LETTER Communicated by Michael Jordan

Robust Full Bayesian Learning for Radial Basis Networks

Christophe Andrieu*
Cambridge University Engineering Department, Cambridge CB2 1PZ, England

Nando de Freitas
Computer Science Division, University of California, Berkeley, CA 94720-1776, U.S.A.

Arnaud Doucet
Cambridge University Engineering Department, Cambridge CB2 1PZ, England

We propose a hierarchical full Bayesian model for radial basis networks. This model treats the model dimension (number of neurons), model parameters, regularization parameters, and noise parameters as unknown random variables. We develop a reversible-jump Markov chain Monte Carlo (MCMC) method to perform the Bayesian computation. We find that the results obtained using this method are not only better than the ones reported previously, but also appear to be robust with respect to the prior specification. In addition, we propose a novel and computationally efficient reversible-jump MCMC simulated annealing algorithm to optimize neural networks. This algorithm enables us to maximize the joint posterior distribution of the network parameters and the number of basis function. It performs a global search in the joint space of the parameters and number of parameters, thereby surmounting the problem of local minima to a large extent. We show that by calibrating the full hierarchical Bayesian prior, we can obtain the classical Akaike information criterion, Bayesian information criterion, and minimum description length model selection criteria within a penalized likelihood framework. Finally, we present a geometric convergence theorem for the algorithm with homogeneous transition kernel and a convergence theorem for the reversible-jump MCMC simulated annealing method.

1 Introduction

Buntine and Weigend (1991) and Mackay (1992) showed that a principled Bayesian learning approach to neural networks can lead to many improvements. In particular, Mackay showed that by approximating the distributions of the weights with gaussians and adopting smoothing priors, it is

* Authorship based on alphabetical order.

possible to obtain estimates of the weights and output variances and to set the regularization coefficients automatically.

Neal (1996) cast the net much further by introducing advanced Bayesian simulation methods, specifically the hybrid Monte Carlo method (Brass, Pendleton, Chen, & Robson, 1993), into the analysis of neural networks. Theoretically, he also proved that certain classes of priors for neural networks, whose number of hidden neurons tends to infinity, converge to gaussian processes. Bayesian sequential Monte Carlo methods have also been shown to provide good training results, especially in time-varying scenarios (de Freitas, Niranjan, Gee, & Doucet, 2000).

An essential requirement of neural network training is the correct selection of the number of neurons. There have been three main approaches to this problem: penalized likelihood, predictive assessment, and growing and pruning techniques. In the penalized likelihood context, a penalty term is added to the likelihood function so as to limit the number of neurons, thereby avoiding overfitting. Classical examples of penalty terms include the well-known Akaike information criterion (AIC), Bayesian information criterion (BIC) and minimum description length (MDL) (Akaike, 1974; Schwarz, 1985; Rissanen, 1987). Penalized likelihood has also been used extensively to impose smoothing constraints by weight decay priors (Hinton, 1987; Mackay, 1992) or functional regularizers that penalize for high-frequency signal components (Girosi, Jones, & Poggio, 1995).

In the predictive assessment approach, the data are split into a training set, a validation set, and possibly a test set. The key idea is to balance the bias and variance of the predictor by choosing the number of neurons so that the errors in each data set are of the same magnitude.

The problem with the previous approaches, known as the model adequacy problem, is that they assume one knows which models to test. To overcome this difficulty, various authors have proposed model selection methods whereby the number of neurons is set by growing and pruning algorithms. Examples of this class of algorithms include the upstart algorithm (Frean, 1990), cascade correlation (Fahlman & Lebiere, 1988), optimal brain damage (Le Cun, Denker, & Solla, 1990) and the resource allocating network (RAN) (Platt, 1991). A major shortcoming of these methods is that they lack robustness in that the results depend on several heuristically set thresholds. For argument's sake, let us consider the case of the RAN algorithm. A new radial basis function is added to the hidden layer each time an input in a novel region of the input space is found. Unfortunately, novelty is assessed in terms of two heuristically set thresholds. The center of the gaussian basis function is then placed at the location of the novel input, while its width depends on the distance between the novel input and the stored patterns. For improved efficiency, the amplitudes of the gaussians may be estimated with an extended Kalman filter (Kadirkamanathan & Niranjan, 1993). Yingwei, Sundararajan, and Saratchandran (1997) have extended the approach by proposing a simple pruning technique. Their strategy is to monitor the

outputs of the gaussian basis functions continuously and compare them to a threshold. If a particular output remains below the threshold over a number of consecutive inputs, then the corresponding basis function is removed.

Recently, Rios Insua and Müller (1998), Marrs (1998), and Holmes and Mallick (1998) have addressed the issue of selecting the number of hidden neurons with growing and pruning algorithms from a Bayesian perspective. In particular, they apply the reversible-jump Markov chain Monte Carlo (MCMC) algorithm of Green (1995; Richardson & Green, 1997) to feedforward sigmoidal networks and radial basis function (RBF) networks to obtain joint estimates of the number of neurons and weights. Once again, their results indicate that it is advantageous to adopt the Bayesian framework and MCMC methods to perform model order selection. In this article, we also apply the reversible-jump MCMC simulation algorithm to RBF networks so as to compute the joint posterior distribution of the radial basis parameters and the number of basis functions. We advance this area of research in three important directions:

- We propose a hierarchical prior for RBF networks. That is, we adopt a full Bayesian model, which accounts for model order uncertainty and regularization, and show that the results appear to be robust with respect to the prior specification.
- We propose an automated growing and pruning reversible-jump MCMC optimization algorithm to choose the model order using the classical AIC, BIC, and MDL criteria. This algorithm estimates the maximum of the joint likelihood function of the radial basis parameters and the number of bases using a reversible-jump MCMC simulated annealing approach. It has the advantage of being more computationally efficient than the reversible-jump MCMC algorithm used to perform the integrations with the hierarchical full Bayesian model.
- We derive a geometric convergence theorem for the homogeneous reversible-jump MCMC algorithm and a convergence theorem for the annealed reversible-jump MCMC algorithm.

In Section 1, we present the approximation model. In section 2, we formalize the Bayesian model and specify the prior distributions. Section 3 is devoted to Bayesian computation. We first propose an MCMC sampler to perform Bayesian inference when the number of basis functions is given. Subsequently, a reversible-jump MCMC algorithm is derived to deal with the case where the number of basis functions is unknown. A reversible-jump MCMC simulated annealing algorithm to perform stochastic optimization using the AIC, BIC, and MDL criteria is proposed in section 5. The convergence of the algorithms is established in section 6. The performance of the proposed algorithms is illustrated by computer simulations in section 7. Finally, some conclusions are drawn in section 8. Appendix A defines the notation used, and the proofs of convergence are given in appendix B.

2 Problem Statement

Many physical processes may be described by the following nonlinear, multivariate input-output mapping:

$$\mathbf{y}_t = \mathbf{f}(\mathbf{x}_t) + \mathbf{n}_t,$$

where $\mathbf{x}_t \in \mathbb{R}^d$ corresponds to a group of input variables, $\mathbf{y}_t \in \mathbb{R}^c$ to the target variables, $\mathbf{n}_t \in \mathbb{R}^c$ to an unknown noise process, and $t = \{1, 2, \ldots\}$ is an index variable over the data. In this context, the learning problem involves computing an approximation to the function $\mathbf{f}$ and estimating the characteristics of the noise process given a set of N input-output observations $\mathcal{O} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N, \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N\}$. Typical examples include regression, where $\mathbf{y}_{1:N,1:c}$[1] is continuous; classification, where $\mathbf{y}$ corresponds to a group of classes; and nonlinear dynamical system identification, where the inputs and targets correspond to several delayed versions of the signals under consideration.

When the exact nonlinear structure of the multivariate function $\mathbf{f}$ cannot be established a priori, it may be synthesized as a combination of parameterized basis functions, that is,

$$\hat{\mathbf{f}}(\mathbf{x}, \boldsymbol{\theta}) = G_k\left(\boldsymbol{\theta}_k; \left(\cdots \sum_j G_j\left(\boldsymbol{\theta}_j; \sum_i G_i(\boldsymbol{\theta}_i; \mathbf{x})\right)\cdots\right)\right), \tag{2.1}$$

where $G_i(\mathbf{x}, \boldsymbol{\theta}_i)$ denotes a multivariate basis function. These multivariate basis functions may be generated from univariate basis functions using radial basis, tensor product, or ridge construction methods. This type of modeling is often referred to as non-parametric regression because the number of basis functions is typically very large. Equation 2.1 encompasses a large number of nonlinear estimation methods, including projection pursuit regression, Volterra series, fuzzy inference systems, multivariate adaptive regression splines (MARS), and many artificial neural network paradigms such as functional link networks, multilayer perceptrons (MLPs), RBF networks, wavelet networks, and hinging hyperplanes (see, e.g., Cheng & Titterington, 1994; de Freitas, 1999; Denison, Mallick, & Smith, 1998; Holmes & Mallick, 2000).

For the purposes of this article, we adopt the approximation scheme of Holmes and Mallick (1998), consisting of a mixture of k RBFs and a linear

[1] $\mathbf{y}_{1:N,1:c}$ is an $N \times c$ matrix, where N is the number of data and c the number of outputs. We adopt the notation $\mathbf{y}_{1:N,j} \triangleq (\mathbf{y}_{1,j}, \mathbf{y}_{2,j}, \ldots, \mathbf{y}_{N,j})'$ to denote all the observations corresponding to the jth output (jth column of $\mathbf{y}$). To simplify the notation, $\mathbf{y}_t$ is equivalent to $\mathbf{y}_{t,1:c}$. That is, if one index does not appear, it is implied that we are referring to all of its possible values. Similarly, $\mathbf{y}$ is equivalent to $\mathbf{y}_{1:N,1:c}$. We favor the shorter notation but invoke the longer notation to avoid ambiguities and emphasize certain dependencies.

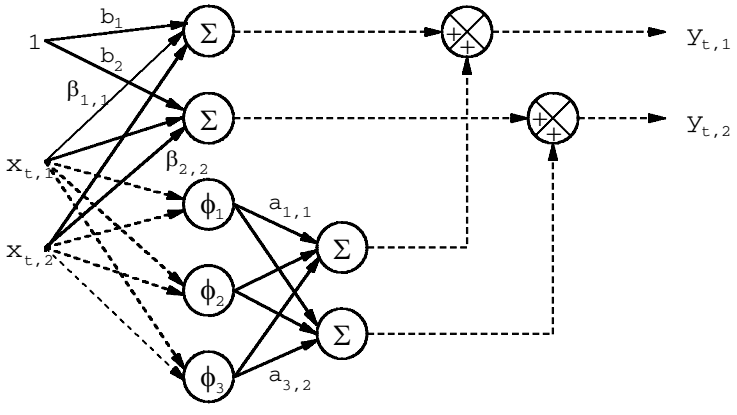

Figure 1: Approximation model with three RBFs, two inputs, and two outputs. The solid lines indicate weighted connections.

regression term. However, the work can be straightforwardly extended to other regression models. More precisely, our model $\mathcal{M}$ is:

$$\begin{aligned} \mathcal{M}_0\text{:} \quad & \mathbf{y}_t = \boldsymbol{b} + \boldsymbol{\beta}'\mathbf{x}_t + \mathbf{n}_t && k = 0 \\ \mathcal{M}_k\text{:} \quad & \mathbf{y}_t = \textstyle\sum_{j=1}^{k} \mathbf{a}_j \phi(\|\mathbf{x}_t - \boldsymbol{\mu}_j\|) + \boldsymbol{b} + \boldsymbol{\beta}'\mathbf{x}_t + \mathbf{n}_t && k \geq 1, \end{aligned}$$

where $\|\cdot\|$ denotes a distance metric (usually Euclidean or Mahalanobis), $\boldsymbol{\mu}_j \in \mathbb{R}^d$ denotes the jth RBF center for a model with k RBFs, $\mathbf{a}_j \in \mathbb{R}^c$ the jth RBF amplitude, and $\boldsymbol{b} \in \mathbb{R}^c$ and $\boldsymbol{\beta} \in \mathbb{R}^d \times \mathbb{R}^c$ the linear regression parameters. The noise sequence $\mathbf{n}_t \in \mathbb{R}^c$ is assumed to be zero-mean white gaussian. Although we have not explicitly indicated the dependency of $\boldsymbol{b}$, $\boldsymbol{\beta}$, and $\mathbf{n}_t$ on k, these parameters are indeed affected by the value of k. Figure 1 depicts the approximation model for $k = 3$, $c = 2$, and $d = 2$ (c is the number of outputs, d is the number of inputs, and k is the number of basis functions). Depending on our a priori knowledge about the smoothness of the mapping, we can choose different types of basis functions (Girosi et al., 1995). The most common choices are:

Linear: $\phi(\varrho) = \varrho$

Cubic: $\phi(\varrho) = \varrho^3$

Thin plate spline: $\phi(\varrho) = \varrho^2 \ln(\varrho)$

Multiquadric: $\phi(\varrho) = \left(\varrho^2 + \lambda^2\right)^{1/2}$

Gaussian: $\phi(\varrho) = \exp\left(-\lambda\varrho^2\right)$

For the last two choices of basis functions, we treat λ as a user set parameter. For convenience, we express our approximation model in vector-matrix form:

$$\mathbf{y} = \mathbf{D}\left(\boldsymbol{\mu}_{1:k,1:d}, \mathbf{x}_{1:N,1:d}\right)\boldsymbol{\alpha}_{1:1+d+k,1:c} + \mathbf{n}_t, \tag{2.2}$$

that is:

$$\begin{bmatrix} \mathbf{y}_{1,1} \cdots \mathbf{y}_{1,c} \\ \mathbf{y}_{2,1} \cdots \mathbf{y}_{2,c} \\ \vdots \\ \mathbf{y}_{N,1} \cdots \mathbf{y}_{N,c} \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{x}_{1,1} \cdots \mathbf{x}_{1,d} & \phi(\mathbf{x}_1, \boldsymbol{\mu}_1) \cdots \phi(\mathbf{x}_1, \boldsymbol{\mu}_k) \\ 1 & \mathbf{x}_{2,1} \cdots \mathbf{x}_{2,d} & \phi(\mathbf{x}_2, \boldsymbol{\mu}_1) \cdots \phi(\mathbf{x}_2, \boldsymbol{\mu}_k) \\ \vdots & \vdots & \vdots \\ 1 & \mathbf{x}_{N,1} \cdots \mathbf{x}_{N,d} & \phi(\mathbf{x}_N, \boldsymbol{\mu}_1) \cdots \phi(\mathbf{x}_N, \boldsymbol{\mu}_k) \end{bmatrix}$$

$$\times \begin{bmatrix} \boldsymbol{b}_1 \cdots \boldsymbol{b}_c \\ \boldsymbol{\beta}_{1,1} \cdots \boldsymbol{\beta}_{1,c} \\ \vdots \\ \boldsymbol{\beta}_{d,1} \cdots \boldsymbol{\beta}_{d,c} \\ \mathbf{a}_{1,1} \cdots \mathbf{a}_{1,c} \\ \vdots \\ \mathbf{a}_{k,1} \cdots \mathbf{a}_{k,c} \end{bmatrix} + \mathbf{n}_{1:N},$$

where the noise process is assumed to be normally distributed as follows:

$$\mathbf{n}_t \sim \mathcal{N}\left(\mathbf{0}_{c\times 1}, \operatorname{diag}\left(\boldsymbol{\sigma}_1^2, \ldots, \boldsymbol{\sigma}_c^2\right)\right).$$

Once again, we stress that $\boldsymbol{\sigma}^2$ depends implicitly on the model order k. We assume here that the number k of basis functions and their parameters $\boldsymbol{\theta} \triangleq \{\boldsymbol{\alpha}_{1:m,1:c}, \boldsymbol{\mu}_{1:k,1:d}, \boldsymbol{\sigma}_{1:c}^2\}$, with $m = 1 + d + k$, are unknown. Given the data set $\{\mathbf{x}, \mathbf{y}\}$, our objective is to estimate k and $\boldsymbol{\theta} \in \boldsymbol{\Theta}_k$.

3 Bayesian Model and Aims

We follow a Bayesian approach where the unknowns k and $\boldsymbol{\theta}$ are regarded as being drawn from appropriate prior distributions. These priors reflect our degree of belief on the relevant values of these quantities (Bernardo & Smith, 1994). Furthermore, we adopt a hierarchical prior structure that enables us to treat the priors' parameters (hyperparameters) as random variables drawn from suitable distributions (hyperpriors). That is, instead of fixing the hyperparameters arbitrarily, we acknowledge that there is an inherent uncertainty in what we think their values should be. By devising probabilistic models that deal with this uncertainty, we are able to implement estimation techniques that are robust to the specification of the hyperpriors.

The remainder of the section is organized as follows. First, we propose a hierarchical model prior that defines a probability distribution over the space of possible structures of the data. Subsequently, we specify the estimation and inference aims. Finally, we exploit the analytical properties of the model to obtain an expression, up to a normalizing constant, of the joint posterior distribution of the basis centers and their number.

3.1 Prior Distributions. The overall parameter space $\boldsymbol{\Theta} \times \boldsymbol{\Psi}$ can be written as a finite union of subspaces $\boldsymbol{\Theta} \times \boldsymbol{\Psi} = (\cup_{k=0}^{k_{\max}} \{k\} \times \boldsymbol{\Theta}_k) \times \boldsymbol{\Psi}$ where $\boldsymbol{\Theta}_0 \triangleq (\mathbb{R}^{d+1})^c \times (\mathbb{R}^+)^c$ and $\boldsymbol{\Theta}_k \triangleq (\mathbb{R}^{d+1+k})^c \times (\mathbb{R}^+)^c \times \boldsymbol{\Omega}_k$ for $k \in \{1, \ldots, k_{\max}\}$. That is, $\boldsymbol{\alpha} \in (\mathbb{R}^{d+1+k})^c$, $\boldsymbol{\sigma} \in (\mathbb{R}^+)^c$, and $\boldsymbol{\mu} \in \boldsymbol{\Omega}_k$. The hyperparameter space $\boldsymbol{\Psi} \triangleq (\mathbb{R}^+)^{c+1}$, with elements $\boldsymbol{\psi} \triangleq \{\Lambda, \boldsymbol{\delta}^2\}$, will be discussed at the end of this section.

The space of the radial basis centers $\boldsymbol{\Omega}_k$ is defined as a compact set that encompasses the input data: $\boldsymbol{\Omega}_k \triangleq \{\boldsymbol{\mu}; \boldsymbol{\mu}_{1:k,i} \in [\min(\mathbf{x}_{1:N,i}) - \iota \Xi_i, \max(\mathbf{x}_{1:N,i}) + \iota \Xi_i]^k$ for $i = 1, \ldots, d$ with $\boldsymbol{\mu}_{j,i} \neq \boldsymbol{\mu}_{l,i}$ for $j \neq l\}$. $\Xi_i = \|\max(\mathbf{x}_{1:N,i}) - \min(\mathbf{x}_{1:N,i})\|$ denotes the Euclidean distance for the ith dimension of the input, and ι is a user-specified parameter that we need to consider only if we wish to place basis functions outside the region where the input data lie. That is, we allow $\boldsymbol{\Omega}_k$ to include the space of the input data and extend it by a factor proportional to the spread of the input data. Typically, researchers either set ι to zero and choose the basis centers from the input data (Holmes & Mallick, 1998; Kadirkamanathan & Niranjan, 1993) or compute the basis centers using clustering algorithms (Moody & Darken, 1988). This strategy is also exploited within the support vector paradigm (Vapnik, 1995). The premise here is that it is better to place the basis functions where the data are dense, not in regions of extrapolation. Moreover, if the input space is very large, then placing basis functions where the data lie reduces the space over which one has to sample the basis locations. However, when one adopts global basis functions, it is no longer clear that this is the case. In fact, if there are outliers, it might be a bad strategy to place basis functions where the data lie. After some experimentation and taking these trade-offs into consideration, we chose to sample the basis centers from the space $\boldsymbol{\Omega}_k$, whose hypervolume is $\Im^k \triangleq (\prod_{i=1}^{d}(1 + 2\iota)\Xi_i)^k$. Figure 2 shows this space for a two-dimensional input.

