QUO VADIS, BAYESIAN IDENTIFICATION?

RUDOLF KULHAVÝ AND PETYA IVANOVA

Honeywell Technology Center Europe, Prague and
Institute of Information Theory and Automation, AS CR, Prague[*]
Tel: +420 2 6605 2313, Fax: +420 22 688 4903
E-mail: kulhavy@htc.honeywell.cz, ivanova@htc.honeywell.cz

[*] The work was supported in part by Grant A2075603 of the Academy of Sciences of CR and Grant 102/97/0466 of the Grant Agency of CR.

SUMMARY

The Bayesian identification of non-linear, non-Gaussian, non-stationary or non-parametric models is notoriously known as computer-intensive and not solvable in a closed form. The paper outlines three major approaches to approximate Bayesian estimation, based on locally weighted smoothing of data, iterative and non-iterative Monte Carlo simulation and direct approximation of an information "distance" between the empirical and model distributions of data. The information-based view of estimation is used throughout to give more insight into the methods and show their mutual relationship.

Key words: Nonlinear estimation, Bayesian methods, local regression, Monte Carlo simulation, information geometry.

1. INTRODUCTION

In 1981, V. Peterka published a monograph chapter *Bayesian Approach to System Identification*[1] where he presented—on just 65 pages—a self-contained tutorial on Bayesian identification. In this chapter, he outlined the fundamentals of Bayesian calculus, Bayesian view of modelling, general philosophy of Bayesian estimation and its application to the multivariate linear ARX model, Bayesian derivation of Kalman filter, Bayesian interpretation of exponential forgetting and Bayesian classification of model structures.

Viewed from today's perspective, little needs to be changed on this work. Peterka's attention to 'details that matter' and his gift of boiling technicalities down to their natural meaning has made the chapter a sort of classics. A couple of generations of research students as well as practising engineers have been using it as an entrance gate to the Bayesian world.

The fact that the chapter looks so complete and comprehensive has one more reason—a careful choice of the topics treated. All essential what can be solved analytically in a closed form in the Bayesian framework is there. What is not treated—and what has become a challenge to Peterka's students and disciples—is identification of models resisting to any analytic solution, namely *non*-stationary, *non*-linear, *non*-Gaussian, or *non*-parametric models.

The lack of an established way of dealing with such models gave birth to a multitude of different approaches, methods and algorithms. The present paper indicates that most of these results are based upon a few general principles of statistical inference. We believe that recognition of these principles brings more insight into the wealth of existing solutions as well as better understanding where the future development is likely to go.

The results presented in the paper are stated without proofs, which can be found in the references. Our choice of topics is necessarily subjective and incomplete, especially with respect to the dramatic development in Bayesian statistics in the last two decades. A number of powerful algorithms have been invented (and reinvented) during this period, stimulated significantly by the requirements of practice to manage very complex models and extremely large datasets.

2. GENERAL REGRESSION MODEL

The traditional view of Bayesian estimation concentrates on the posterior density representing a total description of the parameter uncertainty. Alternatively, Bayesian estimation can be regarded as a process of computing an information "distance" be-

tween the empirical and model distributions of data.[2] The present section makes a summary of both the approaches.

2.1. Model Class

Consider a system on which two sequences of continuous random variables are measured, $U^{N+m} = (U_1, \mathrm{K}\, , U_{N+m})$, $Y^{N+m} = (Y_1, \mathrm{K}\, , Y_{N+m})$, where U_k and Y_k, for $k = 1, \mathrm{K}\, , N+m$ take values in subsets $\boldsymbol{U}$ and $\boldsymbol{Y}$ of $\boldsymbol{R}^{\dim U}$ and $\boldsymbol{R}^{\dim Y}$, respectively. U_k is defined as a directly manipulated input to the system at time k. Y_k is the output, i.e., response of the system at time k to the past history of data represented by the sequences Y^{k-1} and U^k. The above sequences form a *sample* of data. A sequence of observed values $y^{N+m} = (y_1, \mathrm{K}\, , y_{N+m})$, $u^{N+m} = (u_1, \mathrm{K}\, , u_{N+m})$ is called a *realization* of the sample Y^{N+m}, U^{N+m} or an *observed sample.*

Suppose that the output values Y_k depend on the past data Y_{k-m}^{k-1} and U_{k-m}^{k} only through a known vector function, *regressor* $Z_k = z(U^k, Y^{k-1})$, taking values in a subset $\boldsymbol{Z}$ of $\boldsymbol{R}^{\dim Z}$. More precisely, if the dependence is described through a conditional probability density function $s_k(y_k \mid y^{k-1}, u^k)$ of Y_k given $Y^{k-1} = y^{k-1}$ and $U^k = u^k$, the assumption reads

$$s_k(y_k \mid y^{k-1}, u^k) = s_k(y_k \mid z_k)$$

for $k = m+1, \mathrm{K}\, , N+m$. In addition, we assume that the conditional density of Y_k given $Z_k = z_k$ is identical, $s_k(y_k \mid z_k) = s(y_k \mid z_k)$, for all k. Finally, it is assumed that (y_N, z_N) is recursively computable given its last value (y_{N-1}, z_{N-1}) and the latest data (y_N, u_N), i.e., there exists a map F such that

$$(y_N, z_N) = F((y_{N-1}, z_{N-1}), (y_N, u_N)) .$$

In the sequel we assume that the density $s(y \mid z)$ comes from a given family

$$\boldsymbol{S} = \{ s_\theta(y \mid z) : \theta \in \boldsymbol{T} \}$$

parametrized by a vector parameter θ taking values in a subset $\boldsymbol{T}$ of $\boldsymbol{R}^{\dim \theta}$. To simplify introduction of information measures in Section 2.3, we restrict ourselves to the case that $s_\theta(y \mid z) > 0$ for all $(y, z) \in \boldsymbol{Y} \times \boldsymbol{Z}$ and all $\theta \in \boldsymbol{T}$.

