you can recover all the above prior models. Replace `BayesianPriorElicitation.m` by a generic code with input parameters λ_i. Experience new prior models by using different values of λ_i (those that don't reproduce priors presented in the text).

4.2 exercise. Construct a prior with a non-zero Dirichlet boundary condition at $s = 0$ and zero Neumann boundary condition at $s = 1$. Draw a few samples together with the variance curve to see whether your prior model indeed conveys your belief.

4.2 *"Non-smooth" priors*

We first consider the case in which we believe that $f(s)$ is still smooth but may have discontinuities at known locations on the mesh. Can we design a prior to convey this belief? A natural approach is to require that M_j is equal to M_{j-1}

plus a random jump, i.e.,

$$M_j = M_{j-1} + E_j,$$

where $E_j \sim \mathcal{N}\left(0, \gamma^2\right)$, and for simplicity, let us assume that $M_0 = 0$. The prior density in this case would be

$$\pi_{\text{pren}}(m) \propto \exp\left(-\frac{1}{2\gamma^2}\left\|L_N m\right\|^2\right), \tag{19}$$

where

$$L_N = \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{n\times n}.$$

But, if we think that there is a particular big jump, relative to others, from M_{j-1} to M_j, then the mathematical translation of this belief is $E_j \sim \mathcal{N}\left(0, \frac{\gamma^2}{\theta^2}\right)$ with $\theta < 1$. The corresponding prior in this case reads

$$\pi^O_{\text{prior}}(m) \propto \exp\left(-\frac{1}{2\gamma^2}\left\|JL_N m\right\|^2\right), \tag{20}$$

with

$$J = \operatorname{diag}\left(1, \ldots, 1, \underbrace{\theta}_{j\text{th index}}, 1, \ldots, 1\right).$$

Let's draw some random realizations from $\pi^O_{\text{prior}}(m)$ in Figure 6 with $n = 160$, $j = 80$, $\beta = 1$, and $\theta = 0.01$. As desired, all realizations have a sudden jump at $j = 80$, and the standard deviation of the jump is $1/\theta = 100$. In addition, compared to priors in Figure 4 and 5, the realizations from $\pi^O_{\text{prior}}(m)$ are less smooth, which confirms that our belief is indeed conveyed.

4.3 exercise. Use `BayesianPriorElicitation.m` to construct examples with 2 or more sudden jumps and plot a few random realizations to see whether your belief is conveyed.

A more interesting and more practical situation is the one in which we don't know how many jump discontinuities and their locations. A natural

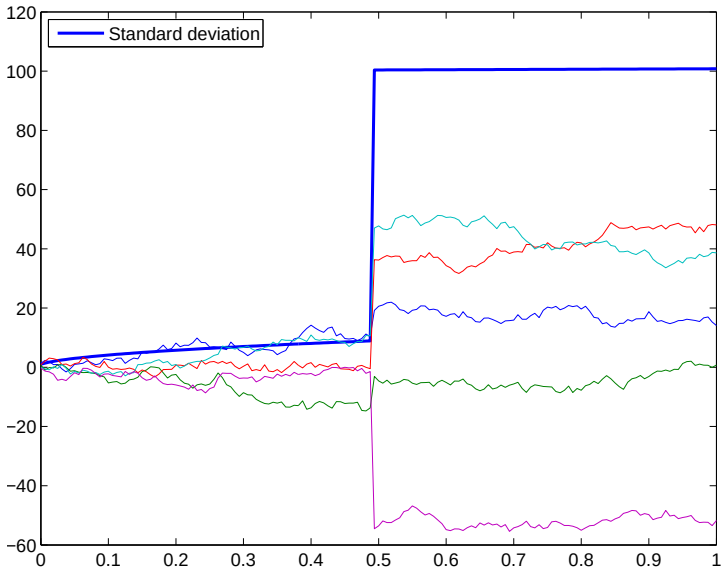


Figure 6: Prior random draws from π^{O}_{prior} together with the standard deviation curve.

prior in this situation is a generalized version of (20), e.g.,

$$\pi^{M}_{\text{prior}}(m) \propto C(M) \exp\left(-\frac{1}{2\gamma^2}\|ML_N m\|^2\right), \tag{21}$$

with

$$M = \operatorname{diag}(\theta_1, \ldots, \theta_n),$$

where θ_i, $i = 1, \ldots, n$, are unknown. In fact, these are called *hyper-parameters* and one can determine them using information from the likelihood; the readers are referred to [1] for the details.

wrong with this?

4.4 exercise. Modify the scheme in Exercise 4.1 to include priors with sudden jumps.

5 Posterior as the solution to Bayesian inverse problems

In this section, we explore the posterior (12), the solution of our Bayesian problem, given the likelihood in Section 3 and priors in Section 4.

To derive results that are valid for all priors discussed so far, we work with

the following generic prior

$$\pi_{\text{prior}}(m) \propto \exp\left(-\frac{1}{2\gamma^2}\left\|\Gamma^{-\frac{1}{2}}m\right\|^2\right),$$

where $\Gamma^{-\frac{1}{2}} \in \{L_D, L_A, HL_N\}$, each of which again presents a different belief. The Bayesian solution (12) can be now written as

$$\pi_{\text{post}}\left(m|y^{obs}\right) \propto \exp\left(-\underbrace{\left[\frac{1}{2\sigma^2}\left\|y^{obs}-\mathcal{A}m\right\|^2+\frac{1}{2\gamma^2}\left\|\Gamma^{-\frac{1}{2}}m\right\|^2\right]}_{T(m)}\right),$$

where $T(m)$ is the familiar (to you I hope) *Tikhonov functional*; it is sometimes called the *potential*. We re-emphasize here that the Bayesian solution is the posterior probability density, and if we draw samples from it, we want to know what the most likely function m is going to be. In other words, we ask for the most probable point m in the posterior distribution. This point is known as the *Maximum A Posteriori (MAP)* estimator/point, namely, the point at which the posterior density is maximized. Let us denote this point as m_{MAP}, and we have

$$m_{MAP} \overset{\text{def}}{=} \arg\max_m \pi_{\text{post}}\left(m|y^{obs}\right) = \arg\min_m T(m).$$

Hence, the MAP point is exactly the deterministic solution of the Tikhonov functional!

This is fundamental. If you have not seen this, prove it!

Since both likelihood and prior are Gaussian, the posterior is also a Gaussian. For our case, the resulting posterior Gaussian reads

$$\begin{aligned}\pi_{\text{post}}\left(m|y^{obs}\right) &\propto \exp\left(-\frac{1}{2}\left\|m-\frac{1}{\sigma^2}H^{-1}\mathcal{A}^Ty^{obs}\right\|_H^2\right)\\ &= \exp\left(-\frac{1}{2}\left(m-\frac{1}{\sigma^2}H^{-1}\mathcal{A}^Ty^{obs}, H\left(m-\frac{1}{\sigma^2}H^{-1}\mathcal{A}^Ty^{obs}\right)\right)\right)\\ &\overset{\text{def}}{=} \exp\left(-\frac{1}{2}\left(m-\frac{1}{\sigma^2}H^{-1}\mathcal{A}^Ty^{obs}, \Gamma_{\text{post}}^{-1}\left(m-\frac{1}{\sigma^2}H^{-1}\mathcal{A}^Ty^{obs}\right)\right)\right)\end{aligned}$$

where

$$\boxed{H = \tfrac{1}{\sigma^2}\mathcal{A}^T\mathcal{A} + \tfrac{1}{\gamma^2}\Gamma^{-1}},$$

is the Hessian of the Tikhonov functional (aka the regularized misfit), and we have used the weighted norm $\|\cdot\|_H^2 = \left\|H^{\frac{1}{2}}\cdot\right\|^2$.

5.1 exercise. Show that the posterior is indeed a Gaussian, i.e.,

$$\pi_{\text{post}}\left(m|y^{obs}\right) \propto \exp\left(-\frac{1}{2}\left\|m - \frac{1}{\sigma^2}H^{-1}\mathcal{A}^T y^{obs}\right\|_H^2\right).$$

The other important point is that the posterior covariance matrix is precisely the inverse of the Hessian of the regularized misfit, i.e.,

$$\boxed{\Gamma_{\text{post}} = H^{-1}}.$$

Last, but not least, we have showed that the MAP point is given by

$$m_{MAP} = \frac{1}{\sigma^2}H^{-1}\mathcal{A}^T y^{obs} = \frac{1}{\sigma^2}\left(\frac{1}{\sigma^2}\mathcal{A}^T\mathcal{A} + \frac{1}{\gamma^2}\Gamma^{-1}\right)^{-1}\mathcal{A}^T y^{obs},$$

which is, again, exactly the solution of the Tikhonov functional for linear inverse problem.

5.2 exercise. Show that m_{MAP} is also the least squares solution of the following over-determined system

$$\begin{bmatrix} \frac{1}{\sigma}\mathcal{A} \\ \frac{1}{\gamma}\Gamma^{-\frac{1}{2}} \end{bmatrix} m = \begin{bmatrix} \frac{1}{\sigma}y^{obs} \\ 0 \end{bmatrix}$$

5.3 exercise. Show that the posterior mean, which is in fact the conditional mean, is precisely the MAP point.