The maximum number of basis functions is defined as $k_{\max} \triangleq (N - (d + 1))$.[2] We also define $\boldsymbol{\Omega} \triangleq \cup_{k=0}^{k_{\max}} \{k\} \times \boldsymbol{\Omega}_k$ with $\boldsymbol{\Omega}_0 \triangleq \emptyset$. There is a natural hierarchical structure to this setup (Richardson & Green, 1997), which we formalize by modeling the joint distribution of all variables as

[2] The constraint $k \leq N - (d + 1)$ is added because otherwise the columns of $\mathbf{D}(\mu_{1:k}, \mathbf{x})$ are linearly dependent and the parameters θ may not be uniquely estimated from the data (see equation 2.2).

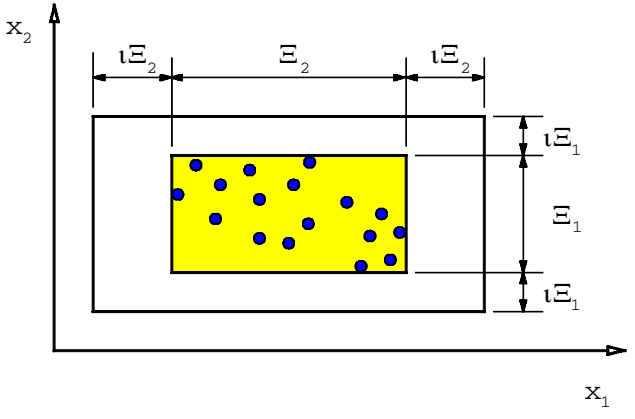

Figure 2: RBF centers space Ω for a two-dimensional input. The circles represent the input data.

$p(k, \boldsymbol{\theta}, \psi, \mathbf{y} \mid \mathbf{x}) = p(\mathbf{y} \mid k, \boldsymbol{\theta}, \psi, \mathbf{x}) p(\boldsymbol{\theta} \mid k, \psi, \mathbf{x}) p(k, \psi \mid \mathbf{x})$, where $p(k, \psi \mid \mathbf{x})$ is the joint model order and hyperparameters' probability, $p(\boldsymbol{\theta} \mid k, \psi, \mathbf{x})$ is the parameters' prior, and $p(\mathbf{y} \mid k, \boldsymbol{\theta}, \psi, \mathbf{x})$ is the likelihood. Under the assumption of independent outputs given $(k, \boldsymbol{\theta})$, the likelihood for the approximation model described in the previous section is:

$$
\begin{aligned}
p(\mathbf{y} \mid k, \boldsymbol{\theta}, \psi, \mathbf{x}) &= \prod_{i=1}^{c} p(\mathbf{y}_{1:N,i} \mid k, \boldsymbol{\alpha}_{1:m,i}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}_i^2, \mathbf{x}) \\
&= \prod_{i=1}^{c} (2\pi \boldsymbol{\sigma}_i^2)^{-N/2} \exp\left(-\frac{1}{2\boldsymbol{\sigma}_i^2} \left(\mathbf{y}_{1:N,i} - \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \boldsymbol{\alpha}_{1:m,i}\right)' \right. \\
&\qquad \left. \times \left(\mathbf{y}_{1:N,i} - \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \boldsymbol{\alpha}_{1:m,i}\right) \right).
\end{aligned}
$$

We assume the following structure for the prior distribution:

$$
\begin{aligned}
&p(k, \boldsymbol{\theta}, \psi) \\
&\quad = p(\boldsymbol{\alpha}_{1:m} \mid k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) p(\boldsymbol{\mu}_{1:k} \mid k, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) \\
&\qquad \times p(k \mid \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) p(\boldsymbol{\sigma}^2 \mid \Lambda, \boldsymbol{\delta}^2) p(\Lambda, \boldsymbol{\delta}^2) \\
&\quad = p(\boldsymbol{\alpha}_{1:m} \mid k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \boldsymbol{\delta}^2) p(\boldsymbol{\mu}_{1:k} \mid k) p(k \mid \Lambda) p(\boldsymbol{\sigma}^2) p(\Lambda) p(\boldsymbol{\delta}^2), \qquad (3.1)
\end{aligned}
$$

where the scale parameters $\boldsymbol{\sigma}_i^2$, $i = 1, \ldots, c$ are assumed to be independent of the hyperparameters (i.e., $p(\boldsymbol{\sigma}^2 \mid \Lambda, \boldsymbol{\delta}^2) = p(\boldsymbol{\sigma}^2)$), independent of each other $(p(\boldsymbol{\sigma}^2) = \prod_{i=1}^{c} p(\boldsymbol{\sigma}_i^2))$, and distributed according to conjugate inverse-gamma prior distributions $\boldsymbol{\sigma}_i^2 \sim \mathcal{IG}(\frac{\upsilon_0}{2}, \frac{\gamma_0}{2})$. When $\upsilon_0 = 0$ and $\gamma_0 = 0$,

we obtain Jeffreys's uninformative prior $p(\boldsymbol{\sigma}_i^2) \propto 1/\boldsymbol{\sigma}_i^2$ (Bernardo & Smith, 1994). Given $\boldsymbol{\sigma}^2$, we introduce the following prior distribution:

$$p(k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k} \mid \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) = p(\boldsymbol{\alpha}_{1:m} \mid k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \boldsymbol{\delta}^2) p(\boldsymbol{\mu}_{1:k} \mid k) p(k \mid \Lambda)$$
$$= \left[\prod_{i=1}^{c} |2\pi \boldsymbol{\sigma}_i^2 \boldsymbol{\Sigma}_i|^{-1/2} \exp\left(-\frac{1}{2\boldsymbol{\sigma}_i^2} \boldsymbol{\alpha}'_{1:m,i} \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\alpha}_{1:m,i} \right) \right]$$
$$\times \left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}_{1:k})}{\Im^k} \right] \left[\frac{\Lambda^k / k!}{\sum_{j=0}^{k_{\max}} \Lambda^j / j!} \right],$$

where $\boldsymbol{\Sigma}_i^{-1} = \boldsymbol{\delta}_i^{-2} \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})$ and $\mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}_{1:k})$ is the indicator function of the set $\boldsymbol{\Omega}$ (1 if $(k, \boldsymbol{\mu}_{1:k}) \in \boldsymbol{\Omega}$, 0 otherwise).

The prior model order distribution $p(k \mid \Lambda)$ is a truncated Poisson distribution. Conditional on k, the RBF centers are uniformly distributed. Finally, conditional on $(k, \boldsymbol{\mu}_{1:k})$, the coefficients $\boldsymbol{\alpha}_{1:m,i}$ are assumed to be zero-mean gaussian with variance $\boldsymbol{\sigma}_i^2 \boldsymbol{\Sigma}_i$. The terms $\boldsymbol{\delta}^2 \in (\mathbb{R}^+)^c$ and $\Lambda \in \mathbb{R}^+$ can be respectively interpreted as the expected signal-to-noise ratios and the expected number of radial basis. The prior for the coefficients has been previously advocated by various authors (George & Foster, 1997; Smith & Kohn, 1996). It corresponds to the popular g-prior distribution (Zellner, 1986) and can be derived using a maximum entropy approach (Andrieu, 1998). An important property of this prior is that it penalizes for basis functions being too close as, in this situation, the determinant of $\boldsymbol{\Sigma}_i^{-1}$ tends to zero.

We now turn our attention to the hyperparameters, which allow us to accomplish our goal of designing robust model selection schemes. We assume that they are independent of each other, that is, $p(\Lambda, \boldsymbol{\delta}^2) = p(\Lambda) p(\boldsymbol{\delta}^2)$. Moreover, $p(\boldsymbol{\delta}^2) = \prod_{i=1}^{c} p(\boldsymbol{\delta}_i^2)$. As $\boldsymbol{\delta}^2$ is a scale parameter, we ascribe a vague conjugate prior density to it: $\boldsymbol{\delta}_i^2 \sim \mathcal{IG}(\alpha_{\delta^2}, \beta_{\delta^2})$ for $i = 1, \ldots, c$, with $\alpha_{\delta^2} = 2$ and $\beta_{\delta^2} > 0$. The variance of this hyperprior with $\alpha_{\delta^2} = 2$ is infinite. We apply the same method to Λ by setting an uninformative conjugate prior (Bernardo & Smith, 1994): $\Lambda \sim \mathcal{G}a(1/2 + \varepsilon_1, \varepsilon_2)$ ($\varepsilon_i \ll 1$ $i = 1, 2$). We can visualize our hierarchical prior (see equation 3.1) with a directed acyclic graphical model (DAG), as shown in Figure 3.

3.2 Estimation and Inference Aims. The Bayesian inference of k, $\boldsymbol{\theta}$, and $\boldsymbol{\psi}$ is based on the joint posterior distribution $p(k, \boldsymbol{\theta}, \boldsymbol{\psi} \mid \mathbf{x}, \mathbf{y})$ obtained from Bayes's theorem. Our aim is to estimate this joint distribution from which, by standard probability marginalization and transformation techniques, one can "theoretically" obtain all posterior features of interest. For instance, we might wish to perform inference with the predictive density:

$$p(\mathbf{y}_{N+1} \mid \mathbf{x}_{1:N+1}, \mathbf{y}_{1:N})$$
$$= \int_{\boldsymbol{\Theta} \times \boldsymbol{\Psi}} p(\mathbf{y}_{N+1} \mid k, \boldsymbol{\theta}, \boldsymbol{\psi}, \mathbf{x}_{N+1}) p(k, \boldsymbol{\theta}, \boldsymbol{\psi} \mid \mathbf{x}_{1:N}, \mathbf{y}_{1:N}) \, \mathrm{d}k \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\boldsymbol{\psi}$$

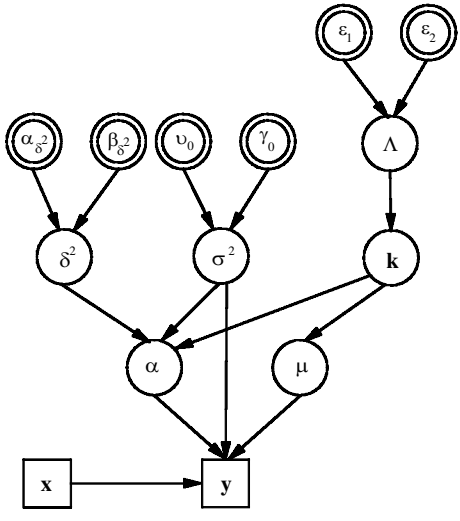

Figure 3: Directed acyclic graphical model for our prior.

and consequently make predictions, such as:

$$\mathbb{E}(\mathbf{y}_{N+1} \mid \mathbf{x}_{1:N+1}, \mathbf{y}_{1:N}) = \int_{\Theta \times \Psi} \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}_{N+1}) \boldsymbol{\alpha}_{1:m} p(k, \boldsymbol{\theta}, \boldsymbol{\psi} \mid \mathbf{x}_{1:N}, \mathbf{y}_{1:N}) \, \mathrm{d}k \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\boldsymbol{\psi}.$$

We might also be interested in evaluating the posterior model probabilities $p(k \mid \mathbf{x}, \mathbf{y})$, which can be used to perform model selection by selecting the model order as $\arg\max_{k \in \{0,\ldots,k_{\max}\}} p(k \mid \mathbf{x}, \mathbf{y})$. In addition, it allows us to perform parameter estimation by computing, for example, the conditional expectation $\mathbb{E}(\boldsymbol{\theta} \mid k, \mathbf{x}, \mathbf{y})$.

However, it is not possible to obtain these quantities analytically, as it requires the evaluation of high-dimensional integrals of nonlinear functions in the parameters, as we shall see in the following section. We propose here to use an MCMC method to perform Bayesian computation. MCMC techniques were introduced in the mid-1950s in statistical physics and started appearing in the fields of applied statistics, signal processing, and neural networks in the 1980s and 1990s (Holmes & Mallick, 1998; Neal, 1996; Rios Insua & Müller, 1998; Robert & Casella, 1999; Tierney, 1994). The key idea is to build an ergodic Markov chain $(k^{(i)}, \boldsymbol{\theta}^{(i)}, \boldsymbol{\psi}^{(i)})_{i \in \mathbb{N}}$ whose equilibrium distribution is the desired posterior distribution. Under weak additional assumptions, the $P \gg 1$ samples generated by the Markov chain are asymptotically distributed according to the posterior distribution and thus allow

easy evaluation of all posterior features of interest—for example:

$$\widehat{p}(k = j \mid \mathbf{x}, \mathbf{y}) = \frac{1}{P} \sum_{i=1}^{P} \mathbb{I}_{\{j\}}(k^{(i)})$$

and

$$\widehat{\mathbb{E}}(\boldsymbol{\theta} \mid k = j, \mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{P} \boldsymbol{\theta}^{(i)} \mathbb{I}_{\{j\}}(k^{(i)})}{\sum_{i=1}^{P} \mathbb{I}_{\{j\}}(k^{(i)})}. \tag{3.2}$$

In addition, we can obtain predictions, such as:

$$\widehat{\mathbb{E}}(\mathbf{y}_{N+1} \mid \mathbf{x}_{1:N+1}, \mathbf{y}_{1:N}) = \frac{1}{P} \sum_{i=1}^{P} \mathbf{D}(\boldsymbol{\mu}_{1:k}^{(i)}, \mathbf{x}_{N+1}) \boldsymbol{\alpha}_{1:m}^{(i)}.$$

3.3 Integration of the Nuisance Parameters. The proposed Bayesian model allows for the integration of the so-called nuisance parameters, $\boldsymbol{\alpha}_{1:m}$ and $\boldsymbol{\sigma}^2$, and subsequently to obtain an expression for $p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y})$ up to a normalizing constant. By applying Bayes's theorem, multiplying the exponential terms of the likelihood and coefficients prior, and completing squares, we obtain the following expression for the full posterior distribution:

$$\begin{aligned}
&p(k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y}) \\
&\quad \propto \left[\prod_{i=1}^{c} (2\pi\boldsymbol{\sigma}_i^2)^{-N/2} \exp\left(-\frac{1}{2\boldsymbol{\sigma}_i^2} \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k} \mathbf{y}_{1:N,i} \right) \right] \\
&\qquad \times \left[\prod_{i=1}^{c} |2\pi\boldsymbol{\sigma}_i^2 \boldsymbol{\Sigma}_i|^{-1/2} \right. \\
&\qquad\quad \left. \times \exp\left(-\frac{1}{2\boldsymbol{\sigma}_i^2} \left(\boldsymbol{\alpha}_{1:m,i} - \mathbf{h}_{i,k} \right)' \mathbf{M}_{i,k}^{-1} \left(\boldsymbol{\alpha}_{1:m,i} - \mathbf{h}_{i,k} \right) \right) \right] \\
&\qquad \times \left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}_{1:k})}{\Im^k} \right] \left[\frac{\Lambda^k / k!}{\sum_{j=0}^{k_{\max}} \Lambda^j / j!} \right] \left[\prod_{i=1}^{c} (\boldsymbol{\sigma}_i^2)^{-(\upsilon_0/2+1)} \exp\left(-\frac{\gamma_0}{2\boldsymbol{\sigma}_i^2} \right) \right] \\
&\qquad \times \left[\prod_{i=1}^{c} (\boldsymbol{\delta}_i^2)^{-(\alpha_{\delta^2}+1)} \exp\left(-\frac{\beta_{\delta^2}}{\boldsymbol{\delta}_i^2} \right) \right] \left[(\Lambda)^{(\varepsilon_1 - 1/2)} \exp(-\varepsilon_2 \Lambda) \right],
\end{aligned}$$

where

$$\begin{aligned}
\mathbf{M}_{i,k}^{-1} &= \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}) + \boldsymbol{\Sigma}_i^{-1}, \qquad \mathbf{h}_{i,k} = \mathbf{M}_{i,k} \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \mathbf{y}_{1:N,i} \\
\mathbf{P}_{i,k} &= \mathbf{I}_N - \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \mathbf{M}_{i,k} \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x}).
\end{aligned}$$

We can now integrate with respect to $\boldsymbol{\alpha}_{1:m}$ (gaussian distribution) and $\boldsymbol{\sigma}_i^2$ (inverse gamma distribution) to obtain the following expression for the posterior:

$$p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y}) \propto \left[\prod_{i=1}^{c} \left(1 + \boldsymbol{\delta}_i^2\right)^{-m/2} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k} \mathbf{y}_{1:N,i}}{2} \right)^{\left(-\frac{N+v_0}{2}\right)} \right]$$

$$\times \left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}_k)}{\Im^k} \right] \left[\frac{\Lambda^k / k!}{\sum_{j=0}^{k_{\max}} \Lambda^j / j!} \right] \left[\prod_{i=1}^{c} (\boldsymbol{\delta}_i^2)^{-(\alpha_{\delta^2}+1)} \exp\left(-\frac{\beta_{\boldsymbol{\delta}^2}}{\boldsymbol{\delta}_i^2} \right) \right]$$

$$\times \left[(\Lambda)^{(\varepsilon_1 - 1/2)} \exp\left(-\varepsilon_2 \Lambda\right) \right]. \tag{3.3}$$

It is worth noticing that the posterior distribution is highly nonlinear in the RBF centers $\boldsymbol{\mu}_k$ and that an expression of $p(k \mid \mathbf{x}, \mathbf{y})$ cannot be obtained in closed form.

4 Bayesian Computation

For clarity, we assume that k is given. After dealing with this fixed-dimension scenario, we present an algorithm where k is treated as an unknown random variable.

4.1 MCMC Sampler for Fixed Dimension. We propose the following hybrid MCMC sampler, which combines Gibbs steps and Metropolis-Hastings (MH) steps (Gilks, Richardson, & Spiegelhalter, 1996; Tierney, 1994):

Fixed-Dimension MCMC Algorithm

1. Initialization. Fix the value of k and set $(\theta^{(0)}, \psi^{(0)})$ and $i = 1$.
2. Iteration i
   - For $j = 1, \ldots, k$
     - — Sample $u \sim \mathcal{U}_{[0,1]}$.
     - — If $u < \varpi$, perform an MH step admitting $p(\mu_{j,1:d} \mid \mathbf{x}, \mathbf{y}, \mu^{(i)}_{-j,1:d})$ as invariant distribution and $q_1(\mu^{\star}_{j,1:d} \mid \mu^{(i)}_{j,1:d})$ as proposal distribution.
     - — Else perform an MH step using $p(\mu_{j,1:d} \mid \mathbf{x}, \mathbf{y}, \mu^{(i)}_{-j,1:d})$ as invariant distribution and $q_2(\mu^{\star}_{j,1:d} \mid \mu^{(i)}_{j,1:d})$ as proposal distribution.

     End For.
   - Sample the nuisance parameters $(\alpha^{(i)}_{1:m}, \sigma^{2(i)})$ using equations 4.1 and 4.2.

- Sample the hyperparameters $(\Lambda^{(i)}, \delta^{2(i)})$ using equations 4.3 and 4.4.

3. $i \leftarrow i + 1$ and go to 2.

The simulation parameter ϖ is a real number satisfying $0 < \varpi < 1$. Its value indicates our belief on which proposal distribution leads to faster convergence. If we have no preference for a particular proposal, we can set it to 0.5. The various steps of the algorithm are detailed in the following sections. In order to simplify the notation, we drop the superscript $\cdot^{(i)}$ from all variables at iteration i.

4.1.1 Updating the RBF Centers. Sampling the RBF centers is difficult because the distribution is nonlinear in these parameters. We have chosen here to sample them one at a time using a mixture of MH steps. An MH step of invariant distribution, say $\pi(\mathbf{z})$, and proposal distribution, say $q(\mathbf{z}^\star \mid \mathbf{z})$, involves sampling a candidate value $\mathbf{z}^\star$ given the current value $\mathbf{z}$ according to $q(\mathbf{z}^\star \mid \mathbf{z})$. The Markov chain then moves toward $\mathbf{z}^\star$ with probability $\mathcal{A}(\mathbf{z}, \mathbf{z}^\star) \triangleq \min\{1, (\pi(\mathbf{z})q(\mathbf{z}^\star \mid \mathbf{z}))^{-1}\pi(\mathbf{z}^\star)\, q(\mathbf{z} \mid \mathbf{z}^\star)\}$; otherwise, it remains equal to $\mathbf{z}$. This algorithm is very general, but to perform well in practice, it is necessary to use "clever" proposal distributions to avoid rejecting too many candidates.

