CAMBRIDGE UNIVERSITY

ENGINEERING DEPARTMENT

ROBUST FULL BAYESIAN LEARNING FOR NEURAL NETWORKS

C Andrieu[1], JFG de Freitas and A Doucet

Draft of CUED/F-INFENG/TR 343

July 14, 1999

Cambridge University Engineering Department
Trumpington Street
Cambridge CB2 1PZ
England

E-mail: {ca226,jfgf,ad2}@eng.cam.ac.uk

[1]Authorship based on alphabetical order

Abstract

In this paper, we propose a hierarchical full Bayesian model for neural networks. This model treats the model dimension (number of neurons), model parameters, regularisation parameters and noise parameters as random variables that need to be estimated. We develop a reversible jump Markov chain Monte Carlo (MCMC) method to perform the necessary computations. 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 optimise neural networks. This algorithm enables us to maximise 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. We show that by calibrating the full hierarchical Bayesian prior, we can obtain the classical AIC, BIC and MDL model selection criteria within a penalised 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.

Contents

1 Introduction

Artificial neural networks are powerful nonlinear approximation tools, which rely on structured combinations of many parameterised basis functions to perform regression, classification and density estimation. They can approximate any continuous function arbitrarily well as the number of neurons (basis functions) increases without bound (Cybenko 1989, Poggio and Girosi 1990). In addition, they have been successfully applied to many and varied complex problems, including speech recognition (Robinson 1994), hand written digit recognition (Le Cun, Boser, Denker, Henderson, Howard and Hubbard 1989), financial modelling (Refenes 1995) and medical diagnosis (Baxt 1990) among others. In these scenarios, it has been shown that it is possible to train networks with hundreds or thousands of parameters using various heuristics and standard techniques, such as back-propagation (Rumelhart, Hinton and Williams 1986) and cross-validation (Wahba and Wold 1969).

In the early nineties, 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 possible to obtain estimates of the weights and output variances and to automatically set the regularisation coefficients.

Neal (1996) cast the net much further by introducing advanced Bayesian simulation methods, specifically the hybrid Monte Carlo method (Brass, Pendleton, Chen and Robson 1993, Duane, Kennedy, Pendleton and Roweth 1987), into the analysis of neural networks. Theoretically, he also proved that certain classes of priors for neural networks, whose number or 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 and Doucet 1999).

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, namely penalised likelihood, predictive assessment and growing and pruning techniques. In the penalised likelihood context, a penalty term is added to the likelihood function so as to limit the number of neurons; thereby avoiding over-fitting. 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). Penalised likelihood has also been used extensively to impose smoothing constraints either via weight decay priors (Hinton 1987, Mackay 1992) or functional regularisers that penalise for high frequency signal components (Girosi, Jones and Poggio 1995).

In the predictive assessment approach, the data is 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 and Lebiere 1988), optimal brain damage (Le Cun, Denker and 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 centre 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 and Niranjan 1993). Yingwei, Sundararajan and Saratchandran (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 (Green 1995, Richardson and Green 1997) to feed-forward 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 paper, 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. However, we advance this area of research in three important directions. Firstly, we propose a hierarchical prior for RBF networks. That is, we adopt a full Bayesian model, which accounts for model order uncertainty and regularisation, and show that the results appear to be robust with respect to the prior specification. Secondly, we propose an automated growing and pruning reversible jump MCMC optimisation 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. Finally, we derive a geometric convergence theorem for the homogeneous reversible jump MCMC algorithm and a convergence theorem for the annealed reversible jump MCMC algorithm.

The remainder of the paper is organised as follows: in Section 1, we present the approximation model. In Section 2, we formalise 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 optimisation 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 in the paper. 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:

$$\mathbf{y}_t = \mathbf{f}(\mathbf{x}_t) + \mathbf{n}_t \tag{1}$$

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 $t = \{1, 2, \cdots\}$ 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:

$$\mathcal{O} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N, \mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_N\}$$

Typical examples include regression, where $\mathbf{y}_{1:N,1:c}$[2] 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 synthesised as a combination of parametrised basis functions. That is:

$$\hat{\mathbf{f}}(\mathbf{x}, \boldsymbol{\theta}) = G_k \left(\boldsymbol{\theta}_k; \left(\cdots \sum_j G_j \left(\boldsymbol{\theta}_j; \sum_i G_i(\boldsymbol{\theta}_i; \mathbf{x}) \right) \cdots \right) \right) \tag{2}$$

[2] $\mathbf{y}_{1:N,1:c}$ is an N by c matrix, where N is the number of data and c the number of outputs. We adopt the notation $\mathbf{y}_{1:N,j} \triangleq (\mathbf{y}_{1,j}, \mathbf{y}_{2,j}, \cdots, \mathbf{y}_{N,j})'$ to denote all the observations corresponding to the j-th output (j-th 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 will favour the shorter notation and only invoke the longer notation to avoid ambiguities and emphasise certain dependencies.

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 modelling is often referred to as "non-parametric" regression because the number of basis functions is typically very large. Equation (2) encompasses a large number of nonlinear estimation methods including projection pursuit regression (Friedman and Stuetzle 1981, Huber 1985), Volterra series (Billings 1980, Mathews 1991), fuzzy inference systems (Jang and Sun 1993), multivariate adaptive regression splines (MARS) (Cheng and Titterington 1994, Denison 1998, Friedman 1991) and many artificial neural network paradigms such as functional link networks (Pao 1989), multi-layer perceptrons (MLPs) (Rosenblatt 1959, Rumelhart et al. 1986), radial basis function networks (Lowe 1989, Moody and Darken 1988, Poggio and Girosi 1990), wavelet networks (Bakshi and Stephanopoulos 1993, Juditsky, Hjalmarsson, Benveniste, Delyon, Ljung and Sjöberg 1995) and hinging hyper-planes (Breiman 1993). For an introduction to neural networks, the reader may consult any of the following books (Bishop 1995, Haykin 1994, Hecht-Nielsen 1990, Ripley 1996).

For the purposes of this paper, we adopt the approximation scheme of Holmes and Mallick (1998), consisting of a mixture of k RBFs and a linear regression term. However, the work can be straight-forwardly extended to other regression models. More precisely, our model $\mathcal{M}$ is:

$$\begin{aligned} \mathcal{M}_0 : &\quad \mathbf{y}_t = \boldsymbol{b} + \boldsymbol{\beta}'\mathbf{x}_t + \mathbf{n}_t & k = 0 \\ \mathcal{M}_k : &\quad \mathbf{y}_t = \sum_{j=1}^{k} \mathbf{a}_j \phi(\|\mathbf{x}_t - \boldsymbol{\mu}_j\|) + \boldsymbol{b} + \boldsymbol{\beta}'\mathbf{x}_t + \mathbf{n}_t & k \geq 1 \end{aligned} \tag{3}$$

where $\|\cdot\|$ denotes a distance metric (usually Euclidean or Mahalanobis), $\boldsymbol{\mu}_j \in \mathbb{R}^d$ denotes the j-th RBF centre for a model with k RBFs, $\mathbf{a}_j \in \mathbb{R}^c$ the j-th 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. It is important to mention that 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 $d = 2$. 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: $\phi(\varrho) = \varrho^2 \ln(\varrho)$
- Multi-quadric: $\phi(\varrho) = \left(\varrho^2 + \lambda^2\right)^{1/2}$
- 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:

$$\mathbf{y} = \mathbf{D}\left(\boldsymbol{\mu}_{1:k,1:d}, \mathbf{x}_{1:N,1:d}\right)\boldsymbol{\alpha}_{1:1+d+k,1:c} + \mathbf{n}_t \tag{4}$$

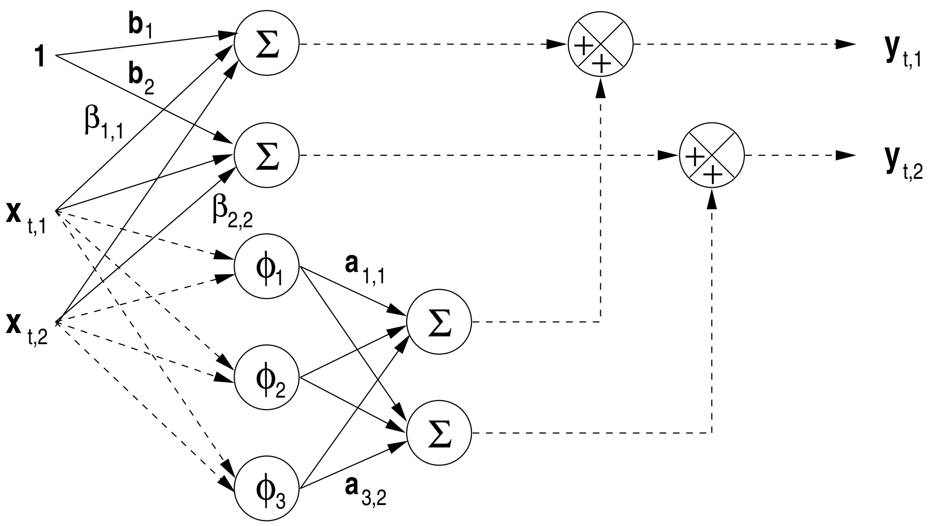

Figure 1: Approximation model with three radial basis functions, two inputs and two outputs. The solid lines indicate weighted connections.

That is:

$$\begin{bmatrix} \mathbf{y}_{1,1}\cdots\mathbf{y}_{1,c} \\ \mathbf{y}_{2,1}\cdots\mathbf{y}_{2,c} \\ \\ \vdots \\ \\ \mathbf{y}_{N,1}\cdots\mathbf{y}_{N,c} \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{x}_{1,1}\cdots\mathbf{x}_{1,d} & \phi(\mathbf{x}_1,\boldsymbol{\mu}_1)\cdots\phi(\mathbf{x}_1,\boldsymbol{\mu}_k) \\ 1 & \mathbf{x}_{2,1}\cdots\mathbf{x}_{2,d} & \phi(\mathbf{x}_2,\boldsymbol{\mu}_1)\cdots\phi(\mathbf{x}_2,\boldsymbol{\mu}_k) \\ \\ \vdots & \vdots & \vdots \\ \\ 1 & \mathbf{x}_{N,1}\cdots\mathbf{x}_{N,d} & \phi(\mathbf{x}_N,\boldsymbol{\mu}_1)\cdots\phi(\mathbf{x}_N,\boldsymbol{\mu}_k) \end{bmatrix} \begin{bmatrix} \boldsymbol{b}_1\cdots\boldsymbol{b}_c \\ \boldsymbol{\beta}_{1,1}\cdots\boldsymbol{\beta}_{1,c} \\ \vdots \\ \boldsymbol{\beta}_{d,1}\cdots\boldsymbol{\beta}_{d,c} \\ \mathbf{a}_{1,1}\cdots\mathbf{a}_{1,c} \\ \vdots \\ \mathbf{a}_{k,1}\cdots\mathbf{a}_{k,c} \end{bmatrix} + \mathbf{n}_{1:N}$$

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

$$\mathbf{n}_t \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\sigma}_1^2 & 0 & \cdots & 0 \\ 0 & \boldsymbol{\sigma}_2^2 & \cdots & 0 \\ & & \ddots & \\ 0 & 0 & \cdots & \boldsymbol{\sigma}_c^2 \end{bmatrix} \right)$$

Once again, we stress that $\boldsymbol{\sigma}^2$ depends implicitly on the model order k. We assume here that the number k of RBFs and their parameters $\boldsymbol{\theta} \triangleq \{\boldsymbol{\alpha}_{1:m,1:c}, \boldsymbol{\mu}_{1:k,1:d}, \boldsymbol{\sigma}^2_{1:c}\}$, 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 and Smith 1994). Furthermore, we adopt a hierarchical prior structure that enables us to treat the priors' parameters (hyper-parameters) as random variables drawn from suitable distributions (hyper-priors). That is, instead of fixing the hyper-parameters 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 hyper-priors.

The remainder of the section is organised as follows: firstly, we propose a hierarchical model prior which 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 normalising constant, of the joint posterior distribution of the basis centres and their number.

3.1 Prior distributions

The overall parameter space $\mathbf{\Theta} \times \mathbf{\Psi}$ can be written as a finite union of subspaces $\mathbf{\Theta} \times \mathbf{\Psi} = \left(\cup_{k=0}^{k_{\max}} \{k\} \times \mathbf{\Theta}_k \right) \times \mathbf{\Psi}$ where $\mathbf{\Theta}_0 \triangleq (\mathbb{R}^{d+1})^c \times (\mathbb{R}^+)^c$ and $\mathbf{\Theta}_k \triangleq (\mathbb{R}^{d+1+k})^c \times (\mathbb{R}^+)^c \times \mathbf{\Omega}_k$ 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 $\boldsymbol{\mu} \in \mathbf{\Omega}_k$. The hyper-parameter space $\mathbf{\Psi} \triangleq (\mathbb{R}^+)^{c+1}$, with elements $\boldsymbol{\psi} \triangleq \{\Lambda, \boldsymbol{\delta}^2\}$, will be discussed at the end of this section.