Since the covariance matrix, generalization of the variance in multi-dimensional spaces, represents the uncertainty, quantifying the uncertainty in the MAP estimator is ready by simply computing the inverse the Hessian matrix. Let's us now numerically explore the Bayesian posterior solution.

We choose $\beta = 0.05$, $n = 100$, and $\gamma = 1/n$. The truth underlying function that we would like to invert for is given by

$$f(t) = 10(t - 0.5)\exp\left(-50(t - 0.5)^2\right) - 0.8 + 1.6t.$$

The noise level is taken to be the 5% of the maximum value of $f(s)$, i.e. $\sigma = 0.05\max_{s\in[0,1]}|f(s)|$.

We first consider the belief described by π^D_{prior} in which we think that $f(s)$ is zero at the boundaries. Figures 7 plots the MAP estimator, the truth function $f(s)$, and the predicted uncertainty. As can be observed, the MAP is in good agreement with the truth function inside the interval $[0, 1]$, though it is far from recovering $f(s)$ at the boundaries. This is the price we have to pay for not admitting our ignorance about the boundary values of $f(s)$. The

likelihood in fact sees this discrepancy in the prior knowledge and tries to make correction by lifting the MAP away from 0, but not enough to be a good reconstruction. The reason is that our incorrect prior is strong enough such that the information from the data y^{obs} cannot help much.

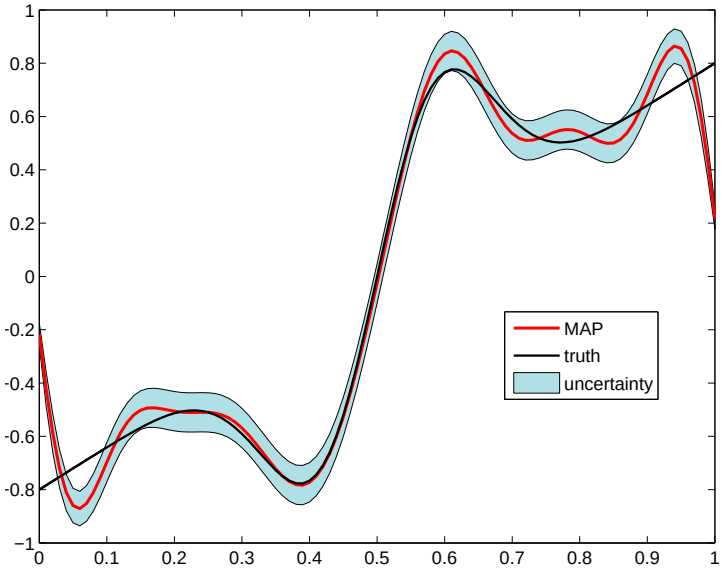


Figure 7: The MAP estimator, the truth function, and the predicted uncertainty (95% credibility region) using π^D_{prior}.

5.4 exercise. Can you make the prior less strong? *Change some parameter to make prior contribution less!* Use `BayesianPosterior.m` to test your answer. Is the prediction better in terms of satisfying the boundary conditions? Is the uncertainty smaller? If not, why?

On the other hand, if we admit this ignorance and use the corresponding prior π^D_{prior}, we see much better reconstruction in Figure 8. In this case, we in fact let the information from the data y^{obs} determine the appropriate values for the Dirichlet boundary conditions rather than setting them to zero. By doing this, we allow the likelihood and the prior to be well-balanced leading to good reconstruction and uncertainty quantification.

5.5 exercise. Play with `BayesianPosterior.m` by varying γ, the data misfit (or the likelihood) contribution, and σ, the regularization (or the prior) contribution.

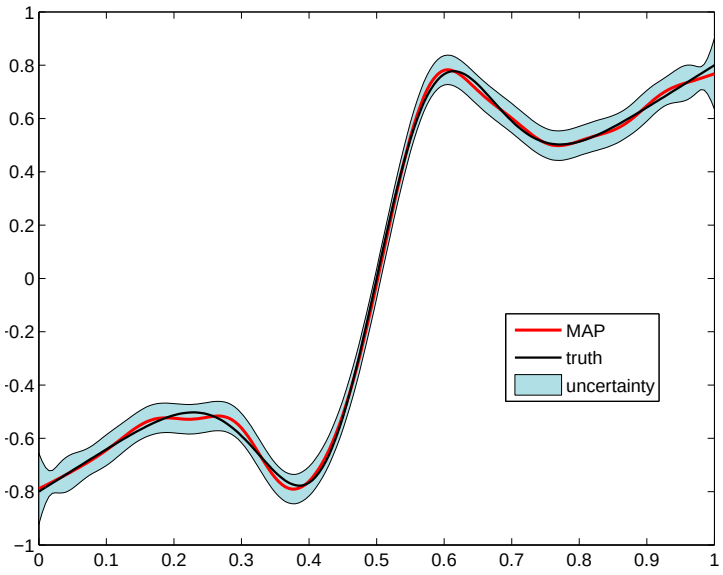


Figure 8: The MAP estimator, the truth function, and the predicted uncertainty (95% credibility region) using π^A_{prior}.

5.6 exercise. Use your favorite deterministic inversion approach to solve the above deconvolution problem and then compare it with the solution in Figure 8.

Now consider the case in which the truth function has a jump discontinuity at $j = 70$. Assume we also know that the magnitude of the jump is 10. In particular, we take the truth function $f(s)$ as the following step function

$$f(s) = \begin{cases} 0 & \text{if } s \leq 0.7 \\ 10 & \text{otherwise} \end{cases}.$$

Since we otherwise have no further information about $f(s)$, let us be more conservative by choosing $\gamma = 1$ and $\theta = 0.1$ at $j = 70$ in π^O_{prior} as we discussed in (20). Figure 9 shows that we are doing pretty well in recovering the jump and other parts of the truth function.

A question you may ask is whether we can do better? The answer is yes by taking smaller γ if the truth function does not vary much everywhere except at the jump discontinuity. We take this prior information into account by taking $\gamma = 1.e-8$, for example, then our reconstruction is almost perfect in Figure 10.

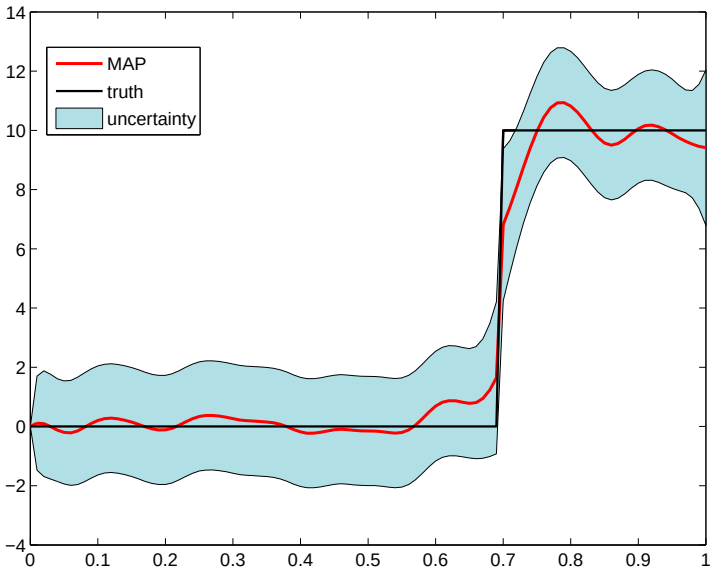


Figure 9: The MAP estimator, the truth function, and the predicted uncertainty (95% credibility region) using π^O_{prior}.

5.7 exercise. Try `BayesianPosteriorJump.m` with γ deceasing from 1 to $1.e-8$ to see the improvement in quality of the reconstruction.

5.8 exercise. Use your favorite deterministic inversion approach to solve the above deconvolution problem with discontinuity and then compare it with the solution in Figure 10.

6 Connection between Bayesian inverse problems and deterministic inverse problems

We have touched upon the relationship between Bayesian inverse problem and deterministic inverse problem in Section 5 by pointing out that the potential of the posterior density is precisely the Tikhonov functional up to a constant. We also point out that the MAP estimator is exactly the solution of the deterministic inverse problem. Note that we derive this relation for a linear likelihood model, but it is in fact true for nonlinear ones (e.g. nonlinear parameter-to-observable map $\mathcal{A}m$). *Can you confirm this?*

Up to this point, you may realize that the Bayesian solution contains much more information than its deterministic counterpart. Instead of having a point estimate, the MAP point, we have a complete posterior distribution to explore. In particular, we can talk about a simple uncertainty quantification by exam-

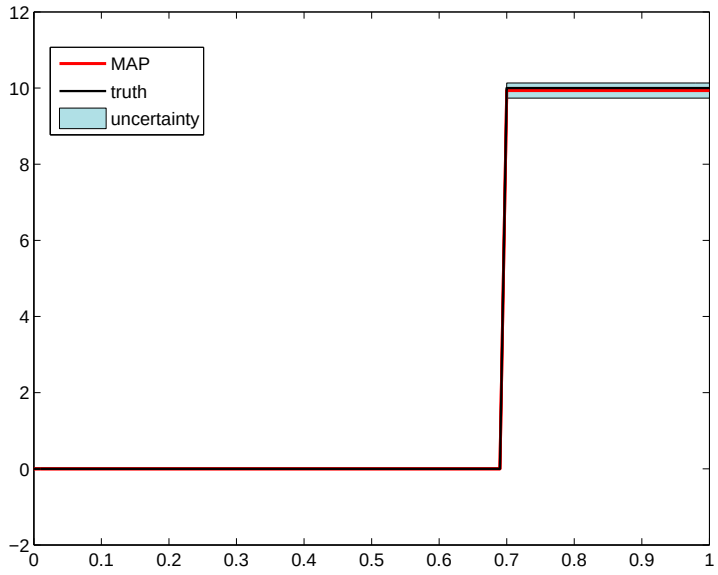


Figure 10: The MAP estimator, the truth function, and the predicted uncertainty (95% credibility region) using π^O_{prior}.

ining the diagonal of the posterior covariance matrix. We can even discuss about the posterior correlation structure by looking at the off diagonal elements, though we are not going to do it here in this lecture note. Since, again, both likelihood and prior are Gaussian, the posterior is a Gaussian distribution, and hence the MAP point (the first order moment) and the covariance matrix (the second order moment) are the complete description of the posterior. If, however, the likelihood is not Gaussian, say when the $\mathcal{A}m$ is nonlinear, then one can explore higher moments.