According to equation 3.3, the target distribution is the full conditional distribution of a basis center:

$$p(\boldsymbol{\mu}_{j,1:d} \mid \mathbf{x}, \mathbf{y}, \boldsymbol{\mu}_{-j,1:d}) \propto \left[\prod_{i=1}^{c} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{2}\right)^{\left(-\frac{N+v_0}{2}\right)}\right] \mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}_{1:k}),$$

where $\boldsymbol{\mu}_{-j,1:d}$ denotes $\{\boldsymbol{\mu}_{1,1:d}, \boldsymbol{\mu}_{2,1:d}, \ldots, \boldsymbol{\mu}_{j-1,1:d}, \boldsymbol{\mu}_{j+1,1:d}, \ldots, \boldsymbol{\mu}_{k,1:d}\}$.

With probability $0 < \varpi < 1$, the proposal $q_1(\boldsymbol{\mu}^\star_{j,1:d} \mid \boldsymbol{\mu}_{j,1:d})$ corresponds to randomly sampling a basis center from the interval $[\min(\mathbf{x}_{1:N,i}) - \iota\Xi_i, \max(\mathbf{x}_{1:N,i}) + \iota\Xi_i]^k$ for $i = 1, \ldots, d$. The motivation for using such a proposal distribution is that the regions where the data are dense are reached quickly. Subsequently, with probability $1-\varpi$, we perform an MH step with proposal distribution $q_2(\boldsymbol{\mu}^\star_{j,1:d} \mid \boldsymbol{\mu}_{j,1:d})$:

$$\boldsymbol{\mu}^\star_{j,1:d} \mid \boldsymbol{\mu}_{j,1:d} \sim \mathcal{N}(\boldsymbol{\mu}_{j,1:d}, \sigma^2_{RW}\mathbf{I}_d).$$

This proposal distribution yields a candidate $\boldsymbol{\mu}^\star_{j,1:d}$, which is a perturbation of the current center. The perturbation is a zero-mean gaussian random variable with variance $\sigma^2_{RW}\mathbf{I}_d$. This random walk is introduced to perform a

local exploration of the posterior distribution. In both cases, the acceptance probability is given by:

$$\mathcal{A}(\boldsymbol{\mu}_{j,1:d}, \boldsymbol{\mu}^{\star}_{j,1:d}) = \min\left\{1, \left[\prod_{i=1}^{c}\left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}^{\star}_{i,k}\mathbf{y}_{1:N,i}}\right)^{\left(\frac{N+\upsilon_0}{2}\right)}\right]\mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}^{\star}_{1:k})\right\},$$

where $\mathbf{P}^{\star}_{i,k}$ and $\mathbf{M}^{\star}_{i,k}$ are similar to $\mathbf{P}_{i,k}$ and $\mathbf{M}_{i,k}$ with $\boldsymbol{\mu}_{1:k,1:d}$ replaced by $\{\boldsymbol{\mu}_{1,1:d}, \boldsymbol{\mu}_{2,1:d}, \ldots, \boldsymbol{\mu}_{j-1,1:d}, \boldsymbol{\mu}^{\star}_{j,1:d}, \boldsymbol{\mu}_{j+1,1:d}, \ldots, \boldsymbol{\mu}_{k,1:d}\}$. We have found that the combination of these proposal distributions works well in practice.

4.1.2 Sampling the Nuisance Parameters. In section 3.3, we derived an expression for $p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y})$ from the full posterior distribution $p(k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y})$ by performing some algebraic manipulations and integrating with respect to $\boldsymbol{\alpha}_{1:m}$ (gaussian distribution) and $\boldsymbol{\sigma}^2$ (inverse gamma distribution). As a result, if we take into consideration that

$$\begin{aligned} p(k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y}) &= p(\boldsymbol{\alpha}_{1:m} \mid k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2, \mathbf{x}, \mathbf{y}) p(k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y}) \\ &= p(\boldsymbol{\alpha}_{1:m} \mid k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2, \mathbf{x}, \mathbf{y}) p(\boldsymbol{\sigma}^2 \mid k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2, \mathbf{x}, \mathbf{y}) \\ &\quad \times p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y}), \end{aligned}$$

it follows that for $i = 1, \ldots, c$, $\boldsymbol{\alpha}_{1:m,i}$, and $\boldsymbol{\sigma}_i^2$ are distributed according to:

$$\boldsymbol{\sigma}_i^2 \mid (k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\delta}^2, \mathbf{x}, \mathbf{y}) \sim \mathcal{IG}\left(\frac{\upsilon_0 + N}{2}, \frac{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{2}\right) \quad (4.1)$$

$$\boldsymbol{\alpha}_{1:m,i} \mid (k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \boldsymbol{\delta}^2, \mathbf{x}, \mathbf{y}) \sim \mathcal{N}(\mathbf{h}_{i,k}, \boldsymbol{\sigma}_i^2\mathbf{M}_{i,k}). \quad (4.2)$$

4.1.3 Sampling the Hyperparameters. By considering $p(k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y})$, we can clearly see that the hyperparameters $\boldsymbol{\delta}_i$ (for $i = 1, \ldots, c$) can be simulated from the full conditional distribution:

$$\boldsymbol{\delta}_i^2 \mid (k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}_i^2, \mathbf{x}, \mathbf{y}) \sim \mathcal{IG}\left(\alpha_{\delta^2} + \frac{m}{2}, \beta_{\delta^2} + \frac{1}{2\boldsymbol{\sigma}_i^2}\boldsymbol{\alpha}'_{1:m,i}\mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})\boldsymbol{\alpha}_{1:m,i}\right). \quad (4.3)$$

On the other hand, an expression for the posterior distribution of Λ is not so straightforward because the prior for k is a truncated Poisson distribution. Λ can be simulated using the MH algorithm with a proposal corresponding to the full conditional that would be obtained if the prior for k was an infinite

Poisson distribution. That is, we can use the following gamma proposal for Λ,

$$q(\Lambda^\star) \propto \Lambda^{\star(1/2+\varepsilon_1+k)} \exp\left(-(1+\varepsilon_2)\Lambda^\star\right), \tag{4.4}$$

and subsequently perform an MH step with the full conditional distribution $p(\Lambda \mid k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\delta}^2, \mathbf{x}, \mathbf{y})$ as invariant distribution.

4.2 MCMC Sampler for Unknown Dimension. Now let us consider the case where k is unknown. Here, the Bayesian computation for the estimation of the joint posterior distribution $p(k, \boldsymbol{\theta}, \psi \mid \mathbf{x}, \mathbf{y})$ is even more complex. One obvious solution would consist of upper bounding k by, say, $k_{\max}$ and running $k_{\max} + 1$ independent MCMC samplers, each being associated to a fixed number $k = 0, \ldots, k_{\max}$. However, this approach suffers from severe drawbacks. First, it is computationally very expensive since $k_{\max}$ can be large. Second, the same computational effort is attributed to each value of k. In fact, some of these values are of no interest in practice because they have a very weak posterior model probability $p(k \mid \mathbf{x}, \mathbf{y})$. Another solution would be to construct an MCMC sampler that would be able to sample directly from the joint distribution on $\boldsymbol{\Theta} \times \boldsymbol{\Psi} = (\cup_{k=0}^{k_{\max}} \{k\} \times \boldsymbol{\Theta}_k) \times \boldsymbol{\Psi}$. Standard MCMC methods are not able to "jump" between subspaces $\boldsymbol{\Theta}_k$ of different dimensions. However, Green (1995) has introduced a new, flexible class of MCMC samplers, the so-called reversible-jump MCMC, that are capable of jumping between subspaces of different dimensions. This is a general state-space MH algorithm (see Andrieu, Djurić, & Doucet, in press, for an introduction). One proposes candidates according to a set of proposal distributions. These candidates are randomly accepted according to an acceptance ratio, which ensures reversibility and thus invariance of the Markov chain with respect to the posterior distribution. Here, the chain must move across subspaces of different dimensions, and therefore the proposal distributions are more complex (see Green, 1995 and Richardson & Green, 1997, for details). For our problem, the following moves have been selected:

1. Birth of a new basis, that is, proposing a new basis function in the interval $[\min(\mathbf{x}_{1:N,i}) - \iota\Xi_i, \max(\mathbf{x}_{1:N,i}) + \iota\Xi_i]^k$ for $i = 1, \ldots, d$.
2. Death of an existing basis, that is, removing a basis function chosen randomly.
3. Merge a randomly chosen basis function and its closest neighbor into a single basis function.
4. Split a randomly chosen basis function into two neighbor basis functions, such that the distance between them is shorter than the distance between the proposed basis function and any other existing basis function. This distance constraint ensures reversibility.
5. Update the RBF centers.

These moves are defined by heuristic considerations, the only condition to be fulfilled being to maintain the correct invariant distribution. A particular choice will have influence only on the convergence rate of the algorithm. The birth and death moves allow the network to grow from k to $k+1$ and decrease from k to $k-1$, respectively. The split and merge moves also perform dimension changes from k to $k+1$ and k to $k-1$. The merge move serves to avoid the problem of placing too many basis functions in the same neighborhood. That is, when amplitudes of many basis functions, in a close neighborhood, add to the amplitude that would be obtained by using fewer basis functions, the merge move combines some of these basis functions. On the other hand, the split move is useful in regions of the data where there are close components. Other moves may be proposed, but we have found that the ones suggested here lead to satisfactory results. In particular, we started with the update, birth, and death moves and noticed that by incorporating split and merge moves, the estimation results improved.

The resulting transition kernel of the simulated Markov chain is then a mixture of the different transition kernels associated with the moves described above. This means that at each iteration, one of the candidate moves—birth, death, merge, split, or update—is randomly chosen. The probabilities for choosing these moves are b_k, d_k, m_k, s_k, and u_k, respectively, such that $b_k + d_k + m_k + s_k + u_k = 1$ for all $0 \leq k \leq k_{\max}$. A move is performed if the algorithm accepts it. For $k = 0$ the death, split, and merge moves are impossible, so that $d_0 \triangleq 0$, $s_0 \triangleq 0$, and $m_0 \triangleq 0$. The merge move is also not permitted for $k = 1$, that is, $m_1 \triangleq 0$. For $k = k_{\max}$, the birth and split moves are not allowed, and therefore $b_{k_{\max}} \triangleq 0$ and $s_{k_{\max}} \triangleq 0$. Except in the cases described above, we adopt the following probabilities:

$$b_k \triangleq c^\star \min\left\{1, \frac{p\left(k+1 \mid \Lambda\right)}{p\left(k \mid \Lambda\right)}\right\},$$

$$d_{k+1} \triangleq c^\star \min\left\{1, \frac{p\left(k \mid \Lambda\right)}{p\left(k+1 \mid \Lambda\right)}\right\}, \tag{4.5}$$

where $p(k \mid \Lambda)$ is the prior probability of model $\mathcal{M}_k$ and $c^\star$ is a parameter that tunes the proportion of dimension and update moves. As Green (1995) pointed out, this choice ensures that $b_k p(k \mid \Lambda)[d_{k+1} p(k+1 \mid \Lambda)]^{-1} = 1$, which means that an MH algorithm in a single dimension, with no observations, would have 1 as acceptance probability. We take $c^\star = 0.25$ and then $b_k + d_k + m_k + s_k \in [0.25, 1]$ for all k (Green, 1995). In addition, we choose $m_k = d_k$ and $s_k = b_k$. We can now describe the main steps of the algorithm as follows:

Reversible-Jump MCMC Algorithm

1. Initialization: set $(k^{(0)}, \theta^{(0)}, \psi^{(0)}) \in \boldsymbol{\Theta} \times \boldsymbol{\Psi}$.

2. Iteration i.

   - Sample $u \sim \mathcal{U}_{[0,1]}$.
   - If $(u \leq b_{k^{(i)}})$
     - — then birth move (see section 4.2.1).
     - — else if $(u \leq b_{k^{(i)}} + d_{k^{(i)}})$ then death move (see section 4.2.1).
     - — else if $(u \leq b_{k^{(i)}} + d_{k^{(i)}} + s_{k^{(i)}})$ then split move (see section 4.2.2).
     - — else if $(u \leq b_{k^{(i)}} + d_{k^{(i)}} + s_{k^{(i)}} + m_{k^{(i)}})$ then merge move (see section 4.2.2).
     - — else update the RBF centers (see section 4.2.3).

     End If.
   - Sample the nuisance parameters $(\sigma_k^{2(i)}, \alpha_k^{(i)})$ using equations 4.1 and 4.2.
   - Simulate the hyperparameters $(\Lambda^{(i)}, \delta^{2(i)})$ using equations 4.3 and 4.4.

3. $i \leftarrow i + 1$ and go to 2.

We expand on these different moves in the following sections. Once again, in order to simplify the notation, we drop the superscript $\cdot^{(i)}$ from all variables at iteration i.

4.2.1 Birth and Death Moves. Suppose that the current state of the Markov chain is in $\{k\} \times \boldsymbol{\Theta}_k \times \boldsymbol{\Psi}$. Then the algorithm for the birth and death moves is as follows:

Birth Move

1. Propose a new basis location at random from the interval $[\min(\mathbf{x}_{1:N,i}) - \iota\Xi_i, \max(\mathbf{x}_{1:N,i}) + \iota\Xi_i]$ for $i = 1, \ldots, d$.
2. Evaluate $\mathcal{A}_{birth}$ (see equation 4.7), and sample $u \sim \mathcal{U}_{[0,1]}$.
3. If $u \leq \mathcal{A}_{birth}$, then the state of the Markov chain becomes $(k + 1, \mu_{1:k+1})$, else it remains equal to $(k, \mu_{1:k})$.

Death move

1. Choose the basis center to be deleted, at random among the k existing basis.
2. Evaluate $\mathcal{A}_{death}$ (see equation 4.7), and sample $u \sim \mathcal{U}_{[0,1]}$.
3. If $u \leq \mathcal{A}_{death}$ then the state of the Markov chain becomes $(k - 1, \mu_{1:k-1})$, else it remains equal to $(k, \mu_{1:k})$.

The acceptance ratio for the proposed birth move is deduced from the following expression (Green, 1995):

$$r_{birth} \triangleq (\textit{posterior distributions ratio}) \times (\textit{proposal ratio}) \times (\textit{Jacobian})$$
$$= \frac{p(k+1, \boldsymbol{\mu}_{1:k+1}, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y})}{p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 | \mathbf{x}, \mathbf{y})} \times \frac{d_{k+1}/(k+1)}{b_k/\Im} \times \left| \frac{\partial(\boldsymbol{\mu}_{1:k+1})}{\partial(\boldsymbol{\mu}_{1:k}, \boldsymbol{\mu}^\star)} \right|.$$

Clearly, the Jacobian is equal to 1, and after simplifications we obtain:

$$r_{birth} = \left[\prod_{i=1}^{c} \frac{1}{(1+\boldsymbol{\delta}_i^2)^{1/2}} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k} \mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k+1} \mathbf{y}_{1:N,i}} \right)^{\left(\frac{N+\upsilon_0}{2}\right)} \right] \frac{1}{(k+1)}.$$

Similarly, for the death move:

$$r_{death} = \frac{p(k-1, \boldsymbol{\mu}_{1:k-1}, \Lambda, \boldsymbol{\delta}^2 | \mathbf{x}, \mathbf{y})}{p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 | \mathbf{x}, \mathbf{y})} \times \frac{b_{k-1}/\Im}{d_k/k} \times \left| \frac{\partial(\boldsymbol{\mu}_{1:k-1}, \boldsymbol{\mu}^\star)}{\partial(\boldsymbol{\mu}_{1:k})} \right|$$
$$= \left[\prod_{i=1}^{c} (1+\boldsymbol{\delta}_i^2)^{1/2} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k} \mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k-1} \mathbf{y}_{1:N,i}} \right)^{\left(\frac{N+\upsilon_0}{2}\right)} \right] k. \tag{4.6}$$

The acceptance probabilities corresponding to the described moves are:

$$\mathcal{A}_{birth} = \min\{1, r_{birth}\}, \qquad \mathcal{A}_{death} = \min\{1, r_{death}\} \tag{4.7}$$

4.2.2 Split and Merge Moves. The merge move involves randomly selecting a basis function ($\boldsymbol{\mu}_1$) and then combining it with its closest neighbor ($\boldsymbol{\mu}_2$) into a single basis function $\boldsymbol{\mu}$, whose new location is

$$\boldsymbol{\mu} = \frac{\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2}{2}. \tag{4.8}$$

The corresponding split move that guarantees reversibility is

$$\boldsymbol{\mu}_1 = \boldsymbol{\mu} - u_{ms} \varsigma^\star$$
$$\boldsymbol{\mu}_2 = \boldsymbol{\mu} + u_{ms} \varsigma^\star, \tag{4.9}$$

where $\varsigma^\star$ is a simulation parameter and $u_{ms} \sim \mathcal{U}_{[0,1]}$. To ensure reversibility, we perform the merge move only if $\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\| < 2\varsigma^\star$. Suppose now that the current state of the Markov chain is in $\{k\} \times \boldsymbol{\Theta}_k \times \boldsymbol{\Psi}$. Then the algorithm for the split and merge moves is as follows:

Split Move

1. Randomly choose an existing RBF center.

2. Substitute it for two neighbor basis functions, whose centers are obtained using equation 4.9. The new centers must be bound to lie in the space $\mathbf{\Omega}_k$, and the distance (typically Euclidean) between them has to be shorter than the distance between the proposed basis function and any other existing basis function.
3. Evaluate $\mathcal{A}_{split}$ (see equation 4.10), and sample $u \sim \mathcal{U}_{[0,1]}$.
4. If $u \leq \mathcal{A}_{split}$ then the state of the Markov chain becomes $(k+1, \mu_{1:k+1})$, else it remains equal to $(k, \mu_{1:k})$.

Merge Move

1. Choose a basis center at random among the k existing basis. Then find the closest basis function to it applying some distance metric, for example, Euclidean.
2. If $\|\mu_1 - \mu_2\| < 2\varsigma^\star$, substitute the two basis functions for a single basis function in accordance with equation 4.8.
3. Evaluate $\mathcal{A}_{merge}$ (see equation 4.10), and sample $u \sim \mathcal{U}_{[0,1]}$.
4. If $u \leq \mathcal{A}_{merge}$, then the state of the Markov chain becomes $(k-1, \mu_{1:k-1})$, else it remains equal to $(k, \mu_{1:k})$.