The space of the radial basis centres $\mathbf{\Omega}_k$ is defined as a compact set that encompasses the input data: $\mathbf{\Omega}_k \triangleq \{\boldsymbol{\mu}; \boldsymbol{\mu}_{1:k,i} \in [\min(\mathbf{x}_{1:N,i}) - \iota\Xi_i, \max(\mathbf{x}_{1:N,i}) + \iota\Xi_i]^k$ 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 i-th dimension of the input and ι is a user specified parameter that we only need to consider if we wish to place basis functions outside the region where the input data lie. That is, we allow $\mathbf{\Omega}_k$ to include the space of the input data and extend it by a factor which is proportional to the spread of the input data. Typically, researchers either set ι to zero and choose the basis centres from the input data (Holmes and Mallick 1998, Kadirkamanathan and Niranjan 1993) or compute the basis centres using clustering algorithms (Moody and Darken 1988). The premise here is that it is better to place the basis functions where the data is dense; not in regions of extrapolation. In our case, we sample the basis centres from the space $\mathbf{\Omega}_k$, whose hyper-volume is $\Im^k \triangleq \left(\prod_{i=1}^{d} (1 + 2\iota)\Xi_i\right)^k$. Figure 2 shows this space for a two-dimensional input.

The maximum number of basis functions is defined as $k_{\max} \triangleq (N - (d+1))$ [3]. We also define $\mathbf{\Omega} \triangleq \cup_{k=0}^{k_{\max}} \{k\} \times \mathbf{\Omega}_k$ with $\mathbf{\Omega}_0 \triangleq \emptyset$. There is a natural hierarchical structure to this setup (Richardson and Green 1997), which we formalise by modelling the joint distribution of all variables as:

$$p(k, \boldsymbol{\theta}, \boldsymbol{\psi}, \mathbf{y}|\mathbf{x}) = p(\mathbf{y}|k, \boldsymbol{\theta}, \boldsymbol{\psi}, \mathbf{x}) p(\boldsymbol{\theta}|k, \boldsymbol{\psi}, \mathbf{x}) p(k, \boldsymbol{\psi}|\mathbf{x}) \tag{5}$$

[3] The constraint $k \leq N - (d+1)$ is added because otherwise the columns of $\mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})$ are linearly dependent and the parameters $\boldsymbol{\theta}$ may not be uniquely estimated from the data (see Equation (4)).

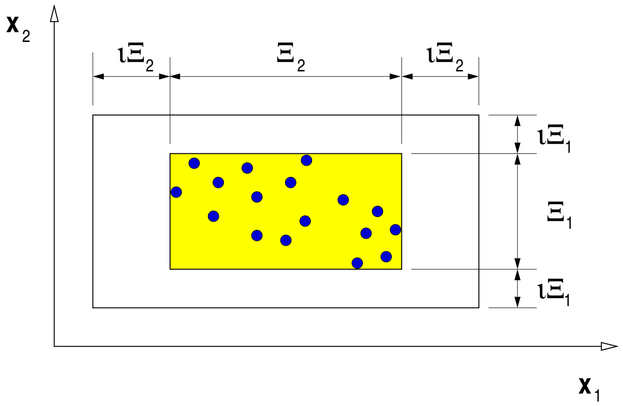

Figure 2: RBF centres space $\boldsymbol{\Omega}$ for a two-dimensional input. The circles represent the input data.

where $p(k, \boldsymbol{\psi}|\mathbf{x})$ is the joint model order and hyper-parameters probability, $p(\boldsymbol{\theta}|k, \boldsymbol{\psi}, \mathbf{x})$ is the parameters' prior and $p(\mathbf{y}|k, \boldsymbol{\theta}, \boldsymbol{\psi}, \mathbf{x})$ 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:

$$
\begin{aligned}
p(\mathbf{y}|k, \boldsymbol{\theta}, \boldsymbol{\psi}, \mathbf{x}) &= \prod_{i=1}^{c} p(\mathbf{y}_{1:N,i}|k, \boldsymbol{\alpha}_{1:m,i}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}_i^2, \mathbf{x}) \\
&= \prod_{i=1}^{c} (2\pi\boldsymbol{\sigma}_i^2)^{-N/2} \exp\left(-\frac{1}{2\boldsymbol{\sigma}_i^2}\left(\mathbf{y}_{1:N,i} - \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})\boldsymbol{\alpha}_{1:m,i}\right)'\left(\mathbf{y}_{1:N,i} - \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x})\boldsymbol{\alpha}_{1:m,i}\right)\right)
\end{aligned}
\tag{6}
$$

We assume the following structure for the prior distribution:

$$
\begin{aligned}
p(k, \boldsymbol{\theta}, \boldsymbol{\psi}) &= p(\boldsymbol{\alpha}_{1:m}|k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) p(\boldsymbol{\mu}_{1:k}|k, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) p(k|\boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) p(\boldsymbol{\sigma}^2|\Lambda, \boldsymbol{\delta}^2) p(\Lambda, \boldsymbol{\delta}^2) \\
&= p(\boldsymbol{\alpha}_{1:m}|k, \boldsymbol{\sigma}^2, \boldsymbol{\delta}^2) p(\boldsymbol{\mu}_{1:k}|k) p(k|\Lambda) p(\boldsymbol{\sigma}^2) p(\Lambda) p(\boldsymbol{\delta}^2)
\end{aligned}
\tag{7}
$$

where the scale parameters $\boldsymbol{\sigma}_i^2$, $i = 1, \ldots, c$, are assumed to be independent of the hyper-parameters (*i.e.* $p(\boldsymbol{\sigma}^2|\Lambda, \boldsymbol{\delta}^2) = p(\boldsymbol{\sigma}^2)$), 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:

$$
\boldsymbol{\sigma}_i^2 \sim \mathcal{IG}\left(\frac{\upsilon_0}{2}, \frac{\gamma_0}{2}\right)
$$

When $\upsilon_0 = 0$ and $\gamma_0 = 0$, we obtain Jeffreys' uninformative prior $p\left(\boldsymbol{\sigma}_i^2\right) \propto 1/\boldsymbol{\sigma}_i^2$ (Bernardo and Smith 1994). Given $\boldsymbol{\sigma}^2$, we introduce the following prior distribution:

$$
\begin{aligned}
p(k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k}|\boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) &= p(\boldsymbol{\alpha}_{1:m}|k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) p(\boldsymbol{\mu}_{1:k}|k, \boldsymbol{\sigma}^2) p(k|\boldsymbol{\sigma}^2, \Lambda, \boldsymbol{\delta}^2) \\
&= \left[\prod_{i=1}^{c} |2\pi\boldsymbol{\sigma}_i^2\boldsymbol{\Sigma}_i|^{-1/2} \exp\left(-\frac{1}{2\boldsymbol{\sigma}_i^2}\boldsymbol{\alpha}'_{1:m,i}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\alpha}_{1:m,i}\right)\right]\left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k, \boldsymbol{\mu}_{1:k})}{\Im^k}\right]\left[\frac{\Lambda^k/k!}{\sum_{j=0}^{k_{\max}} \Lambda^j/j!}\right]
\end{aligned}
\tag{8}
$$

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

The prior model order distribution $p(k|\Lambda)$ is a truncated Poisson distribution. Conditional upon k, the RBF centres are uniformly distributed. Finally, conditional upon $(k,\boldsymbol{\mu}_{1:k})$, the coefficients $\boldsymbol{\alpha}_{1:m,i}$ are assumed to be zero-mean Gaussian with variance $\boldsymbol{\sigma}_i^2\mathbf{\Sigma}_i$. 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 and Foster 1997, Smith and 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 penalises for basis functions being too close as, in this situation, the determinant of $\mathbf{\Sigma}_i^{-1}$ tends to zero.

We now turn our attention to the hyper-parameters, which, as mentioned before, allow us to accomplish our goal of designing robust model selection schemes. We assume that they are independent of each other, *i.e.* $p(\Lambda,\boldsymbol{\delta}^2) = p(\Lambda)p(\boldsymbol{\delta}^2)$. 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 hyper-prior with $\alpha_{\delta^2} = 2$ is infinite. We apply the same method to Λ by setting an uninformative conjugate prior (Bernardo and Smith 1994): $\Lambda \sim \mathcal{G}a\left(1/2+\varepsilon_1,\varepsilon_2\right)$ ($\varepsilon_i \ll 1$ $i = 1,2$). We can visualise our hierarchical prior (equation (7)) 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 $p(k,\boldsymbol{\theta},\boldsymbol{\psi}|\mathbf{x},\mathbf{y})$ obtained from Bayes' theorem. Our aim is to estimate this joint distribution from which, by standard probability marginalisation and transformation techniques, one can "theoretically" obtain all posterior features of interest. For instance, we might wish to perform inference with the predictive density:

$$p(\mathbf{y}_{N+1}|\mathbf{x}_{1:N+1},\mathbf{y}_{1:N}) = \int_{\mathbf{\Theta}\times\mathbf{\Psi}} p(\mathbf{y}_{N+1}|k,\boldsymbol{\theta},\boldsymbol{\psi},\mathbf{x}_{N+1})p(k,\boldsymbol{\theta},\boldsymbol{\psi}|\mathbf{x}_{1:N},\mathbf{y}_{1:N})\mathrm{d}k\mathrm{d}\boldsymbol{\theta}\mathrm{d}\boldsymbol{\psi} \tag{9}$$

and consequently make predictions, such as:

$$\mathbb{E}(\mathbf{y}_{N+1}|\mathbf{x}_{1:N+1},\mathbf{y}_{1:N}) = \int_{\mathbf{\Theta}\times\mathbf{\Psi}} \mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x}_{N+1})\boldsymbol{\alpha}_{1:m}p(k,\boldsymbol{\theta},\boldsymbol{\psi}|\mathbf{x}_{1:N},\mathbf{y}_{1:N})\mathrm{d}k\mathrm{d}\boldsymbol{\theta}\mathrm{d}\boldsymbol{\psi} \tag{10}$$

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

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 subsection. We propose here to use an MCMC method to perform Bayesian

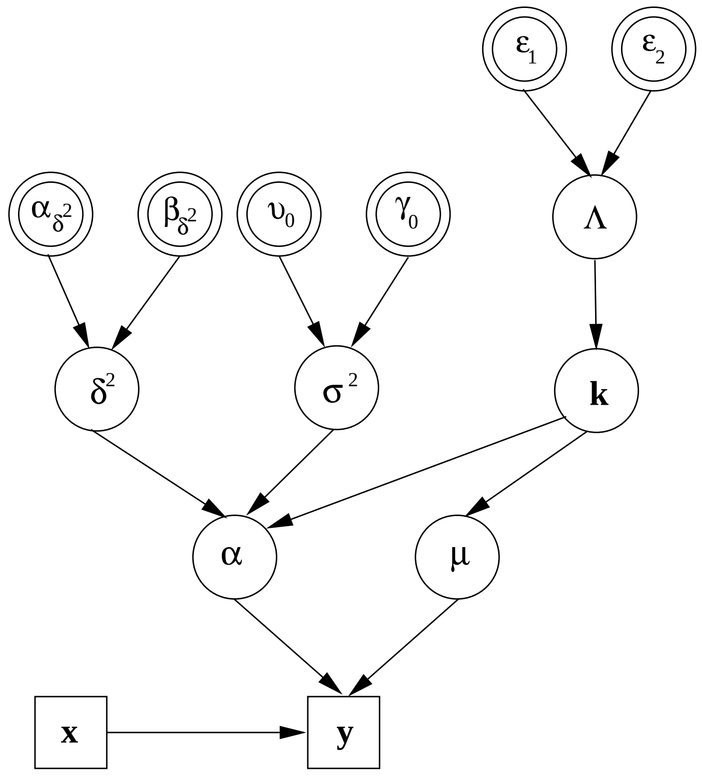

Figure 3: Directed acyclic graphical model for our prior.

computation. MCMC techniques were introduced in the mid 1950's in statistical physics and started appearing in the fields of applied statistics, signal processing and neural networks in the 1980's and 1990's (Besag, Green, Hidgon and Mengersen 1995, Holmes and Mallick 1998, Müller and Rios Insua 1998, Neal 1996, Rios Insua and Müller 1998, 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:

$$\widehat{p}(k=j|\mathbf{x},\mathbf{y}) = \frac{1}{P}\sum_{i=1}^{P}\mathbb{I}_{\{j\}}(k^{(i)}) \text{ and } \widehat{\mathbb{E}}(\boldsymbol{\theta}|k=j,\mathbf{x},\mathbf{y}) = \frac{\sum_{i=1}^{P}\boldsymbol{\theta}^{(i)}\mathbb{I}_{\{j\}}(k^{(i)})}{\sum_{i=1}^{P}\mathbb{I}_{\{j\}}(k^{(i)})} \tag{11}$$

In addition, we can obtain predictions, such as:

$$\widehat{\mathbb{E}}(\mathbf{y}_{N+1}|\mathbf{x}_{1:N+1},\mathbf{y}_{1:N}) = \frac{1}{P}\sum_{i=1}^{P}\mathbf{D}(\boldsymbol{\mu}_{1:k}^{(i)},\mathbf{x}_{N+1})\boldsymbol{\alpha}_{1:m}^{(i)} \tag{12}$$

As shown in the next subsection, we can integrate $\boldsymbol{\alpha}_{1:m}$ analytically. Consequently, we can reduce the variance of the predictions by employing the following Rao Blackwellised estimate

(Liu, Wong and Kong 1994):

$$\widehat{\mathbb{E}}(\mathbf{y}_{N+1}|\mathbf{x}_{1:N+1},\mathbf{y}_{1:N}) = \frac{1}{P}\sum_{i=1}^{P}\mathbf{D}(\boldsymbol{\mu}_{1:k}^{(i)},\mathbf{x}_{N+1})\mathbb{E}(\boldsymbol{\alpha}_{1:m}|k^{(i)},\boldsymbol{\mu}_{1:k}^{(i)},\sigma_k^{2(i)},\boldsymbol{\delta}^{2(i)},\mathbf{x},\mathbf{y})$$