We hope the arguments above convince you that the Bayesian solution provide information far beyond the deterministic counterpart. In the remainder of this section, let us dig into details the connection between the MAP point and the deterministic solution, particularly in the context of the deconvolution problem. Recall the definition of the MAP point

$$\begin{aligned} m_{MAP} &\overset{\text{def}}{=} \arg\min_m T(m) = \sigma^2 \left(\frac{1}{2} \left\| y^{obs} - \mathcal{A}m \right\|^2 + \frac{1}{2} \frac{\sigma^2}{\gamma^2} \left\| \Gamma^{-\frac{1}{2}} m \right\|^2 \right) \\ &= \arg\min_m T(m) = \sigma^2 \left(\frac{1}{2} \left\| y^{obs} - y \right\|^2 + \frac{1}{2} \kappa \left\| R^{\frac{1}{2}} m \right\|^2 \right), \end{aligned}$$

where we have defined $\kappa = \sigma^2/\gamma^2$, $R^{\frac{1}{2}} = \Gamma^{-\frac{1}{2}}$, and $y = \mathcal{A}m$.

We begin our discussion with zero Dirichlet boundary condition prior

$\pi^D_{\text{prior}}(m)$ in (17). Recall in (14) and (16) that $L_D M$ is proportional to a discretization of the Laplacian operator with zero boundary conditions using second order finite difference method. Therefore, our Tikhonov functional is in fact a discretization, up to a constant, of the following potential in the infinite dimensional setting

$$T_\infty(f) = \frac{1}{2}\left\|y - y^{obs}\right\|^2 + \frac{1}{2}\kappa \|\Delta f\|^2_{L^2(0,1)},$$

where $\|\cdot\|^2_{L^2(0,1)} \overset{\text{def}}{=} \int_0^1 (\cdot)^2 \, ds$. Rewrite the preceding equation informally as

$$T_\infty(f) = \frac{1}{2}\left\|y - y^{obs}\right\|^2 + \frac{1}{2}\kappa \left(f, \Delta^2 f\right)_{L^2(0,1)},$$

and we immediately realize that the potential in our prior description, namely $\|L_D m\|^2$, is in fact a discretization of Tikhonov regularization using the biharmonic operator. This is another explanation for the smoothness of the prior realizations and the name smooth prior, since biharmonic regularization is very smooth. [13]

The power of the statistical approach lies in the construction of prior $\pi^R_{\text{prior}}(m)$. Here, the interpretation of rows corresponding to interior nodes s_j is still the discretization of the biharmonic regularization, but the design of those corresponding to the boundary points is purely statistics, for which we have no corresponding deterministic counterpart (or at least it is not clear how to construct it from a purely deterministic point of view). As the results in Section 5 showed, $\pi^R_{\text{prior}}(m)$ provided much more satisfactory results both in the prediction and in uncertainty quantification.

As for the "non-smooth"priors in Section 4.2, a simple inspection shows that $L_N m$ is, up to a constant, a discretization of ∇f. Similar to the above discussion, the potential in our prior description, namely $\|L_D m\|^2$, is now in fact a discretization of Tikhonov regularization using the Laplacian operator.[14] As a result, the current prior is less smooth than the previous one with harmonic operator. Nevertheless, all the prior realizations corresponding to $\pi_{\text{pren}}(m)$

[13] From a functional analysis point of view, $\|\Delta f\|^2_{L^2(0,1)}$ is finite if $f \in H^2(0,1)$, and by Sobolev imbedding theorem we know that in fact $f \in C^{1,1/2-\varepsilon}$, the space of continuous differential functions whose first derivative is in the Hölder space of continuous function $C^{1/2-\varepsilon}$, for any $0 < \varepsilon < \frac{1}{2}$. So indeed f is more than continuously differentiable.

[14] Again, Sobolev embedding theorem shows that $f \in C^{1/2-\varepsilon}$ for $\|\nabla f\|^2_{L^2(0,1)}$ to be finite. Hence, all prior realizations corresponding to $\pi_{\text{pren}}(m)$ are at least continuous. The prior $\pi^O_{\text{prior}}(m)$ is different, due to the scaling matrix J. As long as θ stays away from zero, prior realizations are still in $H^1(0,1)$, and hence continuous though having steep gradient at s_j as shown in Figures 9 and 10. But as θ approaches zero, prior realizations are leaving $H^1(0,1)$, and therefore may be no longer continuous. Note that in one dimension, $H^{\frac{1}{2}+\varepsilon}$ is enough to be embedded in the space of C^ε-Hölder continuous functions. If you like to know a bit about the Sobolev embedding theorem, see [3].

are at least continuous, though may have steep gradient at s_j as shown in Figures 9 and 10. The rigorous arguments for the prior smoothness require the Sobolev embedding theorem, but we avoid the details.

For those who have not seen the Sobolev embedding theorem, you only loose the insight on why $\pi^O_{\text{prior}}(m)$ could give very steep gradient realizations (which is the prior belief we start with). Nevertheless, you still can see that $\pi^O_{\text{prior}}(m)$ gives less smooth realizations than $\pi^D_{\text{prior}}(m)$ does, since, at least, the MAP point corresponding to $\pi^O_{\text{prior}}(m)$ only requires finite first derivative of f while second derivative of f needs to be finite at the MAP point if $\pi^D_{\text{prior}}(m)$ is used.

7 Markov chain Monte Carlo

In the last section, we have shown that if the paramter-to-observable map is linear, i.e. $h(m) = \mathcal{A}m$, and both the noise and the prior models are Gaussian, then the MAP point and the posterior covariance matrix are exactly the solution and the inverse of the Hessian of the Tikhonov functional, respectively. Moreover, since the posterior is Gaussian, the MAP point is identically the mean, and hence the posterior distribution is completely characterized. In practice, $h(m)$ is typically nonlinear. Consequently, the posterior distribution is no longer Gaussian. Nevertheless, the MAP point is still the solution of the Tikhonov functional, though the mean and the convariance matrix are to be determined. The question is how to estimate the mean and the covariance matrix of a non-Gaussian density.

We begin by recalling the definition the mean

$$\overline{m} = \mathbb{E}[M],$$

and a natural idea is to approximate the integral by some numerical integration. For example, suppose $S = [0,1]$ and then we can divide S into N intervals, each of which has length of $1/N$. Using a rectangle rule gives

$$\overline{m} \approx \frac{(M_1 + \ldots + M_N)}{N}. \tag{22}$$

But this kind of method cannot be extended to $S = \mathbb{R}^n$. This is where the central limit theorem and law of large numbers come to rescue. They say that the simple formula (22) is still valid with a simple error estimation expression.

7.1 *Some classical limit theorems*

7.1 theorem (central limit theorem (CLT)). *Assume that real valued random variables $M_1, \ldots$ are independent and identically distributed (iid), each with expectation*

$\overline{m}$ and variance σ^2. Then

$$Z_N = \frac{1}{\sigma\sqrt{N}} \left(M_1 + M_2 + \cdots + M_N \right) - \frac{\overline{m}}{\sigma}\sqrt{N}$$

converges, in distribution[15]*, to a standard normal random variable. In particular,*

$$\lim_{N\to\infty} \mathbb{P}\left[Z_N \leq m \right] = \frac{1}{2\pi} \int_{-\infty}^{m} \exp\left(-\frac{t^2}{2} \right) dt \tag{23}$$

Proof. The proof is elementary, though technical, using the concept of characteristic function (Fourier transform of a random variable). You can consult [4] for the complete proof. □

7.2 theorem (Strong law of large numbers (LLN)). *Assume random variables $M_1, \ldots$ are independent and identically distributed (iid), each with finite expectation $\overline{m}$ and finite variance σ^2. Then*

$$\lim_{N\to\infty} S_N = \frac{1}{N} \left(M_1 + M_2 + \cdots + M_N \right) = \overline{m} \tag{24}$$

almost surely[16]*.*

Proof. A beautiful, though not classical, proof of this theorem is based on backward martingale, tail σ-algebra, and uniform integrability. Let's accept it in this note and see [4] for the complete proof. □

7.3 remark. The central limit theorem says that no matter what the underlying common distribution looks like, the sum of iid random variables, when properly scaled and centralized, converges in distribution to a standard normal distribution. The strong law of large numbers, on the other hand, states that the average of the sum is, as expected in the limit, precisely the mean of the common distribution with probability one.