The acceptance ratio for the proposed split move is given by:

$$r_{split} = \frac{p(k+1, \boldsymbol{\mu}_{1:k+1}, \Lambda, \boldsymbol{\delta}^2 | \mathbf{x}, \mathbf{y})}{p(k, \boldsymbol{\mu}_{1:k}, k, \Lambda, \boldsymbol{\delta}^2 | \mathbf{x}, \mathbf{y})} \times \frac{m_{k+1}/(k+1)}{p(u_{ms}) s_k / k} \times \left| \frac{\partial(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)}{\partial(\boldsymbol{\mu}, u_{ms})} \right|.$$

In this case, the Jacobian is equal to:

$$J_{split} = \left| \frac{\partial(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)}{\partial(\boldsymbol{\mu}, u_{ms})} \right| = \begin{vmatrix} 1 & 1 \\ -\varsigma^\star & \varsigma^\star \end{vmatrix} = 2\varsigma^\star,$$

and, after simplifications, we obtain:

$$r_{split} = \left[\prod_{i=1}^{c} \frac{1}{(1+\boldsymbol{\delta}_i^2)^{1/2}} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k} \mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k+1} \mathbf{y}_{1:N,i}} \right)^{\left(\frac{N+\upsilon_0}{2}\right)} \right] \frac{k\varsigma^\star}{\Im(k+1)}.$$

Similarly, for the merge move:

$$r_{merge} = \frac{p(k-1, \boldsymbol{\mu}_{1:k-1}, \Lambda, \boldsymbol{\delta}^2 | \mathbf{x}, \mathbf{y})}{p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 | \mathbf{x}, \mathbf{y})} \times \frac{s_{k-1}/(k-1)}{m_k / k} \times \left| \frac{\partial(\boldsymbol{\mu}, u_{ms})}{\partial(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)} \right|,$$

and, since $J_{merge} = 1/2\varsigma^{\star}$, it follows that

$$r_{merge} = \left[\prod_{i=1}^{c} (1+\delta_i^2)^{1/2} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k} \mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k-1} \mathbf{y}_{1:N,i}} \right)^{\left(\frac{N+v_0}{2}\right)} \right] \frac{k\Im}{\varsigma^{\star}(k-1)}.$$

The acceptance probabilities for the split and merge moves are:

$$\mathcal{A}_{split} = \min\left\{1, r_{split}\right\}, \qquad \mathcal{A}_{merge} = \min\left\{1, r_{merge}\right\}. \tag{4.10}$$

4.2.3 Update Move. The update move does not involve changing the dimension of the model. It requires an iteration of the fixed dimension MCMC sampler presented in section 4.1.1.

The method presented so far can be very accurate, yet it can be computationally demanding. In the following section, we present a method that requires optimization instead of integration to obtain estimates of the parameters and model dimension. This method, although less accurate, as shown in section 7, is less computationally demanding. The choice of one method over the other should ultimately depend on the modeling constraints and specifications.

5 Reversible-Jump Simulated Annealing

In this section, we show that traditional model selection criteria within a penalized likelihood framework, such as AIC, BIC, and MDL (Akaike, 1974; Schwarz, 1985; Rissanen, 1987), can be shown to correspond to particular hyperparameter choices in our hierarchical Bayesian formulation. (As pointed out by one of the reviewers, AIC is not restricted to the situation where the true model belongs to the set of model candidates.) That is, we can calibrate the prior choices so that the problem of model selection within the penalized likelihood context can be mapped exactly to a problem of model selection via posterior probabilities. This technique has been previously applied to the problem of variable selection (George & Foster, 1997).

After resolving the calibration problem, we perform maximum likelihood estimation, with the model selection criteria, by maximizing the calibrated posterior distribution. To accomplish this goal, we adopt an MCMC simulated annealing algorithm, which makes use of the homogeneous reversible-jump MCMC kernel as proposal distribution. This approach has the advantage that we can start with an arbitrary model order, and the algorithm will perform dimension jumps until it finds the "true" model order. That is, we do not have to resort to the more expensive task of running a fixed-dimension algorithm for each possible model order and subsequently select the best model.

5.1 Penalized Likelihood Model Selection. Traditionally, penalized likelihood model order selection strategies, based on standard information criteria, require the evaluation of the maximum likelihood (ML) estimates for each model order. The number of required evaluations can be prohibitively expensive unless appropriate heuristics are available. Subsequently, a particular model $\mathcal{M}_s$ is selected if it is the one that minimizes the sum of the the log-likelihood and a penalty term that depends on the model dimension (Djurić, 1998; Gelfand & Dey, 1997). In mathematical terms, this estimate is given by:

$$\mathcal{M}_s = \underset{\mathcal{M}_k:\ k\in\{0,\ldots,k_{\max}\}}{\arg\min} \left\{ -\log(p(\mathbf{y} \mid k, \widehat{\boldsymbol{\theta}}, \mathbf{x})) + \mathcal{P} \right\}, \tag{5.1}$$

where $\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{\alpha}}_{1:m}, \widehat{\boldsymbol{\mu}}_{1:k}, \widehat{\boldsymbol{\sigma}}_k^2)$ is the ML estimate of $\boldsymbol{\theta}$ for model $\mathcal{M}_k$. $\mathcal{P}$ is a penalty term that depends on the model order. Examples of ML penalties include the well-known AIC, BIC, or MDL information criteria (Akaike, 1974; Schwarz, 1985; Rissanen, 1987). The expressions for these in the case of gaussian observation noise are

$$\mathcal{P}_{\text{AIC}} = \xi \qquad \text{and} \qquad \mathcal{P}_{\text{BIC}} = \mathcal{P}_{\text{MDL}} = \frac{\xi}{2}\log(N),$$

where ξ denotes the number of model parameters ($k(c+1)+c(1+d)$) in the case of an RBF network). These criteria are motivated by different factors: AIC is based on expected information, BIC is an asymptotic Bayes factor, and MDL involves evaluating the minimum information required to transmit some data and a model, which describes the data, over a communications channel.

Using the conventional estimate of the variance for gaussian distributions,

$$\begin{aligned}\widehat{\boldsymbol{\sigma}}_i^2 &= \frac{1}{N}\left(\mathbf{y}_{1:N,i} - \mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k}, \mathbf{x})\widehat{\boldsymbol{\alpha}}_{1:m,i}\right)'\left(\mathbf{y}_{1:N,i} - \mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k}, \mathbf{x})\widehat{\boldsymbol{\alpha}}_{1:m,i}\right)\\ &= \frac{1}{N}\mathbf{y}'_{1:N,i}\mathbf{P}^*_{i,k}\mathbf{y}_{1:N,i},\end{aligned}$$

where $\mathbf{P}^*_{i,k}$ is the least-squares orthogonal projection matrix,

$$\mathbf{P}^*_{i,k} = \mathbf{I}_N - \mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k}, \mathbf{x})\left[\mathbf{D}'(\widehat{\boldsymbol{\mu}}_{1:k}, \mathbf{x})\mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k}, \mathbf{x})\right]^{-1}\mathbf{D}'(\widehat{\boldsymbol{\mu}}_{1:k}, \mathbf{x}),$$

we can expand equation 5.1 as follows:

$$\mathcal{M}_s = \underset{\mathcal{M}_k:k\in\{0,\ldots,k_{\max}\}}{\arg\min} \left\{ -\log\left[\prod_{i=1}^{c}(2\pi\widehat{\boldsymbol{\sigma}}_i^2)^{-N/2}\right.\right.$$

$$\times \exp\left(-\frac{1}{2\widehat{\boldsymbol{\sigma}}_i^2}\left(\mathbf{y}_{1:N,i} - \mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k}, \mathbf{x})\widehat{\boldsymbol{\alpha}}_{1:m,i}\right)'\right.$$

$$\left.\left.\times \left(\mathbf{y}_{1:N,i} - \mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k}, \mathbf{x})\widehat{\boldsymbol{\alpha}}_{1:m,i}\right)\right)\right] + \mathcal{P}\Bigg\}$$

$$= \underset{\mathcal{M}_k : k \in \{0,\ldots,k_{\max}\}}{\arg\max} \left\{\left[\prod_{i=1}^{c} (\mathbf{y}'_{1:N,i}\mathbf{P}^*_{i,k}\mathbf{y}_{1:N,i})^{-N/2}\right]\exp(-\mathcal{P})\right\}. \tag{5.2}$$

In the following section, we show that calibrating the priors in our hierarchical Bayes model will lead to the expression given by equation 5.2.

5.2 Calibration. It is useful and elucidating to impose some restrictions on the Bayesian hierarchical prior (see equation 3.3) to obtain the AIC and MDL criteria. We begin by assuming that the hyperparameter $\boldsymbol{\delta}$ is fixed to a particular value, say $\overline{\boldsymbol{\delta}}$, and that we no longer have a definite expression for the model prior $p(k)$, so that

$$p(k, \boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y}) \propto \left[\prod_{i=1}^{c} (1 + \overline{\boldsymbol{\delta}}_i^2)^{-m/2} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{2}\right)^{\left(-\frac{N+\upsilon_0}{2}\right)}\right]\left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}_k)}{\Im^k}\right] p(k).$$

Furthermore, we set $\upsilon_0 = 0$ and $\gamma_0 = 0$ to obtain Jeffreys's uninformative prior $p(\boldsymbol{\sigma}_i^2) \propto 1/\boldsymbol{\sigma}_i^2$. Consequently, we obtain the following expression:

$$p(k, \boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y}) \propto \left[\prod_{i=1}^{c} \left(1 + \overline{\boldsymbol{\delta}}_i^2\right)^{-k/2} \left(\mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}\right)^{-\frac{N}{2}}\right]\left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}_k)}{\Im^k}\right] p(k),$$

where $\mathbf{M}_{i,k}^{-1} = (1 + \overline{\boldsymbol{\delta}}_i^{-2})\mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})$, $\mathbf{h}_{i,k} = \mathbf{M}_{i,k}\mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x})\mathbf{y}_{1:N,i}$, and $\mathbf{P}_{i,k} = \mathbf{I}_N - \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})\mathbf{M}_{i,k}\mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x})$. Finally, we can select $\overline{\boldsymbol{\delta}}_i^2$ and $p(k)$ such that

$$\left[\prod_{i=1}^{c} \left(1 + \overline{\boldsymbol{\delta}}_i^2\right)^{-k/2}\right]\left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}_k)}{\Im^k}\right] p(k) = \exp(-\mathcal{P}) \propto \exp(-\mathcal{C}k),$$

thereby ensuring that the expression for the calibrated posterior distribution $p(k, \boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y})$ corresponds to the term that needs to be maximized in the penalized likelihood framework (see equation 5.2). Note that for the purposes of optimization, we only need the proportionality condition with $\mathcal{C} = c + 1$ for the AIC criterion and $\mathcal{C} = (c + 1)\log(N)/2$ for the MDL and

BIC criteria. We could, for example, satisfy the proportionality by remaining in the compact set Ω and choosing the prior $p(k) = \frac{\Lambda^k}{\sum_{j=0}^{k_{\max}} \Lambda^j}$, with the following fixed value for Λ:

$$\overline{\Lambda} = \left[\prod_{i=1}^{c} \left(1 + \overline{\boldsymbol{\delta}}_i^2 \right)^{\frac{1}{2}} \right] \Im \exp\left(-\mathcal{C}\right). \tag{5.3}$$

In addition, we have to let $\overline{\boldsymbol{\delta}} \to \infty$ so that $\mathbf{P}_{i,k} \to \mathbf{P}^*_{i,k}$.

We have thus shown that by calibrating the priors in the hierarchical Bayesian formulation, in particular by treating Λ and $\boldsymbol{\delta}^2$ as fixed quantities instead of as random variables, letting $\overline{\boldsymbol{\delta}} \to \infty$, choosing an uninformative Jeffreys's prior for $\boldsymbol{\sigma}^2$ and setting Λ as in equation 5.3, we can obtain the expression that needs to be maximized in the classical penalized likelihood formulation with AIC, MDL, and BIC model selection criteria. Consequently, we can interpret the penalized likelihood framework as a problem of maximizing the joint posterior distribution $p(k, \boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y})$. Effectively, we can obtain this MAP estimate as follows:

$$\begin{aligned}
(k, \boldsymbol{\mu}_{1:k})_{\text{MAP}} &= \arg\max_{k, \mu_{1:k} \in \boldsymbol{\Omega}} p(k, \boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y}) \\
&= \arg\max_{k, \mu_{1:k} \in \boldsymbol{\Omega}} \left\{ \left[\prod_{i=1}^{c} (\mathbf{y}'_{1:N,i} \mathbf{P}^*_{i,k} \mathbf{y}_{1:N,i})^{-N/2} \right] \exp(-\mathcal{P}) \right\}.
\end{aligned}$$

The sufficient conditions that need to be satisfied so that the distribution $p(k, \boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y})$ is proper are not overly restrictive. First, $\boldsymbol{\Omega}$ has to be a compact set, which is not a problem in our setting. Second, $\mathbf{y}'_{1:N,i} \mathbf{P}^*_{i,k} \mathbf{y}_{1:N,i}$ has to be larger than zero for $i = 1, \ldots, c$. In appendix B, lemma 1, we show that this is the case unless $\mathbf{y}_{1:N,i}$ spans the space of the columns of $\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})$, in which case $\mathbf{y}'_{1:N,i} \mathbf{P}^*_{i,k} \mathbf{y}_{1:N,i} = 0$. This event has probability zero.

5.3 Reversible-Jump Simulated Annealing. From an MCMC perspective, we can solve the stochastic optimization problem posed in the previous section by adopting a simulated annealing strategy (Geman & Geman, 1984; Van Laarhoven & Arts, 1987). The simulated annealing method involves simulating a nonhomogeneous Markov chain whose invariant distribution at iteration i is no longer equal to $\pi(\mathbf{z})$, but to $\pi_i(\mathbf{z}) \propto \pi^{1/T_i}(\mathbf{z})$, where T_i is a decreasing cooling schedule with $\lim_{i \to +\infty} T_i = 0$. The reason for doing this is that under weak regularity assumptions on $\pi(\mathbf{z})$, $\pi^{\infty}(\mathbf{z})$ is a probability density that concentrates itself on the set of global maxima of $\pi(\mathbf{z})$.

As with the MH method, the simulated annealing method with distribution $\pi(\mathbf{z})$ and proposal distribution $q(\mathbf{z}^\star \mid \mathbf{z})$ involves sampling a candidate value $\mathbf{z}^\star$ given the current value $\mathbf{z}$ according to $q(\mathbf{z}^\star \mid \mathbf{z})$. The Markov chain

moves toward the candidate $\mathbf{z}^\star$ with probability $\mathcal{A}_{\mathrm{SA}}(\mathbf{z}, \mathbf{z}^\star) = \min\{1, (\pi^{1/T_i}(\mathbf{z})q(\mathbf{z}^\star \mid \mathbf{z}))^{-1}\pi^{1/T_i}(\mathbf{z}^\star)q(\mathbf{z} \mid \mathbf{z}^\star)\}$; otherwise, it remains equal to $\mathbf{z}$. If we choose the homogeneous transition kernel $\mathcal{K}(\mathbf{z}, \mathbf{z}^\star)$ of the reversible-jump algorithm as the proposal distribution and use the reversibility property,

$$\pi(\mathbf{z}^\star)\mathcal{K}(\mathbf{z}^\star, \mathbf{z}) = \pi(\mathbf{z})\mathcal{K}(\mathbf{z}, \mathbf{z}^\star),$$

it follows that

$$\mathcal{A}_{\mathrm{RJSA}} = \min\left\{1, \frac{\pi^{(1/T_i-1)}(\mathbf{z}^\star)}{\pi^{(1/T_i-1)}(\mathbf{z})}\right\}. \tag{5.4}$$

Consequently, the following algorithm, with $b_k = d_k = m_k = s_k = u_k = 0.2$, can find the joint MAP estimate of $\boldsymbol{\mu}_{1:k}$ and k:

Reversible-Jump Simulated Annealing

1. Initialization: set $(k^{(0)}, \theta^{(0)}) \in \boldsymbol{\Theta}$.
2. Iteration i.

   - Sample $u \sim \mathcal{U}_{[0,1]}$, and set the temperature according to the cooling schedule.
   - If $(u \leq b_{k^{(i)}})$
     - — then birth move (see equation 5.6).
     - — else if $(u \leq b_{k^{(i)}} + d_{k^{(i)}})$ then death move (see equation 5.6).
     - — else if $(u \leq b_{k^{(i)}} + d_{k^{(i)}} + s_{k^{(i)}})$ then split move (see equation 5.7).
     - — else if $(u \leq b_{k^{(i)}} + d_{k^{(i)}} + s_{k^{(i)}} + m_{k^{(i)}})$ then merge move (see equation 5.7).
     - — else update the RBF centers (see equation 5.5).

     End If.
   - Perform an MH step with the annealed acceptance ratio (see equation 5.4).
3. $i \leftarrow i + 1$ and go to 2.
4. Compute the coeffcients $\alpha_{1:m}$ by least squares (optimal in this case):

$$\widehat{\alpha}_{1:m,i} = [\mathbf{D}'(\mu_{1:k}, \mathbf{x})\mathbf{D}(\mu_{1:k}, \mathbf{x})]^{-1}\mathbf{D}'(\mu_{1:k}, \mathbf{x})\mathbf{y}_{1:N,i}.$$

We explain the simulated annealing moves in the following sections. To simplify the notation, we drop the superscript $\cdot^{(i)}$ from all variables at iteration i.

5.4 Moves. We sample the radial basis centers in the same way as explained in section 4.1.1. However, the target distribution is given by

$$p(\boldsymbol{\mu}_{j,1:d} \mid \mathbf{x}, \mathbf{y}, \boldsymbol{\mu}_{-j,1:d}) \propto \left[\prod_{i=1}^{c} (\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i})^{\left(-\frac{N}{2}\right)}\right] \exp(-\mathcal{P}),$$

and, consequently, the acceptance probability is

$$\mathcal{A}_{\text{update}}(\boldsymbol{\mu}_{j,1:d}, \boldsymbol{\mu}^{\star}_{j,1:d}) = \min\left\{1, \left[\prod_{i=1}^{c}\left(\frac{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i}\mathbf{P}^{\star}_{i,k}\mathbf{y}_{1:N,i}}\right)^{\left(\frac{N}{2}\right)}\right]\right\}, \tag{5.5}$$

where $\mathbf{P}^{\star}_{i,k}$ is similar to $\mathbf{P}^{*}_{i,k}$ with $\boldsymbol{\mu}_{1:k,1:d}$ replaced by $\{\boldsymbol{\mu}_{1,1:d}, \boldsymbol{\mu}_{2,1:d}, \ldots, \boldsymbol{\mu}_{j-1,1:d}, \boldsymbol{\mu}^{\star}_{j,1:d}, \boldsymbol{\mu}_{j+1,1:d}, \ldots, \boldsymbol{\mu}_{k,1:d}\}$. The birth and death moves are similar to the ones proposed in section 4.2.1, except that the expressions for r_{birth} and r_{death} (with $b_k = d_k = 0.2$) become

$$r_{birth} = \left[\prod_{i=1}^{c}\left(\frac{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k+1}\mathbf{y}_{1:N,i}}\right)^{\left(\frac{N}{2}\right)}\right]\frac{\Im\exp(-\mathcal{C})}{k+1}.$$

Similarly,

$$r_{death} = \left[\prod_{i=1}^{c}\left(\frac{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k-1}\mathbf{y}_{1:N,i}}\right)^{\left(\frac{N}{2}\right)}\right]\frac{k\exp(\mathcal{C})}{\Im}.$$

Hence, the acceptance probabilities corresponding to the described moves are

$$\mathcal{A}_{birth} = \min\{1, r_{birth}\}, \quad \mathcal{A}_{death} = \min\{1, r_{death}\}. \tag{5.6}$$

Similarly, the split and merge moves are analogous to the ones proposed in section 4.2.2, except that the expressions for r_{split} and r_{merge} (with $m_k = s_k = 0.2$) become

$$r_{split} = \left[\prod_{i=1}^{c}\left(\frac{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k+1}\mathbf{y}_{1:N,i}}\right)^{\left(\frac{N}{2}\right)}\right]\frac{k\varsigma^{\star}\exp(-\mathcal{C})}{k+1}$$

and

$$r_{merge} = \left[\prod_{i=1}^{c} \left(\frac{\mathbf{y}'_{1:N,i} \mathbf{P}^*_{i,k} \mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i} \mathbf{P}^*_{i,k-1} \mathbf{y}_{1:N,i}} \right)^{\left(\frac{N}{2}\right)} \right] \frac{k \exp(\mathcal{C})}{\varsigma^\star(k-1)}.$$

The acceptance probabilities for these moves are

$$\mathcal{A}_{split} = \min\left\{1, r_{split}\right\}, \qquad \mathcal{A}_{merge} = \min\left\{1, r_{merge}\right\}. \tag{5.7}$$

6 Convergence Results

It is easy to prove that the reversible-jump MCMC algorithm applied to the full Bayesian model converges, in other words, that the Markov chain $(k^{(i)}, \boldsymbol{\mu}^{(i)}_{1:k}, \Lambda^{(i)}, \boldsymbol{\delta}^{2(i)})_{i\in\mathbb{N}}$ is ergodic. We prove here a stronger result by showing that $(k^{(i)}, \boldsymbol{\mu}^{(i)}_{1:k}, \Lambda^{(i)}, \boldsymbol{\delta}^{2(i)})_{i\in\mathbb{N}}$ converges to the required posterior distribution at a geometric rate. For the homogeneous kernel, we have the following result:

Theorem 1. *Let $(k^{(i)}, \boldsymbol{\mu}^{(i)}_{1:k}, \Lambda^{(i)}, \boldsymbol{\delta}^{2(i)})_{i\in\mathbb{N}}$ be the Markov chain whose transition kernel has been described in section 3. This Markov chain converges to the probability distribution $p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y})$. Furthermore, this convergence occurs at a geometric rate, that is, for $p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y})$-almost every initial point $(k^{(0)}, \boldsymbol{\mu}^{(0)}_{1:k}, \Lambda^{(0)}, \boldsymbol{\delta}^{2(0)}) \in \boldsymbol{\Omega} \times \boldsymbol{\Psi}$ there exists a function $C(k^{(0)}, \boldsymbol{\mu}^{(0)}_{1:k}, \Lambda^{(0)}, \boldsymbol{\delta}^{2(0)}) > 0$ and a constant $\rho \in [0, 1)$ such that:*

$$\begin{aligned} &\left\| p^{(i)}\left(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2\right) - p\left(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 \middle| \mathbf{x}, \mathbf{y}\right) \right\|_{TV} \\ &\quad \leq C\left(k^{(0)}, \boldsymbol{\mu}^{(0)}_{1:k}, \Lambda^{(0)}, \boldsymbol{\delta}^{2(0)}\right) \rho^{\lfloor i/k_{\max} \rfloor} \end{aligned} \tag{6.1}$$

where $p^{(i)}(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2)$ is the distribution of $(k^{(i)}, \boldsymbol{\mu}^{(i)}_{1:k}, \Lambda^{(i)}, \boldsymbol{\delta}^{2(i)})$ and $\|\cdot\|_{TV}$ is the total variation norm (Tierney, 1994).