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 $p(k,\boldsymbol{\mu}_{1:k},\Lambda,\boldsymbol{\delta}^2|\mathbf{x},\mathbf{y})$ up to a normalising constant. According to Bayes theorem:

$$\begin{aligned}
&p(k,\boldsymbol{\alpha}_{1:m},\boldsymbol{\mu}_{1:k},\boldsymbol{\sigma}^2,\Lambda,\boldsymbol{\delta}^2|\mathbf{x},\mathbf{y}) \propto p(\mathbf{y}|k,\boldsymbol{\alpha}_{1:m},\boldsymbol{\mu}_{1:k},\boldsymbol{\sigma}^2,\Lambda,\boldsymbol{\delta}^2,\mathbf{x})p(k,\boldsymbol{\alpha}_{1:m},\boldsymbol{\mu}_{1:k},\boldsymbol{\sigma}^2,\Lambda,\boldsymbol{\delta}^2)\\
&\propto \left[\prod_{i=1}^{c}(2\pi\boldsymbol{\sigma}_i^2)^{-N/2}\exp\Big(-\frac{1}{2\boldsymbol{\sigma}_i^2}\big(\mathbf{y}_{1:N,i}-\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x})\boldsymbol{\alpha}_{1:m,i}\big)'\big(\mathbf{y}_{1:N,i}-\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x})\boldsymbol{\alpha}_{1:m,i}\big)\Big)\right]\\
&\times\left[\prod_{i=1}^{c}|2\pi\boldsymbol{\sigma}_i^2\boldsymbol{\Sigma}_i|^{-1/2}\exp\Big(-\frac{1}{2\boldsymbol{\sigma}_i^2}\boldsymbol{\alpha}_{1:m,i}'\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\alpha}_{1:m,i}\Big)\right]\left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k,\boldsymbol{\mu}_{1:k})}{\Im^k}\right]\left[\frac{\Lambda^k/k!}{\sum_{j=0}^{k_{\max}}\Lambda^j/j!}\right]\\
&\times\left[\prod_{i=1}^{c}(\boldsymbol{\sigma}_i^2)^{-(v_0/2+1)}\exp\Big(-\frac{\gamma_0}{2\boldsymbol{\sigma}_i^2}\Big)\right]\left[\prod_{i=1}^{c}(\boldsymbol{\delta}_i^2)^{-(\alpha_{\delta^2}+1)}\exp\Big(-\frac{\beta_{\delta^2}}{\boldsymbol{\delta}_i^2}\Big)\right]\\
&\times\left[(\Lambda)^{(\varepsilon_1-1/2)}\exp\Big(-\varepsilon_2\Lambda\Big)\right]
\end{aligned}\tag{13}$$

We can then proceed to multiply the exponential terms of the likelihood and coefficients prior and complete squares to obtain:

$$\begin{aligned}
p(k,\boldsymbol{\alpha}_{1:m},\boldsymbol{\mu}_{1:k},\boldsymbol{\sigma}^2,\Lambda,\boldsymbol{\delta}^2|\mathbf{x},\mathbf{y}) \propto &\left[\prod_{i=1}^{c}(2\pi\boldsymbol{\sigma}_i^2)^{-N/2}\exp\Big(-\frac{1}{2\boldsymbol{\sigma}_i^2}\mathbf{y}_{1:N,i}'\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}\Big)\right]\\
\times&\left[\prod_{i=1}^{c}|2\pi\boldsymbol{\sigma}_i^2\boldsymbol{\Sigma}_i|^{-1/2}\exp\Big(-\frac{1}{2\boldsymbol{\sigma}_i^2}\big(\boldsymbol{\alpha}_{1:m,i}-\mathbf{h}_{i,k}\big)'\mathbf{M}_{i,k}^{-1}\big(\boldsymbol{\alpha}_{1:m,i}-\mathbf{h}_{i,k}\big)\Big)\right]\\
\times&\left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k,\boldsymbol{\mu}_{1:k})}{\Im^k}\right]\left[\frac{\Lambda^k/k!}{\sum_{j=0}^{k_{\max}}\Lambda^j/j!}\right]\left[\prod_{i=1}^{c}(\boldsymbol{\sigma}_i^2)^{-(v_0/2+1)}\exp\Big(-\frac{\gamma_0}{2\boldsymbol{\sigma}_i^2}\Big)\right]\\
\times&\left[\prod_{i=1}^{c}(\boldsymbol{\delta}_i^2)^{-(\alpha_{\delta^2}+1)}\exp\Big(-\frac{\beta_{\delta^2}}{\boldsymbol{\delta}_i^2}\Big)\right]\left[(\Lambda)^{(\varepsilon_1-1/2)}\exp\Big(-\varepsilon_2\Lambda\Big)\right]
\end{aligned}$$

where:

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

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

$$p(k,\boldsymbol{\mu}_{1:k},\Lambda,\boldsymbol{\delta}^2|\mathbf{x},\mathbf{y}) \propto \left[\prod_{i=1}^{c}\left(1+\boldsymbol{\delta}_i^2\right)^{-m/2}\left(\frac{\gamma_0+\mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{2}\right)^{\left(-\frac{N+v_0}{2}\right)}\right]$$
$$\times\left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k,\boldsymbol{\mu}_k)}{\Im^k}\right]\left[\frac{\Lambda^k/k!}{\sum_{j=0}^{k_{\max}}\Lambda^j/j!}\right]\left[\prod_{i=1}^{c}(\boldsymbol{\delta}_i^2)^{-(\alpha_{\delta^2}+1)}\exp\left(-\frac{\beta_{\delta^2}}{\boldsymbol{\delta}_i^2}\right)\right]$$
$$\times\left[(\Lambda)^{(\varepsilon_1-1/2)}\exp\left(-\varepsilon_2\Lambda\right)\right] \tag{14}$$

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

4 Bayesian Computation

For the sake of clarity, we firstly assume that k is given. After dealing with this fixed dimension scenario, we move on to 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 (Besag et al. 1995, Gilks, Richardson and Spiegelhalter 1996, Tierney 1994):

Fixed dimension MCMC algorithm

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

     End For.
   - Sample the nuisance parameters $(\boldsymbol{\alpha}_{1:m}^{(i)},\boldsymbol{\sigma}^{2(i)})$ using equations (18) and (19).
   - Sample the hyper-parameters $(\Lambda^{(i)},\boldsymbol{\delta}^{2(i)})$ using equations (20) and (21) .
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 subsections. In order to simplify the notation, we drop the superscript $\cdot^{(i)}$ from all variables at iteration i.

4.1.1 Updating the RBF centres

Sampling the RBF centres 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 $q(\mathbf{z}^\star|\,\mathbf{z})$, involves sampling a candidate value $\mathbf{z}^\star$ given the current value $\mathbf{z}$ according to $q(\mathbf{z}^\star|\,\mathbf{z})$. The Markov chain then moves towards $\mathbf{z}^\star$ with probability $\mathcal{A}(\mathbf{z},\mathbf{z}^\star) \triangleq \min\{1, (\pi(\mathbf{z})q(\mathbf{z}^\star|\mathbf{z}))^{-1}\pi(\mathbf{z}^\star)\; q(\mathbf{z}|\mathbf{z}^\star)\}$, 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 (14), the target distribution is the full conditional distribution of a basis centre:

$$p(\boldsymbol{\mu}_{j,1:d}|\mathbf{x},\mathbf{y},\boldsymbol{\mu}_{-j,1:d}) \propto \left[\prod_{i=1}^{c}\left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{2}\right)^{(-\frac{N+v_0}{2})}\right]\mathbb{I}_{\boldsymbol{\Omega}}(k,\boldsymbol{\mu}_{1:k}) \tag{15}$$

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

With probability $0 < \varpi < 1$, the proposal $q_1(\boldsymbol{\mu}^\star_{j,1:d}|\boldsymbol{\mu}_{j,1:d})$ corresponds to randomly sampling a basis centre 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 is dense are reached quickly. Subsequently, with probability $1-\varpi$, we perform an MH step with proposal distribution $q_2(\boldsymbol{\mu}^\star_{j,1:d}|\boldsymbol{\mu}_{j,1:d})$:

$$\boldsymbol{\mu}^\star_{j,1:d}|\boldsymbol{\mu}_{j,1:d} \sim \mathcal{N}(\boldsymbol{\mu}_{j,1:d}, \sigma^2_{RW}\mathbf{I}_d) \tag{16}$$

This proposal distribution yields a candidate $\boldsymbol{\mu}^\star_{j,1:d}$ which is a perturbation of the current centre. 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:

$$\mathcal{A}(\boldsymbol{\mu}_{j,1:d},\boldsymbol{\mu}^\star_{j,1:d}) = \min\left\{1, \left[\prod_{i=1}^{c}\left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}^\star_{i,k}\mathbf{y}_{1:N,i}}\right)^{(\frac{N+v_0}{2})}\right]\mathbb{I}_{\boldsymbol{\Omega}}(k,\boldsymbol{\mu}^\star_{1:k})\right\} \tag{17}$$

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 $\{\boldsymbol{\mu}_{1,1:d},\boldsymbol{\mu}_{2,1:d},\cdots,$ $\boldsymbol{\mu}_{j-1,1:d},\boldsymbol{\mu}^\star_{j,1:d},\boldsymbol{\mu}_{j+1,1:d},\cdots,\boldsymbol{\mu}_{k,1:d}\}$. 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 $p(k,\boldsymbol{\mu}_{1:k},\Lambda,\boldsymbol{\delta}^2|\mathbf{x},\mathbf{y})$ from the full posterior distribution $p(k,\boldsymbol{\alpha}_{1:m},\boldsymbol{\mu}_{1:k},\boldsymbol{\sigma}^2,\Lambda,\boldsymbol{\delta}^2|\mathbf{x},\mathbf{y})$ 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:

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

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

$$\boldsymbol{\sigma}_i^2 | (k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\delta}^2, \mathbf{x}, \mathbf{y}) \sim \mathcal{IG}\left(\frac{\upsilon_0 + N}{2}, \frac{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k} \mathbf{y}_{1:N,i}}{2} \right) \tag{18}$$

$$\boldsymbol{\alpha}_{1:m,i} | (k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}^2, \boldsymbol{\delta}^2, \mathbf{x}, \mathbf{y}) \sim \mathcal{N}(\mathbf{h}_{i,k}, \boldsymbol{\sigma}_i^2 \mathbf{M}_{i,k}) \tag{19}$$

4.1.3 Sampling the hyper-parameters

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

$$\boldsymbol{\delta}_i^2 | (k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}_i^2, \mathbf{x}, \mathbf{y}) \sim \mathcal{IG}\left(\alpha_{\delta^2} + \frac{m}{2}, \beta_{\delta^2} + \frac{1}{2\boldsymbol{\sigma}_i^2} \boldsymbol{\alpha}'_{1:m,i} \mathbf{D}'(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \mathbf{D}(\boldsymbol{\mu}_{1:k}, \mathbf{x}) \boldsymbol{\alpha}_{1:m,i} \right) \tag{20}$$

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:

$$q(\Lambda^\star) \propto \Lambda^{\star(1/2 + \varepsilon_1 + k)} \exp\left(-(1 + \varepsilon_2) \Lambda^\star \right) \tag{21}$$

and subsequently perform an MH step with the full conditional distribution $p(\Lambda | k, \boldsymbol{\mu}_{1:k}, \boldsymbol{\delta}^2, \mathbf{x}, \mathbf{y})$ 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 $p(k, \boldsymbol{\theta}, \boldsymbol{\psi} | \mathbf{x}, \mathbf{y})$ 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 $k = 0, ..., k_{\max}$. However, this approach suffers from severe drawbacks. Firstly, it is computationally very expensive since $k_{\max}$ can be large. Secondly, 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 $p(k | \mathbf{x}, \mathbf{y})$. Another solution would be to construct an MCMC sampler that would be able to sample directly from the joint distribution on $\boldsymbol{\Theta} \times \boldsymbol{\Psi} = \left(\cup_{k=0}^{k_{\max}} \{k\} \times \boldsymbol{\Theta}_k \right) \times \boldsymbol{\Psi}$. Standard MCMC methods are not able to "jump" between subspaces $\boldsymbol{\Theta}_k$ of different dimensions. However, recently, Green 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 (Green 1995). This is a general state-space MH algorithm (see (Andrieu, Djurić and Doucet 1999) 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, Richardson and Green 1997) for details. For our problem, the following moves have been selected:

1. Birth of a new basis, *i.e.* 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, *i.e.* removing a basis function chosen randomly.
3. Merge a randomly chosen basis function and its closest neighbour into a single basis function.
4. Split a randomly chosen basis function into two neighbour 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 centres.

These moves are defined by heuristic considerations, the only condition to be fulfilled being to maintain the correct invariant distribution. A particular choice will only have influence 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 neighbourhood. That is, when amplitudes of many basis functions, in a close neighbourhood, add to the amplitude that would be obtained by using less 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.