Both the central limit theorem (CLT) and the strong law of large numbers (LLN) are useful, particularly LLN, and we use them routinely. For example, if you are given an iid sample $\{M_1, M_2, \cdots, M_N\}$ from a common distribution $\pi(m)$, the first thing you should do is to compute the the sample mean S_N to estimate the actual mean $\overline{m}$. From LLN we know that the sample mean can be as close as desired if N is sufficiently large. A question immediately arises is whether we can estimate the error between the sample mean and the

[15] Convergence in distribution is also known as weak convergence and it is beyond the scope of this introductory note. You can think of the distribution of Z_n is more and more like the standard normal distribution as $n \to \infty$, and it is precisely (23).

[16] Almost sure convergence is the same as convergence with probability one, that is, the event on which the convergence (24) does not happen has zero probability.

truth mean, given a finite N. Let us first give an answer based on a simple application of the CLT. Since the sample $\{M_1, M_2, \cdots, M_N\}$ satisfies the condition of the CLT, we know that Z_N converges to $\mathcal{N}(0,1)$. It follows that, at least for sufficiently large N, the mean squared error between z_N and 0 can be estimated as

$$\|z_N - 0\|^2_{L^2(S,\mathbb{P})} \overset{\text{def}}{=} \mathbb{E}\left[(z_N - 0)^2\right] \overset{\text{def}}{=} \mathbb{V}ar\left[Z_N - 0\right] \approx 1,$$

which, after some simple algebra manipulations, can be rewritten as

$$\|S_N - \overline{m}\|^2_{L^2(S,\mathbb{P})} \overset{\text{def}}{=} \mathbb{V}ar\left[S_N - \overline{m}\right] \approx \frac{\sigma^2}{N} \tag{25}$$

7.4 exercise. Show that (25) holds.

7.5 remark. The result (25) shows that the error of the sample mean S_N in the $L^2(S,\mathbb{P})$-norm goes to zero like $1/\sqrt{N}$. One should be aware of the popular statement that the error goes to zero like $1/\sqrt{N}$ independent of dimension is not entirely correct because the variance σ^2, and hence the standard deviation σ, of the underlying distribution $\pi(m)$ may depend on the dimension n.

If you are a little bit delicate, you may not feel completely comfortable with the error estimate (25) since you can rewrite it as

$$\|S_N - \overline{m}\|_{L^2(S,\mathbb{P})} = C\frac{\sigma}{\sqrt{N}},$$

and you are not sure how big C is and the dependence of C on N. Let us try to make you happy. We have

$$\begin{aligned}\|S_N - \overline{m}\|^2_{L^2(S,\mathbb{P})} &= \frac{1}{N^2}\mathbb{E}\left[\left(\sum_{i=1}^N (M_i - \overline{m})\right)\left(\sum_{j=1}^N (M_j - \overline{m})\right)\right] \\ &= \frac{1}{N^2}\mathbb{E}\left[\left(\sum_{i=1}^N (M_i - \overline{m})^2\right)\right] = \frac{1}{N^2}\sum_{i=1}^N \sigma^2 = \frac{\sigma^2}{N},\end{aligned}$$

where we have used $\overline{m} = \frac{1}{N}\sum_{i=1}^N \overline{m}$ in the first equality, $\mathbb{E}\left[(M_i - \overline{m})(M_j - \overline{m})\right] = 0$ if $i \neq j$ in the second equality since M_i, $i = 1, \ldots, N$ are iid random variables, and the definition of variance in the third equality. So in fact $C = 1$, and we hope that you feel pleased by now.

In practice, we rarely work with M directly but indirectly via some mapping $g : S \to T$. We have that $g(M_i)$, $i = 1, \ldots, N$ are iid[17] if M_i, $i = 1, \ldots, N$ are iid.

[17] We avoid technicalities here, but g needs to be a Borel function for the statement to be true.

7.6 exercise. Suppose the density of M is $\pi(m)$ and $z = g(m)$ is differentially invertible, i.e. $m = g^{-1}(z)$ exists and differentiable, what is the density of $g(M)$?

Perhaps, one of the most popular and practical problems is to evaluate the mean of g, i.e.,

$$I \overset{\text{def}}{=} \mathbb{E}[G(M)] = \int_S g(m)\,\pi(m)\,dm, \tag{26}$$

which is an integral in $\mathbb{R}^n$.

7.7 exercise. Define $z = g(m) \in T$, the definition of the mean in (7) gives

$$\mathbb{E}[G(M)] \equiv \mathbb{E}[Z] \overset{\text{def}}{=} \int_T z\pi_Z(z)\,dz.$$

Derive formula (26).

Again, we emphasize that using any numerical integration methods that you know of for integral (26) is either infeasible or prohibitedly expensive when the dimension n is large, and hence not scalable. The LLN provides a reasonable answer if we can draw iid samples $\{g(M_1), \ldots, g(M_N)\}$ since we know that

$$\lim_{N\to\infty} \underbrace{\frac{1}{N}\left(g(M_1) + \ldots + g(M_N)\right)}_{I_N} = I$$

with probability 1. Moreover, as showed above, the mean squared error is given by *Do you trivially see this?*

$$\|I_N - I\|^2_{L^2(T,\mathbb{P})} = \mathbb{E}\left[(I_N - I)^2\right] = \frac{\mathbb{V}ar[G(M)]}{N}.$$

Again, the error decreases to zero like $1/\sqrt{N}$ "independent" of the dimension of T, but we need to be careful with such a statement unless $\mathbb{V}ar[G(M)]$ DOES NOT depend on the dimension.

A particular function g of interest is the following

$$g(m) = (m - \overline{m})(m - \overline{m})^T,$$

whose expectation is precisely the covariance matrix

$$\Gamma = cov(M) = \mathbb{E}\left[(M - \overline{m})(M - \overline{m})^T\right] = \mathbb{E}[G].$$

The average I_N in this case is known as the sample (aka empirical) covariance matrix. Denote

$$\hat{\Gamma} = \frac{1}{N} \sum_{i=1}^{N} (m_i - \overline{m}) (m_i - \overline{m})^T$$

as the sample covariance matrix. Note that $\overline{m}$ is typically not available in practice, and we have to resort to a computable approximation

$$\hat{\Gamma} = \frac{1}{N} \sum_{i=1}^{N} (m_i - \hat{m}) (m_i - \hat{m})^T ,$$

with $\hat{m}$ denoting the sample mean.

7.2 *Independent and identically distributed random draws*

Sampling methods discussed in this note are based on two fundamental iid random generators that are available as built-in functions in Matlab. The first one is `rand.m` function which can draw iid random numbers (vectors) from the uniform distribution in $[0, 1]$, denoted as $U[0, 1]$, and the second one is `randn.m` function that generates iid numbers (vectors) from standard normal distribution $\mathcal{N}(0, I)$, where I is the identity matrix of appropriate size.

The most trivial task is how to draw iid samples $\{M_1, M_2, \ldots, M_N\}$ from a multivariate Gaussian $\mathcal{N}(\overline{m}, \Gamma)$. This can be done through a so-called *whitening* process. The first step is to carry out the following decomposition

$$\Gamma = RR^T,$$

which can be done, for example, using Cholesky factorization. The second step is to define a new random variable as

$$Z = R^{-1} (M - \overline{m}) ,$$

at Z is a standard multivariate n.

then Z is a standard multivariate Gaussian, i.e. its density is $\mathcal{N}(0, I)$, for which `randn.m` can be used to generate iid samples

$$\{Z_1, Z_2, \ldots, Z_N\} = \texttt{randn(n,N)}.$$

We now generate iid samples M_i by solving

$$M_i = \overline{m} + RZ_i.$$

7.8 exercise. Look at `BayesianPriorElicitation.m` to see how we apply the above whitening process to generate multivariate Gaussian prior random realizations.