Proof. See appendix B.

Corollary 1. *Since at each iteration i, one simulates the nuisance parameters $(\boldsymbol{\alpha}_{1:m}, \boldsymbol{\sigma}^2_k)$, the distribution of the series $(k^{(i)}, \boldsymbol{\alpha}^{(i)}_{1:m}, \boldsymbol{\mu}^{(i)}_{1:k}, \boldsymbol{\sigma}^{2(i)}_k, \Lambda^{(i)}, \boldsymbol{\delta}^{2(i)})_{i\in\mathbb{N}}$ converges geometrically toward $p(k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2_k, \Lambda, \boldsymbol{\delta}^2 \mid \mathbf{x}, \mathbf{y})$ at the same rate ρ.*

In other words, the distribution of the Markov chain converges at least at a geometric rate, dependent on the initial state, to the required equilibrium distribution $p(k, \boldsymbol{\theta}, \psi \mid \mathbf{x}, \mathbf{y})$.

Remark 1. In practice one cannot evaluate ρ, but theorem 1 proves its existence. This type of convergence ensures that a central limit theorem for ergodic averages is valid (Meyn & Tweedie, 1993; Tierney, 1994). Moreover, in practice, there is empirical evidence that the Markov chain converges quickly.

We have the following convergence theorem for the reversible-jump MCMC simulated annealing algorithm:

Theorem 2. *Under certain assumptions found in Andrieu, Breyer, and Doucet (1999), the series of* $(\boldsymbol{\theta}^{(i)}, k^{(i)})$ *converges in probability to the set of global maxima* $(\boldsymbol{\theta}^{\max}, k^{\max})$, *that is, for any* $\epsilon > 0$, *it follows that*

$$\lim_{i\to\infty} \Pr\left(\frac{p(\boldsymbol{\theta}^{(i)}, k^{(i)})}{p(\boldsymbol{\theta}^{\max}, k^{\max})} \geq 1 - \epsilon\right) = 1.$$

Proof. If we follow the same steps as in proposition 1 of appendix B, with $\boldsymbol{\delta}^2$ and Λ fixed, it is easy to show that the transition kernels for each temperature are uniformly geometrically ergodic. Hence, as a corollary of Andrieu et al. (1999, theorem 1), the convergence result for the simulated annealing MCMC algorithm follows.

7 Experiments

When implementing the reversible-jump MCMC algorithm discussed in section 4, one might encounter problems of ill conditioning, in particular for high-dimensional parameter spaces. There are two satisfactory ways of overcoming this problem.[3] First, we can introduce a ridge regression component so that the expression for $\mathbf{M}_{i,k}^{-1}$ in section 3.3 becomes

$$\mathbf{M}_{i,k}^{-1} = \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}) + \boldsymbol{\Sigma}_i^{-1} + \hbar\mathbf{I}_m,$$

where $\hbar$ is a small number. Alternatively, we can introduce a slight modification of the prior for $\boldsymbol{\alpha}_{1:m}$:

$$p(\boldsymbol{\alpha}_{1:m} \mid k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) = \left[\prod_{i=1}^{c} |2\pi\boldsymbol{\sigma}_i^2\boldsymbol{\delta}_i^2\mathbf{I}_m|^{-1/2} \exp\left(-\frac{1}{2\boldsymbol{\sigma}_i^2\boldsymbol{\delta}_i^2}\boldsymbol{\alpha}'_{1:m,i}\boldsymbol{\alpha}_{1:m,i}\right)\right].$$

[3] The software is available online at http://www.cs.berkeley.edu/~jfgf.

We have found that although both strategies can deal with the problem of limited numerical precision, the second approach seems to be more stable. In addition, the second approach does not oblige us to select a value for the simulation parameter $\hbar$.

7.1 Experiment 1: Signal Detection. The problem of detecting signal components in noisy signals has occupied the minds of many researchers for a long time (Djurić, 1996). Here, we consider the rather simple toy problem of detecting gaussian components in a noisy signal. Our aim is to compare the performance of the hierarchical Bayesian model selection scheme and the penalized likelihood model selection criteria (AIC, MDL) when the amount of noise in the signal varies.

The data were generated from the following univariate function using 50 covariate points uniformly on $[-2, 2]$:

$$y = x + 2\exp(-16x^2) + 2\exp(-16(x - 0.7)^2) + n,$$

where $n \sim \mathcal{N}(0, \sigma^2)$. The data were then rescaled to make the input data lie in the interval $[0, 1]$. We used the reversible-jump MCMC algorithm, for the full Bayesian model, and the simulated annealing algorithms to estimate the number of components in the signal for different levels of noise. We repeated the experiment 100 times for each noise level. We chose gaussian radial basis functions with the same variance as the gaussian signal components. For the simulated annealing method, we adopted a linear cooling schedule: $T_i = a - bi$, where $a, b \in \mathbb{R}^+$ and $T_i > 0$ for $i = 1, 2, 3, \ldots$ In particular, we set the initial and final temperatures to 1 and 1×10^{-5}. For the Bayesian model, we selected diffuse priors ($\alpha_{\delta^2} = 2, \beta_{\delta^2} = 10$ [see experiment 2], $\upsilon_0 = 0, \gamma_0 = 0, \varepsilon_1 = 0.001$ and $\varepsilon_2 = 0.0001$). Finally, we set the simulation parameters $k_{\max}$, ι, σ^2_{RW}, and $\varsigma^\star$ to 20, 0.1, 0.001, and 0.1.

Figure 4 shows the typical fits that were obtained for training and validation data sets. By varying the variance of the noise σ^2, we estimated the main mode and fractions of unexplained variance. For the AIC and BIC/MDL criteria, the main mode corresponds to the one for which the posterior is maximized, while for the Bayesian approach, the main mode corresponds to the MAP of the model order probabilities $\widehat{p}(k \mid \mathbf{x}, \mathbf{y})$, computed as suggested in section 3.2.

The fractions of unexplained variance (fv) were computed as follows:

$$\text{fv} = \frac{1}{100}\sum_{i=1}^{100}\frac{\sum_{t=1}^{50}(y_{t,i} - \widehat{y}_{t,i})^2}{\sum_{t=1}^{50}(y_{t,i} - \overline{y}_i)^2},$$

where $\widehat{y}_{t,i}$ denotes the tth prediction for the ith trial and $\overline{y}_i$ is the estimated mean of y_i. The normalization in the fv error measure makes it independent of the size of the data set. If the estimated mean was to be used as the

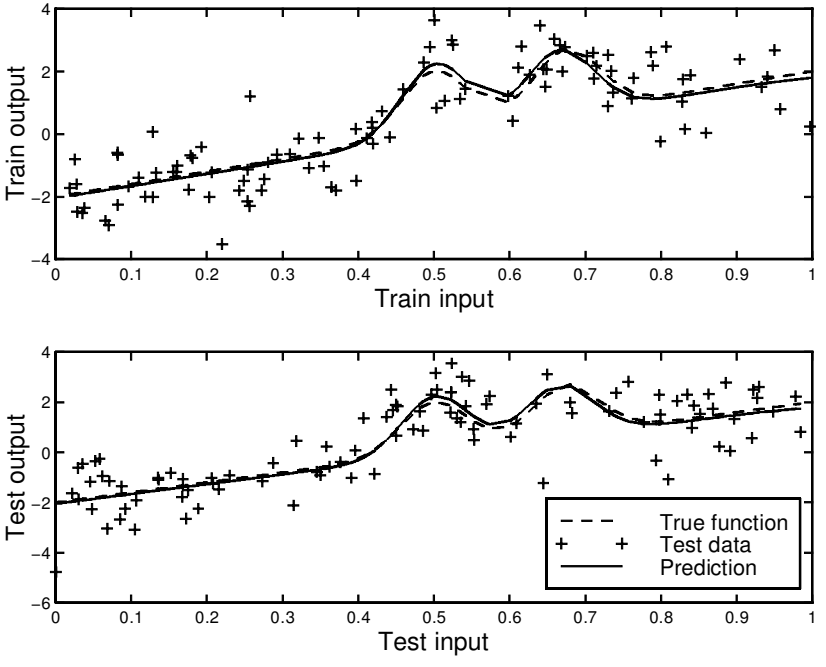

Figure 4: Performance of the reversible-jump MCMC algorithm on the signal detection problem. Despite the large noise variance, the estimates of the true function and noise process are very accurate, thereby leading to good generalisation (no overfitting).

Table 1: Fraction of Unexplained Variance for Different Values of the Noise Variance, Averaged over 100 Test Sets.

σ^2	AIC	BIC/MDL	Bayes
0.01	0.0070	0.0076	0.0069
0.1	0.0690	0.0732	0.0657
1	0.6083	0.4846	0.5105

predictor of the data, the fv would be equal to 1. The results obtained are shown in Figure 5 and Table 1. The fv for each model selection approach are very similar. This result is expected since the problem under consideration is rather simple and the error variations could possibly be attributed to the fact that we use only 100 realizations of the noise process for each σ^2. What is important is that even in this scenario, it is clear that the full Bayesian model provides more accurate estimates of the model order.

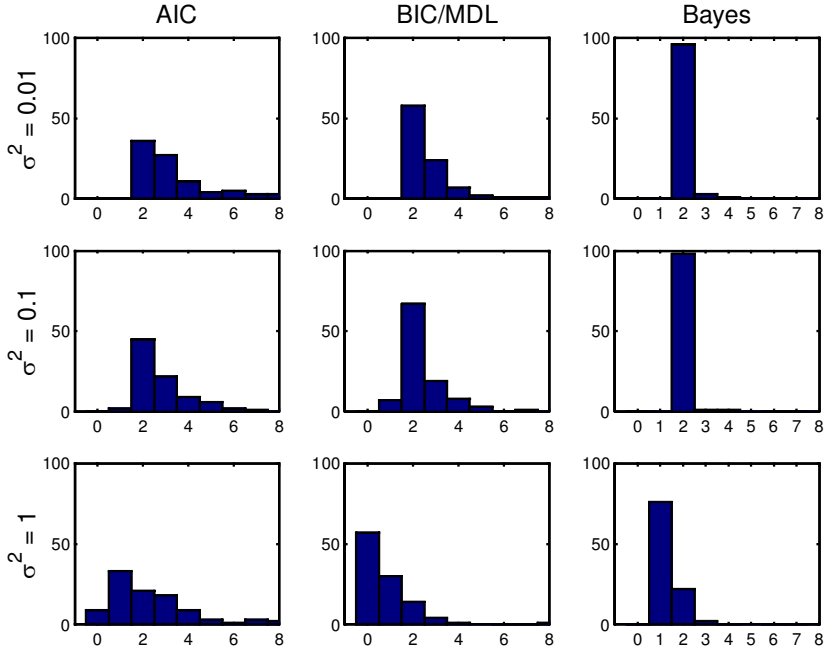

Figure 5: Histograms of the main mode $\widehat{p}(k \mid \mathbf{x}, \mathbf{y})$ for 100 trials of each noise level in experiment 1. The Bayes solution provides a better estimate of the true number of basis (2) than the MDL/BIC and AIC criteria.

7.2 Experiment 2: Robot Arm Data. This data set is often used as a benchmark to compare neural network algorithms.[4] It involves implementing a model to map the joint angle of a robot arm (x_1, x_2) to the position of the end of the arm (y_1, y_2). The data were generated from the following model:

$$\begin{aligned} y_1 &= 2.0\cos(x_1) + 1.3\cos(x_1 + x_2) + \epsilon_1 \\ y_2 &= 2.0\sin(x_1) + 1.3\sin(x_1 + x_2) + \epsilon_2 \end{aligned} \tag{7.1}$$

where $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$; $\sigma = 0.05$. We use the first 200 observations of the data set to train our models and the last 200 observations to test them.

First, we assessed the performance of the reversible-jump algorithm with the Bayesian model. In all the simulations, we chose to use cubic basis functions. Figure 6 shows the three-dimensional plots of the training data and the contours of the training and test data. The contour plots also include

[4] The data set is available online at http://wol.ra.phy.cam.ac.uk/mackay/.

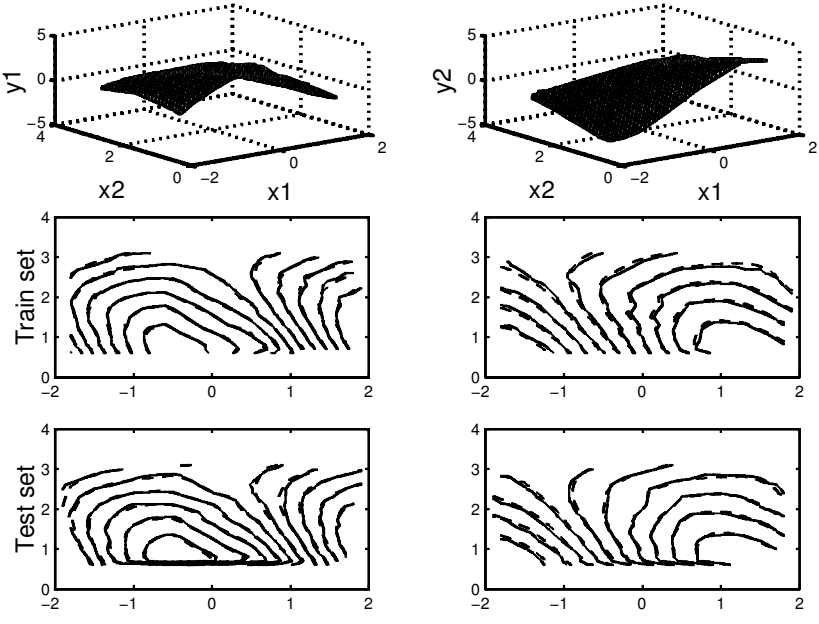

Figure 6: (Top) Training data surfaces corresponding to each coordinate of the robot arm's position. (Middle, bottom) Training and validation data (solid lines) and respective RBF network mappings (dotted lines).

the typical approximations that were obtained using the algorithm. To assess convergence, we plotted the probabilities of each model order $\widehat{p}(k \mid \mathbf{x}, \mathbf{y})$ in the chain (using equation 3.2) for 50,000 iterations, as shown in Figure 7. As the model orders begin to stabilize after 30,000 iterations, we decided to run the Markov chains for 50,000 iterations with a burn-in of 30,000 iterations. It is possible to design more complex convergence diagnostic tools; however, this topic is beyond the scope of this article.

We chose uninformative priors for all the parameters and hyperparameters. In particular, we used the values shown in Table 2. To demonstrate the robustness of our model, we chose different values for β_{δ^2} (the only critical hyperparameter as it quantifies the mean of the spread $\boldsymbol{\delta}$ of $\boldsymbol{\alpha}_k$). The obtained mean square errors (see Table 2) and probabilities for $\boldsymbol{\delta}_1$, $\boldsymbol{\delta}_2$, $\boldsymbol{\sigma}^2_{1,k}$, $\boldsymbol{\sigma}^2_{2,k}$, and k, shown in Figure 8, clearly indicate that our model is robust with respect to prior specification.

As shown in Table 3, our mean square errors are slightly better than the ones reported by other researchers (Holmes & Mallick, 1998; Mackay, 1992; Neal, 1996; Rios Insua & Müller, 1998). Yet the main point we are trying to make is that our model exhibits the important quality of being robust to the

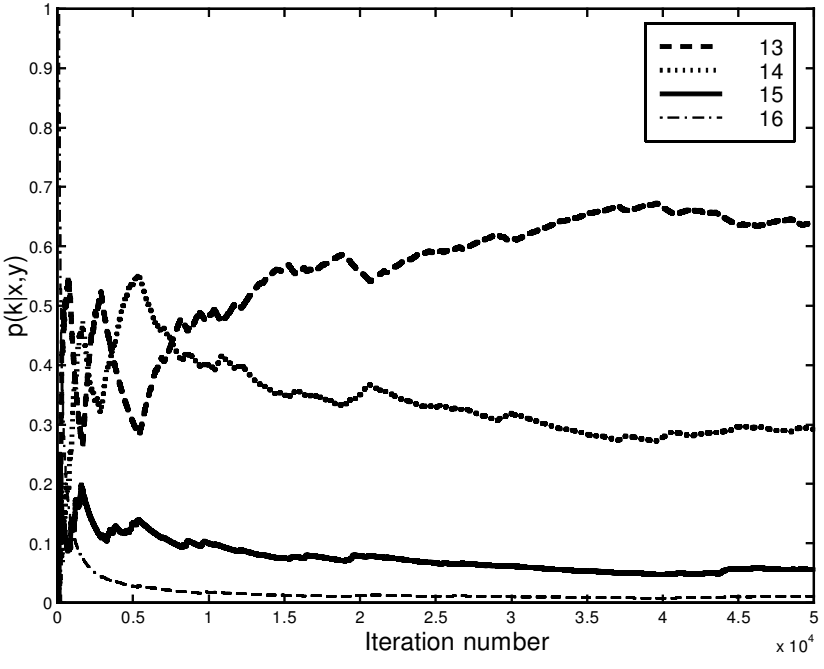

Figure 7: Convergence of the reversible-jump MCMC algorithm for RBF networks. The plot shows the probability of each model order given the data. The model orders begin to stabilize after 30,000 iterations.

Table 2: Simulation Parameters and Mean Square Errors for the Robot Arm Data (Test Set) Using the Reversible-Jump MCMC Algorithm and the Bayesian Model.

α_{δ^2}	β_{δ^2}	υ_0	γ_0	ε_1	ε_2	Mean Square Error
2	0.1	0	0	0.0001	0.0001	0.00505
2	10	0	0	0.0001	0.0001	0.00503
2	100	0	0	0.0001	0.0001	0.00502

prior specification and statistically significant. Moreover, it leads to more parsimonious models than the ones previously reported.