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:

$$b_k \triangleq c^\star \min\left\{1, \frac{p(k+1)}{p(k)}\right\}, \; d_{k+1} \triangleq c^\star \min\left\{1, \frac{p(k)}{p(k+1)}\right\} \tag{22}$$

where $p(k)$ is the prior probability of model $\mathcal{M}_k$ and $c^\star$ is a parameter which tunes the proportion of dimension/update moves. As pointed out in (Green 1995) this choice ensures that $b_k p(k) [d_{k+1} p(k+1)]^{-1} = 1$, 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. Initialisation: set $(k^{(0)}, \boldsymbol{\theta}^{(0)}, \boldsymbol{\psi}^{(0)}) \in \boldsymbol{\Theta} \times \boldsymbol{\Psi}$.
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 centres (See Section 4.2.3).

     End If.
   - Sample the nuisance parameters $\left(\boldsymbol{\sigma}_k^{2(i)}, \boldsymbol{\alpha}_k^{(i)}\right)$ using equations (18) and (19).
   - Simulate the hyper-parameters $(\Lambda^{(i)}, \boldsymbol{\delta}^{2(i)})$ using equations (20) and (21) .
3. $i \leftarrow i + 1$ and go to 2. ∎

We expand on these different moves in the following subsections. 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:

Birth move

- Propose a new RBF centre at random from the interval $[\min(\mathbf{x}_{1:N,i})-\iota\Xi_i, \max(\mathbf{x}_{1:N,i})+\iota\Xi_i]$ for $i = 1,\ldots,d$.
- Evaluate $\mathcal{A}_{birth}$, see equation(26), and sample $u \sim \mathcal{U}_{[0,1]}$.
- If $u \leq \mathcal{A}_{birth}$ then the state of the Markov chain becomes $\left(k+1, \boldsymbol{\mu}_{1:k+1}\right)$, else it remains equal to $(k, \boldsymbol{\mu}_{1:k})$.

■

Now, assume that the current state of the Markov chain is in $\{k\} \times \boldsymbol{\Theta}_k \times \boldsymbol{\Psi}$, then:

Death move

- Choose the basis centre, to be deleted, at random among the k existing basis.
- Evaluate $\mathcal{A}_{death}$, see equation (26), and sample $u \sim \mathcal{U}_{[0,1]}$.
- If $u \leq \mathcal{A}_{death}$ then the state of the Markov chain becomes $\left(k-1, \boldsymbol{\mu}_{1:k-1}\right)$, else it remains equal to $(k, \boldsymbol{\mu}_{1:k})$.

■

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

$$r_{birth} \triangleq (posterior\ distributions\ ratio) \times (proposal\ ratio) \times (Jacobian) \tag{23}$$

That is:

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

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

$$r_{birth} = \left[\prod_{i=1}^{c} \frac{1}{(1+\boldsymbol{\delta}_i^2)^{1/2}} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k+1}\mathbf{y}_{1:N,i}}\right)^{\left(\frac{N+v_0}{2}\right)}\right] \frac{1}{(k+1)} \tag{24}$$

Similarly, for the death move:

$$r_{death} = \frac{p(k-1, \boldsymbol{\mu}_{1:k-1}, \Lambda, \boldsymbol{\delta}^2|\mathbf{x}, \mathbf{y})}{p(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2|\mathbf{x}, \mathbf{y})} \times \frac{b_{k-1}/\Im}{d_k/k} \times \left|\frac{\partial(\boldsymbol{\mu}_{1:k-1}, \boldsymbol{\mu}^\star)}{\partial(\boldsymbol{\mu}_{1:k})}\right|$$

and consequently:

$$r_{death} = \left[\prod_{i=1}^{c} (1+\boldsymbol{\delta}_i^2)^{1/2} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k-1}\mathbf{y}_{1:N,i}}\right)^{\left(\frac{N+v_0}{2}\right)}\right] k \tag{25}$$

The acceptance probabilities corresponding to the described moves are:

$$\mathcal{A}_{birth} = \min\{1, r_{birth}\}, \ \mathcal{A}_{death} = \min\{1, r_{death}\} \tag{26}$$

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 neighbour ($\boldsymbol{\mu}_2$) into a single basis function $\boldsymbol{\mu}$, whose new location is:

$$\boldsymbol{\mu} = \frac{\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2}{2} \tag{27}$$

The corresponding split move that guarantees reversibility is:

$$\begin{aligned} \boldsymbol{\mu}_1 &= \boldsymbol{\mu} - u_{ms}\varsigma^\star \\ \boldsymbol{\mu}_2 &= \boldsymbol{\mu} + u_{ms}\varsigma^\star \end{aligned} \tag{28}$$

where $\varsigma^\star$ is a simulation parameter and $u_{ms} \sim \mathcal{U}_{[0,1]}$. Note that to ensure reversibility, we only perform the merge move 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:

Split move

- Randomly choose an existing RBF centre.
- Substitute it for two neighbour basis functions, whose centres are obtained using equation (28). The new centres must be bound to lie in the space $\boldsymbol{\Omega}_k$ 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.
- Evaluate $\mathcal{A}_{split}$, see equation(31), and sample $u \sim \mathcal{U}_{[0,1]}$.
- If $u \leq \mathcal{A}_{split}$ then the state of the Markov chain becomes $(k+1, \boldsymbol{\mu}_{1:k+1})$, else it remains equal to $(k, \boldsymbol{\mu}_{1:k})$.

Now, assume that the current state of the Markov chain is in $\{k\} \times \boldsymbol{\Theta}_k \times \boldsymbol{\Psi}$, then:

Merge move

- Choose a basis centre at random among the k existing basis. Then find the closest basis function to it applying some distance metric, *e.g.* Euclidean.
- If $\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\| < 2\varsigma^\star$, substitute the two basis functions for a single basis function in accordance with equation (27).
- Evaluate $\mathcal{A}_{merge}$, see equation (31)), and sample $u \sim \mathcal{U}_{[0,1]}$.
- If $u \leq \mathcal{A}_{merge}$ then the state of the Markov chain becomes $(k-1, \boldsymbol{\mu}_{1:k-1})$, else it remains equal to $(k, \boldsymbol{\mu}_{1:k})$.

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:

$$r_{split} = \left[\prod_{i=1}^{c} \frac{1}{(1+\boldsymbol{\delta}_i^2)^{1/2}} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k} \mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k+1} \mathbf{y}_{1:N,i}} \right)^{\left(\frac{N+v_0}{2}\right)} \right] \frac{k\varsigma^\star}{\Im(k+1)} \tag{29}$$

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:

$$r_{merge} = \left[\prod_{i=1}^{c} (1+\boldsymbol{\delta}_i^2)^{1/2} \left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k} \mathbf{y}_{1:N,i}}{\gamma_0 + \mathbf{y}'_{1:N,i} \mathbf{P}_{i,k-1} \mathbf{y}_{1:N,i}} \right)^{\left(\frac{N+v_0}{2}\right)} \right] \frac{k\Im}{\varsigma^\star(k-1)} \tag{30}$$

The acceptance probabilities for the split and merge moves are:

$$\mathcal{A}_{split} = \min\left\{1, r_{split}\right\}, \; \mathcal{A}_{merge} = \min\left\{1, r_{merge}\right\} \tag{31}$$

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 optimisation 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 modelling constraints and specifications.

5 Reversible Jump Simulated Annealing

In this section, we show that traditional model selection criteria within a penalised likelihood framework, such as AIC, BIC and MDL (Akaike 1974, Schwarz 1985, Rissanen 1987), can be shown to correspond to particular hyper-parameter choices in our hierarchical Bayesian formulation. That is, we can calibrate the prior choices so that the problem of model selection within the penalised likelihood context can be mapped exactly to a problem of model selection via posterior probabilities. This technique has been previously applied in the areas of variable selection (George and Foster 1997) and the detection of harmonics in noisy signals (Andrieu and Doucet 1998).

After resolving the calibration problem, we perform maximum likelihood estimation, with the aforementioned model selection criteria, by maximising 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. 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 Penalised likelihood model selection

Traditionally, penalised 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 minimises the sum of the the log-likelihood and a penalty term that depends on the model dimension (Djurić 1998, Gelfand and Dey 1997). In mathematical terms, this estimate is given by:

$$\mathcal{M}_s = \underset{\mathcal{M}_k : k \in \{0, \ldots, k_{\max}\}}{\arg\min} \left\{ -\log(p(\mathbf{y}|k, \widehat{\boldsymbol{\theta}}, \mathbf{x})) + \mathcal{P} \right\} \tag{32}$$

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:

$$\mathcal{P}_{\text{AIC}} = \xi \qquad \text{and} \qquad \mathcal{P}_{\text{BIC}} = \mathcal{P}_{\text{MDL}} = \frac{\xi}{2}\log(N) \tag{33}$$

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:

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

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 (32) as follows:

$$\begin{aligned}
\mathcal{M}_s &= \underset{\mathcal{M}_k:k\in\{0,\dots,k_{\max}\}}{\arg\min}\Bigg\{-\log\Bigg[\prod_{i=1}^{c}(2\pi\widehat{\boldsymbol{\sigma}}_i^2)^{-N/2}\exp\Bigg(-\frac{1}{2\widehat{\boldsymbol{\sigma}}_i^2}\big(\mathbf{y}_{1:N,i}-\mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k},\mathbf{x})\widehat{\boldsymbol{\alpha}}_{1:m,i}\big)' \\
&\qquad\qquad \big(\mathbf{y}_{1:N,i}-\mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k},\mathbf{x})\widehat{\boldsymbol{\alpha}}_{1:m,i}\big)\Bigg)\Bigg]+\mathcal{P}\Bigg\} \\
&= \underset{\mathcal{M}_k:k\in\{0,\dots,k_{\max}\}}{\arg\min}\Bigg\{\frac{N}{2}\sum_{i=1}^{c}\log\big(2\pi\widehat{\boldsymbol{\sigma}}_i^2\big)+\sum_{i=1}^{c}\frac{1}{2\widehat{\boldsymbol{\sigma}}_i^2}\big(\mathbf{y}_{1:N,i}-\mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k},\mathbf{x})\widehat{\boldsymbol{\alpha}}_{1:m,i}\big)' \\
&\qquad\qquad \big(\mathbf{y}_{1:N,i}-\mathbf{D}(\widehat{\boldsymbol{\mu}}_{1:k},\mathbf{x})\widehat{\boldsymbol{\alpha}}_{1:m,i}\big)+\mathcal{P}\Bigg\} \\
&= \underset{\mathcal{M}_k:k\in\{0,\dots,k_{\max}\}}{\arg\min}\Bigg\{\frac{N}{2}\sum_{i=1}^{c}\log\big(\widehat{\boldsymbol{\sigma}}_i^2\big)+\mathcal{P}\Bigg\} \\
&= \underset{\mathcal{M}_k:k\in\{0,\dots,k_{\max}\}}{\arg\min}\Bigg\{\frac{N}{2}\sum_{i=1}^{c}\log\big(\mathbf{y}'_{1:N,i}\mathbf{P}^*_{i,k}\mathbf{y}_{1:N,i}\big)+\mathcal{P}\Bigg\} \\
&= \underset{\mathcal{M}_k:k\in\{0,\dots,k_{\max}\}}{\arg\max}\Bigg\{\Bigg[\prod_{i=1}^{c}(\mathbf{y}'_{1:N,i}\mathbf{P}^*_{i,k}\mathbf{y}_{1:N,i})^{-N/2}\Bigg]\exp(-\mathcal{P})\Bigg\} \qquad (34)
\end{aligned}$$

In the following subsection, we show that calibrating the priors in our hierarchical Bayes model will lead to the expression given by equation (34).

5.2 Calibration

It is useful and elucidating to impose some restrictions on the Bayesian hierarchical prior (equation (14)) to obtain the AIC and MDL criteria. We begin by assuming that the hyper-parameter $\boldsymbol{\delta}$ 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:

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

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

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

where:

$$\begin{aligned}
\mathbf{M}_{i,k}^{-1} &= \big(1+\overline{\boldsymbol{\delta}}_i^{-2}\big)\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x}) \\
\mathbf{h}_{i,k} &= \mathbf{M}_{i,k}\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{y}_{1:N,i} \\
\mathbf{P}_{i,k} &= \mathbf{I}_N-\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{M}_{i,k}\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})
\end{aligned}$$

Finally, we can select $\overline{\boldsymbol{\delta}}_i^2$ and $p(k)$ such that:

$$\Bigg[\prod_{i=1}^{c}\big(1+\overline{\boldsymbol{\delta}}_i^2\big)^{-k/2}\Bigg]\Bigg[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k,\boldsymbol{\mu}_k)}{\Im^k}\Bigg]p(k) = \exp(-\mathcal{P}) \propto \exp(-\mathcal{C}k) \qquad (35)$$

thereby ensuring that the expression for the calibrated posterior distribution $p(k, \boldsymbol{\mu}_{1:k}|\mathbf{x}, \mathbf{y})$ corresponds to the term that needs to be maximised in the penalised likelihood framework (equation (34)). Note that for the purposes of optimisation, 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 $\boldsymbol{\Omega}$ and choosing the prior:

$$p(k) = \frac{\Lambda^k}{\sum_{j=0}^{k_{\max}} \Lambda^j}$$

with the following fixed value for Λ:

$$\overline{\Lambda} = \left[\prod_{i=1}^{c} \left(1 + \overline{\boldsymbol{\delta}}_i^2\right)^{\frac{1}{2}}\right] \Im \exp\left(-\mathcal{C}\right) \tag{36}$$

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' prior for $\boldsymbol{\sigma}^2$ and setting Λ as in equation (36), we can obtain the expression that needs to be maximised in the classical penalised likelihood formulation with AIC, MDL and BIC model selection criteria. Consequently, we can interpret the penalised likelihood framework as a problem of maximising the joint posterior posterior distribution $p(k, \boldsymbol{\mu}_{1:k}|\mathbf{x}, \mathbf{y})$. Effectively, we can obtain this MAP estimate as follows:

$$\begin{aligned}
(k, \boldsymbol{\mu}_{1:k})_{\mathrm{MAP}} &= \operatorname*{arg\,max}_{k, \mu_{1:k} \in \boldsymbol{\Omega}} p(k, \boldsymbol{\mu}_{1:k}|\mathbf{x}, \mathbf{y}) \\
&= \operatorname*{arg\,max}_{k, \mu_{1:k} \in \boldsymbol{\Omega}} \left\{ \left[\prod_{i=1}^{c} (\mathbf{y}'_{1:N,i}\mathbf{P}^*_{i,k}\mathbf{y}_{1:N,i})^{-N/2}\right] \exp(-\mathcal{P}) \right\}
\end{aligned} \tag{37}$$

The sufficient conditions that need to be satisfied so that the distribution $p(k, \boldsymbol{\mu}_{1:k}|\mathbf{x}, \mathbf{y})$ is proper are not overly restrictive. Firstly, $\boldsymbol{\Omega}$ has to be a compact set, which is not a problem in our setting. Secondly, $\mathbf{y}'_{1:N,i}\mathbf{P}^*_{i,k}\mathbf{y}_{1:N,i}$ 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 is rather unlikely to occur in our approximation framework, yet we can safeguard against it happening by choosing a very large value for $\overline{\boldsymbol{\delta}}$ in the simulations. This is a standard least squares trick known as ridge regression (Marquardt and Snee 1975, Wetherill 1986).