2.2. Bayesian Estimation

Generally speaking, the dependence of the input U_k on the past data Y^{k-1}, U^{k-1} and the parameter θ can be expressed through a conditional density $\gamma_k(u_k \mid y^{k-1}, u^{k-1}, \theta)$. In most cases of practical interest, we may adopt the simplifying assumption, introduced by Peterka[1] as 'natural conditions of control', that the only information about θ

used for computation of the new input is the information contained in the past data. More precisely, we assume that at $k = m+1,\mathrm{K}\,, N+m$

$$\gamma_k(u_k \mid y^{k-1}, u^{k-1}, \theta) = \gamma_k(u_k \mid y^{k-1}, u^{k-1}).$$

Provided the unknown parameter θ is interpreted as a random variable Θ, it is possible then to describe its uncertainty through the posterior density conditional on the observed samples y^{N+m}, u^{N+m}

$$p_N(\theta) \overset{def}{=} p(\theta \mid y^{N+m}, u^{N+m}).$$

Here the subscript N indicates conditioning on N data points $(y_{m+1}, z_{m+1}),\mathrm{K}\,,$ (y_{N+m}, z_{N+m}). Given a prior density conditional on available prior information and m initial values y^m, u^m (the latter is often not considered in practice)

$$p_0(\theta) \overset{def}{=} p(\theta \mid y^m, u^m),$$

the posterior density follows by application of Bayes theorem and natural conditions of control

$$p_N(\theta) \propto p_0(\theta) \prod_{k=m+1}^{N+m} s_\theta(y_k \mid z_k) \qquad (1)$$

where $\propto$ stands for equality up to a normalizing factor.

2.3. Bayesian Estimation via Inaccuracy

The Bayesian estimation can be regarded alternatively as measuring an information "distance" between the empirical density of data and densities within the *model* class $\mathbf{S}$. The information-based view yields a good starting point for all subsequent approximations.[2]

Given the observed sample y^{N+m}, u^{N+m}, we define a joint *empirical* density of (Y, Z) as

$$r_N(y, z) \overset{def}{=} \frac{1}{N} \sum_{k=m+1}^{N+m} \delta(y - y_k, z - z_k)$$

where $\delta(y, z)$ is a Dirac function satisfying $\delta(y, z) = 0$ for $y \neq 0$ or $z \neq 0$ and

$$\iint_{\mathbf{Y}\times\mathbf{Z}} \delta(y, z)\, \mathrm{d}y\, \mathrm{d}z = 1.$$

Next we introduce a (conditional) inaccuracy[2] of the empirical density $r_N(y, z)$ relative to the model density $s_\theta(y \mid z)$

$$K(r_N : s_\theta) =_{\text{def}} -\iint r_N(y,z) \log s_\theta(y \mid z)\, \mathrm{d}y\, \mathrm{d}z\,.$$

Combining the definitions of the empirical density and conditional inaccuracy, we obtain an alternative expression of the posterior density

$$p_N(\theta) \propto p_0(\theta) \exp\bigl(-N\, K(r_N : s_\theta)\bigr) \tag{2}$$

which distinguishes clearly the basic ingredients of Bayesian estimation—the prior density of the parameter θ, the amount of data N, the empirical (joint) density $r_N(y,z)$ and the theoretical (conditional) density $s_\theta(y \mid z)$.

The expression (2) can be further simplified if the prior density is chosen in the following *conjugate*[1,2] form

$$p_0(\theta) \propto \exp\bigl(-\nu_0\, K(\rho_0 : s_\theta)\bigr) \tag{3}$$

where $\rho_0(y,z)$ stands for a "prior" density of (Y,Z) built upon prior information and m initial values $y^m,\ u^m$. The nonnegative scalar ν_0 can be interpreted as the number of actual or fake observations $\rho_0(y,z)$ is built on. The density (3) then becomes a posterior density for a uniform prior distribution on θ and ν_0 observations with the empirical density $\rho_0(y,z)$. In general, the scalar ν_0 need not be integer. Its practical meaning is simply to put an appropriate weight on prior knowledge expressed through $\rho_0(y,z)$.

With the conjugate prior density (3), the posterior density (2) takes the form

$$p_N(\theta) \propto \exp\bigl(-\nu_N\, K(\rho_N : s_\theta)\bigr) \tag{4}$$

where the scalar $\nu_N > 0$ counts the total "number" of data and ρ_N is a convex combination of the "prior" density ρ_0 and the empirical density r_N

$$\begin{aligned} \nu_N &= \nu_0 + N\,, \\ \rho_N(y,z) &= \frac{\nu_0}{\nu_N}\, \rho_0(y,z) + \frac{N}{\nu_N}\, r_N(y,z)\,. \end{aligned} \tag{5}$$

2.4. Bayesian Prediction

In many cases, the ultimate goal of Bayesian inference is model-based prediction rather than pure parameter estimation. The predictive density of Y given $Z = z$ conditional on the previous observations $(y_{m+1}, z_{m+1}), \mathrm{K}, (y_{N+m}, z_{N+m})$ follows by elementary rules of probability calculus

$$s_N(y \mid z) = \int s_\theta(y \mid z)\; p_N(\theta)\, \mathrm{d}\theta\ .$$

After substituting for $p_N(\theta)$ from (4), we obtain the following compact expression

$$s_N(y \mid z) \propto \int \exp\bigl(-(\nu_N + 1)\, K(\rho_{N;y,z} : s_\theta)\bigr)\, \mathrm{d}\theta \tag{6}$$

where the $\rho_{N;y,z}$ is an empirical density updated by the data pair (y, z)

$$\rho_{N;\, y^*,\, z^*}(y,z) = \frac{\nu_N}{\nu_N + 1}\ \rho_N(y,z) + \frac{1}{\nu_N + 1}\ \delta(y - y^*, z - z^*)\,. \tag{7}$$

2.5. Linear Normal Regression

For a linear normal ARX model with the sampling density

$$s_\theta(y \mid z) = \frac{1}{\sqrt{2\pi\sigma^2}}\ \exp\left(-\frac{1}{2\sigma^2}(y - \theta' z)^2\right),$$

the conditional inaccuracy can be calculated analytically

$$K(\rho_N : s_\theta) = \frac{1}{2}\ \log 2\pi\sigma^2 + \frac{1}{2\sigma^2}\ V_N + \frac{1}{2\sigma^2}\ (\theta - \hat{\theta}_N)' C_N (\theta - \hat{\theta}_N)$$

using the statistics

$$\begin{aligned} \hat{\theta}_N &= C_N{}^{-1} E_N(ZY)\,, \\ V_N &= E_N(Y^2) - E_N(YZ')\, C_N{}^{-1} E_N(ZY)\,, \\ C_N &= E_N(ZZ')\,. \end{aligned}$$

The expectation $E_N(\cdot)$ is taken with respect to the density $\rho_N(\theta)$ defined by the convex combination (5) of the "prior" density ρ_0 and the empirical density r_N

$$\begin{aligned} E_N(ZY) &= \iint y\, z\, \rho_N(y,z)\, \mathrm{d}y\, \mathrm{d}z \\ &= \frac{\nu_0}{\nu_0 + N}\ \iint y\, z\, \rho_0(y,z)\, \mathrm{d}y\, \mathrm{d}z + \frac{N}{\nu_0 + N}\ \frac{1}{N} \sum_{k=m+1}^{N+m} y_k\, z_k\,. \end{aligned}$$

The prior matrix C_0 is supposed positive definite. Note that the expectation with respect to the "prior" density ρ_0 is what distinguishes the Bayesian inference from the least-squares solution.