We also tested the reversible-jump simulated annealing algorithms with the AIC and MDL criteria on this problem. The results for the MDL criterion are depicted in Figure 9. We note that the posterior increases stochastically with the number of iterations and eventually converges to a maximum. The figure also illustrates the convergence of the train and test set errors for each

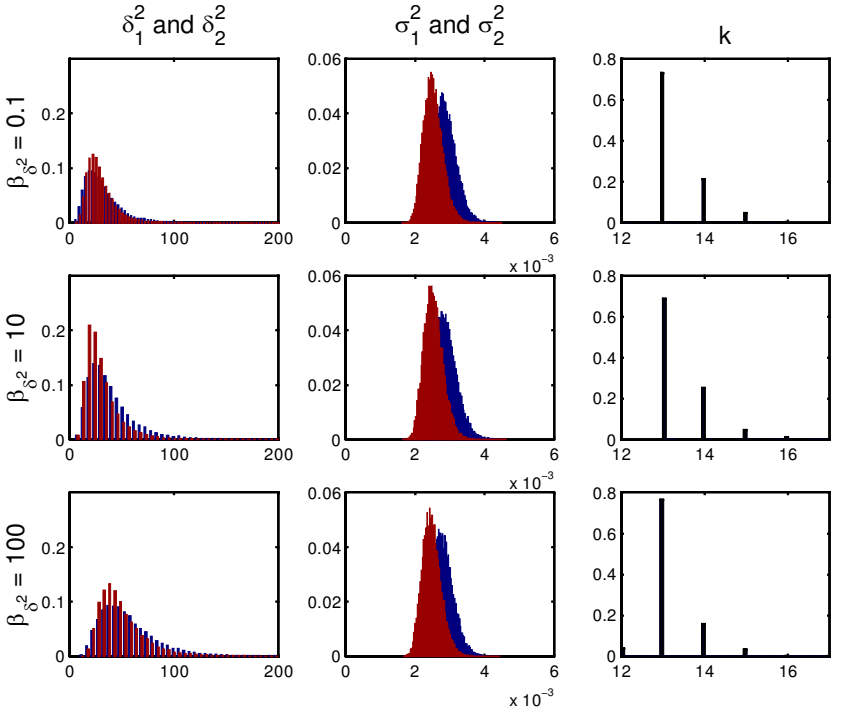

Figure 8: Histograms of smoothness constraints for each output (δ_1 and δ_2), noise variances ($\sigma^2_{1,k}$ and $\sigma^2_{2,k}$), and model order (k) for the robot arm data simulation using three different values for β_{δ^2}. The plots confirm that the model is robust to the setting of β_{δ^2}.

network in the Markov chain. For the final network, we chose the one that maximized the posterior (the MAP estimate). This network consisted of 12 basis functions and incurred an error of 0.00512 in the test set. Following the same procedure, the AIC network consisted of 27 basis functions and incurred an error of 0.00520 in the test set. These results indicate that the full Bayesian model, with model averaging, provides more accurate results. Moreover, it seems that the information criteria, in particular the AIC, can lead to overfitting of the data.

These results confirm the well-known fact that suboptimal techniques, such as the simulated annealing method with information criteria penalty terms and a rapid cooling schedule, can allow for faster computation at the expense of accuracy.

Table 3: Mean Square Errors and Number of Basis Functions for the Robot Arm Data.

Method	Mean Square Error
Mackay's (1992) gaussian approximation with highest evidence	0.00573
Mackay's (1992) gaussian approximation with lowest test error	0.00557
Neal's (1996) hybrid Monte Carlo	0.00554
Neal's (1996) hybrid Monte Carlo with ARD	0.00549
Rios Insua and Müller's (1998) MLP with reversible-jump MCMC	0.00620
Holmes and Mallick's (1998) RBF with reversible-jump MCMC	0.00535
Reversible-jump MCMC with Bayesian model	0.00502
Reversible-jump MCMC with MDL	0.00512
Reversible-jump MCMC with AIC	0.00520

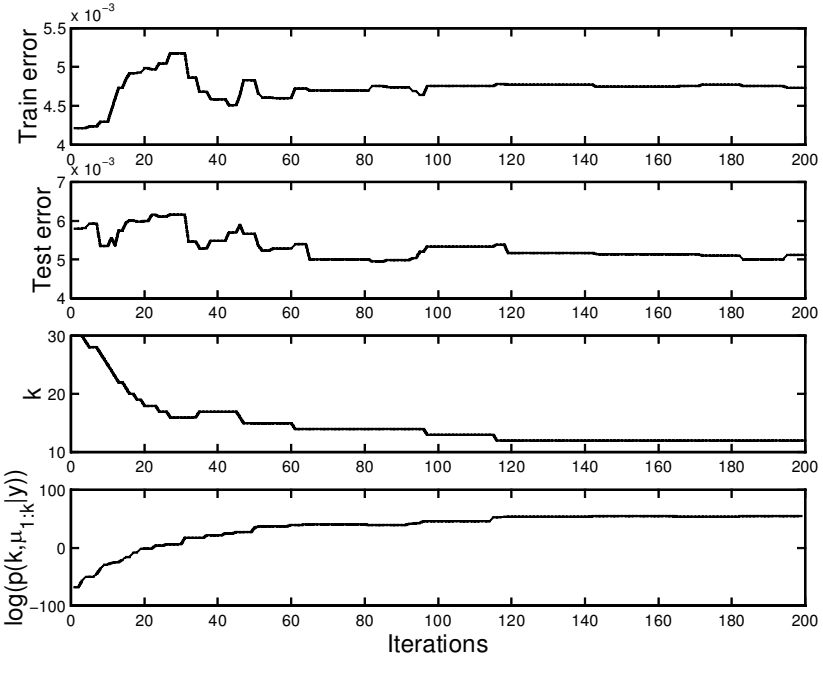

Figure 9: Performance of the reversible-jump simulated annealing algorithm for 200 iterations on the robot arm data, with the MDL criterion.

7.3 Experiment 3: Classification with Discriminants. Here we consider an interesting nonlinear classification data set[5] collected as part of a study to identify patients with muscle tremor (Roberts, Penny, & Pillot, 1996; Spyers-

[5] The data are available online at http://www.ee.ic.ac.uk/hp/staff/sroberts.html.

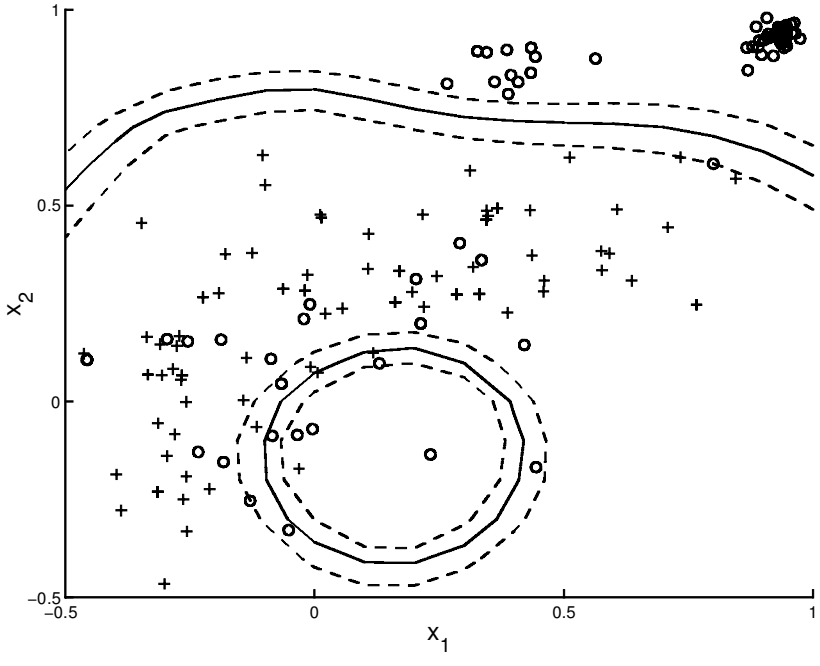

Figure 10: Classification boundaries (solid line) and confidence intervals (dashed line) for the RBF classifier. The circles indicate patients, and the crosses represent the control group.

Ashby, Bain, & Roberts, 1998). The data were gathered from a group of patients (nine with, primarily, Parkinson's disease or multiple sclerosis) and from a control group (not exhibiting the disease). Arm muscle tremor was measured with a 3D mouse and a movement tracker in three linear and three angular directions. The time series of the measurements were parameterized using a set of autoregressive models. The number of features was then reduced to two (Roberts et al., 1996). Figure 10 shows a plot of these features for patient and control groups. The figure also shows the classification boundaries and confidence intervals obtained with our model, using thin-plate spline hidden neurons and an output linear neuron. We should point out, however, that having an output linear neuron leads to a classification framework based on discriminants. An alternative approach, which we do not pursue here, is to use a logistic output neuron so that the classification scheme is based on probabilities of class membership. It is, however, possible to extend our approach to this probabilistic classification setting by adopting the generalized linear models framework with logistic, probit

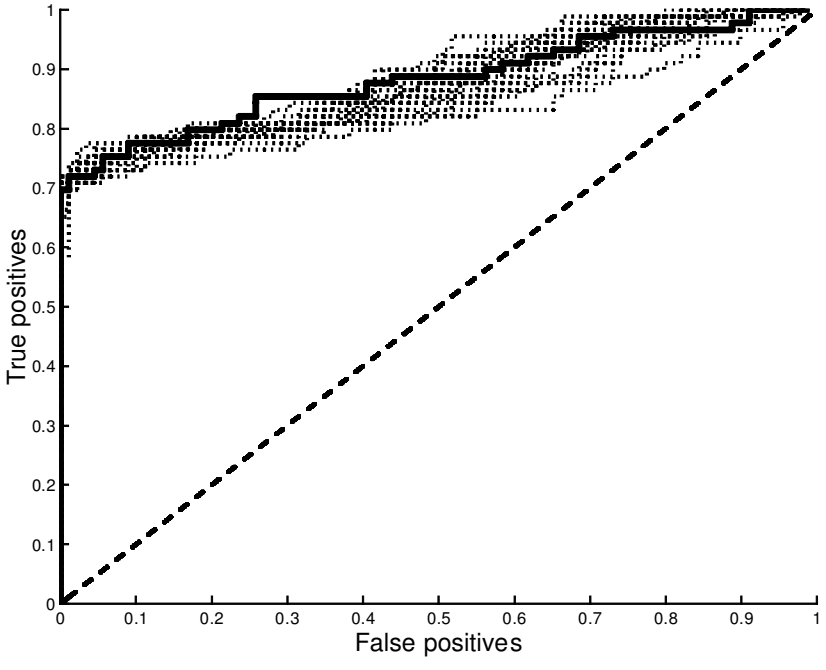

Figure 11: Receiver operating characteristic (ROC) of the classifier for the tremor data. The solid line is the ROC curve for the posterior mean classifier, and the dotted lines correspond to the curves obtained for various classifiers in the Markov chain.

or softmax link functions (Gelman, Carlin, Stern, & Rubin, 1995; Holmes & Mallick, 1999; Nabney, 1999).

The size of the confidence intervals for the decision boundary is given by the noise variance (σ^2). These intervals are a measure of uncertainty on the threshold that we apply to the linear output neuron. Our confidence of correctly classifying a sample occurring within these intervals should be very low. The receiver operating characteristic (ROC) curve, shown in Figure 11, indicates that using the Neyman Pearson criterion, we can expect to detect patients with a 69% confidence and without making any mistakes (Hand, 1997). In our application, the ROC curve was obtained by averaging all the predictions for each classifier in the Markov chain.

The percentage of classification errors in the test set was found to be 14.60. This error is of the same magnitude as previously reported results (de Freitas, Niranjan, & Gee, 1998; Roberts & Penny, 1998). Finally, the estimated probabilities of the signal-to-noise ratio (δ^2), noise variance (σ^2), and model order (k) for this application are depicted in Figure 12.

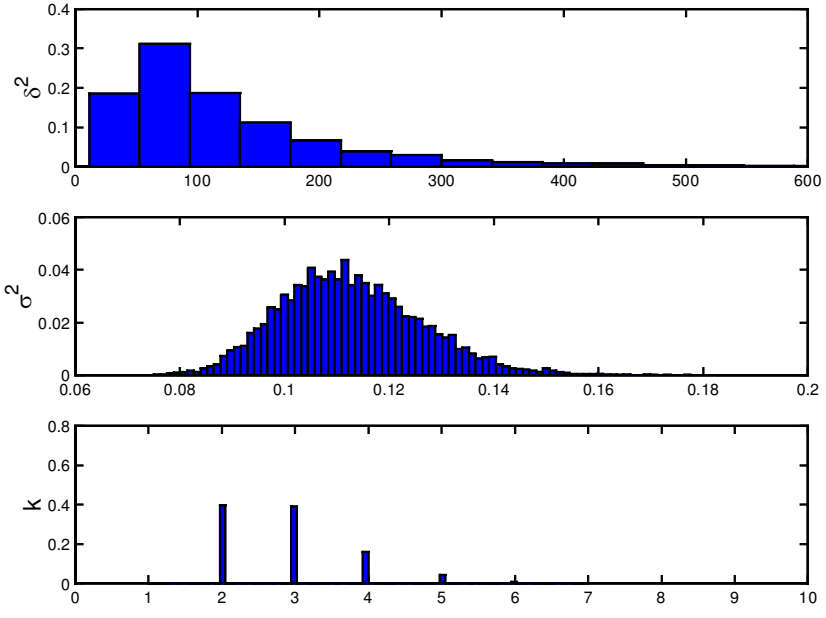

Figure 12: Estimated probabilities of the signal-to-noise ratio (δ^2), noise variance (σ^2), and model order (k) for the classification example.

8 Conclusions

We have proposed a robust full Bayesian model for estimating, jointly, the noise variance, parameters, and number of parameters of an RBF model. We also considered the problem of stochastic optimization for model order selection and proposed a solution that makes use of a reversible-jump simulated annealing algorithm and classical information criteria. Moreover, we gave proofs of geometric convergence for the reversible-jump algorithm for the full Bayesian model and convergence for the simulated annealing algorithm.

Contrary to previously reported results, our experiments suggest that our Bayesian model is robust with respect to the specification of the prior. In addition, we obtained more parsimonious RBF networks and slightly better approximation errors than the ones previously reported in the literature. We also presented a comparison between Bayesian model averaging and penalized likelihood model selection with the AIC and MDL criteria. We found that the Bayesian strategy led to more accurate results. Yet the optimization strategy using the AIC and MDL criteria and a reversible-jump simulated annealing algorithm was shown to converge faster for a specific cooling schedule.

There are many avenues for further research. These include estimating the type of basis functions required for a particular task, performing input variable selection, considering other noise models, adopting Bernoulli and multinomial output distributions for probabilistic classification by incorporating ideas from the generalized linear models field, and extending the framework to sequential scenarios. A solution to the first problem can be easily formulated using the reversible-jump MCMC framework presented here. Variable selection schemes can also be implemented via the reversible-jump MCMC algorithm. Finally, we are working on a sequential version of the algorithm that allows us to perform model selection in nonstationary environments (Andrieu, de Freitas, & Doucet, 1999a, 1999b). We also believe that the algorithms need to be tested on additional real-world problems. For this purpose, we have made the software available online at http://www.cs.berkeley.edu/~jfgf.

Appendix A: Notation

$\mathbf{A}_{i,j}$: entry of the matrix $\mathbf{A}$ in the ith row and jth column.

$\mathbf{A}'$: transpose of matrix $\mathbf{A}$.

$|\mathbf{A}|$: determinant of matrix $\mathbf{A}$.

If $\mathbf{z} \triangleq (z_1, \ldots, z_{j-1}, z_j, z_{j+1}, \ldots, z_k)'$
then $\mathbf{z}_{-j} \triangleq (z_1, \ldots, z_{j-1}, z_{j+1}, \ldots, z_k)'$.

$\mathbf{I}_n$: identity matrix of dimension $n \times n$.

$\mathbb{I}_E(\mathbf{z})$: indicator function of the set E (1 if $\mathbf{z} \in E$, 0 otherwise).

$\lfloor z \rfloor$: highest integer strictly less than z.

$\mathbf{z} \sim p(\mathbf{z})$: $\mathbf{z}$ is distributed according to $p(\mathbf{z})$.

$\mathbf{z} \mid \mathbf{y} \sim p(\mathbf{z})$: the conditional distribution of $\mathbf{z}$ given $\mathbf{y}$ is $p(\mathbf{z})$.

Probability Distribution	$\mathcal{F}$	$f_{\mathcal{F}}(\cdot)$
Inverse gamma	$\mathcal{IG}(\alpha, \beta)$	$\frac{\beta^\alpha}{\Gamma(\alpha)} z^{-\alpha-1} \exp(-\beta/z)\, \mathbb{I}_{[0,+\infty)}(z)$
Gamma	$\mathcal{G}a(\alpha, \beta)$	$\frac{\beta^\alpha}{\Gamma(\alpha)} z^{\alpha-1} \exp(-\beta z)\, \mathbb{I}_{[0,+\infty)}(z)$
Gaussian	$\mathcal{N}(\mathbf{m}, \Sigma)$	$\lvert 2\pi\Sigma\rvert^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{z}-\mathbf{m})'\Sigma^{-1}(\mathbf{z}-\mathbf{m})\right)$
Poisson	$\mathcal{P}n(\lambda)$	$\frac{\lambda^z}{z!} \exp(-\lambda) \mathbb{I}_{\mathbb{N}}(z)$
Uniform	$\mathcal{U}_A$	$\left[\int_A d\mathbf{z}\right]^{-1} \mathbb{I}_A(\mathbf{z})$

Appendix B: Proof of Theorem 1

The proof of theorem 1 relies on the following theorem, which is a result of theorems 14.0.1 and 15.0.1 in Meyn and Tweedie (1993):

Theorem 3. *Suppose that a Markovian transition kernel P on a space* $\mathbf{Z}$

1. *Is a* $\phi-$*irreducible (for some measure* ϕ*) aperiodic Markov transition kernel with invariant distribution* π.
2. *Has geometric drift toward a small set C with drift function* V: $\mathbf{Z} \to [1, +\infty)$, *that is, there exists* $0 < \lambda < 1$, $b > 0$, k_0 *and an integrable measure* ν *such that:*

$$PV(\mathbf{z}) \leq \lambda V(\mathbf{z}) + b\mathbb{I}_C(\mathbf{z}) \tag{B.1}$$

$$P^{k_0}(\mathbf{z}, d\mathbf{z}') \geq \mathbb{I}_C(\mathbf{z})\nu(d\mathbf{z}'). \tag{B.2}$$

Then for π*-almost all* $\mathbf{z}_0$, *some constants* $\rho < 1$ *and* $R < +\infty$, *we have:*

$$\|P^n(\mathbf{z}_0, \cdot) - \pi(\cdot)\|_{TV} \leq RV(\mathbf{z}_0)\rho^n. \tag{B.3}$$

That is, P is geometrically ergodic.

We need to prove five lemmas that will allow us to prove the different conditions required to apply theorem 2. These lemmas will enable us to prove proposition 1, which will establish the minorization condition, equation B.2, for some k_0 and measure ϕ (to be described). The ϕ-irreducibility and aperiodicity of the Markov chain are then proved in corollary 3, thereby ensuring the simple convergence of the Markov chain. To complete the proof, proposition 2 will establish the drift condition, equation B.1. To simplify the presentation, we consider only one network output. The proof for multiple outputs follows trivially.