You may ask what if the distribution under consideration is not Gaussian, which is true for most practical applications. Well, if the target density $\pi(m)$ is one dimensional or multivariate with independent components (in this case, we can draw samples from individual components separately), then we still can draw iid samples from $\pi(m)$, but this time via the standard uniform distribution $U[0,1]$. If you have not seen it before, here is the definition: $U[0,1]$ has 1 as its density function, i.e.,

$$\mu_U(A) = \int_A ds, \quad \forall A \subset [0,1]. \tag{27}$$

Now suppose that we would like to draw iid samples from a one dimensional ($S = \mathbb{R}$) distribution with density $\pi(m) > 0$. We still allow $\pi(m)$ to be zero, but only at isolated points on $\mathbb{R}$, and you will see the reason in a moment. Define the cumulative distribution function (CDF) as

$$\Phi(w) = \int_{-\infty}^{w} \pi(m)\, dm, \tag{28}$$

Why? then it is clearly that $\Phi(w)$ is non-decreasing and $0 \leq \Phi(w) \leq 1$. Let us define a new random variable Z as

$$Z = \Phi(M). \tag{29}$$

Our next step is to prove that Z is actually a standard uniform random variable, i.e. $Z \sim U[0,1]$, and then show how to draw M via Z. We begin by the following observation

$$\mathbb{P}[Z < a] = \mathbb{P}[\Phi(M) < a] = \mathbb{P}\left[M < \Phi^{-1}(a)\right] = \int_{-\infty}^{\Phi^{-1}(a)} \pi(m)\, dm, \tag{30}$$

where we have used (29) in the first equality, the monotonicity of $\Phi(M)$ in the second equality, and the definition of CDF (28) in the last equality. Now, we can view (29) as the change of variable formula $z = \Phi(m)$, then combining this fact with (28) to have

$$dz = d\Phi(m) = \pi(m)\, dm, \text{ and } z = a \text{ when } x = \Phi^{-1}(a).$$

Consequently, (30) becomes *Do you see the second equality?*

$$\mathbb{P}[Z < a] = \int_{-\infty}^{a} dz = \mu_Z(Z < a),$$

which says that the density of Z is 1, and hence Z must be a standard uniform random variable. In terms of our language at the end of Section 7.1, we can define $M = g(Z) = \Phi^{-1}(Z)$, then drawing iid samples for M is simple by first drawing iid samples from Z, then mapping them through g. Let us summarize the idea in Algorithm 1.

Algorithm 1 CDF-based sampling algorithm

1. Draw $z \sim U[0,1]$,
2. Compute the inverse of the CDF to draw m, i.e. $m = \Phi^{-1}(z)$. Go back to Step 1.

The above method works perfectly if one can compute the analytical inverse of the CDF easily and efficiently; it is particularly efficient for discrete random variables, as we shall show. You may say that you can always compute the inverse CDF numerically. Yes, you are right, but you need to be careful about this. Note that the CDF is an integral operation, and hence its inverse is some kind of differentiation. The fact is that numerical differentiation is an ill-posed problem! You don't want to add extra ill-posedness on top of the original ill-posed inverse problem that you started with, do you? If not, let us introduce to you a simpler but more robust algorithm that works for multivariate distribution without requiring the independence of individual components. We shall first introduce the algorithm and then analyze it to show you why it works.

Suppose that you want to draw iid samples from a target density $\pi(m)$, but you only know it up to a constant $C > 0$, that is, you only know $C\pi(m)$. (This is perfect for our Bayesian inversion framework since we typically know the posterior up to a constant as in (12).) Assume that we have a *proposal distribution* $q(m)$ at hand, for which we know how to sample easily and efficiently. This is not a limitation since we can always take either the standard normal distribution or uniform distribution as the proposal distribution. We further assume that there exists $D > 0$ such that

$$C\pi(m) \leq Dq(m), \tag{31}$$

then we can draw a sample from $\pi(m)$ by the rejection-acceptance sampling Algorithm 2.

In practice, we carry out Step 3 of Algorithm 2 by flipping an "α-coin". In particular, we draw u from $U[0,1]$, then accept m if $\alpha > u$. It may seem to be magic to you why Algorithm 2 provides random samples from $\pi(m)$. Let us confirm this with you using the Bayes formula (9).

7.9 proposition. *Accepted m is distributed by the target density π.*

Algorithm 2 Rejection-Acceptance sampling algorithm

1. Draw m from the proposal $q(m)$,
2. Compute the *acceptance probability*

$$\alpha = \frac{C\pi(m)}{Dq(m)},$$

3. Accept m with probability α or reject it with probability $1-\alpha$. Go back to Step 1.

Make sure you understand this proof since we will reuse most of it for the Metropolis-Hastings algorithm!

Proof. Denote B as the event of accepting a draw q (or the acceptance event). Algorithm 2 tells us that the probability of B given m, which is precisely the acceptance probability, is

$$\mathbb{P}[B|m] = \alpha = \frac{C\pi(m)}{Dq(m)}. \tag{32}$$

On the other hand, the prior probability of m in the incremental event $dA = [m', m'+dm]$ in Step 1 is $q(m)\,dm$. Applying the Bayes formula for conditional probability (4) yields the distribution of a draw m provided that it has been already accepted

$$\mathbb{P}[m \in dA|B] = \frac{\mathbb{P}[B|m]\,q(m)\,dm}{\mathbb{P}[B]} = \pi(m)\,dm,$$

where we have used (32) and $\mathbb{P}[B]$, the probability of accepting a draw from q, is the following marginal probability

$$\mathbb{P}[B] = \int_S \mathbb{P}[B|m]\,q(m)\,dm = \frac{C}{D}\int_S \pi(m)\,dm = \frac{C}{D}.$$

Note that

$$\mathbb{P}[B, m \in dm] = \pi(B, m)\,dm = \mathbb{P}[B|m]\,\pi_{\text{prior}}(m) = \mathbb{P}[B|m]\,q(m)\,dm,$$

an application of (3), is the probability of the joint event of drawing an m from $q(m)$ and accept it. The probability of B, the acceptance event, is the total of accepting probability, which is exactly the marginal probability. As a result, we have

$$\mathbb{P}[m \in A|B] = \int_A \pi(m)\,dm,$$

which, by definition (6), says that the accepted m in Step 3 of Algorithm 2 is distributed by $\pi(m)$, and this is the desired result. □

Algorithm 2 is typically slow in practice in the sense that a large portion of samples is rejected, particularly for high dimensional problem, though it provides iid samples from the true underlying density. Another problem with this algorithm is the computation of D. Clearly, we can take very large D and the condition (31) would be satisfied. However, the larger D is the smaller the acceptance probability α, making Algorithm 2 inefficient since most of draws from $q(m)$ will be rejected. As a result, we need to minimize D and this could be nontrivial depending the complexity of the target density.

7.10 exercise. You are given the following target density

$$\pi(m) = \frac{g(m)}{C} \exp\left(-\frac{m^2}{2}\right),$$

where C is some constant independent of m, and

$$g(m) = \begin{cases} 1 & \text{if } x > a \\ 0 & \text{otherwise} \end{cases}, \quad a \in \mathbb{R}.$$

Take the proposal density as $q(m) = \mathcal{N}(0, 1)$.

1. Find the smallest D that satisfies condition (31).
2. Implement the rejection-acceptance sampling Algorithm 2 in Matlab and draw 10000 samples, by taking $a = 1$. Use Matlab `hist.m` to plot the histogram. Does its shape resemble the exact density shape?
3. Increase a as much as you can, is there any problem with Algorithm 2? Can you explain why?

7.11 exercise. You are given the following target density

$$\pi(m) \propto \exp\left(-\frac{1}{2\sigma^2}\left(\sqrt{m_1^2 + m_2^2} - 1\right)^2 - \frac{1}{2\delta^2}(m_2 - 1)^2\right),$$

where $\sigma = 0.1$ and $\delta = 1$. Take the proposal density as $q(m) = \mathcal{N}(0, I_2)$, where I_2 is the 2×2 identity matrix.

1. Find a reasonable D, using any means you like, that satisfies condition (31).
2. Implement the rejection-acceptance sampling Algorithm 2 in Matlab and draw 10000 samples. Plot a contour plot for the target density, and you should see the horse-shoe shape, then plot all the samples as dots on top of the contour. Do most of the samples sit on the horse-shoe?

7.3 *Markov chain Monte Carlo*

We have presented a few methods to draw iid samples from a target distribution $q(m)$. The most robust method that works in any dimension is the rejection-acceptance sampling algorithm though it may be slow in practice. In this section, we introduce the Markov chain Monte Carlo scheme which is the most popular sampling approach. It is in general more effective than any methods discussed so far, particularly for complex target density in high dimensions, though it has its own problems. One of them is that we no longer have iid samples but correlated ones. Let us start the motivation by considering the following web-page ranking problem.

Assume that we have a set of Internet websites that may be linked to the others. We represent these sites as nodes and mutual linkings by directed arrows connecting nodes such as in Figure 11. Now you are seeking sites that

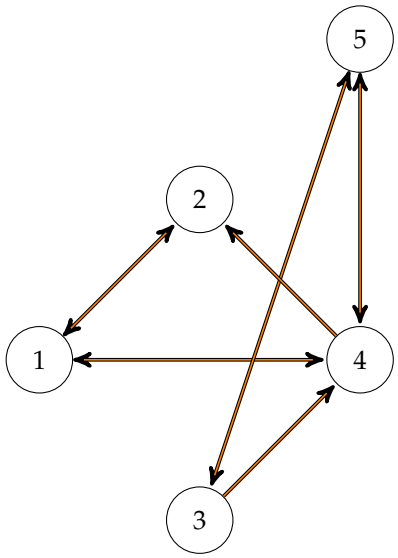


Figure 11: Five internet websites and their connections.

contains a keyword of interest for which all the nodes, and hence websites, contain. A good search engine will show you all these websites. The question is now which website should be ranked first, second, and so on? You may guess that node 4 should be the first one in the list. Let us present a probabilistic method to see whether your guess is correct or not. We first assign the

network of nodes a *transition matrix* P as

$$P = \begin{bmatrix} 0 & 1 & 0 & 1/3 & 0 \\ 1/2 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 \\ 1/2 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 1/2 & 1/3 & 0 \end{bmatrix}. \tag{33}$$

The jth column of P is contains the probability of moving from the jth node to the rest. For example, the first column says that if we start from node 1, we can move to either node 2 or node 4, each with probability $\frac{1}{2}$. Note that we have treated all the nodes equally, that is, the transition probability from one node to other linked nodes is the same (a node is not linked to itself in this model). Note that the sum of each column is 1, meaning that a website must have a link to a website in the network.