2.6. General Regression

For non-linear, non-Gaussian, non-stationary, or non-parametric models $s(y \mid z)$, the conditional inaccuracy $K(\rho_N : s_\theta)$ can rarely be computed analytically. In the following sections, we show three approaches to approximate Bayesian inference.

1. *Local regression* captures only the local behaviour of data. The empirical distribution of data is replaced with a locally weighted distribution that can be fitted with a simpler and easier-to-estimate model distribution.

2. *Monte Carlo simulation* replaces the posterior distribution with a large enough sample. The target features of the posterior distribution are explored via the corresponding features of the sample.

3. *Information geometry* approximates directly the information "distance" between the empirical and model distributions of data. Additional constraints can be imposed on the approximation, in terms of information inequality.

3. LOCAL-IN-TIME REGRESSION

The local regression concept is actually used very often in practice—to address the situation when the parameter θ varies in time. Identification of a non-stationary system can be addressed in two basic ways. Either a global model describing the system behaviour at all time instants is built, or a local model capturing the system behaviour around the time instant of interest is fitted to the data. The latter approach is much easier to implement and usually sufficient for the purpose of response prediction.

3.1. Exponential Discounting

Intuitively, in order to focus on the recent data, we must assign the older data smaller weights. The simplest choice is to make the weight on the data point (y_k, z_k) exponentially decreasing with its age $N+m-k$. More specifically, the data point (y_k, z_k) is assigned the weight λ^{N+m-k} where $\lambda \in (0,1)$ acts as a discounting or "forgetting" factor. With this choice, the statistic (ν_N, ρ_N) modifies as follows,

$$\begin{aligned} \nu_N &= \nu_0 + \sum_{k=m+1}^{N+m} \lambda^{N+m-k}, \\ \rho_N(y,z) &= \frac{\nu_0}{\nu_N}\rho_0(y,z) + \frac{\nu_N - \nu_0}{\nu_N} \frac{\sum_{k=m+1}^{N+m} \lambda^{N+m-k}\ \delta(y-y_k, z-z_k)}{\sum_{k=m+1}^{N+m} \lambda^{N+m-k}}. \end{aligned} \qquad (8)$$

The density $\rho_N(y,z)$ is a convex combination of the "prior" density $\rho_0(y,z)$ and a *time-discounted* empirical density. The weight on the prior part approaches

$$\frac{\nu_0}{\nu_N} \to \frac{\nu_0}{\nu_0 + \frac{1}{1-\lambda}} \text{ as } \mathrm{N} \to \infty.$$

The prior information thus effectively regularizes estimation even in case that the data does not carry temporarily enough information.

The scalar ν_N and the density $\rho_N(y,z)$ can be updated recursively

$$\begin{aligned} \nu_k &= \lambda\nu_{k-1} + (1-\lambda)\nu_0 + 1\,, \\ \rho_k(y,z) &= \frac{\lambda\nu_{k-1}}{\nu_k}\,\rho_{k-1}(y,z) + \frac{(1-\lambda)\,\nu_0}{\nu_k}\,\rho_0(y,z) + \frac{1}{\nu_k}\,\delta(y-y_k, z-z_k)\,. \end{aligned}$$

Note again that due to the regularization the prior information is not lost as a result of discounting.

The exponential discounting was introduced first in the forecasting literature.[3,4] Later it has become a standard tool in adaptive control and signal processing. It allows multiple interpretations—it can be regarded as filtering of data,[5] flattening of posterior density[1] or as a result of minimization of Kullback-Leibler divergence[6,7].

3.2. Kernel-Based Discounting

The idea of discounting the data according to their relative importance can be extended so as to use a general weighting profile. Consider a kernel function $K(x)$ that equals to 1 at $x = 0$ and decreases to 0 as $|x|$ increases. Examples of such functions include the Gaussian kernel $K(x) = \exp(-x^2)$ or the Epanechnikov kernel $K(x) = \max(1 - x^2, 0)$.

Provided k^* is the time instant of interest, we assign to the data point (y_k, z_k) the weight $K((k-k^*)/h)$. The scalar $h > 0$ is a smoothing factor that determines how quickly the weight on the data point approaches zero as $|k-k^*|$ increases.

With the above choice of the weighting profile, the statistic (ν_N, ρ_N) takes the form

$$\begin{aligned} \nu_N &= \nu_0 + \sum_{k=m+1}^{N+m} K\left(\tfrac{k-k^*}{h}\right), \\ \rho_N(y,z) &= \frac{\nu_0}{\nu_N}\,\rho_0(y,z) + \frac{\nu_N - \nu_0}{\nu_N}\,\frac{\sum_{k=m+1}^{N+m} K\left(\tfrac{k-k^*}{h}\right)\,\delta(y-y_k, z-z_k)}{\sum_{k=m+1}^{N+m} K\left(\tfrac{k-k^*}{h}\right)}\,. \end{aligned} \qquad (9)$$

Compared with exponential discounting, the kernel-based discounting cannot be implemented recursively. On the other hand, it is far more flexible, and for the past time instants, it uses data from a two-sided neighborhood of the point of interest.

4. LOCAL-IN-SPACE REGRESSION

To compute a reliable prediction $s(y \mid z^*)$ for a particular value z^* of the regressor vector, it is often sufficient to fit only the data points within a neighbourhood of z^*. The above idea of discounting data according to their "age" can be extended straightforwardly to discounting data according to "similarity" of the regressor z_k to the regressor of interest z^*.

4.1. Locally Weighted Smoothing

Assuming the kernel function $K(x)$ introduced in Section 3.2, we assign to the data point (y_k, z_k) the weight $K(\| z_k - z^* \|_H)$ dependent on the Euclidean distance

$$\| z_k - z^* \|_H^2 = (z_k - z^*)' H^{-1} (z_k - z^*).$$

The symmetric, positive definite matrix H is introduced to reshape further the neighbourhood of the z^* -point, emphasizing the relative importance of the individual entries of regressor z .