Before presenting the various lemmas and their respective proofs, we need to introduce some notation. Let $\mathcal{K}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_2}, d\delta^2_{k_2}, k_2, d\boldsymbol{\mu}_{1:k_2})$ denote the transition kernel of the Markov chain.[6] Thus, for fixed $(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}) \in \mathbb{R}^{+2} \times \boldsymbol{\Omega}$, we have:

$$\Pr\left((\Lambda_{k_2}, \delta^2_{k_2}, k_2, \boldsymbol{\mu}_{1:k_2}) \in A_{k_2} \mid (\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1})\right)$$
$$= \int_{A_{k_2}} \mathcal{K}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_2}, d\delta^2_{k_2}, k_2, d\boldsymbol{\mu}_{1:k_2}),$$

where $A_{k_2} \in \mathcal{B}(\mathbb{R}^{+2} \times \{k_2\} \times \boldsymbol{\Omega}_{k_2})$. This transition kernel is by construction (see section 4.2) a mixture of transition kernels. Hence:

$$\mathcal{K}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_2}, d\delta^2_{k_2}, k_2, d\boldsymbol{\mu}_{1:k_2})$$

[6] We will use the notation Λ_k, δ^2_k, when necessary, for ease of presentation. This does not mean that these variables depend on the dimension k.

$$
\begin{aligned}
&= \Big(b_{k_1} \mathcal{K}_{birth}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta^2_{k_1}, k_1+1, d\boldsymbol{\mu}_{1:k_1+1}) \\
&\quad + d_{k_1} \mathcal{K}_{death}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta^2_{k_1}, k_1-1, d\boldsymbol{\mu}_{1:k_1-1}) \\
&\quad + s_{k_1} \mathcal{K}_{split}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta^2_{k_1}, k_1+1, d\boldsymbol{\mu}_{1:k_1+1}) \\
&\quad + m_{k_1} \mathcal{K}_{merge}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta^2_{k_1}, k_1-1, d\boldsymbol{\mu}_{1:k_1-1}) \\
&\quad + (1 - b_{k_1} - d_{k_1} - s_{k_1} - m_{k_1}) \\
&\quad \times \mathcal{K}_{update}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta^2_{k_1}, k_1, d\boldsymbol{\mu}^*_{1:k_1}) \Big) \\
&\quad \times p(\delta^2_{k_2} \mid \delta^2_{k_1}, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) p(\Lambda_{k_2} \mid k_2) d\Lambda_{k_2} d\delta^2_{k_2},
\end{aligned}
$$

where $\mathcal{K}_{birth}$ and $\mathcal{K}_{death}$ correspond to the reversible jumps described in section 4.2.1, $\mathcal{K}_{split}$ and $\mathcal{K}_{merge}$ to the reversible jumps described in section 4.2.2, and $\mathcal{K}_{update}$ is described in section 4.2.3. The different steps for sampling the parameters $\delta^2_{k_2}$ and Λ_{k_2} are described in section 4.1.3.

Lemma 1. *We denote $\mathbf{P}^*_k$ the matrix $\mathbf{P}_k$ for which $\delta^2 \to +\infty$. Let $\mathbf{v} \in \mathbb{R}^N$, then $\mathbf{v}'\mathbf{P}^*_k\mathbf{v} = 0$ if and only if $\mathbf{v}$ belongs to the space spanned by the columns of $\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})$, with $\boldsymbol{\mu}_{1:k} \in \boldsymbol{\Omega}_k$.*

Then, noting that $\mathbf{y}'\mathbf{P}_k\mathbf{y} = \frac{1}{1+\delta^2}\mathbf{y}'\mathbf{y} + \frac{\delta^2}{1+\delta^2}\mathbf{y}'\mathbf{P}^*_k\mathbf{y}$, we obtain the following corollary:

Corollary 2. *If the observed data $\mathbf{y}$ are really noisy, that is, cannot be described as the sum of k basis functions and a linear mapping, then there exists a number $\varepsilon > 0$ such that for all $k \leq k_{\max}$, $\delta^2 \in \mathbb{R}^+$, and $\boldsymbol{\mu}_{1:k} \in \boldsymbol{\Omega}_k$, we have $\mathbf{y}'\mathbf{P}_k\mathbf{y} \geq \varepsilon > 0$.*

Lemma 2. *For all $k \leq k_{\max}$, $\delta^2 \in \mathbb{R}^+$, and $\boldsymbol{\mu}_{1:k} \in \boldsymbol{\Omega}_k$, we have $\mathbf{y}'\mathbf{P}_k\mathbf{y} \leq \mathbf{y}'\mathbf{y}$.*

Lemma 3. *Let K_1 be the transition kernel corresponding to $\mathcal{K}$ such that Λ and δ^2 are kept fixed. Then there exists $M_1 > 0$ such that for any M_2 sufficiently large, any $\delta^2 \in \mathbb{R}^+$ and $k_1 = 1, \ldots, k_{\max}$:*

$$
\begin{aligned}
&K_1(\Lambda, \delta^2, k_1, \boldsymbol{\mu}_{1:k_1}; \Lambda, \delta^2, k_1-1, d\boldsymbol{\mu}_{1:k_1-1}) \\
&\quad \geq \frac{c^\star \mathbb{I}_{\{\Lambda; \Lambda < M_2\}}(\Lambda)}{M_1 M_2 k_1} \delta_S \boldsymbol{\mu}_{1:k_1} (d\boldsymbol{\mu}_{1:k_1-1})
\end{aligned}
$$

with $c^\star > 0$ as defined in equation 4.5.

Proof. According to the definition of the transition kernel, for all variables $((k_1, \boldsymbol{\mu}_{1:k_1}), (k_2, \boldsymbol{\mu}_{1:k_2})) \in \boldsymbol{\Omega}^2$, one has the following inequality:

$$K_1(\Lambda, \delta^2, k_1, \boldsymbol{\mu}_{1:k_1}; \Lambda, \delta^2, k_2, d\boldsymbol{\mu}_{1:k_2}) \geq \min\{1, r_{death}\} d_{k_1} \frac{\delta_{S_{\boldsymbol{\mu}_{1:k_1}}}(d\boldsymbol{\mu}_{1:k_2})}{k_1},$$

where $1/k_1$ is the probability of choosing one of the basis functions for the purpose of removing it and $S_{\boldsymbol{\mu}_{1:k_1}} \triangleq \{\boldsymbol{\mu}' \in \boldsymbol{\Omega}_{k_1-1} / \exists l \in \{1, \ldots, k_1\}$ such that $\boldsymbol{\mu}' = \boldsymbol{\mu}_{-l}\}$. Then from equation 4.6 and for all $k_1 = 1, \ldots, k_{\max}$, we have

$$r_{death}^{-1} = \left(\frac{\gamma_0 + \mathbf{y}'_{1:N} \mathbf{P}_{k_1 - 1} \mathbf{y}_{1:N}}{\gamma_0 + \mathbf{y}'_{1:N} \mathbf{P}_{k_1} \mathbf{y}_{1:N}} \right)^{\left(\frac{N+\upsilon_0}{2}\right)} \frac{1}{k_1 (1 + \delta^2)^{1/2}}.$$

As a result, we can use lemmas 1 and 2 to obtain ε and M_1 such that

$$r_{death}^{-1} \leq \left(\frac{\gamma_0 + \mathbf{y}'_{1:N} \mathbf{y}_{1:N}}{\varepsilon} \right)^{\left(\frac{N+\upsilon_0}{2}\right)} \frac{1}{k_1 (1 + \delta^2)^{1/2}} < M_1 < +\infty.$$

Thus, there exists M_1 sufficiently large such that for any M_2 sufficiently large (from equation 4.5), $\delta^2 \in \mathbb{R}^+$, $1 \leq k_1 \leq k_{\max}$, and $\boldsymbol{\mu}_{1:k_1} \in \boldsymbol{\Omega}_{k_1}$

$$\begin{aligned} &K_1(\Lambda, \delta^2, k_1, \boldsymbol{\mu}_{1:k_1}; \Lambda, \delta^2, k_1 - 1, d\boldsymbol{\mu}_{1:k_1-1}) \\ &\quad \geq \mathbb{I}_{\{\Lambda; \Lambda < M_2\}}(\Lambda) \frac{c^\star}{M_2} \frac{1}{M_1 k_1} \delta_{S_{\boldsymbol{\mu}_{1:k_1}}}(d\boldsymbol{\mu}_{1:k_1-1}). \end{aligned} \tag{B.4}$$

Lemma 4. *The transition kernel $\mathcal{K}$ satisfies the following inequality for $k = 0$:*

$$\mathcal{K}(\Lambda_0, \delta_0^2, 0, \boldsymbol{\mu}_0; d\Lambda_0^*, d\delta_0^{*2}, 0, d\boldsymbol{\mu}_0) \geq \zeta \varphi(\delta_0^{*2} \mid 0) p(\Lambda_0 \mid 0) d\delta_0^{*2} d\Lambda_0, \tag{B.5}$$

with $\zeta > 0$ and φ a probability density.

Proof. From the definition of the transition kernel $\mathcal{K}$, we have:[7]

$$\begin{aligned} &\mathcal{K}(\Lambda_0, \delta_0^2, 0, \boldsymbol{\mu}_0; d\Lambda_0^*, d\delta_0^{*2}, 0, d\boldsymbol{\mu}_0) \\ &\quad \geq u_0 p(\delta_0^{*2} \mid \delta_0^2, 0, d\boldsymbol{\mu}_0, \mathbf{x}, \mathbf{y}) p(\Lambda_0 \mid 0) d\delta_0^{*2} d\Lambda_0 \\ &\quad \geq (1 - c^\star) p(\delta_0^{*2} \mid \delta_0^2, 0, d\boldsymbol{\mu}_0, \mathbf{x}, \mathbf{y}) p(\Lambda_0 \mid 0) d\delta_0^{*2} d\Lambda_0 \end{aligned}$$

as $0 < 1 - c^\star \leq u_0 \leq 1$, and we adopt the notation $\varphi(\delta^{*2} \mid 0) \triangleq p(\delta^{*2} \mid \delta^2, 0, \boldsymbol{\mu}_0, \mathbf{x}, \mathbf{y})$.

[7] When $k = 0$, we keep for notational convenience the same notation for the transition kernel even if μ_0 does not exist.

Lemma 5. *There exists a constant $\xi > 0$ and a probability density φ such that for all $\delta^2 \in \mathbb{R}^+$, $0 \leq k \leq k_{\max}$, and $\boldsymbol{\mu}_{1:k} \in \boldsymbol{\Omega}_k$, one obtains:*

$$p(\delta^{*2} \mid \delta^2, k, \boldsymbol{\mu}_{1:k}, \mathbf{x}, \mathbf{y}) \geq \xi \varphi(\delta^{*2} \mid k). \tag{B.6}$$

Proof. From section 4.1.2, to update δ^2 at each iteration, one draws from the distribution $p(\boldsymbol{\alpha}_{1:m}, \sigma^2 \mid \delta^2, k, \boldsymbol{\mu}_{1:k}, \mathbf{x}, \mathbf{y})$, that is, one draws σ^2 from

$$p(\sigma^2 \mid \delta^2, k, \boldsymbol{\mu}_{1:k}, \mathbf{x}, \mathbf{y}) = \frac{\left(\frac{\gamma_0 + \mathbf{y}'_{1:N}\mathbf{P}_k\mathbf{y}_{1:N}}{2}\right)^{\frac{N+\upsilon_0}{2}}}{\Gamma\left(\frac{N+\upsilon_0}{2}\right)(\sigma^2)^{\frac{N+\upsilon_0}{2}+1}} \exp\left(\frac{-1}{2\sigma^2}(\gamma_0 + \mathbf{y}'_{1:N}\mathbf{P}_k\mathbf{y}_{1:N})\right);$$

then $\boldsymbol{\alpha}_{1:m}$ from

$$p(\boldsymbol{\alpha}_{1:m} \mid \delta^2, k, \boldsymbol{\mu}_{1:k}, \sigma^2, \mathbf{x}, \mathbf{y}) = \frac{1}{|2\pi\sigma^2\mathbf{M}_k|^{1/2}} \exp\left(\frac{-1}{2\sigma^2}(\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k)'\mathbf{M}_k^{-1}(\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k)\right);$$

and finally one draws δ^{*2} according to

$$p(\delta^{*2} \mid \delta^2, k, \boldsymbol{\theta}, \mathbf{x}, \mathbf{y}) = \frac{\left(\frac{\boldsymbol{\alpha}'_{1:m}\mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})\boldsymbol{\alpha}_{1:m}}{2\sigma^2} + \beta_{\delta^2}\right)^{m/2+\alpha_{\delta^2}}}{\Gamma(m/2 + \alpha_{\delta^2})(\delta^{*2})^{m/2+\alpha_{\delta^2}+1}} \times \exp\left(\frac{-1}{\delta^{*2}}\left(\frac{\boldsymbol{\alpha}'_{1:m}\mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})\boldsymbol{\alpha}_{1:m}}{2\sigma^2} + \beta_{\delta^2}\right)\right).$$

Consequently:

$$p(\delta^{*2} \mid \delta^2, k, \boldsymbol{\theta}, \mathbf{x}, \mathbf{y})p(\boldsymbol{\alpha}_{1:m}, \sigma^2 \mid \delta^2, k, \boldsymbol{\mu}_{1:k}, \mathbf{x}, \mathbf{y}) = \frac{\left(\frac{\gamma_0 + \mathbf{y}'_{1:N}\mathbf{P}_k\mathbf{y}_{1:N}}{2}\right)^{\frac{N+\upsilon_0}{2}} \left(\frac{\boldsymbol{\alpha}'_{1:m}\mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})\boldsymbol{\alpha}_{1:m}}{2\sigma^2} + \beta_{\delta^2}\right)^{m/2+\alpha_{\delta^2}}}{\Gamma(\frac{N+\upsilon_0}{2})\Gamma(m/2 + \alpha_{\delta^2})(2\pi)^{m/2}(\sigma^2)^{(N+\upsilon_0+m)/2+1}(\delta^{*2})^{m/2+\alpha_{\delta^2}+1}|\mathbf{M}_k|^{1/2}}$$
$$\times \exp\left(\frac{-1}{2\sigma^2}\left[(\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k)'\mathbf{M}_k^{-1}(\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k) + \gamma_0 + \mathbf{y}'_{1:N}\mathbf{P}_k\mathbf{y}_{1:N} + \frac{\boldsymbol{\alpha}'_{1:m}\mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})\boldsymbol{\alpha}_{1:m}}{\delta^{*2}}\right] - \frac{\beta_{\delta^2}}{\delta^{*2}}\right).$$

We can obtain the minorization condition, given by equation B.6, by integrating with respect to the nuisance parameters $\boldsymbol{\alpha}_{1:m}$ and σ^2. To accomplish

this, we need to perform some algebraic manipulations to obtain the following relation,

$$
\begin{aligned}
&(\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k)' \mathbf{M}_k^{-1} (\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k) + (\gamma_0 + \mathbf{y}_{1:N} \mathbf{P}_k \mathbf{y}_{1:N}) \\
&\qquad + \frac{\boldsymbol{\alpha}'_{1:m} \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \boldsymbol{\alpha}_{1:m}}{\delta^{*2}} \\
&\quad = (\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k^{\bullet})' \mathbf{M}_k^{\bullet -1} (\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k^{\bullet}) + \gamma_0 + \mathbf{y}_{1:N} \mathbf{P}_k^{\bullet} \mathbf{y}_{1:N},
\end{aligned}
$$

with:

$$
\begin{aligned}
\mathbf{M}_k^{\bullet -1} &= \left(1 + \frac{1}{\delta^2} + \frac{1}{\delta^{*2}}\right) \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \\
\mathbf{h}_k^{\bullet} &= \mathbf{M}_k^{\bullet} \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \mathbf{y}_{1:N} \\
\mathbf{P}_k^{\bullet} &= \mathbf{I}_N - \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \mathbf{M}_k^{\bullet} \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x}).
\end{aligned}
$$

We can now integrate with respect to $\boldsymbol{\alpha}_{1:m}$ (gaussian distribution) and $\boldsymbol{\sigma}^2$ (inverse gamma distribution) to obtain the minorization condition for δ^{*2}:

$$
\begin{aligned}
&p(\delta^{*2} \mid \delta^2, k, \boldsymbol{\mu}_{1:k}, \mathbf{x}, \mathbf{y}) \\
&\quad \geq \int_{\mathbb{R}^m \times \mathbb{R}^+} \frac{\left(\frac{\gamma_0 + \mathbf{y}'_{1:N} \mathbf{P}_k \mathbf{y}_{1:N}}{2}\right)^{\frac{N+\upsilon_0}{2}} (\beta_{\delta^2})^{m/2+\alpha_{\delta^2}}}{\Gamma\left(\frac{N+\upsilon_0}{2}\right) \Gamma(m/2 + \alpha_{\delta^2}) (2\pi)^{m/2}} \\
&\qquad\qquad \times (\sigma^2)^{(N+\upsilon_0+m)/2+1} (\delta^{*2})^{m/2+\alpha_{\delta^2}+1} |\mathbf{M}_k|^{1/2} \\
&\qquad \times \exp\left(\frac{-1}{2\sigma^2}\left[(\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k^{\bullet})' \mathbf{M}_k^{\bullet -1} (\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k^{\bullet}) + \gamma_0 + \mathbf{y}'_{1:N} \mathbf{P}_k^{\bullet} \mathbf{y}_{1:N}\right]\right. \\
&\qquad\qquad \left. - \frac{\beta_{\delta^2}}{\delta^{*2}}\right) d\boldsymbol{\alpha}_{1:m} d\sigma^2 \\
&\quad = \frac{|\mathbf{M}_k^{\bullet}|^{1/2}}{|\mathbf{M}_k|^{1/2}} \frac{\left(\frac{\gamma_0 + \mathbf{y}'_{1:N} \mathbf{P}_k \mathbf{y}_{1:N}}{2}\right)^{\frac{N+\upsilon_0}{2}} (\beta_{\delta^2})^{m/2+\alpha_{\delta^2}}}{\Gamma(m/2 + \alpha_{\delta^2}) \left(\frac{\gamma_0 + \mathbf{y}'_{1:N} \mathbf{P}_k^{\bullet} \mathbf{y}_{1:N}}{2}\right)^{\frac{N+\upsilon_0}{2}} (\delta^{*2})^{m/2+\alpha_{\delta^2}+1}} \exp\left(-\frac{\beta_{\delta^2}}{\delta^{*2}}\right) \\
&\quad \geq \left(\frac{1 + \frac{1}{\delta^2}}{1 + \frac{1}{\delta^2} + \frac{1}{\delta^{*2}}}\right)^{m/2} \frac{\varepsilon^{\frac{N+\upsilon_0}{2}} \beta_{\delta^2}^{m/2+\alpha_{\delta^2}}}{(\gamma_0 + \mathbf{y}'_{1:N} \mathbf{y}_{1:N})^{\frac{N+\upsilon_0}{2}} \Gamma(m/2 + \alpha_{\delta^2})} \frac{1}{(\delta^{*2})^{m/2+\alpha_{\delta^2}+1}} \\
&\qquad \times \exp\left(-\frac{\beta_{\delta^2}}{\delta^{*2}}\right) \\
&\quad \geq \left(\frac{1}{1 + \delta^{*2}}\right)^{(k_{\max}+d+1)/2} \frac{\varepsilon^{\frac{N+\upsilon_0}{2}} \min_{k \in \{0,\ldots,k_{\max}\}} \beta_{\delta^2}^{m/2+\alpha_{\delta^2}}}{(\gamma_0 + \mathbf{y}'_{1:N} \mathbf{y}_{1:N})^{\frac{N+\upsilon_0}{2}} \Gamma\left((k_{\max} + d + 1)/2 + \alpha_{\delta^2}\right)}
\end{aligned}
$$

$$\times \frac{1}{(\delta^{*2})^{\frac{k_{\max}+d+1}{2}+\alpha_{\delta^2}+1}} \exp\left(-\frac{\beta_{\delta^2}}{\delta^{*2}}\right),$$

where we have made use of lemma 1, its corollary and lemma 2.

Proposition 1. *For any M_2 sufficiently large, there exists an $\eta_{M_2} > 0$ such that for all $((\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}), (\Lambda_{k_2}, \delta^2_{k_2}, k_2, \boldsymbol{\mu}_{1:k_2})) \in (\mathbb{R}^{+2} \times \boldsymbol{\Omega})^2$*

$$\begin{aligned}
&\mathcal{K}^{(k_{\max})}\Lambda_{k_1}, \delta^2_1, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_2}, d\delta^2_{k_2}, k_2, d\boldsymbol{\mu}_{1:k_2}) \\
&\quad \geq \mathbb{I}_{\{\Lambda_{k_1};\Lambda_{k_1}<M_2\}}(\Lambda_{k_1})\eta_{M_2}\phi(d\Lambda_{k_2}, d\delta^2_{k_2}, k_2, d\boldsymbol{\mu}_{1:k_2})
\end{aligned}$$

where $\phi(d\Lambda, d\delta^2, k, d\boldsymbol{\mu}_{1:k}) \triangleq p(\Lambda \mid k)d\Lambda\varphi(\delta^2 \mid k)d\delta^2\mathbb{I}_{\{0\}}(k)\delta_{\{\boldsymbol{\mu}_0\}}(d\boldsymbol{\mu}_{1:k})$.