Assume we are initially at node 4, and we represent the initial probability density as

$$\pi_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix},$$

that is, we are initially at node 4 with certainty. In order to know the next node to visit, we first compute the probability density of the next state by

$$\pi_1 = P\pi_0,$$

then randomly move to a node by drawing a sample from the (discrete) probability density π_j (see Exercise 7.12). In general, the probability density after k steps is given by

$$\pi_k = P\pi_{k-1} = \ldots = P^k\pi_0, \tag{34}$$

where the jth component of π_k is the probability of moving to the jth node.

Observing (34) you may wonder what happens if k approaches infinity. Assume, on credit, the limit probability density π_∞ exists, then it ought to satisfy

$$\pi_\infty = P\pi_\infty. \tag{35}$$

It follows that π_∞ is the "normalized" eigenvector of P corresponding to unity eigenvalue. Here, normalization means taking that eigenvector and then dividing by the sum of its components so that the result is a probability density.

Figure 12 shows the visiting frequency (blue) for each node after $N = 1500$ moves. Here, visiting frequency of a node is the number of visits to that node divided by N. We expect that numerical visiting frequencies approximate the visiting probabilities in the limit. We confirm this expectation by also plotting the components of π_∞ (red) in Figure 12. By the way, π_{1500} is equal to π_∞ up to machine zero, meaning that a draw from π_N, $N \geq 1500$, is distributed by the limit distribution π_∞. We are now in the position to answer our ranking question. Figure 12 shows that node 1 is the most visited one, and hence should appear at the top of the website list coming from the search engine.

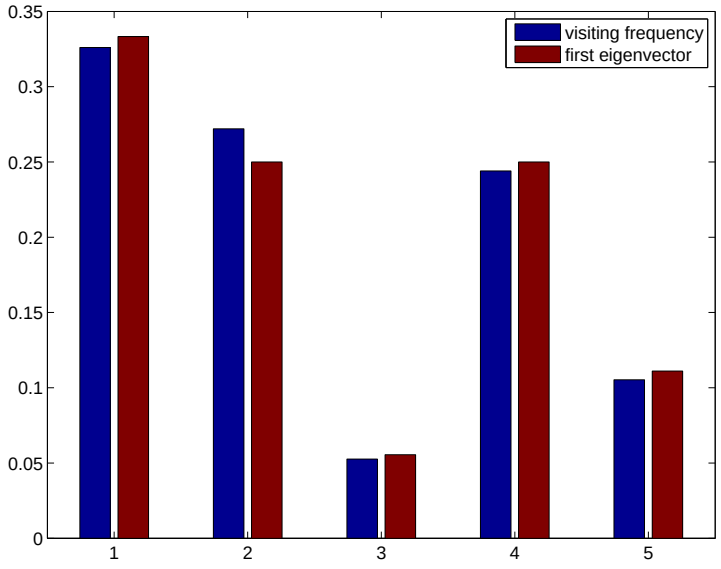


Figure 12: Visiting frequency after $N = 1500$ moves and the first eigenvector.

7.12 exercise. Use the CDF-based sampling algorithm, namely Algorithm 1, to reproduce Figure 12. Compare the probability density π_{1500} with the limit density, are they the same? Generate 5 figures corresponding to starting nodes $1, \ldots, 5$, what do you observe?

7.13 exercise. Using the above probabilistic method to determine the probability that the economy, as shown in Figure 13, is in recession.

The limit probability density π_∞ is known as the *invariant density*. Invariance here means that the action of P on π_∞ returns exactly π_∞. In summary, we start with the transition matrix P and then find the invariant probability density by taking the limit. Drawings from π_k are eventually distributed as the invariant density (see Exercise 7.12). A Markov chain Monte Carlo (MCMC)

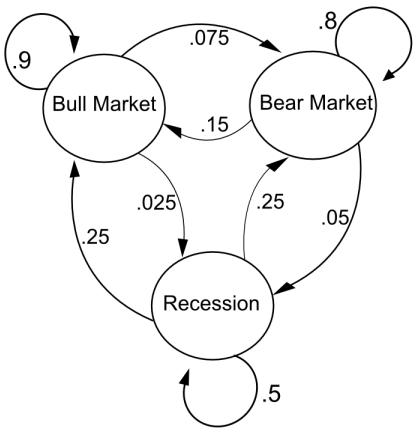


Figure 13: Economy states extracted from `http://en.wikipedia.org/wiki/Markov_chain`

method reverses the process. In particular, we start with a target density and look for the transition matrix such that MCMC samples are eventually distributed as the target density. Before presenting the popular Metropolis-Hastings MCMC method, we need to explain what "Markov" means.

Let us denote $m_0 = 4$ the initial starting node, then the next move m_1 is either 1 or 2 or 5 since the probability density is

$$\pi_1(m_1|m_0) = [1/3, 1/3, 0, 0, 1/3]^T, \tag{36}$$

where we have explicitly pointed out that π_1 is a conditional probability density given known initial state m_0. Similarly, (34) should have been written as

$$\pi_k\left(m_k|m_0\right) = P\pi_{k-1}\left(m_{k-1}|m_0\right) = \ldots = P^k\pi_0.$$

Now assume that we know all states up to m_{k-1}, say $m_{k-1} = 1$. It follows that

$$\pi_{k-1}\left(m_{k-1}|m_{k-2}, \ldots, m_0\right) = [1, 0, 0, 0, 0]^T,$$

since we know m_{k-1} for certain. Consequently,

$$\pi_k\left(m_k|m_{k-1}, \ldots, m_0\right) = P\pi_{k-1}\left(m_{k-1}|m_{k-2}, \ldots, m_0\right) = P\left[1, 0, 0, 0, 0\right]^T$$

regardless of the values of the states $m_{k-2}, \ldots, m_0$. More generally,

$$\pi_k\left(m_k | m_{k-1}, \ldots, m_0\right) = \pi_k\left(m_k | m_{k-1}\right),$$

which is known as *Markov property*, namely, "tomorrow depends on the past only through today".

7.14 definition. A collection $\{m_0, m_1, \ldots, m_N, \ldots\}$ is called *Markov chain* if the distribution of m_k depends only on the immediate previous state m_{k-1}.

Next, let us introduce some notations to study MCMC methods.

7.15 definition. We call the probability of m_k in A starting from m_{k-1} as the *transition probability* and denote it as $P\left(m_{k-1}, A\right)$. With an abuse of notation, we introduce the *transition kernel* $P\left(m_{k-1}, m\right)$ such that

What does $P\left(m_{k-1}, dm\right)$ mean?

$$P\left(m_{k-1}, A\right) \stackrel{\text{def}}{=} \int_A P\left(m_{k-1}, m\right)\, dm = \int_A P\left(m_{k-1}, dm\right).$$

Clearly $P\left(m, S\right) = \int_S P\left(m, p\right)\, dp = 1$.

7.16 example. Let $m_{k-1} = 4$ in our website ranking problem, then the probability kernel $P\left(m_{k-1} = 4, m\right)$ is exactly the probability density in (36).

What is the transition probability $P\left(m_{k-1} = 4, m_k = 1\right)$?

7.17 definition. We call $\mu\left(dm\right) = \pi\left(m\right) dm$ the invariant distribution and $\pi\left(m\right)$ invariant density of the transition probability $P\left(m_{k-1}, dm\right)$ if

$$\mu\left(dm\right) = \pi\left(m\right) dm = \int_S P\left(p, dm\right) \pi\left(p\right)\, dp. \tag{37}$$

7.18 example. The discrete version of (37), applying to our website ranking problem, reads

$$\pi_\infty\left(j\right) = \sum_{k=1}^{5} P\left(j, k\right) \pi_\infty\left(k\right) = P(j, :)\pi_\infty, \quad \forall j = 1, \ldots, 5,$$

which is exactly (35).

7.19 definition. A Markov chain $\{m_0, m_1, \ldots, m_N, \ldots\}$ is *reversible* if

$$\pi\left(m\right) P\left(m, p\right) = \pi\left(p\right) P\left(p, m\right). \tag{38}$$

The reversibility relation (38) is also known as *detailed balanced equation*. You can think of the reversibility saying that the probability of moving from m to p is equal to the probability of moving from p to m.

7.20 exercise. What is the discrete version of (38)? Does the transition matrix in the website ranking problem satisfy the reversibility? If not, why? How about the transition matrix in Exercise 7.13.

The reversibility of a Markov chain is useful since we can immediately conclude that $\pi(m)$ is its invariant density.