With the above choice of the weighting profile, the statistic (ν_N, ρ_N) takes the form

$$\begin{aligned} \nu_N &= \nu_0 + \sum_{k=m+1}^{N+m} K(\| z_k - z^* \|_H), \\ \rho_N(y,z) &= \frac{\nu_0}{\nu_N}\rho_0(y,z) + \frac{\nu_N - \nu_0}{\nu_N} \frac{\sum_{k=m+1}^{N+m} K(\| z_k - z^* \|_H)\, \delta(y - y_k, z - z_k)}{\sum_{k=m+1}^{N+m} K(\| z_k - z^* \|_H)} \end{aligned} \qquad (10)$$

The above idea has been presented in various contexts—as locally-weighted smoothing,[8,9] memory-based learning,[10,11] or just-in-time estimation.[12,13] The theory of non-parametric regression yields a general framework for this approach.[14,15]

4.2. Local Representation of Data

The local (in time or space) regression replaces the true empirical distribution of data with a locally-weighted distribution that stresses the data points close to the time or regressor of interest and that suppresses the points far away. Purposeful modification of the empirical distribution looks like a feature that goes totally beyond the Bayesian paradigm. But even in the Bayesian setting, we make decisions about which variables to consider in the model or what is the range of values to be modelled. The local weighting of the data can be regarded as a smooth way of determining the optimum model structure.

As there are many ways of locally weighting the data, the smoothing needs to be optimized. Cross-validation, bias-variance trade-off and minimization of Mallow's C_p statistic are the most frequently used techniques in practice.[15]

4.3. Implementation Issues

The major advantage of locally-weighted regression is that a simpler model can be used to describe the local behaviour of data. Often models linear in parameters but

nonlinear in data (e.g., polynomial fit) are used. As we have seen in Section 2.5, a closed-form solution exists for such models.

A part of the price is that the locally-weighted regression cannot be implemented recursively. Moreover, when dealing with data stored in a large database, a proper technology must be used to ensure that the required data are retrieved as quickly as possible. This entails the use of a proper database management system, storing of all regressor entries in the database, use of SQL queries to retrieve "similar" data and proper indexing of all regressor variables. The retrieved data are processed then in one shot. A detailed discussion of the database part of the algorithm goes beyond the scope of this paper.

4.4. Applications

Atkeson and co-workers[16] provided a survey on application of the methodology in robot modeling and control. Gorinevsky and Connolly[17] compared several different approximation schemes, such as neural networks, Kohonen maps, radial basis functions, and local polynomial fits, on simulated robot inverse kinematics with added noise, and showed that the local polynomial fits were more accurate than the other methods. Lawrence and co-workers[18] compare neural networks and local regression methods on several benchmark problems; local methods outperformed neural networks on half the problems.

Several researchers have applied locally weighted averaging and regression to free-form 2D and 3D deformation, morphing, and image interpolation in computer graphics[19,20,21]. Coughran, Jr. and Grosse[22] described the use of locally weighted regression in scientific visualization of data.

Numerous researchers reported successful practical applications of locally weighted regression, including Hammond[23] in model fermentation, Ge and co-workers[24] in prediction of the cell density in a fermentation process, Næs and Isaksson[25] and Wang and co-workers[26] in analytical chemistry, Tamada and co-workers[27] in water demand forecasting, Townshend[28] in the analysis, modelling, coding and prediction of speech signals, Kozek[29] in modelling of automobile emissions, Meese and Rose[30] and LeBaron[31] in economics and econometrics, Farmer and Sidorowich[32,33] in modelling and prediction of chaotic dynamic systems.

5. NON-ITERATIVE MONTE CARLO SIMULATION

The idea of non-iterative Monte Carlo simulation is to draw samples of θ from an approximate density $\pi(\theta)$ and then correct the draws so as to approximate better the target posterior density $p_N(\theta)$. Below we suppose that the posterior density $p_N(\theta)$ can be evaluated for any particular point θ even though it is not available in a closed form. This is the case when the observed sample $y^{N+m} = (y_1, \mathrm{K}, y_{N+m})$, $u^{N+m} = (u_1, \mathrm{K}, u_{N+m})$ is short enough for the posterior density to be easily computed for any particular parameter value.

More specifically, the algorithm proceeds as follows.

1. Draw a sample $\theta^1, \mathrm{K}, \theta^M$ from an approximate density $\pi(\theta) \approx p_N(\theta)$.
2. Calculate for $j = 1, \mathrm{K}, M$

$$w_j = \frac{p_N(\theta^j)}{\pi(\theta^j)},$$

$$q_j = \frac{w_j}{\sum_{l=1}^{M} w_l}.$$

3. Draw θ^* from the discrete distribution over $\{\theta^1, \mathrm{K}, \theta^M\}$ which assigns probability q_j at θ^j.

The above algorithm generates θ^* that is distributed approximately according to $p_N(\theta)$. The approximation is improving as M increases. Note that both the posterior density $p_N(\theta)$ and the approximate density $\pi(\theta)$ need to be known with precision up to the normalizing constant.

The above algorithm is known in the literature as *sampling-importance resampling* scheme[34] or *weighted bootstrap*[35].

The algorithm is ideally suited for recursive Bayesian estimation[36]

$$p_k(\theta) \propto p_{k-1}(\theta)\, s_\theta(y_k \mid z_k)$$

where the density $p_{k-1}(\theta)$ is a natural candidate for the approximate density $\pi(\theta)$. The recursive algorithm proceeds as follows.

1. Draw a sample $\theta^{0,1}, \mathrm{K}, \theta^{0,M}$ from the prior density $p_0(\theta)$.
2. For $k = m+1, \mathrm{K}, N+m$:

   (a) For $j = 1, \mathrm{K}, M$, calculate

$$q_j = \frac{s_{\theta^{k-1,j}}(y_k \mid z_k)}{\sum_{l=1}^{M} s_{\theta^{k-1,l}}(y_k \mid z_k)}.$$

(b) Draw a sample $\theta^{k,1},\mathrm{K},\theta^{k,M}$ from the discrete distribution over the points $\{\theta^{k-1,1},\mathrm{K},\theta^{k-1,M}\}$ which places mass q_j at $\theta^{k-1,j}$.

Due to the unequal weighting q_j, some points are resampled more often than others. The number of different values within the sample $\theta^{k,1},\mathrm{K},\theta^{k,M}$ thus decreases with the increasing number of observations. In a recursive setting, the sample may quickly degenerate to a couple of points within the region of high probability. To prevent it, we can sample from a kernel-smoothed approximation of the density $p_k(\theta)$. The step (2b) then modifies as follows.

For $j=1,\mathrm{K},M$:

(b1) Draw a sample $\tilde{\theta}^{k,j}$ from the discrete distribution over the points $\{\theta^{k-1,1},\mathrm{K},\theta^{k-1,M}\}$ which places mass q_j at $\theta^{k-1,j}$.