Proof. From lemmas 3 and 5, one obtains for $k_1 = 1, \ldots, k_{\max}$:

$$\begin{aligned}
&\mathcal{K}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1-1}, d\delta^2_{k_1-1}, k_1-1, d\boldsymbol{\mu}_{k_1-1}) \\
&\quad \geq \mathbb{I}_{\{\Lambda_{k_1};\Lambda_{k_1}<M_2\}}(\Lambda_{k_1})\frac{c^\star}{M_2}\frac{1}{M_1 k_1}\xi p(\Lambda_{k_1-1} \mid k_1-1)d\Lambda_{k_1-1}\varphi(\delta^2_{k_1-1} \mid k_1-1) \\
&\qquad \times d\delta^2_{k_1-1}\delta_{S_{\boldsymbol{\mu}_{1:k_1}}}(d\boldsymbol{\mu}_{k_1-1}).
\end{aligned}$$

Consequently for $k_1 = 1, \ldots, k_{\max}$, when one iterates the kernel $\mathcal{K}$ $k_{\max}$ times, the resulting transition kernel denoted $\mathcal{K}^{(k_{\max})}$ satisfies:

$$\begin{aligned}
&\mathcal{K}^{(k_{\max})}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0) \\
&\quad = \int_{\mathbb{R}^+\times\mathbb{R}^+\times\boldsymbol{\Omega}} \mathcal{K}^{(k_1)}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_l, d\delta^2_l, l, d\boldsymbol{\mu}_{1:l})\mathcal{K}^{(k_{\max}-k_1)} \\
&\qquad \times (\Lambda_l, \delta^2_l, l, \boldsymbol{\mu}_{1:l}; d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0) \\
&\quad \geq \int_{\mathbb{R}^+\times\mathbb{R}^+} \int_{\{0\}\times\boldsymbol{\Omega}_0} \mathcal{K}^{(k_1)}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_l, d\delta^2_l, l, d\boldsymbol{\mu}_{1:l})\mathcal{K}^{(k_{\max}-k_1)} \\
&\qquad \times (\Lambda_l, \delta^2_l, l, \boldsymbol{\mu}_{1:l}; d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0) \\
&\quad = \mathcal{K}^{(k_1)}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_0, d\delta^2_0, 0, d\boldsymbol{\mu}_0)\mathcal{K}^{(k_{\max}-k_1)} \\
&\qquad \times (\Lambda_0, \delta^2_0, 0, \boldsymbol{\mu}_0; d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0) \\
&\quad \geq \mathbb{I}_{\{\Lambda_{k_1};\Lambda_{k_1}<M_2\}}(\Lambda_{k_1})M_3^{k_1-1}\left(\frac{\xi c^\star}{M_1 M_2}\right)^{k_1}\varsigma^{k_{\max}-k_1}\phi(d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0),
\end{aligned}$$

where we have used lemma 4 and $M_3 = \min_{k=1,\ldots,k_{\max}} \int_{\{\Lambda;\Lambda<M_2\}} p(\Lambda \mid k)d\Lambda > 0$. The conclusion follows with $\eta_{M_2} \triangleq \min\{\varsigma^{k_{\max}}, \min_{k\in\{1,\ldots,k_{\max}\}} M_3^{k-1}(\frac{\xi c^\star}{M_1 M_2})^k \varsigma^{k_{\max}-k}\} > 0$.

Corollary 3. *The transition kernel $\mathcal{K}$ is ϕ-irreducible. In addition, we know that $p(d\Lambda, d\delta^2, k, d\boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y})$ is an invariant distribution of $\mathcal{K}$ and the Markov chain is ϕ-irreducible; then according to Tierney (1994, theorem 1*) the Markov chain is $p(d\Lambda, d\delta^2, k, d\boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y})$-irreducible. Aperiodicity is straightforward. Indeed there is a nonzero probability of choosing the update move in the empty configuration from equation B.5 and to move anywhere in $\mathbb{R}^2 \times \{0\} \times \{\boldsymbol{\mu}_0\}$. Therefore, the Markov chain admits $p(d\Lambda, d\delta^2, k, d\boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y})$ as unique equilibrium distribution (Tierney, 1994, theorem 1*).*

We will now prove the drift condition:

Proposition 2. *Let $V(\Lambda, \delta^2, k, \boldsymbol{\mu}_{1:k}) \triangleq \max\{1, \Lambda^{\upsilon}\}$ for $\upsilon > 0$; then*

$$\lim_{\Lambda \to +\infty} \mathcal{K}V(\Lambda, \delta^2, k, \boldsymbol{\mu}_{1:k}) / V(\Lambda, \delta^2, k, \boldsymbol{\mu}_{1:k}) = 0,$$

where by definition,

$$\mathcal{K}V(\Lambda, \delta^2, k, \boldsymbol{\mu}_{1:k}) \triangleq \int_{\mathbb{R}^+ \times \mathbb{R}^+ \times \boldsymbol{\Omega}} \mathcal{K}(\Lambda, \delta^2, k, \boldsymbol{\mu}_{1:k}; d\Lambda^*, d\delta^{*2}, k^*, d\boldsymbol{\mu}^*_{1:k}) V(\Lambda^*, \delta^{*2}, k^*, \boldsymbol{\mu}^*_{1:k})$$

Proof. The transition kernel of the Markov chain is of the form (we remove some arguments for convenience):

$$\begin{aligned}\mathcal{K} = \big(&b_{k_1}\mathcal{K}_{birth} + d_{k_1}\mathcal{K}_{death} + m_{k_1}\mathcal{K}_{merge} + s_{k_1}\mathcal{K}_{split} \\ &+ (1 - b_{k_1} - d_{k_1} - s_{k_1} - m_{k_1})\,\mathcal{K}_{update}\big) p(\delta^2_{k_2} \mid \delta^2_{k_1}, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) \\ &\times p(\Lambda_{k_2} \mid k_2).\end{aligned}$$

Now, study the following expression:

$$\begin{aligned}&\mathcal{K}V(\Lambda_{k_1}, \delta^2_1, k_1, \boldsymbol{\mu}_{1:k_1}) \\ &= b_{k_1} \sum_{k_2 \in \{k_1, k_1+1\}} \int_{\Phi_{k_2}} \mathcal{K}_{birth} \int_{\mathbb{R}^+} p(\delta^2_{k_2} \mid \delta^2_{k_1}, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) d\delta^2_{k_2} \\ &\quad \times \int_{\mathbb{R}^+} p(\Lambda_{k_2} \mid k_2) \Lambda^{\upsilon}_{k_2} d\Lambda_{k_2} \\ &\quad + d_{k_1} \sum_{k_2 \in \{k_1, k_1-1\}} \int_{\Phi_{k_2}} \mathcal{K}_{death} \int_{\mathbb{R}^+} p(\delta^2_{k_2} \mid \delta^2_{k_1}, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) d\delta^2_{k_2} \\ &\quad \times \int_{\mathbb{R}^+} p(\Lambda_{k_2} \mid k_2) \Lambda^{\upsilon}_{k_2} d\Lambda_{k_2} \\ &\quad + s_{k_1} \sum_{k_2 \in \{k_1, k_1+1\}} \int_{\Phi_{k_2}} \mathcal{K}_{split} \int_{\mathbb{R}^+} p(\delta^2_{k_2} \mid \delta^2_{k_1}, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) d\delta^2_{k_2}\end{aligned}$$

$$\times \int_{\mathbb{R}^+} p(\Lambda_{k_2} \mid k_2)\Lambda_{k_2}^{\upsilon} d\Lambda_{k_2}$$

$$+ m_{k_1} \sum_{k_2 \in \{k_1, k_1-1\}} \int_{\Phi_{k_2}} \mathcal{K}_{merge} \int_{\mathbb{R}^+} p(\delta_{k_2}^2 \mid \delta_{k_1}^2, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) d\delta_{k_2}^2$$

$$\times \int_{\mathbb{R}^+} p(\Lambda_{k_2} \mid k_2)\Lambda_{k_2}^{\upsilon} d\Lambda_{k_2}$$

$$+ (1 - b_{k_1} - d_{k_1} - s_{k_1} - m_{k_1}) \int_{\boldsymbol{\Omega}_{k_1}} \mathcal{K}_{update}$$

$$\times \int_{\mathbb{R}^+} p(\delta_{k_1}^{*2} \mid \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}^*, \mathbf{x}, \mathbf{y}) d\delta_{k_1}^2 \int_{\mathbb{R}^+} p(\Lambda_{k_1}^* \mid k_1)\Lambda_{k_1}^{*\upsilon} d\Lambda_{k_1}^*$$

$$= b_{k_1} \sum_{k_2 \in \{k_1, k_1+1\}} \int_{\mathbb{R}^+} p(\Lambda_{k_2} \mid k_2)\Lambda_{k_2}^{\upsilon} d\Lambda_{k_2}$$

$$+ d_{k_1} \sum_{k_2 \in \{k_1, k_1-1\}} \int_{\mathbb{R}^+} p(\Lambda_{k_2} \mid k_2)\Lambda_{k_2}^{\upsilon} d\Lambda_{k_2}$$

$$+ s_{k_1} \sum_{k_2 \in \{k_1, k_1+1\}} \int_{\mathbb{R}^+} p(\Lambda_{k_2} \mid k_2)\Lambda_{k_2}^{\upsilon} d\Lambda_{k_2}$$

$$+ m_{k_1} \sum_{k_2 \in \{k_1, k_1-1\}} \int_{\mathbb{R}^+} p(\Lambda_{k_2} \mid k_2)\Lambda_{k_2}^{\upsilon} d\Lambda_{k_2}$$

$$+ (1 - b_{k_1} - d_{k_1} - s_{k_1} - m_{k_1}) \int_{\mathbb{R}^+} p(\Lambda_{k_1}^* \mid k_1)\Lambda_{k_1}^{*\upsilon} d\Lambda_{k_1}^*.$$

As $p(\Lambda \mid k)$ is a gamma distribution, for any $0 \leq k \leq k_{\max}$, one obtains the inequality $\int_{\mathbb{R}^+} p(\Lambda \mid k)\Lambda^{\upsilon} d\Lambda < +\infty$, and then the result follows immediately.

Proof of Theorem 3. By construction, the transition kernel $\mathcal{K}(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_2}, d\delta_{k_2}^2, k_2, d\boldsymbol{\mu}_{1:k_2})$ admits the probability distribution $p(d\Lambda, d\delta^2, k, d\boldsymbol{\mu}_{1:k} \mid \mathbf{x}, \mathbf{y})$ as invariant distribution. Proposition 1 proved the ϕ-irreducibility and the minorization condition with $k_0 = k_{\max}$, and proposition 2 proved the drift condition. Thus, theorem 3 applies.

Acknowledgments

We thank Mark Coates, Bill Fitzgerald, and Jaco Vermaak (Cambridge University), Chris Holmes (Imperial College of London), David Lowe (Aston University), David Melvin (Cambridge Clinical School), Stephen Roberts, and Will Penny (University of Oxford) for very useful discussions and comments.

References

Akaike, H. (1974). A new look at statistical model identification. *IEEE Transactions on Automatic Control, AC-19*, 716–723.

Andrieu, C. (1998). *MCMC methods for the Bayesian analysis of nonlinear parametric regression models*. Unpublished doctoral dissertation, University Cergy-Pontoise, Paris.

Andrieu, C., Breyer, L. A., & Doucet, A. (1999). *Convergence of simulated annealing using Foster-Lyapunov criteria* (Tech. Rep. No. CUED/F-INFENG/TR 346). Cambridge: Cambridge University Engineering Department.

Andrieu, C., de Freitas, J. F. G., & Doucet, A. (1999a). *Sequential Bayesian estimation and model selection applied to neural networks* (Tech. Rep. No. CUED/F-INFENG/TR 341). Cambridge: Cambridge University Engineering Department.

Andrieu, C., de Freitas, J. F. G., & Doucet, A. (1999b). Sequential MCMC for Bayesian model selection. In *IEEE Higher Order Statistics Workshop* (pp. 130–134), Caesarea, Israel.

Andrieu, C., Djurić, P. M., & Doucet, A. (in press). Model selection by MCMC computation. *Signal Processing*.

Bernardo, J. M., & Smith, A. F. M. (1994). *Bayesian theory*. New York: Wiley.

Brass, A., Pendleton, B. J., Chen, Y., & Robson, B. (1993). Hybrid Monte Carlo simulations theory and initial comparison with molecular dynamics. *Biopolymers, 33*(8), 1307–1315.

Buntine, W. L., & Weigend, A. S. (1991). Bayesian back-propagation. *Complex Systems, 5*, 603–643.

Cheng, B., & Titterington, D. M. (1994). Neural networks: A review from a statistical perspective. *Statistical Science, 9*(1), 2–54.

de Freitas, J. F. G. (1999). *Bayesian methods for neural networks*. Unpublished doctoral dissertation, Cambridge University, Cambridge, UK.

de Freitas, J. F. G., Niranjan, M., & Gee, A. H. (1998). *The EM algorithm and neural networks for nonlinear state space estimation* (Tech. Rep. No. CUED/F-INFENG/TR 313). Cambridge: Cambridge University Engineering Department.

de Freitas, J. F. G., Niranjan, M., Gee, A. H., & Doucet, A. (2000). Sequential Monte Carlo methods to train neural network models. *Neural Computation, 12*(4), 955–993.

Denison, D., Mallick, B. K., & Smith, A. F. M. (1998). Bayesian MARS, *Statistics and Computing, 8*, 337–346.

Djurić, P. M. (1996). A model selection rule for sinusoids in white gaussian noise. *IEEE Transactions on Signal Processing, 44*(7), 1744–1751.

Djurić, P. M. (1998). Asymptotic MAP criteria for model selection. *IEEE Transactions on Signal Processing, 46*(10), 2726–2735.

Fahlman, S. E., & Lebiere, C. (1988). The cascade-correlation learning architecture. In D. S. Touretzky (Ed.), *Proceedings of the Connectionist Models Summer School* (Vol. 2, pp. 524–532) San Mateo, CA.

Frean, M. (1990). The upstart algorithm: A method for constructing and training feedforward neural networks. *Neural Computation, 2*(2), 198–209.

Gelfand, A. E., & Dey, D. K. (1997). Bayesian model choice: Asymptotics and exact calculations. *Journal of the Royal Statistical Society B, 56*(3), 501–514.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (1995). *Bayesian data analysis*. London: Chapman and Hall.

Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 6*(6), 721–741.

George, E. I., & Foster, D. P. (1997). *Calibration and empirical Bayes variable selection*. Unpublished manuscript, Department of Management Science and Information Systems, University of Texas.

Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (Eds.). (1996). *Markov chain Monte Carlo in practice*. London: Chapman and Hall.

Girosi, F., Jones, M., & Poggio, T. (1995). Regularization theory and neural networks architectures. *Neural Computation, 7*(2), 219–269.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika, 82*, 711–732.

Hand, D. J. (1997). *Construction and assessment of classification rules*. New York: Wiley.

Hinton, G. (1987). Learning translation invariant recognition in massively parallel networks. In J. W. de Bakker, A. J. Nijman, & P. C. Treleaven (Eds.), *Proceedings of the Conference on Parallel Architectures and Languages* (pp. 1–13) Berlin.

Holmes, C. C., & Mallick, B. K. (1998). Bayesian radial basis functions of variable dimension. *Neural Computation, 10*(5), 1217–1233.

Holmes, C. C., & Mallick, B. K. (1999). *Generalised nonlinear modelling with multivariate smoothing splines*. Unpublished manuscript, Statistics Section, Department of Mathematics, Imperial College of London.

Holmes, C. C., & Mallick, B. K. (2000). Bayesian wavelet networks for nonparametric regression. *IEEE Transactions on Neural Networks, 11*(1), 27–35.

Kadirkamanathan , V., & Niranjan, M. (1993). A function estimation approach to sequential learning with neural networks. *Neural Computation, 5*(6), 954–975.

Le Cun, Y., Denker, J. S., & Solla, S. A. (1990). Optimal brain damage. In D. S. Touretzky (Ed.), *Advances in neural information processing systems, 2* (pp. 598–605). San Mateo, CA: Morgan Kaufmann.

Mackay, D. J. C. (1992). A practical Bayesian framework for backpropagation networks. *Neural Computation, 4*(3), 448–472.

Marrs, A. D. (1998). An application of reversible-jump MCMC to multivariate spherical Gaussian mixtures. In M. I. Jordan, M. J. Kearns, & S. A. Solla (Eds.), *Advances in neural information processing systems, 10* (pp. 577–583). Cambridge, MA: MIT Press.

Meyn, S. P., & Tweedie, R. L. (1993). *Markov chains and stochastic stability*. New York: Springer-Verlag.

Moody, J., & Darken, C. (1988). Learning with localized receptive fields. In G. Hinton, T. Sejnowski, & D. Touretzsky (Eds.), *Proceedings of the 1988 Connectionist Models Summer School* (pp. 133–143). Palo Alto, CA.

Müller, P., & Rios Insua, D. (1998). Issues in Bayesian analysis of neural network models. *Neural Computation, 10*, 571–592.

Nabney, I. T. (1999). *Efficient training of RBF networks for classification* (Tech. Rep. No. NCRG/99/002). Birmingham, UK: Neural Computing Research Group, Aston University.

Neal, R. M. (1996). *Bayesian learning for neural networks*. New York: Springer-Verlag.

Platt, J. (1991). A resource allocating network for function interpolation. *Neural Computation, 3*, 213–225.

Richardson, S., & Green, P. J. (1997). On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society B, 59*(4), 731–792.

Rios Insua, D., & Müller, P. (1998). Feedforward neural networks for nonparametric regression. In D. K. Dey, P. Müller, & D. Sinha (Eds.), *Practical nonparametric and semiparametric Bayesian statistics* (pp. 181–191). New York: Springer Verlag.

Rissanen, J. (1987). Stochastic complexity. *Journal of the Royal Statistical Society, 49*, 223–239.

Robert, C. P., & Casella, G. (1999). *Monte Carlo statistical methods*. New York: Springer-Verlag.

Roberts, S. J., & Penny, W. D. (1998). Bayesian neural networks for classification: How useful is the evidence framework? *Neural Networks, 12*, 877–892.

Roberts, S. J., Penny, W. D., & Pillot, D. (1996). Novelty, confidence and errors in connectionist systems. *IEE Colloquium on Intelligent Sensors and Fault Detection*, no. 261, 10/1–10/6.

Schwarz, G. (1985). Estimating the dimension of a model. *Annals of Statistics, 6*(2), 461–464.

Smith, M., & Kohn, R. (1996). Nonparametric regression using Bayesian variable selection. *Journal of Econometrics, 75*(2), 317–343.

Spyers-Ashby, J. M., Bain, P., & Roberts, S. J. (1998). A comparison of fast Fourier transform (FFT) and autoregressive (AR) spectral estimation techniques for the analysis of tremor data. *Journal of Neuroscience Methods, 83*(1), 35–43.

Tierney, L. (1994). Markov chains for exploring posterior distributions. *The Annals of Statistics, 22*(4), 1701–1762.

Van Laarhoven, P. J., & Arts, E. H. L. (1987). *Simulated annealing: Theory and applications*, Amsterdam: Reidel P.

Vapnik, V. (1995). *The Nature of statistical learning theory*. New York: Springer-Verlag.

Yingwei, L., Sundararajan, N., & Saratchandran, P. (1997). A sequential learning scheme for function approximation using minimal radial basis function neural networks. *Neural Computation, 9*, 461–478.

Zellner, A. (1986). On assessing prior distributions and Bayesian regression analysis with g-prior distributions. In P. Goel & A. Zellner (Eds.), *Bayesian inference and decision techniques* (pp. 233–243) Amsterdam: Elsevier.

Received August 25, 1999; accepted November 17, 2000.