5.3 Reversible jump simulated annealing

From an MCMC perspective, we can solve the stochastic optimisation problem posed in the previous subsection by adopting a simulated annealing strategy (Geman and Geman 1984, Van Laarhoven and Arts 1987). The simulated annealing method involves simulating a non-homogeneous 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 $q(\mathbf{z}^{\star}|\,\mathbf{z})$ involves sampling a candidate value $\mathbf{z}^{\star}$ given the current value $\mathbf{z}$ according to $q(\mathbf{z}^{\star}|\,\mathbf{z})$. The Markov chain moves towards $\mathbf{z}^{\star}$ with probability $\mathcal{A}_{\mathrm{SA}}(\mathbf{z},\mathbf{z}^{\star}) = \min\left\{1, \left(\pi^{1/T_i}(\mathbf{z})\, q(\mathbf{z}^{\star}|\,\mathbf{z})\right)^{-1} \pi^{1/T_i}(\mathbf{z}^{\star})\, q(\mathbf{z}|\,\mathbf{z}^{\star})\right\}$, 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:

$$\mathcal{A}_{\mathrm{RJSA}} = \min\left\{1, \frac{\pi^{(1/T_i-1)}(\mathbf{z}^{\star})}{\pi^{(1/T_i-1)}(\mathbf{z})}\right\} \tag{38}$$

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. Initialisation: set $\left(k^{(0)}, \theta^{(0)}\right) \in \boldsymbol{\Theta}$.
2. Iteration i.
   - Sample $u \sim \mathcal{U}_{[0,1]}$ and set the temperature with a cooling schedule.
   - If $(u \leq b_{k^{(i)}})$
     - then "birth" move (See Section 5.5).
     - else if $(u \leq b_{k^{(i)}} + d_{k^{(i)}})$ then "death" move (See Section 5.5).
     - else if $(u \leq b_{k^{(i)}} + d_{k^{(i)}} + s_{k^{(i)}})$ then "split" move (See Section 5.6).
     - else if $(u \leq b_{k^{(i)}} + d_{k^{(i)}} + s_{k^{(i)}} + m_{k^{(i)}})$ then "merge" move (See Section 5.6).
     - else update the RBF centres (See Section 5.4).

     End If.
   - Perform an MH step with the annealed acceptance ratio (equation (38)).
3. $i \leftarrow i + 1$ and go to 2.
4. Compute the coefficients $\boldsymbol{\alpha}_{1:m}$ by least squares (optimal in this case):

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

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

5.4 Update move

We sample the radial basis centres in the same way as explained in Section 4.1.1. However, the target distribution is given by:

$$p(\boldsymbol{\mu}_{j,1:d}|\mathbf{x},\mathbf{y},\boldsymbol{\mu}_{-j,1:d}) \propto \left[\prod_{i=1}^{c}(\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i})^{(-\frac{N}{2})}\right]\exp(-\mathcal{P}) \tag{39}$$

and, consequently, the acceptance probability is:

$$\mathcal{A}_{\mathrm{RJSA}}(\boldsymbol{\mu}_{j,1:d},\boldsymbol{\mu}^{\star}_{j,1:d}) = \min\left\{1,\left[\prod_{i=1}^{c}\left(\frac{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i}\mathbf{P}^{\star}_{i,k}\mathbf{y}_{1:N,i}}\right)^{(\frac{N}{2})}\right]\right\} \tag{40}$$

where $\mathbf{P}^{\star}_{i,k}$ is similar to $\mathbf{P}^{*}_{i,k}$ with $\boldsymbol{\mu}_{1:k,1:d}$ replaced by $\{\boldsymbol{\mu}_{1,1:d},\boldsymbol{\mu}_{2,1:d},\ldots,\boldsymbol{\mu}_{j-1,1:d},\boldsymbol{\mu}^{\star}_{j,1:d},\boldsymbol{\mu}_{j+1,1:d},\ldots,\boldsymbol{\mu}_{k,1:d}\}$.

5.5 Birth and death moves

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:

$$r_{birth} = \left[\prod_{i=1}^{c}\left(\frac{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k+1}\mathbf{y}_{1:N,i}}\right)^{(\frac{N}{2})}\right]\frac{\Im\exp(-\mathcal{C})}{k+1} \tag{41}$$

Similarly,

$$r_{death} = \left[\prod_{i=1}^{c}\left(\frac{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k-1}\mathbf{y}_{1:N,i}}\right)^{(\frac{N}{2})}\right]\frac{k\exp(\mathcal{C})}{\Im} \tag{42}$$

Hence, the acceptance probabilities corresponding to the described moves are:

$$\mathcal{A}_{birth} = \min\{1, r_{birth}\}, \ \mathcal{A}_{death} = \min\{1, r_{death}\} \tag{43}$$

5.6 Split and merge moves

Again 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:

$$r_{split} = \left[\prod_{i=1}^{c}\left(\frac{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k+1}\mathbf{y}_{1:N,i}}\right)^{(\frac{N}{2})}\right]\frac{k\varsigma^{\star}\exp(-\mathcal{C})}{k+1} \tag{44}$$

and

$$r_{merge} = \left[\prod_{i=1}^{c}\left(\frac{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k}\mathbf{y}_{1:N,i}}{\mathbf{y}'_{1:N,i}\mathbf{P}^{*}_{i,k-1}\mathbf{y}_{1:N,i}}\right)^{(\frac{N}{2})}\right]\frac{k\exp(\mathcal{C})}{\varsigma^{\star}(k-1)} \tag{45}$$

The acceptance probabilities for these moves are:

$$\mathcal{A}_{split} = \min\{1, r_{split}\}, \ \mathcal{A}_{merge} = \min\{1, r_{merge}\} \tag{46}$$

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 $\left(k^{(i)}, \boldsymbol{\mu}_{1:k}^{(i)}, \Lambda^{(i)}, \boldsymbol{\delta}^{2(i)}\right)_{i\in\mathbb{N}}$ is ergodic. We prove here a stronger result by showing that $\left(k^{(i)}, \boldsymbol{\mu}_{1:k}^{(i)}, \Lambda^{(i)}, \boldsymbol{\delta}^{2(i)}\right)_{i\in\mathbb{N}}$ converges to the required posterior distribution at a geometric rate.

For the homogeneous kernel, we have the following result:

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

$$\left\| p^{(i)}\left(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2\right) - p\left(k, \boldsymbol{\mu}_{1:k}, \Lambda, \boldsymbol{\delta}^2 \middle| \mathbf{x}, \mathbf{y}\right)\right\|_{TV} \leq C\left(k^{(0)}, \boldsymbol{\mu}_{1:k}^{(0)}, \Lambda^{(0)}, \boldsymbol{\delta}^{2(0)}\right) \rho^{\lfloor i/k_{\max}\rfloor} \quad (47)$$

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

Proof. See Appendix B ■

Corollary 1 *If for each iteration i one simulates the nuisance parameters $(\boldsymbol{\alpha}_{1:m}, \boldsymbol{\sigma}_k^2)$ then the distribution of the series $(k^{(i)}, \boldsymbol{\alpha}_{1:m}^{(i)}, \boldsymbol{\mu}_{1:k}^{(i)}, \boldsymbol{\sigma}_k^{2(i)}, \Lambda^{(i)}, \boldsymbol{\delta}^{2(i)})_{i\in\mathbb{N}}$ converges geometrically towards $p(k, \boldsymbol{\alpha}_{1:m}, \boldsymbol{\mu}_{1:k}, \boldsymbol{\sigma}_k^2, \Lambda, \boldsymbol{\delta}^2 | \mathbf{x}, \mathbf{y})$ 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 $p(k, \boldsymbol{\theta}, \psi | \mathbf{x}, \mathbf{y})$.

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 and 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:*

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

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, Breyer and Doucet 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[4]. Firstly, 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 $\boldsymbol{\alpha}_{1:m}$:

$$p(\boldsymbol{\alpha}_{1:m}|k,\boldsymbol{\mu}_{1:k},\boldsymbol{\sigma}^2,\Lambda,\boldsymbol{\delta}^2) = \left[\prod_{i=1}^{c} |2\pi\boldsymbol{\sigma}_i^2\boldsymbol{\delta}_i^2\mathbf{I}_m|^{-1/2}\exp\left(-\frac{1}{2\boldsymbol{\sigma}_i^2\boldsymbol{\delta}_i^2}\boldsymbol{\alpha}'_{1:m,i}\boldsymbol{\alpha}_{1:m,i}\right)\right] \tag{48}$$

In doing so, the marginal posterior distribution becomes:

$$\begin{aligned} p(k,\boldsymbol{\mu}_{1:k},\Lambda,\boldsymbol{\delta}^2|\mathbf{x},\mathbf{y}) \propto & \left[\prod_{i=1}^{c}(\boldsymbol{\delta}_i^2)^{-m/2}|\mathbf{M}_{i,k}|^{1/2}\left(\frac{\gamma_0 + \mathbf{y}'_{1:N,i}\mathbf{P}_{i,k}\mathbf{y}_{1:N,i}}{2}\right)^{\left(-\frac{N+v_0}{2}\right)}\right] \\ & \times \left[\frac{\mathbb{I}_{\boldsymbol{\Omega}}(k,\boldsymbol{\mu}_k)}{\Im^k}\right]\left[\frac{\Lambda^k/k!}{\sum_{j=0}^{k_{\max}}\Lambda^j/j!}\right]\left[\prod_{i=1}^{c}(\boldsymbol{\delta}_i^2)^{-(\alpha_{\delta^2}+1)}\exp\left(-\frac{\beta_{\boldsymbol{\delta}^2}}{\boldsymbol{\delta}_i^2}\right)\right] \\ & \times \left[(\Lambda)^{(\varepsilon_1-1/2)}\exp\left(-\varepsilon_2\Lambda\right)\right] \end{aligned} \tag{49}$$

where:

$$\begin{aligned} \mathbf{M}_{i,k}^{-1} &= \mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x}) + \boldsymbol{\delta}_i^{-2}\mathbf{I}_m \\ \mathbf{h}_{i,k} &= \mathbf{M}_{i,k}\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{y}_{1:N,i} \\ \mathbf{P}_{i,k} &= \mathbf{I}_N - \mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{M}_{i,k}\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x}) \end{aligned}$$

We 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 $\hbar$. The results presented henceforth were obtained using this approach.

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, Fisher 1929, Hannan 1961). 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 penalised likelihood model selection criteria (AIC, MDL) when the amount of noise in the signal varies.

[4] The software is available at `http://www-svr.eng.cam.ac.uk/~jfgf`.

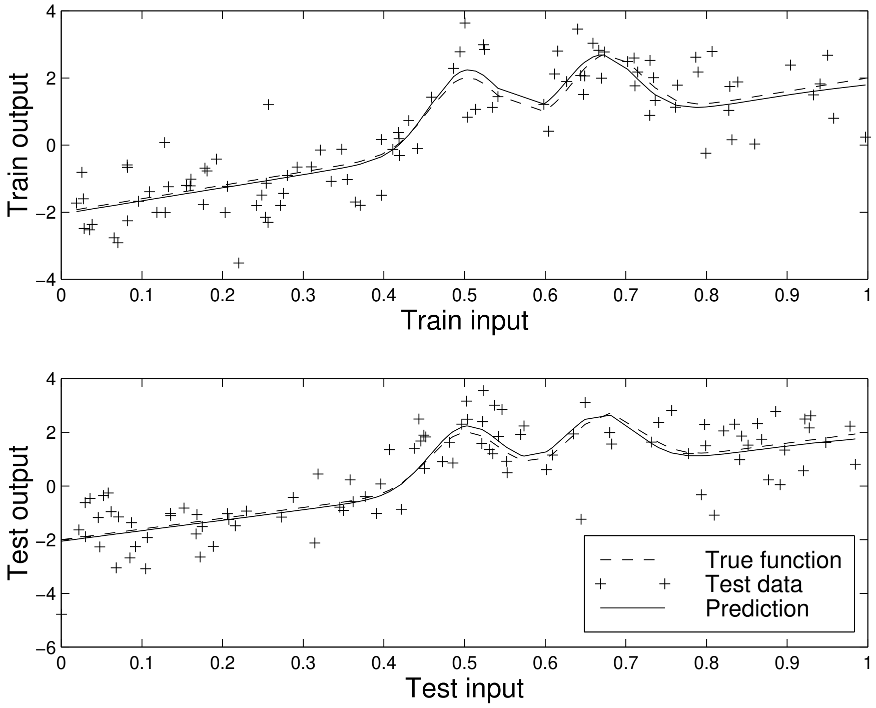

Figure 4: Performance of the reversible jump MCMC algorithm on the signal detection problem. Despite the large noise variance, the estimates of the true function and noise process are very accurate; thereby leading to good generalisation (no over-fitting).