(b2) Draw a sample $\theta^{k,j}$ from the kernel density $K(\theta\,|\,\tilde{\theta}^{k,j})$.

The kernel smoothing can be regarded as adding a jitter to the samples drawn.[37]

6. ITERATIVE MONTE CARLO SIMULATION

In iterative Monte Carlo simulation, the samples are drawn sequentially, with the distribution of the sampled draws depending on the last value drawn. Hence, the draws form a Markov chain. Several variants of Markov chain simulation are used in practice.

5.1. Metropolis Algorithm

The algorithm proceeds as follows.

1. Draw a starting point θ^0 from a proper starting density (concentrated around the mode of the posterior density).
2. For $j=1,2,\mathrm{K}$:

   (a) Sample a candidate point θ^* from a jumping density $\pi(\theta^*\,|\,\theta^{j-1})$. The jumping density must be symmetric, i.e., $\pi(\theta\,|\,\theta')=\pi(\theta'\,|\,\theta)$ for all θ, θ'.

   (b) Calculate the density ratio $w=\dfrac{p_N(\theta^*)}{p_N(\theta^{j-1})}$.

   (c) Set $\theta^j=\begin{cases}\theta^* & \text{with probability } \min(w,1),\\ \theta^{j-1} & \text{otherwise.}\end{cases}$

A simple example of the underlying Markov process is a random walk $\theta^j=\theta^{j-1}+v_j$ with a zero-mean, normally-distributed, white noise v_j.

For the computation of the relative importance ratio w, the posterior density $p_N(\theta)$ needs to be known with precision up to the normalizing constant.

The efficiency of the Metropolis algorithm is determined by the ratio of the accepted samples to the total number of generated samples. This depends on how well the underlying Markov chain explores the regions of high p_N - probability. Both too small and too large variances of the driving noise may result in inefficient sampling.

The Metropolis algorithm was proposed in the early 1950s.[38] It took three decades for the algorithm to be reinvented in simulated annealing.[39]

After substituting for the posterior density from (4), the relative importance ratio takes the form

$$w = \exp\left(-\nu_N\left[K(\rho_N : s_{\theta^*}) - K(\rho_N : s_{\theta^{j-1}})\right]\right).$$

The Metropolis algorithm accepts θ^* whenever the inaccuracy of ρ_N relative to s_{θ^*} increases compared with the inaccuracy of ρ_N relative to $s_{\theta^{j-1}}$. In the opposite case, the chance to accept θ^* decreases with the increasing inaccuracy drop and the increasing ν_N.

5.2. Metropolis-Hastings Algorithm

In the Metropolis-Hastings algorithm, the jumping densities $\pi_j(\theta \mid \theta')$ need no longer be symmetric. To correct for the asymmetry in the jumping rule, the relative importance ratio w in the Metropolis algorithm is replaced by a ratio of importance ratios

$$w = \frac{p_N(\theta^*)/\pi(\theta^* \mid \theta^{j-1})}{p_N(\theta^{j-1})/\pi(\theta^{j-1} \mid \theta^*)}.$$

The above extension was proposed by Hastings.[40] A number of other variants of the Metropolis algorithm have been proposed in a similar vein.[41]

A special case of the Metropolis-Hastings algorithm occurs when the candidate samples are taken from a fixed density $\pi(\theta)$.

1. Draw a starting point θ^0 from a proper starting density.
2. For $j = 1,2,\mathrm{K}$:

   (a) Sample a candidate point θ^* from the approximate density $\pi(\theta)$.

   (b) Calculate the density ratio $w = \dfrac{p_N(\theta^*)/\pi(\theta^*)}{p_N(\theta^{j-1})/\pi(\theta^{j-1})}$.

   (c) Set $\theta^j = \begin{cases} \theta^* & \text{with probability } \min(w,1), \\ \theta^{j-1} & \text{otherwise.} \end{cases}$

Clearly, the closer the approximate density $\pi(x)$ is to the target density $p_N(x)$, the closer the number of accepted samples is to the total number of generated samples.

5.3. Gibbs Sampler

The Gibbs sampler uses the concept of alternating conditional sampling. Suppose that the parameter vector is composed of three entries, $\theta = (\theta_1, \theta_2, \theta_3)$. Then the Gibbs sampler works as follows.

1. Draw a starting point θ^0 from a proper starting density.
2. For $j = 1,2,\mathrm{K}$:
   (a) Draw a sample θ_1^j from $p_N(\theta_1 \mid \theta_2^{j-1}, \theta_3^{j-1})$.
   (b) Draw a sample θ_2^j from $p_N(\theta_2 \mid \theta_1^j, \theta_3^{j-1})$.
   (c) Draw a sample θ_3^j from $p_N(\theta_3 \mid \theta_1^j, \theta_2^j)$.

The algorithm extends straightforwardly to more dimensions. When appropriate, the parameter vector can be subdivided into subvectors rather than scalar entries.

The complete conditionals are lower-dimensional and thus much easier to sample from. The Gibbs sampler is a natural solution to estimation of hierarchic or structured models. Consider, e.g., the problem of estimating the time of a signal change. The Gibbs algorithm suggests alternating sampling of (1) the initial level given the terminal level and change time, (2) the terminal level given the initial level and change time, (3) the change time given the initial and terminal levels.

The Gibbs sampler appeared in the image processing literature in 1984.[42] In the early 1990s it entered Bayesian statistics.[43] Today it has become a *de facto* standard in Bayesian computations.

Note that to sample from the univariate full conditionals, we can use the Metropolis-Hastings algorithm where the candidate sample is taken from a kernel-smoothed approximation to $p_N(\theta)$ based on its values over a grid of fixed points. More specifically, the approximating density $\pi(\theta_i)$ for the i-th entry of θ is defined as

$$\pi(\theta_i) \propto \sum\nolimits_j K(\theta_i \mid \bar{\theta}_i^{\,j})\; p_N(\theta_1^k, \mathrm{K}\,, \theta_{i-1}^k, \bar{\theta}_i^{\,j}, \theta_{i+1}^{k-1}, \mathrm{K}\,, \theta_n^{k-1})$$

where $\bar{\theta}_i^{\,j}$ denotes the j-th grid value of the i-th entry of the parameter vector θ. The algorithm is known as a griddy Gibbs sampler.[44]

An idea similar to the Gibbs sampler is used in the hit-and-run algorithm[45] where sampling is made in randomly chosen directions rather than dimension-by-dimension.