7.21 proposition. *If the Markov chain $\{m_0, m_1, \ldots, m_N, \ldots\}$ is reversible with respect to $\pi(m)$, then $\pi(m)$ is the invariant density.*

Proof. We need to prove (37), but it is straightforward since

$$\int_S \pi(p)\, P(p, dm)\; dp \overset{\text{reversibility}}{=} \pi(m)\, dm \int_S P(m, p)\; dp = \pi(m)\, dm.$$

□

The above discussion shows that if a Markov chain is reversible then eventually the states in the chain are distributed by the underlying invariant distribution. A question you may ask is how to construct a transition kernel such that reversibility holds. This is exactly the question Markov chain Monte Carlo methods are designed to answer. Let us now present the Metropolis-Hastings MCMC method in Algorithm 3.

Algorithm 3 Metropolis-Hastings MCMC Algorithm

Choose initial m_0
for $k = 0, \ldots, N$ **do**

1. Draw a sample p from the proposal density $q(m_k,\, p\,)$
2. Compute $\pi(p)$, $q(m_k, p)$, and $q(p, m_k)$
3. Compute the acceptance probability

$$\alpha(m_k, p) = \min\left\{1, \frac{\pi(p) q(p, m_k)}{\pi(m_k) q(m_k, p)}\right\}$$

4. **Accept** and set $m_{k+1} = p$ with probability $\alpha(m_k, p)$. Otherwise, **reject** and set $m_{k+1} = m_k$

end for

The idea behind the Metropolis-Hastings Algorithm 3 is very similar to that of rejection-acceptance sampling algorithm. That is, we first draw a sample from an "easy" distribution $q(m_k, p)$, then make correction so that it is distributed more like the target density $\pi(p)$. However, there are two main

differences. First, the proposal distribution $q(m_k, p)$ is a function of the last state m_k. Second, the acceptance probability involves both the last state m_k and the proposal move p. As a result, a chain generated from Algorithm 3 is in fact a Markov chain.

What remains to be done is to show that the transition kernel of Algorithm 3 indeed satisfies the reversibility condition (38). This is the focus of the next proposition.

7.22 proposition. *Markov chains generated by Algorithm 3 are reversible.*

Proof. We proceed in two steps. In the first step, we consider the case in which the proposal p is accepted. Denote B as the event of accepting a draw q (or the acceptance event). Following the same proof of Proposition 7.9, we have

$$\mathbb{P}[B|p] = \alpha(m_k, p),$$

Make sure you see this!

leading to

$$\pi(B, p) = \mathbb{P}[B|p]\, \pi_{\text{prior}}(p) = \alpha(m_k, p) q(m_k, p),$$

which is exactly $P(m_k, p)$, the probability density of the joint event of drawing p from $q(m_k, p)$ and accept it, starting from m_k. It follows that the reversibility holds since

$$\begin{aligned}
\pi(m_k) P(m_k, p) &= \pi(m_k) q(m_k, p) \min\left\{1, \frac{\pi(p) q(p, m_k)}{\pi(m_k) q(m_k, p)}\right\} \\
&= \min\{\pi(m_k) q(m_k, p), \pi(p) q(p, m_k)\} \\
&= \min\left\{\frac{\pi(m_k) q(m_k, p)}{\pi(p) q(p, m_k)}, 1\right\} \pi(p) q(p, m_k) \\
&= \pi(p) P(p, m_k).
\end{aligned}$$

In the second step, we remain at m_k, i.e., $m_{k+1} = m_k$, then the reversibility is trivially satisfied no matter what the transition kernel $P(m_k, p)$ is. This is the end of the proof. □ *Explain why*

7.23 exercise. What is the probability of staying put at m_k?

As you can see the Metropolis-Hastings algorithm is simple and elegant, but provides us a reversible transition kernel, which is exactly what we are looking for. The keys behind this are Steps 3 and 4 in Algorithm 3, known as Metropolized steps. At this point, we should be able to implement the algorithm except for one small detail: what should we choose for the proposal

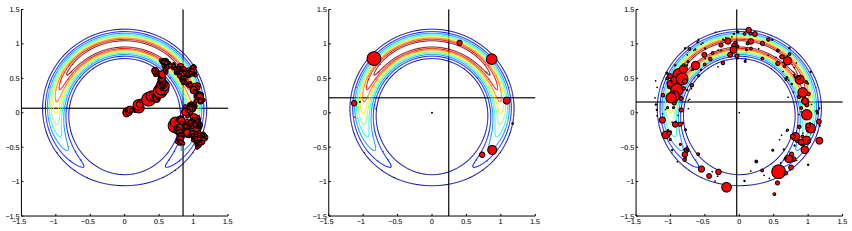
density? Let us choose the following Gaussian kernel

$$q(m, p) = \frac{1}{\sqrt{2\pi\gamma^2}} \exp\left(-\frac{1}{2\gamma^2} \|m - p\|^2\right),$$

which is the most popular choice. Metropolis-Hastings algorithm with above isotropic Gaussian proposal is known as *Random Walk Metropolis-Hastings* (RWMH) algorithm. For this particular method, the acceptance probability is very simple, i.e.,

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

We are now in the position to implement the method. For concreteness, we apply the RWMH algorithm on the horse-shoe shape in Exercise 7.11. We take the origin as the starting point m_0. Let us first be conservative by choosing a small proposal variance $\gamma^2 = 0.02^2$ so that the proposal p is very close to the current state m_k. In order to see how the MCMC chain evolves, we plot each state m_k as a circle (red) centered at m_k with radius proportional to the number of staying-puts. Figure 14(a) shows the results for $N = 1000$. We observe that the chain takes about 200 MCMC simulations to enter the high probability density region. This is known as *burn-in* time in MCMC literature, which tells us how long a MCMC chain takes to start exploring the density. In other words, after the burn-in time, a MCMC begins to distribute like the target density. As can be seen, the chain corresponding to small proposal variance γ^2 explores the target density very slowly. If we approximate the average acceptance rate by taking the ratio of the number of accepted proposal over N, it is 0.905 for this case. That is, almost all the proposals p are accepted, but exploring a very small region of high probability density.



(a) $\gamma = 0.02$ (b) $\gamma = 5$ (c) $\gamma = 0.5$

Figure 14: RWMH with different proposal variance γ^2.

Let us now increase the proposal stepsize γ to 5, and we show the corresponding chain in Figure 14(b). This time, the chain immediately explores the

target density without any burn-in time. However, it does so in an extremely slow manner. Most of the time the proposal p is rejected, resulting in a few big circles in Figure 14(b). The average acceptance rate in this case is 0.014, which shows that most of the time we reject proposals p.

The results in Figures 14(a) and 14(b) are two extreme cases, both of which explore the target density in a very lazy manner since the chain either accepts all the small moves with very high acceptance rate or rejects big moves with very low acceptance rate. This suggests that there must be an *optimal* acceptance rate for the RWMH algorithm. Indeed, one can show that the optimal acceptance rate is 0.234 [5]. For our horse-shoe target, it turns out that the corresponding optimal stepsize is approximately $\gamma = 0.5$. To confirm this, we generate a new chain with this stepsize, again with $N = 1000$, and show the result in Figure 14(c). As can be seen, the samples spread out nicely over the horse-shoe.

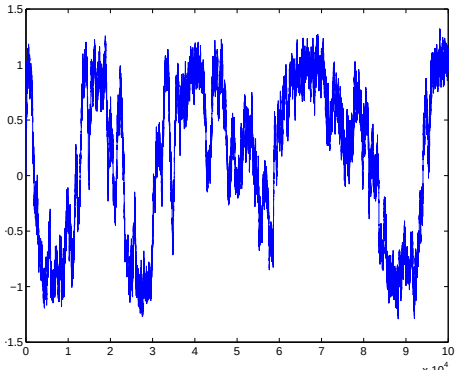
We have judged the quality and convergence of a MCMC chain by looking at the scatter plot of the samples. Another simple approach is to look at the trace plot of components of m. For example, we show the trace plot of the first component in Figure 15 for the above three stepsizes. The rule of thumb is that a Markov chain is considered to be good if its trace plot is close to a white noise one, a "fuzzy worm", in which all the samples are completely uncorrelated. Based on this criteria, we again conclude that $\gamma = 0.5$ is the best compared to the other two extreme cases.

Plot a trace plot for a one dimensional Gaussian white noise to see how it looks like!

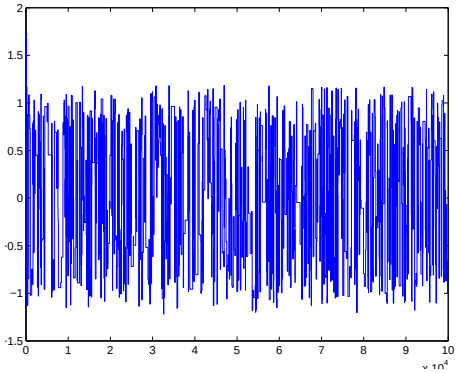
Nevertheless, the above two simple criteria are neither rigorous nor possible in high dimensions. This observation immediately reminds us the strong law of large number in computing the mean and its dimension-independent error analysis using the central limit theorem. Since the target is symmetric about the vertical axis, the first component of the mean must be zero. Let us use the strong law of large number to estimate the means for the above three stepsizes and show them as cross signs in Figures 14(a), 14(b), and 14(c). As can be seen and expected, the sample mean for the optimal stepsize is the most accurate, though it is not exactly on the vertical axis since $\overline{m}_1 = -0.038$. This implies that the sample size of $N = 1000$ is small. If we take $N = 10000$, the sample mean gives $\overline{m}_1 = 0.003$, signifying the convergence when N increases.