The data was 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 was then rescaled to make the input data lie in the interval $[0, 1]$. We used the full Bayesian and 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 $T_i > 0 \quad \text{for } i = 1, 2, 3, \ldots$. 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 maximised, while for the Bayesian approach, the main mode corresponds to the MAP of the model order probabilities $\widehat{p}(k|\mathbf{x}, \mathbf{y})$, computed as suggested in Section 3.2.

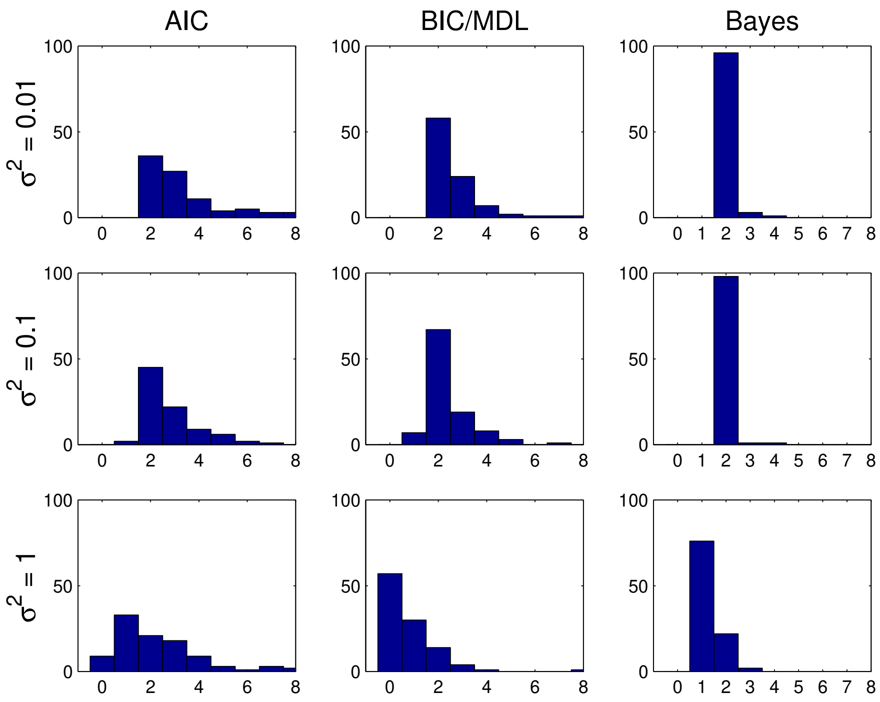

Figure 5: Histograms of the main mode $\widehat{p}(k|\mathbf{x},\mathbf{y})$ for 100 trials of each noise level in Experiment 1. The Bayes solution provides a better estimate of the true number of basis (2) than the MDL/BIC and AIC criteria.

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 t-th prediction for the i-th trial and $\overline{y}_i$ is the estimated mean of y_i. The normalisation 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 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

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

Table 1: Fraction of unexplained variance for different values of the noise variance, averaged over 100 test sets.

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 realisations 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.

7.2 Experiment 2: Robot arm data

This data set is often used as a benchmark to compare neural network algorithms[5]. 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:

$$\begin{aligned} y_1 &= 2.0\cos(x_1) + 1.3\cos(x_1 + x_2) + \epsilon_1 \\ y_2 &= 2.0\sin(x_1) + 1.3\sin(x_1 + x_2) + \epsilon_2 \end{aligned} \tag{50}$$

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.

Firstly, 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 3D plots of the training data and the contours of the training and test data. The contour plots

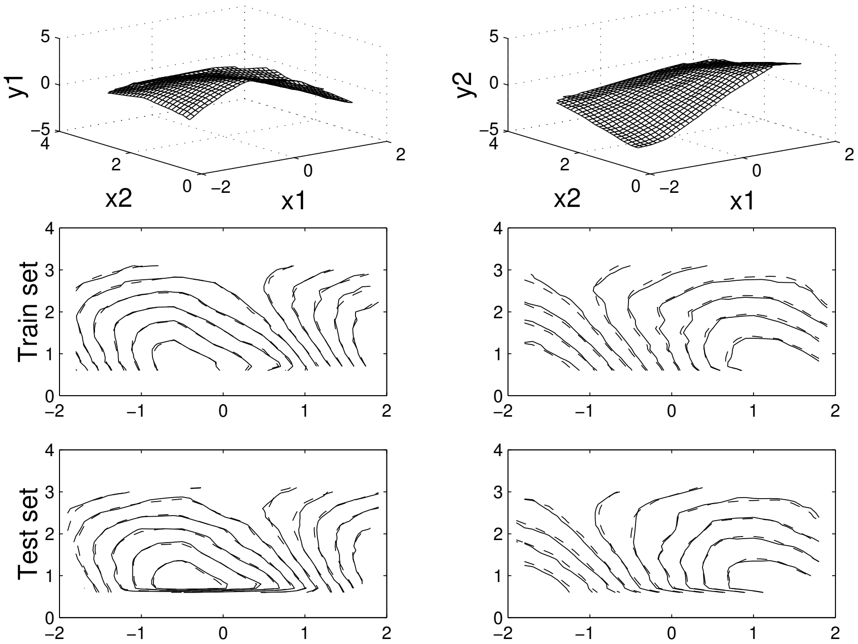

Figure 6: The top plots show the training data surfaces corresponding to each coordinate of the robot arm's position. The Middle and bottom plots show the training and validation data [- -] and the respective RBF network mappings [—].

also include the typical approximations that were obtained using the algorithm. To assess

[5]The data set can be found in David Mackay's home page: `http://wol.ra.phy.cam.ac.uk/mackay/`

convergence, we simply plotted the probabilities of each model order $\widehat{p}(k|\mathbf{x},\mathbf{y})$ in the chain (using equation (11)) for 50000 iterations, as shown in Figure 7. As the model orders begin to stabilise after 30000 iterations, we decided to run the Markov chains for 50000 iterations with a burn in of 30000 iterations. It is possible to design more complex convergence diagnostic tools, however this topic is beyond the scope of this paper.

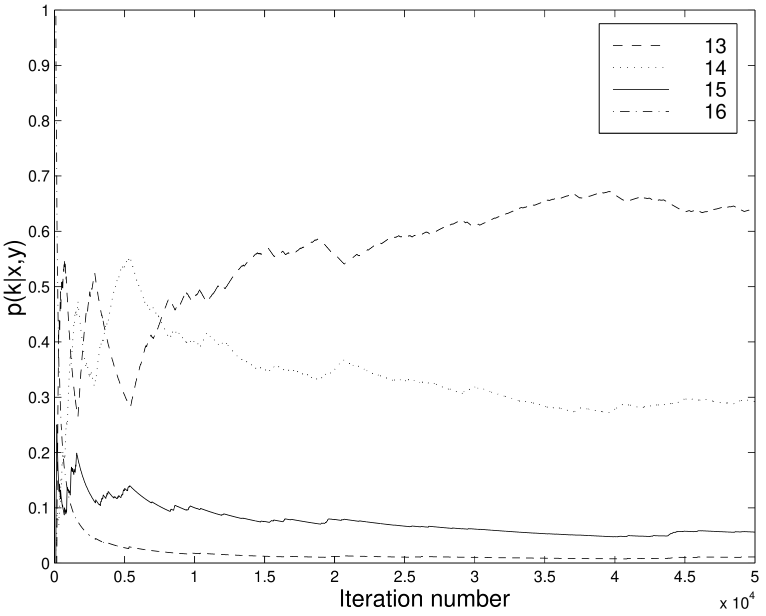

Figure 7: Convergence of the reversible jump MCMC algorithm for RBF networks. The plot shows the probability of each model order given the data. The model orders begin to stabilise after 30000 iterations.

We chose uninformative priors for all the parameters and hyper-parameters. In particular, we used the values shown in Table 2. To demonstrate the robustness of our algorithm, we

α_{δ^2}	β_{δ^2}	υ_0	γ_0	ε_1	ε_2	MS 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

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.

chose different values for β_{δ^2} (the only critical hyper-parameter as it quantifies the mean of the spread $\boldsymbol{\delta}$ of $\boldsymbol{\alpha}_k$). The obtained mean square errors (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 algorithm is robust with respect to prior specification.

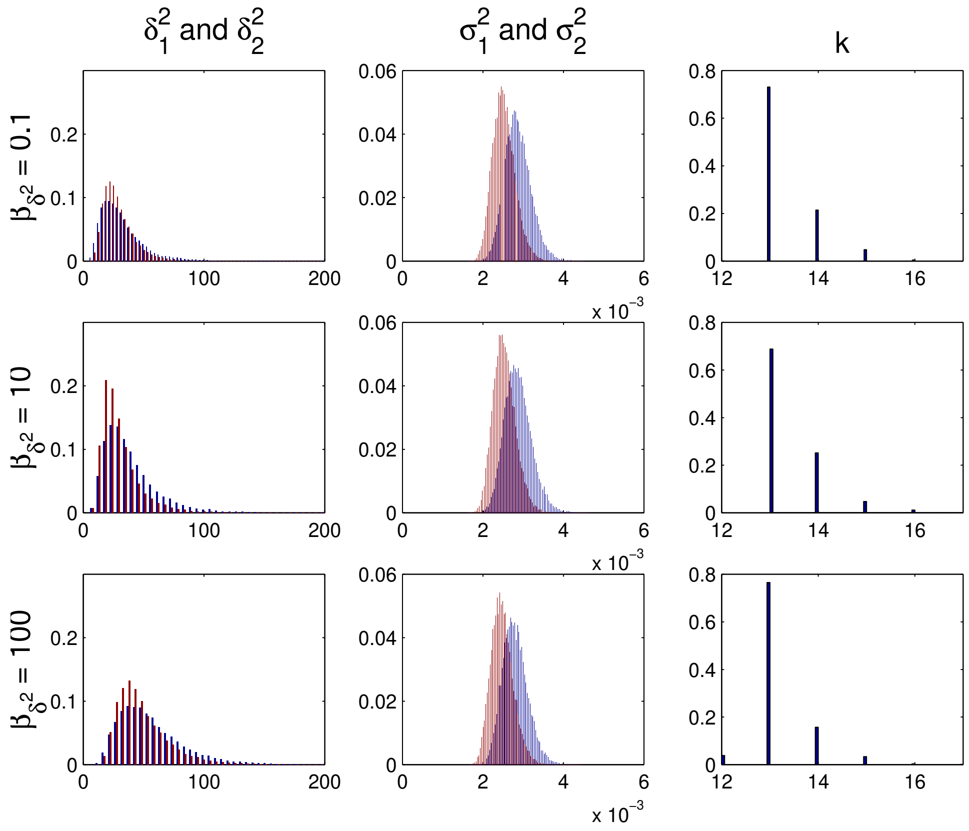

Figure 8: Histograms of smoothness constraints for each output ($\boldsymbol{\delta}_1$ and $\boldsymbol{\delta}_2$), noise variances ($\boldsymbol{\sigma}^2_{1,k}$ and $\boldsymbol{\sigma}^2_{2,k}$) and model order ($k$) for the robot arm data simulation using 3 different values for β_{δ^2}. The plots confirm that the algorithm is robust to the setting of β_{δ^2}.

Our mean square errors are of the same magnitude as the ones reported by other researchers (Holmes and Mallick 1998, Mackay 1992, Neal 1996, Rios Insua and Müller 1998); slightly better (Not by more than 10%). Yet, the main point we are trying to make is that our algorithm exhibits the important quality of being robust to the 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 network in the Markov chain. For the final network, we chose the one that maximised the posterior. 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 provides more accurate models. Moreover, it seems that the

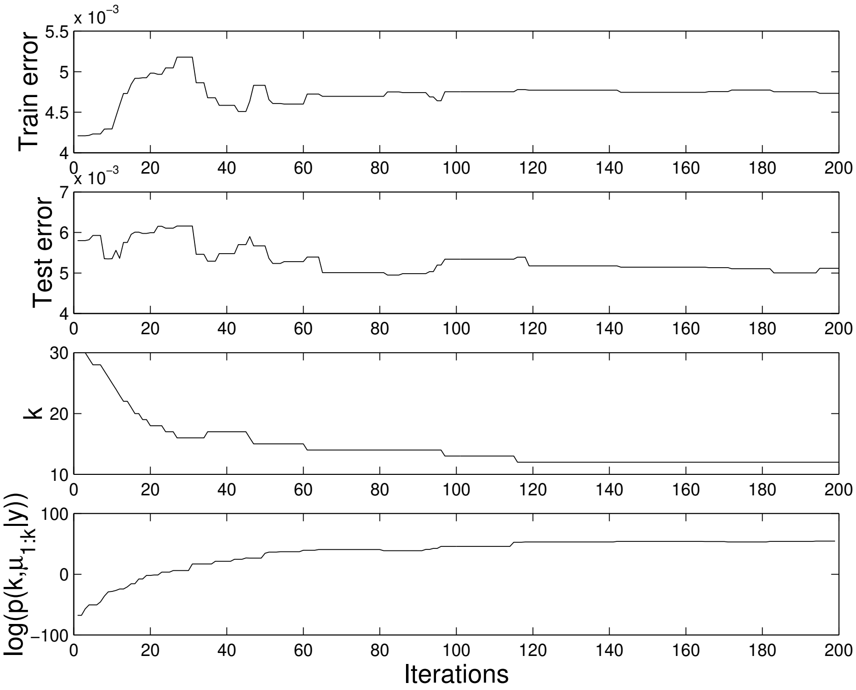

Figure 9: Performance of the reversible jump simulated annealing algorithm for 200 iterations on the robot arm data, with the MDL criterion.

information criteria, in particular the AIC, can lead to over-fitting of the data.