5.4. Langevin Sampler

The algorithm is based on simulation of the Langevin stochastic differential equation

$$\mathrm{d}\theta^{\mathrm{t}} = \mathrm{d}t\, \nabla log\ p_N(\theta^t) + \sqrt{2}\ \mathrm{d}w_t$$

where $p_N(\theta)$ is the target density, w_t is a standard Brownian motion and ∇ stands for gradient with respect to θ. The solution to the equation is known to be asymptotically distributed according to $p_N(\theta)$.

The Langevin equation needs to be discretized before it can be implemented on digital computer

$$\theta^{t+\Delta t} = \theta^t + \Delta t\, \nabla \log\ p_N(\theta^t) + \sqrt{2\Delta t}\ \mathrm{d}v_t\,.$$

The choice of the discretization period Δt crucially affects the sampler performance; a too long period results in a significant deviation of the sample distribution from the target one, a too short period calls for unnecessarily many samples.

After substituting for the posterior density from (4), the gradient of log-posterior takes the appealing form

$$\nabla \log\ p_N(\theta^t) = -\nu_N\, \nabla K(\rho_N : s_{\theta^t})$$

so that the Langevin algorithm can be rewritten as

$$\theta^{t+\Delta t} = \theta^t - \nu_N\, \Delta t\, \nabla K(\rho_N : s_{\theta^t}) + \sqrt{2\Delta t}\ \mathrm{d}v_t\,.$$

5.5. Implementation Issues

Any application of the Monte Carlo simulation algorithms requires solving a number of practical issues such as:

- Which algorithm fits best the problem in question?
- How long simulation is long enough?
- What should the initial “burn-in” period be?
- Is it better to run one long simulation or a bunch of shorter ones?
- How to choose the starting point?
- Can a different parameterization be of help?

No simple answers exist to these questions; the Monte Carlo simulation provides a set of tools that need to be used thoughtfully and with care. There is no “free-lunch” answer to the general nonlinear estimation problem. Yet, no other technology can compete today with the Monte Carlo simulation in the complexity of problems successfully resolved.

5.6. Applications

Starting with the work of Buntine and Weigend[46] and MacKay[47,48], there is a growing interest in Bayesian analysis of neural networks. Efficient Monte Carlo schemes for inference and prediction have been suggested in the works by MacKay[49] and Neal[50].

Many successful practical applications of Markov chain Monte Carlo methods were reported[51]. The scope of problems addressed is illustrated by the works of Benzuini[52] and Mollie[53] in medicine, Thomas and Gauderman[54] in genetics, Green[55] in image analysis, Litton and Buck in archaeology[56].

7. INFORMATION GEOMETRY

In many practical situations, the amount of information we store from data is not sufficient to determine the density $\rho_N(y,z)$ completely. Instead, in addition to N, only the expectation of a certain vector statistic $h(Y,Z)$ with respect to $\rho_N(y,z)$

$$\begin{aligned} \bar{h}_N &= \iint h(y,z)\,\rho_N(y,z)\,\mathrm{d}y\,\mathrm{d}z \\ &= \frac{\nu_0}{\nu_0+N}\iint h(y,z)\,\rho_0(y,z)\,\mathrm{d}y\,\mathrm{d}z + \frac{N}{\nu_0+N}\,\frac{1}{N}\sum_{k=m+1}^{N+m} h(y_k,z_k) \end{aligned}$$

is supposed available (an example of such statistic was shown in Section 2.5). As $\rho_N(y,z)$ is known only to belong to the set

$$\boldsymbol{R}_N = \left\{\rho : \iint h(y,z)\,\rho(y,z)\,\mathrm{d}y\,\mathrm{d}z = \bar{h}_N\right\},$$

the inaccuracy $K(\rho_N : s_\theta)$ cannot be calculated directly. It can be decomposed, however, into sum of two terms one of which is (approximately, at least) independent of θ while the other depends on $\rho_N(y,z)$ through $\bar{h}_N$ only.

7.1. Pythagorean Relationship

The decomposition is constructed as follows.

1. For every density $s_\theta(y\,|\,z)$, $\theta \in \boldsymbol{T}$, an *approximating* exponential family $\boldsymbol{S}_{\theta;h}$ is constructed as composed of the joint densities

   $$s_{\theta,\lambda}(y,z) \propto s_\theta(y\,|\,z)\,\exp\left(\lambda' h(y,z)\right)$$

   where $h : \boldsymbol{Y}\times\boldsymbol{Z} \to \boldsymbol{R}^n$ is a canonical statistic of the exponential family and $\lambda \in \boldsymbol{R}^n$ is its natural parameter. The functions $1, h_1(y,z), \mathrm{K}\,, h_n(y,z)$ are assumed linearly independent.

2. A *h-projection* $s_{\theta,\lambda^*}(y,z)$ of the density $\rho_N(y,z)$ onto the exponential family $\boldsymbol{S}_{\theta;h}$ is defined through the equality

$$\iint s_{\theta,\lambda^*}(y,z)\, h(y,z)\, \mathrm{d}y\, \mathrm{d}z = \bar{h}_N .$$

3. A *Kullback-Leibler distance*[57] of $s_{\theta,\lambda^*}(y,z)$ and $s_\theta(y \mid z)$ is defined as

$$D(s_{\theta,\lambda^*} \parallel s_\theta) = \iint s_{\theta,\lambda^*}(y,z) \log \frac{s_{\theta,\lambda^*}(y,z)}{s_\theta(y \mid z)}\, \mathrm{d}y\, \mathrm{d}z .$$

Note that the density in the denominator is not normalized with respect to z.

The h-projection $s_{\theta,\lambda^*}(y,z)$ then satisfies the following relationship[2]

$$K(\rho_N : s_\theta) = K(r_N : s_{\theta,\lambda^*}) + D(s_{\theta,\lambda^*} \parallel s_\theta) \qquad (11)$$

which can be regarded as generalization of the Pythagorean theorem known to hold for Kullback-Leibler distances of probability distributions in the case of independent identically distributed observations.[58,59,60]

7.2. Approximation of Inaccuracy

By a proper choice of the statistic $h(y,z)$, the inaccuracy

$$K(\rho_N : s_{\theta,\lambda^*})$$

can be made "nearly" constant for all θ. Owing to the Pythagorean relationship (11), the inaccuracy $K(\rho_N : s_\theta)$ can be approximated then as follows

$$K(\rho_N : s_\theta) \approx D(s_{\theta,\lambda^*} \parallel s_\theta) + \text{const.}$$

In addition, the Pythagorean relationship (11) implies that the Kullback-Leibler distance on the right-hand side minimizes Kullback-Leibler distance between all densities $\rho \in \boldsymbol{R}_N$ and the model point s_θ

$$D(s_{\theta,\lambda^*} \parallel s_\theta) = \min_{\rho \in \boldsymbol{R}_N} D(\rho \parallel s_\theta) = D(\boldsymbol{R}_N \parallel s_\theta) .$$

As a result , we obtain the following posterior approximation

$$\hat{p}_N(\theta) \propto \exp\left(-\nu_N\, D(\boldsymbol{R}_N \parallel s_\theta)\right) .$$

The "information inequality" $D(\boldsymbol{R}_N \parallel s_\theta) \le D(\rho \parallel s_\theta)$ for all $\rho \in \boldsymbol{R}_N$ implies that the capability to discriminate between the empirical and model distributions of data cannot improve as a result of approximation.