Take $N = 100000$ for the optimal stepsize case, and again compute the sample mean using `BayesianMCMC.m`. *Is the sample mean better? If not, why?*

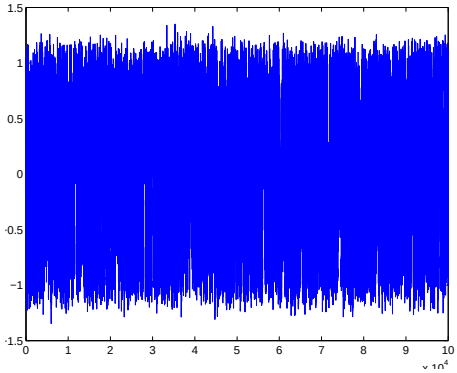
However, the application of LLN and CLT is very limited for Markov chains since they don't provide iid samples. Indeed, as in the above Markov chain theory, the states of the chain eventually identically distributed by $\pi(m)$, but they are always correlated instead of independent since any state in the chain depends on the previous one. What we could hope for is that the current state is effectively independent from its kth previous state. In that case, the effective number of iid samples is N/k, and the mean square error, by the central limit theorem, decays as $\sqrt{k/N}$. As the result, if k is large, the decay rate is very slow. How to estimate k is the goal of the *autocorrelation* study, as we now discuss.



(a) $\gamma = 0.02$



(b) $\gamma = 5$



(c) $\gamma = 0.5$

Figure 15: Trace plots of the first component of m with different γ^2.

We shall compute the autocorrelation for each component of m separately, therefore, without loss of generality, assume that $m \in \mathbb{R}^1$ and that the Markov chain $\{m_j\}_{j=0}^N$ has zero mean. Consider the following discrete convolution quantities

$$c_k = \sum_{j=0}^{N-k} m_{j+k} m_j, \quad , k = 0, \ldots, N-1,$$

and define the autocorrelation of m with *lag* k as

$$\hat{c}_k = \frac{c_k}{c_0}, \quad k = 0, \ldots, N-1.$$

If $\hat{c}_k$ is zero, then we say that the correlation length of the Markov chain is approximately k, that is, any state m_j is considered to be insignificantly correlated to m_{j-k} (and hence any state before m_{j-k}), and to m_{j+k} (and hence any state after m_{j+k}). In other words, every kth sample point can be considered to be approximately independent. Note that this is simply a heuristic and one should be aware that independece implies un-correlation but not vice versa.

Let us now approximately compute the correlation length for three Markov chains corresponding to $\gamma = 0.02$, $\gamma = 0.5$, and $\gamma = 5$, respectively, with $N = 100000$. We first subtract away the sample mean as

$$z_j = m_j - \frac{1}{N+1} \sum_{i=0}^{N} m_i.$$

Then, we plot the autocorrelation functions $\hat{c}_k$ for each component of the zero mean sample $\{z_j\}_{j=0}^N$ in Figure 16. As can be observed, the autocorrelation length for the chain with optimal stepsize $\gamma = 0.5$ is about $k = 100$, while the others are much larger (not shown here). That is, every 100th sample point can be considered to be independent for $\gamma = 0.5$. The case with $\gamma = 0.02$ is the worst, indicating slow move around the target density. The stepsize of $\gamma = 5$ is better, but so big that the chain remains at each state for a long period of time, and hence autocorrelation length is still significant relatively to that of $\gamma = 0.5$.

Extensive MCMC methods including improvements on the standard RWMH algorithm can be found in [6]. Let us introduce two simple modifications through the following two exercises.

7.24 exercise. Consider the following target density

$$\pi(m) \propto \exp\left(-\frac{1}{2\delta^2}\|m\|^2 - \frac{1}{2\sigma^2}\|y - h(m)\|^2\right), \tag{39}$$

where

$$h(m) = \begin{bmatrix} m_1^2 - m_2 \\ m_2/5 \end{bmatrix}, \quad y = \begin{bmatrix} -0.2 \\ 0.1 \end{bmatrix}.$$

Take $\delta = 1$ and $\sigma = 0.1$.

1. Modify `BayesianMCMC.m` to simulate the target density in (39) with $N = 5000$.

2. Tune the proposal stepsize γ so that the average acceptance probability is about 0.234. Show the scatter, trace, and autocorrelation plots for the optimal stepsize.

7.25 exercise. So far the proposal density $q(m, p)$ is isotropic and independent of the target density $\pi(m)$. For anisotropic target density, isotropic proposal is not a good idea, intuitively. The reason is that the proposal is distributed equally in all directions, whereas it is not in the target density. A natural idea is to shape the proposal density to make it locally resemble the target density. A simple idea in this direction is to linearize $h(m)$, and then define the proposal density as

$$q(m_k, p) \propto \exp\left(-\frac{1}{2\delta^2}\|p\|^2 - \frac{1}{2\sigma^2}\|y - h(m_k) - \nabla h(m_k)(p - m_k)\|^2\right),$$

1. Determine $H(m_k)$ such that $q(m_k, p) = \mathcal{N}\left(m_k, H(m_k)^{-1}\right)$, by keeping only the quadratic term in $p - m_k$.

2. Modify `BayesianMCMC.m` to simulate the target density in (39) using the proposal density $q(m_k, p) = \mathcal{N}\left(m_k, H(m_k)^{-1}\right)$. Show the scatter, trace, and autocorrelation plots. Is it better than the isotropic proposal density?

7.26 exercise. Another idea to improve the standard RWMH algorithm is by adaptation. Let's investigate a simple adaptation strategy. Use the resulting sample in Exercise 7.24 to compute the empirical covariance $\hat{\Gamma}$, then use it to construct the proposal density $q(m, p) = \mathcal{N}(m, \hat{\Gamma})$. Show the scatter, trace, and autocorrelation plots. Is it better than the isotropic proposal density?

8 Matlab codes

A set of Matlab codes that can be used to reproduce most of the figures in the note can be downloaded from

`http://users.ices.utexas.edu/~tanbui/teaching/Bayesian`

References

[1] D. Calvetti and E. Somersalo. *Introduction to Bayesian Scientific Computing: Ten Lectures on Subjective Computing*. Springer, New York, 2007.

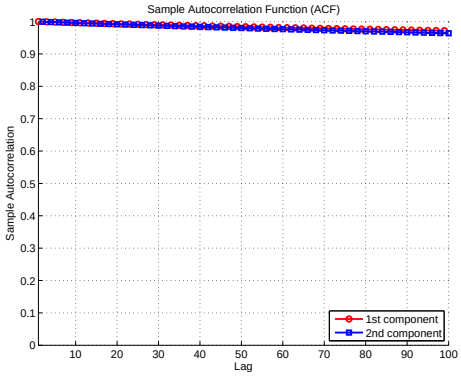
[2] Jari Kaipio and Erkki Somersalo. *Statistical and Computational Inverse Problems*, volume 160 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 2005.

[3] Todd Arbogast and Jerry L. Bona. *Methods of Applied Mathematics*. University of Texas at Austin, 2008. Lecture notes in applied mathematics.

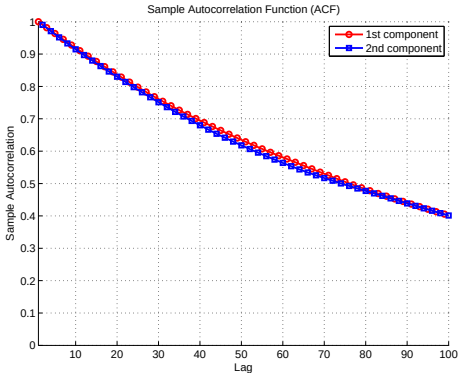
[4] Rick Durret. *Probability: theory and examples*. Cambidge University Press, 2010.

[5] Gareth O. Roberts and Jeffrey S. Rosenthal. Optimal scaling for various metropolis-hastings algorithms. *Statistical Science*, 16(4):pp. 351–367, 2001.

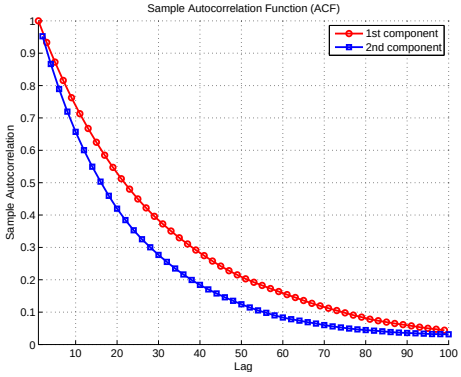
[6] Christian P. Robert and George Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.



(a) $\gamma = 0.02$



(b) $\gamma = 5$



(c) $\gamma = 0.5$

Figure 16: Autocorrelation function plot for both components of m with different γ^2.