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

7.3 Experiment 3: Classification with discriminants

Here, we consider an interesting nonlinear classification data set[6] collected as part of a study to identify patients with muscle tremor (Roberts, Penny and Pillot 1996, Spyers-Ashby, Bain and Roberts 1998). The data was gathered from a group of patients (9 with, primarily, Parkinson's disease or multiple sclerosis) and from a control group (not exhibiting the disease). Arm muscle tremor was measured with a 3-D mouse and a movement tracker in three linear and three angular directions. The time series of the measurements were parameterised 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 decision boundaries (solid lines) and confidence intervals (dashed lines) 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 and more principled approach, which we

[6] The data is available at Stephen Roberts' home page: `http://www.ee.ic.ac.uk/hp/staff/sroberts.html`

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 generalised linear models framework with logistic, probit or softmax link functions (Gelman, Carlin, Stern and Rubin 1995, Holmes 1999, Nabney 1999).

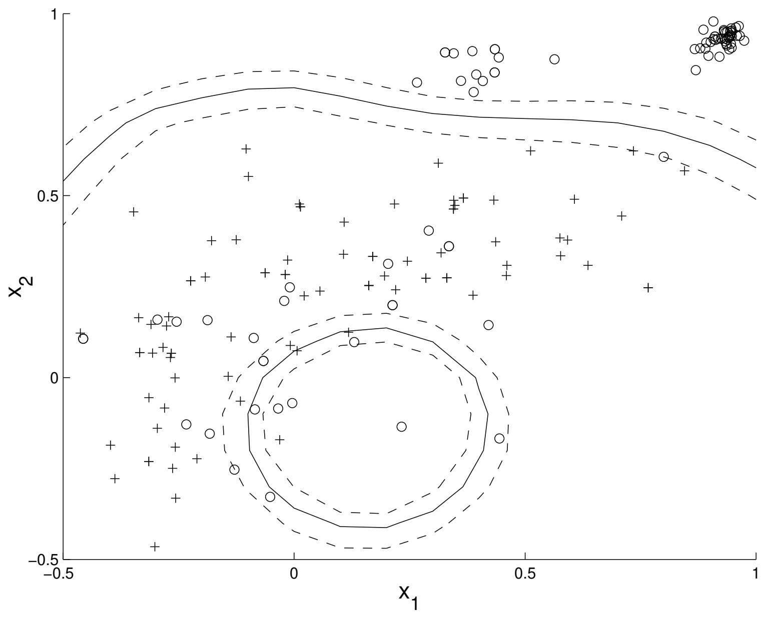

Figure 10: Classification boundaries (–) and confidence intervals (- -) for the RBF classifier. The circles indicate patients, while the crosses represent the control group.

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, Swets 1963). In our application, the ROC curve was obtained by averaging all the predictions for each classifier in the Markov chain. However, the classification performance could be improved by using the convex hull of the ROC curves for each classifier in the chain. This would yield the maximum realisable classifier according to the Neyman Pearson criterion (Andrieu, de Freitas and Doucet 1999a, Scott, Niranjan and Prager 1998).

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 and Gee 1998, Roberts and 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.

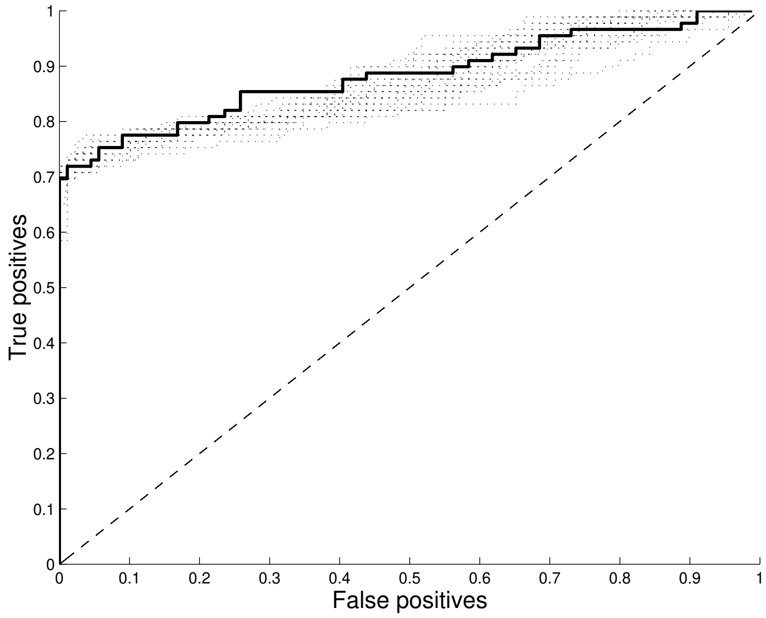

Figure 11: Receiver operating characteristic (ROC) of the classifier for the tremor data. The solid line is the ROC curve for the posterior mean classifier, while the dotted lines correspond to the curves obtained for various classifiers in the Markov chain.

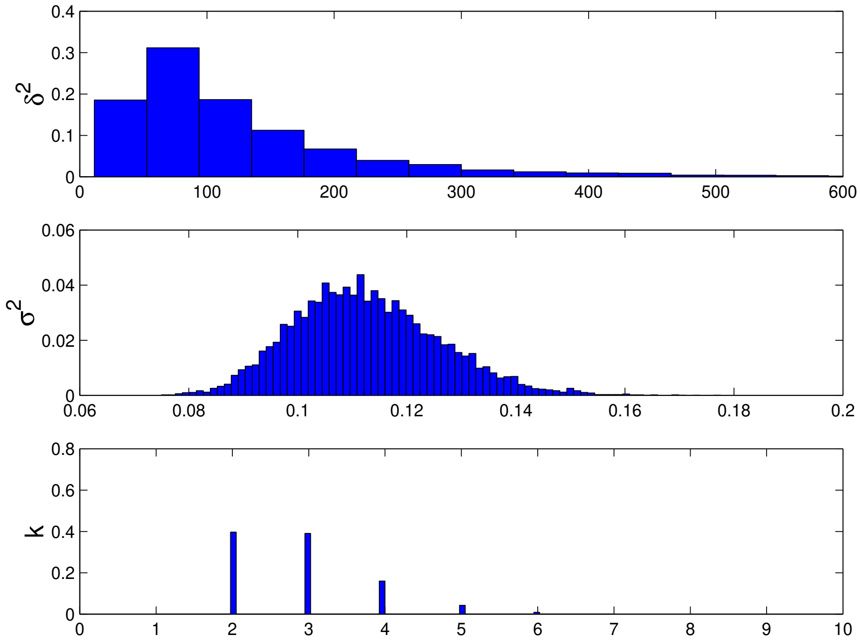

Figure 12: Estimated probabilities of the signal to noise ratio (δ^2), noise variance (σ^2) and model order (k) for the classification example.

8 Conclusions

In this paper, we presented a general methodology for estimating, jointly, the noise variance, parameters and number of parameters of an RBF model. In adopting a Bayesian model and a reversible jump MCMC algorithm to perform the necessary integrations, we demonstrated that the method is very accurate, informative and robust. We also considered the problem of stochastic optimisation 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 previous reported results, our experiments indicate that our model is robust with respect to the specification of the prior. In addition, we obtained more parsimonious RBF networks and better approximation errors than the ones previously reported in the literature.

There are many avenues for further research. These include estimating the type of basis functions that is 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 generalised 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 in this paper. Variable selection schemes can also be implemented via the reversible jump MCMC algorithm. Finally, we are presently working on a sequential version of the algorithm that allows us to perform model selection in non-stationary environments (Andrieu, de Freitas and Doucet 1999c, Andrieu, de Freitas and Doucet 1999b). We also believe that the algorithms need to be tested on additional real world problems. For this purpose, we have made the softare available at `http://www-svr.eng.cam.ac.uk/~jfgf`.

9 Acknowledgements

We would like to thank Mark Coates and Jaco Vermaak (Cambridge University), Chris Holmes (Imperial College of London), David Melvin (Cambridge Clinical School), Stephen Roberts and Will Penny (Imperial College of London) for very useful discussions and comments. Christophe Andrieu is financially supported by AT&T Labs, Cambridge. Nando de Freitas is financially supported by two University of the Witwatersrand Merit Scholarships, a Foundation for Research Development Scholarship (South Africa), an ORS award and a Trinity College External Research Studentship (Cambridge). Arnaud Doucet is financially supported by an EPSRC grant, UK.

A Notation

- $\mathbf{A}_{i,j}$: entry of the matrix $\mathbf{A}$ in the i^{th} row and j^{th} column.
- $\mathbf{A}'$: transpose of matrix $\mathbf{A}$.
- $|\mathbf{A}|$: determinant of matrix $\mathbf{A}$.
- If $\mathbf{z} \triangleq (z_1, ..., z_{j-1}, z_j, z_{j+1}, ..., z_k)'$ then $\mathbf{z}_{-j} \triangleq (z_1, ..., z_{j-1}, z_{j+1}, ..., z_k)'$.
- $\mathbf{I}_n$: 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 $p(\mathbf{z})$.
- $\mathbf{z}|\,\mathbf{y} \sim p(\mathbf{z})$: 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	$\mathcal{U}_A$	$\left[\int_A d\mathbf{z}\right]^{-1} \mathbb{I}_A(\mathbf{z})$

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 towards a small set C with drift function $V : \mathbf{Z} \to [1, +\infty)$* i.e. *there exists $0 < \lambda < 1$, $b > 0$, k_0 and an integrable measure ν such that:*

$$PV(\mathbf{z}) \leq \lambda V(\mathbf{z}) + b\mathbb{I}_C(\mathbf{z}) \tag{51}$$

$$P^{k_0}(\mathbf{z}, d\mathbf{z}') \geq \mathbb{I}_C(\mathbf{z})\nu(d\mathbf{z}') \tag{52}$$

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

$$\|P^n(\mathbf{z}_0, \cdot) - \pi(\cdot)\|_{TV} \leq RV(\mathbf{z}_0)\rho^n \tag{53}$$

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 minorisation condition (52) 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 (51). 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 $\mathcal{K}(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_2}, d\delta_{k_2}^2, k_2, d\boldsymbol{\mu}_{1:k_2})$ denote[7] the transition kernel of the Markov chain. Thus, for fixed $(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}) \in \mathbb{R}^{+2} \times \boldsymbol{\Omega}$, we have:

$$\Pr\big((\Lambda_{k_2}, \delta_{k_2}^2, k_2, \boldsymbol{\mu}_{1:k_2}) \in A_{k_2} | (\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1})\big) = \int_{A_{k_2}} \mathcal{K}(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_2}, d\delta_{k_2}^2, k_2, d\boldsymbol{\mu}_{1:k_2}) \tag{54}$$

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

$$\begin{aligned}
&\mathcal{K}(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_2}, d\delta_{k_2}^2, k_2, d\boldsymbol{\mu}_{1:k_2}) \\
&\qquad = \Big(b_{k_1} \mathcal{K}_{birth}(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta_{k_1}^2, k_1+1, d\boldsymbol{\mu}_{1:k_1+1}) \\
&\qquad + d_{k_1} \mathcal{K}_{death}(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta_{k_1}^2, k_1-1, d\boldsymbol{\mu}_{1:k_1-1}) \\
&\qquad + s_{k_1} \mathcal{K}_{split}(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta_{k_1}^2, k_1+1, d\boldsymbol{\mu}_{1:k_1+1}) \\
&\qquad + m_{k_1} \mathcal{K}_{merge}(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta_{k_1}^2, k_1-1, d\boldsymbol{\mu}_{1:k_1-1}) \\
&\qquad + (1 - b_{k_1} - d_{k_1} - s_{k_1} - m_{k_1}) \mathcal{K}_{update}(\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1}, d\delta_{k_1}^2, k_1, d\boldsymbol{\mu}_{1:k_1}^*) \Big) \\
&\qquad \times p(\delta_{k_2}^2 | \delta_{k_1}^2, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) p(\Lambda_{k_2} | k_2)
\end{aligned} \tag{55}$$

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 $\delta_{k_2}^2$ 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 $\boldsymbol{\mu}_{1:k} \in \boldsymbol{\Omega}_k$.*