7.3 Implementation Issues

In general, the minimum Kullback-Leibler distance $D(\boldsymbol{R}_N \parallel s_\theta)$ cannot be evaluated in an explicit form as a function of θ. But we can compute it for a selection of

parameter points of interest or even better, we can apply the iterative Monte Carlo simulation to generate a sample $\theta^1, \mathrm{K}, \theta^M$ from $p_N(\theta)$.

The minimum Kullback-Leibler distan ce can be computed by solving the dual optimization problem

$$D(\boldsymbol{R}_N \,\|\, s_\theta) = \max_{\lambda} \left[\lambda' \bar{h}_N - \log \iint s_\theta(y \mid z) \exp\left(\lambda' h(y,z)\right) \mathrm{d}y \, \mathrm{d}z \right].$$

Here the difficult part is the multivariate numerical integration, which is related to computation of the normalizing constant of the density $s_\theta(y \mid z)$. Once again, in more dimensions the only working solution is Monte Carlo simulation.[61]

8. CONCLUDING REMARKS

The three major approaches to approximate Bayesian estimation address different kinds of problems. Local-in-space regression is a convenient tool of managing extremely large data sets because relatively simple models are usually sufficient to fit the local data behaviour. Non-iterative sampling is ideally suited to recursive estimation of fairly complex models even though some tricks like adding a jitter to the samples seem still inevitable for practical estimation. Iterative sampling is a method of choice for complex hierarchic or structured models provided the number of samples is relatively small or a sufficient statistic of limited dimension exists for the model considered. Direct approximation using information measures is a systematic and consistent but computer-intensive way of estimating complex models from compressed data.

The fact that the approaches complement in a sense each other makes it possible to combine them if necessary. For instance, the application of iterative sampling to compressed data requires first to approximate the posterior density where the information-based approach can be of immediate help. On the other hand, the information approximation of inaccuracy or posterior density can be evaluated realistically for a limited number of parameter points only, ideally being a sample from the posterior density found via Monte Carlo simulation. Similarly, local regression for complex models may require the use of iterative sampling for approximate estimation.

Whatever the approach, the major challenge for theory, in our view, is to find a systematic way of quantifying the increase of parameter and prediction uncertainty due to the approximation. We can admit uncertainty in computation as long as we keep it under control. The view of parameter estimation via information measures may bring additional insight in this respect.

REFERENCES

1. Peterka, V., Bayesian approach to system identification, in P. Eykhoff (Ed.), *Trends and Progress in System Identification*, Chap. 8, pp. 239–304, Pergamon Press, Elmsford, N.Y., 1981.
2. Kulhavý, R., *Recursive Nonlinear Estimation: A Geometric Approach*, Springer, London, 1996.
3. Brown, R. G., *Statistical Forecasting for Inventory Control*, McGraw-Hill, New York, 1959.
4. Brown, R. G., *Smoothing, Forecasting and Prediction of Discrete Time Series*, Prentice-Hall, Englewood Cliffs, NJ, 1962.
5. Jazwinski, A. H., *Stochastic Processes and Filtering Theory*, Academic Press, New York, 1970.
6. Kulhavý, R. and M. B. Zarrop, 'On a general concept of forgetting', *Int. J. Control*, **58**, 905–924 (1993).
7. Kulhavý, R. and F. J. Kraus, 'On duality of regularized exponential and linear forgetting', *Automatica*, **32**, 1403–1415 (1996).
8. Cleveland, W. S., 'Robust locally-weighted regression and smoothing scatterplots'. *J. Amer. Statist. Assoc.*, **74**, 828–836 (1979).
9. Cleveland, W. S., S. J. Devlin and E. Grosse, 'Regression by local fitting: methods, properties, and computational algorithms', *J. Econometrics*, **37**, 87–114 (1988).
10. Bottou, L. and V. Vapnik, 'Local learning algorithms', *Neural Computation*, **4**, 888–900 (1992).
11. Schaal, S. and C. G. Atkeson, 'Robot juggling: an implementation of memory-based learning', *Control Systems Magazine*, **14**, 57–71 (1994).
12. Cybenko, G., 'Just-in-time learning and estimation', in S. Bittanti and G. Picci (eds.), *Identification, Adaptation, Learning*, pp. 423–434, NATO ASI Series, Springer-Verlag, 1996.
13. Stenman, A., *Just-in-Time Models with Applications to Dynamical Systems*, PhD Dissertation, LIU-TEK-LIC-1997:02, Linköping University, 1997.
14. Härdle, W., *Applied Non-parametric Regression*, Cambridge University Press, 1990.
15. Hastie, T. J., and R. J. Tibshirani, *Generalized Additive Models*, Chapman & Hall, London, 1990.
16. Moore, A.W., C. G. Atkeson and S. A. Schaal, 'Locally weighted learning for control', *Artificial Intelligence Review*, **11,** 75–113 (1997).
17. Gorinevsky, D. and T. H. Connolly, 'Comparison of some neural network and scattered data approximations: the inverse manipulator kinematics example', *Neural Computation*, **6**, 521–542 (1994).
18. Lawrence, S., A.C. Tsoi and A.D. Black, 'Function approximation with neural networks and local methods: bias, variance and smoothness', in P. Bartlett, A. Burkitt, and R. Williamson (Eds.), *Australian Conference on Neural Networks*, Australian National University, Australia, pp. 16–21, 1996.

19. Goshtasby, A., 'Image registration by local approximation methods', *Image and Vision Computing*, **6**, 255–261 (1988).
20. Wolberg, G., *Digital Image Warping*, IEEE Computer Society Press, Los Alamitos, CA, 1990.
21. Ruprecht, D. and H. Müller, 'Deformed cross-dissolves for image interpolation in scientific visualization', *The Journal of Visualization and Computer Animation*, **5**, 167–181 (1994).
22. Coughran, Jr, W.M. and E. Grosse, 'Seeing and hearing dynamic loess surfaces', In *Interface'91 Proceedings*, 224–228, Springer-Verlag, 1991.
23. Hammond, S.V. 'Nir analysis of antibiotic fermentations', in I. Murray and I.A. Cowe (Eds.), *Making Light Work: Advances in Near Infrared Spectroscopy*, 584–589, VCH, New York, NY, 1991.
24. Ge, Z., A.G. Cavitano and J.B. Callis, 'Noninvasive spectroscopy for monitoring cell density in a fermentation process', *Analytical Chemistry*, **66**, 1354–1362 (1994).
25. Næs, T. and T. Isaksson, 'Locally weighted regression in diffuse near-infrared transmittance spectroscopy', *Applied Spectroscopy*, **46**, 34–43 (1992).
26. Wang, Z., T. Isaksson and B.R. Kowalski, 'New approach for distance measurement in locally weighted regression', *Analytical Chemistry*, **66**, 249–260 (1994).
27. Tamada, T., M. Maruyama, Y. Nakamura, S. Abe and K. Maeda, 'Water demand forecasting by memory based learning', *Water Science and Technology*, **28**, 133–140 (1993).
28. Townshend, B., 'Nonlinear prediction of speech signals', in M. Casdagli and S. Eubank (Eds.), *Nonlinear Modeling and Forecasting*, pp. 433–453, Addison Wesley, New York, NY, 1992.
29. Kozek, A.S., 'A new nonparametric estimation method: local and nonlinear', *Interface*, **24**, 389–393 (1992).
30. Meese, R.A. and A.K. Rose, 'Nonlinear, nonparametric, nonessential exchange rate estimation', *The American Economic Review*, 192–196 (1990).
31. LeBaron, B. (1992), Nonlinear forecasts for the S&P stock index in M. Casdagli and S. Eubank (Eds.), *Nonlinear Modeling and Forecasting*, pp. 381–393, Addison Wesley, New York, NY, 1992.
32. Farmer, J.D. and J. Sidorowich, 'Exploiting chaos to predict the future and reduce noise', in W.C. Lee (Ed.), *Evolution, Learning, and Cognition*, pp. 277–330, World Scientific, Singapore, 1988.
33. Farmer, J.D. and J.J. Sidorowich, 'Predicting chaotic time series', *Phys. Rev. Lett.*, **59**, 845–848 (1987).
34. Rubin, D. B., 'Using the SIR algorithm to simulate posterior distributions' (with discussion, in J. M. Bernardo, M. H. DeGroot, D. V. Lindley and A. F. M. Smith (Eds.), *Bayesian Statistics 3*, pp. 395–402, Oxford University Press, Oxford, 1988.
35. Smith, A. F. M. and A. E. Gelfand, 'Bayesian statistics without tears: a sampling–resampling perspective', *Amer. Statistician*, **46**, 84–88 (1992).
36. Liu, J. S. and R. Chen, 'Sequential Monte Carlo methods for dynamic systems', to appear in *J. Amer. Statist. Assoc.*
37. Gordon, N. J., D. J. Salmond and A. F. M. Smith, 'A novel approach to nonlinear/non-Gaussian Bayesian state estimation', *Proc. IEE-F*, **140**,107–113 (1993).

38. Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller, 'Equations of state calculations by fast computing machines', *J. Chem. Phys*., **21**, 1087–1091 (1953).
39. Kirkpatrick, S., C. D. Gelatt and M. P. Vecchi, 'Optimization by simulated annealing', *Science*, **220**, 671–680 (1983).
40. Hastings, W. K., 'Monte Carlo sampling methods using Markov chains and their applications', *Biometrika*, **57**, 97–109 (1970).
41. Tierney, L., 'Markov chains for exploring posterior distributions', *Ann. Statist.*, **22**, 1701–1762 (1994).
42. Geman, S. and D. Geman, 'Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images', *IEEE Trans. Pattern Anal. Machine Intell.*, 6, 721–741 (1984).
43. Gelfand, A.E. and A.F.M. Smith, 'Sampling based approaches to calculating marginal densities', *J. Amer. Statist. Soc.*, **85**, 398–409 (1990).
44. Tanner, M. A., *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*, 2nd ed., Springer-Verlag, New York, 1993.
45. Schmeiser, B. and M.-H. Chen, 'General hit-and-run Monte Carlo sampling for evaluating multidimensional integrals', *Technical Report*, School of Industrial Engineering, Purdue University, 1991.
46. Buntine, W. and A. Weigend, 'Bayesian back-propagation', *Complex Systems*, **5**, 603–643 (1991).
47. MacKay, D.J.C., 'A practical Bayesian framework for backpropagation networks', *Neural Computation*, **4**, 448–472 (1992).
48. MacKay, D.J.C., 'Bayesian interpolation', *Neural Computation*, **4**, 415–447 (1992).
49. MacKay, D.J.C., 'Probable networks and plausible predictions—a review of practical Bayesian methods for supervised neural networks', *Network: Computation in Neural Systems*, **6**, 469–505 (1995).
50. Neal, R.M., *Bayesian Learning for Neural Networks*, Lecture Notes in Statistics, vol. 118, Springer-Verlag, New York, 1996.
51. Gilks, W.R., S. Richardson and D.J. Spiegelhalter, *Markov Chain Monte Carlo in Practice*, Chapman & Hall, London, 1996.
52. Benzuini, C., 'Medical monitoring', in Ref. 51, pp. 321–338.
53. Mollie A., 'Bayesian mapping of disease', in Ref. 51, pp. 359–380.
54. Thomas, D.C. and W. J. Gauderman, 'Gibbs sampling methods in genetics', in Ref. 51, 419–440.
55. Green P.J., 'MCMC in image analysis', in Ref. 51, pp. 381–400.
56. Litton, C. and C. Buck, 'An archaeological example: radoicarbon dating', in Ref. 51, pp. 321–338.
57. Kullback, S. and R. A. Leibler, 'On information and sufficiency', *Ann. Math. Statist.*, **22**, 79–86 (1951).
58. Čencov, N. N., *Statistical Decision Rules and Optimal Inference*, Amer. Math. Soc. Transl. 53, AMS, Providence, RI, 1982.
59. Csiszár, I., 'I-divergence geometry of probability distributions and minimization problems', *Ann. Probab.*, **3**, 146–158 (1975).

60. Amari, S., *Differential-Geometrical Methods in Statistics*, Springer, New York, 1985.
61. Geyer, C. J., Markov chain Monte Carlo maximum likelihood, *Computing Science and Statistics*, **23**, 156–163 (1991).
62. Geyer, C. J., Estimating Normalizing Constants and Reweighting Mixtures in Markov Chain Monte Carlo, *Technical Report*, No. 568, School of Statistics, University of Minnesota, 1994.