■

[7] In what follows we will use the notation Λ_k, δ_k^2, when necessary, for ease of presentation. This does not mean that these variables depend on the dimension 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* $\mathbf{y}$ *is really noisy, i.e. it 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* $\boldsymbol{\mu}_{1:k} \in \mathbf{\Omega}_k$

$$\mathbf{y}'\mathbf{P}_k\mathbf{y} \geq \varepsilon > 0 \tag{56}$$

■

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

$$\mathbf{y}'\mathbf{P}_k\mathbf{y} \leq \mathbf{y}'\mathbf{y} \tag{57}$$

■

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* $k_1 = 1, \ldots, k_{\max}$*:*

$$K_1(\Lambda, \delta^2, k_1, \boldsymbol{\mu}_{1:k_1}; \Lambda, \delta^2, k_1 - 1, d\boldsymbol{\mu}_{1:k_1-1}) \geq \frac{c^\star \mathbb{I}_{\{\Lambda;\Lambda<M_2\}}(\Lambda)}{M_1 M_2 k_1} \delta_{S_{\boldsymbol{\mu}_{1:k_1}}}(d\boldsymbol{\mu}_{1:k_1-1}) \tag{58}$$

with $c^\star > 0$ *as defined in equation (22).*

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

$$K_1(\Lambda, \delta^2, k_1, \boldsymbol{\mu}_{1:k_1}; \Lambda, \delta^2, k_2, d\boldsymbol{\mu}_{1:k_2}) \geq \min\{1, r_{death}\} d_{k_1} \frac{\delta_{S_{\boldsymbol{\mu}_{1:k_1}}}(d\boldsymbol{\mu}_{1:k_2})}{k_1} \tag{59}$$

where $1/k_1$ is the probability of choosing one of the basis functions for the purpose or removing it and $S_{\boldsymbol{\mu}_{1:k_1}} \triangleq \{\boldsymbol{\mu}' \in \mathbf{\Omega}_{k_1-1} / \exists l \in \{1, \ldots, k_1\}$ such that $\boldsymbol{\mu}' = \boldsymbol{\mu}_{-l}\}$. Then from equation (25) and for all $k_1 = 1, \ldots, k_{\max}$, we have:

$$r_{death}^{-1} = \left(\frac{\gamma_0 + \mathbf{y}_{1:N}'\mathbf{P}_{k_1-1}\mathbf{y}_{1:N}}{\gamma_0 + \mathbf{y}_{1:N}'\mathbf{P}_{k_1}\mathbf{y}_{1:N}} \right)^{(\frac{N+v_0}{2})} \frac{1}{k_1(1+\delta^2)^{1/2}}$$

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

$$r_{death}^{-1} \leq \left(\frac{\gamma_0 + \mathbf{y}_{1:N}'\mathbf{y}_{1:N}}{\varepsilon} \right)^{(\frac{N+v_0}{2})} \frac{1}{k_1(1+\delta^2)^{1/2}} < M_1 < +\infty \tag{60}$$

Thus there exists M_1 sufficiently large such that for any M_2 sufficiently large (from equation (22)), $\delta^2 \in \mathbb{R}^+$, $1 \leq k_1 \leq k_{\max}$ and $\boldsymbol{\mu}_{1:k_1} \in \mathbf{\Omega}_{k_1}$

$$K_1(\Lambda, \delta^2, k_1, \boldsymbol{\mu}_{1:k_1}; \Lambda, \delta^2, k_1 - 1, d\boldsymbol{\mu}_{1:k_1-1}) \geq \mathbb{I}_{\{\Lambda;\Lambda<M_2\}}(\Lambda) \frac{c^\star}{M_2} \frac{1}{M_1 k_1} \delta_{S_{\boldsymbol{\mu}_{1:k_1}}}(d\boldsymbol{\mu}_{1:k_1-1}) \tag{61}$$

■

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

$$\mathcal{K}(\Lambda_0,\delta_0^2,0,\boldsymbol{\mu}_0;d\Lambda_0^*,d\delta_0^{*2},0,d\boldsymbol{\mu}_0)\geq\zeta\varphi(\delta_0^{*2}|0)p(\Lambda_0|0)d\delta_0^{*2}d\Lambda_0 \tag{62}$$

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

Proof. From the definition of the transition kernel $\mathcal{K}$, we have [8]:

$$\begin{aligned}\mathcal{K}(\Lambda_0,\delta_0^2,0,\boldsymbol{\mu}_0;d\Lambda_0^*,d\delta_0^{*2},0,d\boldsymbol{\mu}_0) &\geq u_0p(\delta_0^{*2}|\delta_0^2,0,d\boldsymbol{\mu}_0,\mathbf{x},\mathbf{y})p(\Lambda_0|0)d\delta_0^{*2}d\Lambda_0\\ &\geq (1-c^\star)p(\delta_0^{*2}|\delta_0^2,0,d\boldsymbol{\mu}_0,\mathbf{x},\mathbf{y})p(\Lambda_0|0)d\delta_0^{*2}d\Lambda_0\end{aligned} \tag{63}$$

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

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:*

$$p(\delta^{*2}|\delta^2,k,\boldsymbol{\mu}_{1:k},\mathbf{x},\mathbf{y})\geq\xi\varphi(\delta^{*2}|k) \tag{64}$$

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

$$p(\sigma^2|\delta^2,k,\boldsymbol{\mu}_{1:k},\mathbf{x},\mathbf{y})=\frac{\left(\frac{\gamma_0+\mathbf{y}'_{1:N}\mathbf{P}_k\mathbf{y}_{1:N}}{2}\right)^{\frac{N+v_0}{2}}}{\Gamma\left(\frac{N+v_0}{2}\right)(\sigma^2)^{\frac{N+v_0}{2}+1}}\exp\left(\frac{-1}{2\sigma^2}(\gamma_0+\mathbf{y}'_{1:N}\mathbf{P}_k\mathbf{y}_{1:N})\right) \tag{65}$$

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

$$p(\boldsymbol{\alpha}_{1:m}|\delta^2,k,\boldsymbol{\mu}_{1:k},\sigma^2,\mathbf{x},\mathbf{y})=\frac{1}{|2\pi\sigma^2\mathbf{M}_k|^{1/2}}\exp\left(\frac{-1}{2\sigma^2}(\boldsymbol{\alpha}_{1:m}-\mathbf{h}_k)'\mathbf{M}_k^{-1}(\boldsymbol{\alpha}_{1:m}-\mathbf{h}_k)\right) \tag{66}$$

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

$$\begin{aligned}p(\delta^{*2}|\delta^2,k,\boldsymbol{\theta},\mathbf{x},\mathbf{y}) &= \frac{\left(\frac{\boldsymbol{\alpha}'_{1:m}\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x})\boldsymbol{\alpha}_{1:m}}{2\sigma^2}+\beta_{\delta^2}\right)^{m/2+\alpha_{\delta^2}}}{\Gamma(m/2+\alpha_{\delta^2})(\delta^{*2})^{m/2+\alpha_{\delta^2}+1}}\\ &\quad\times\exp\left(\frac{-1}{\delta^{*2}}\left(\frac{\boldsymbol{\alpha}'_{1:m}\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x})\boldsymbol{\alpha}_{1:m}}{2\sigma^2}+\beta_{\delta^2}\right)\right)\end{aligned} \tag{67}$$

Consequently:

$$\begin{aligned}&p(\delta^{*2}|\delta^2,k,\boldsymbol{\theta},\mathbf{x},\mathbf{y})p(\boldsymbol{\alpha}_{1:m},\sigma^2|\delta^2,k,\boldsymbol{\mu}_{1:k},\mathbf{x},\mathbf{y})\\ &\quad=\frac{\left(\frac{\gamma_0+\mathbf{y}'_{1:N}\mathbf{P}_k\mathbf{y}_{1:N}}{2}\right)^{\frac{N+v_0}{2}}\left(\frac{\boldsymbol{\alpha}'_{1:m}\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x})\boldsymbol{\alpha}_{1:m}}{2\sigma^2}+\beta_{\delta^2}\right)^{m/2+\alpha_{\delta^2}}}{\Gamma(\frac{N+v_0}{2})\Gamma(m/2+\alpha_{\delta^2})(2\pi)^{m/2}(\sigma^2)^{(N+v_0+m)/2+1}(\delta^{*2})^{m/2+\alpha_{\delta^2}+1}|\mathbf{M}_k|^{1/2}}\\ &\quad\times\exp\left(\frac{-1}{2\sigma^2}\left[(\boldsymbol{\alpha}_{1:m}-\mathbf{h}_k)'\mathbf{M}_k^{-1}(\boldsymbol{\alpha}_{1:m}-\mathbf{h}_k)+\gamma_0+\mathbf{y}'_{1:N}\mathbf{P}_k\mathbf{y}_{1:N}\right.\right.\\ &\qquad\left.\left.+\frac{\boldsymbol{\alpha}'_{1:m}\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x})\boldsymbol{\alpha}_{1:m}}{\delta^{*2}}\right]-\frac{\beta_{\delta^2}}{\delta^{*2}}\right)\end{aligned} \tag{68}$$

[8] When $k=0$, we keep for notational convenience the same notation for the transition kernel even if $\boldsymbol{\mu}_0$ does not exist.

We can obtain the minorisation condition, given by equation (63), 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:

$$\begin{aligned}
& (\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k)'\mathbf{M}_k^{-1}(\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k) + (\gamma_0 + \mathbf{y}_{1:N}\mathbf{P}_k\mathbf{y}_{1:N}) + \frac{\boldsymbol{\alpha}'_{1:m}\mathbf{D}'(\boldsymbol{\mu}_{1:k},\mathbf{x})\mathbf{D}(\boldsymbol{\mu}_{1:k},\mathbf{x})\boldsymbol{\alpha}_{1:m}}{\delta^{*2}} \\
= \; & (\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k^{\bullet})'\mathbf{M}_k^{\bullet-1}(\boldsymbol{\alpha}_{1:m} - \mathbf{h}_k^{\bullet}) + \gamma_0 + \mathbf{y}_{1:N}\mathbf{P}_k^{\bullet}\mathbf{y}_{1:N} \qquad (69)
\end{aligned}$$

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 with respect to $\boldsymbol{\sigma}^2$ (inverse Gamma distribution) to obtain the minorisation condition for δ^{*2}:

$$\begin{aligned}
& p(\delta^{*2}|\delta^2, k, \boldsymbol{\mu}_{1:k}, \mathbf{x}, \mathbf{y}) \\
\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+v_0}{2}} \left(\beta_{\delta^2}\right)^{m/2+\alpha_{\delta^2}}}{\Gamma(\frac{N+v_0}{2})\Gamma(m/2+\alpha_{\delta^2})(2\pi)^{m/2}(\sigma^2)^{(N+v_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^{\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] - \frac{\beta_{\delta^2}}{\delta^{*2}}\right) d\boldsymbol{\alpha}_{1:m} d\sigma^2 \\
= \; & \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+v_0}{2}} \left(\beta_{\delta^2}\right)^{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+v_0}{2}} (\delta^{*2})^{m/2+\alpha_{\delta^2}+1}} \exp\left(-\frac{\beta_{\delta^2}}{\delta^{*2}}\right) \\
\geq \; & \left(\frac{1 + \frac{1}{\delta^2}}{1 + \frac{1}{\delta^2} + \frac{1}{\delta^{*2}}}\right) \frac{\varepsilon^{\frac{N+v_0}{2}} \beta_{\delta^2}^{m/2+\alpha_{\delta^2}}}{(\gamma_0 + \mathbf{y}'_{1:N}\mathbf{y}_{1:N})^{\frac{N+v_0}{2}}\Gamma(m/2+\alpha_{\delta^2})} \frac{1}{(\delta^{*2})^{m/2+\alpha_{\delta^2}+1}} \\
& \times \exp\left(-\frac{\beta_{\delta^2}}{\delta^{*2}}\right) \\
\geq \; & \left(\frac{1}{1+\delta^{*2}}\right)^{k_{\max}/2} \frac{\varepsilon^{\frac{N+v_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+v_0}{2}}\Gamma\left((k_{\max}+d+1)/2 + \alpha_{\delta^2}\right)} \\
& \times \frac{1}{(\delta^{*2})^{\frac{k_{\max}+d+1}{2}+\alpha_{\delta^2}+1}} \exp\left(-\frac{\beta_{\delta^2}}{\delta^{*2}}\right) \qquad (70)
\end{aligned}$$

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 $((\Lambda_{k_1}, \delta_{k_1}^2, k_1, \boldsymbol{\mu}_{1:k_1}), (\Lambda_{k_2}, \delta_{k_2}^2, k_2, \boldsymbol{\mu}_{1:k_2})) \in (\mathbb{R}^{+2} \times \boldsymbol{\Omega})^2$*

$$\mathcal{K}^{(k_{\max})}(\Lambda_{k_1}, \delta_1^2, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_2}, d\delta_{k_2}^2, k_2, d\boldsymbol{\mu}_{1:k_2}) \geq \mathbb{I}_{\{\Lambda_{k_1}; \Lambda_{k_1} < M_2\}}(\Lambda_{k_1})\eta_{M_2}\phi(d\Lambda_{k_2}, d\delta_{k_2}^2, k_2, d\boldsymbol{\mu}_{1:k_2}) \qquad (71)$$

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

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

$$\mathcal{K}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_{k_1-1}, d\delta^2_{k_1-1}, k_1-1, d\boldsymbol{\mu}_{k_1-1}) \geq \mathbb{I}_{\{\Lambda_{k_1};\Lambda_{k_1}<M_2\}}(\Lambda_{k_1}) \frac{c^\star}{M_2} \frac{1}{M_1 k_1} \times \xi p(\Lambda_{k_1-1}|k_1-1) d\Lambda_{k_1-1} \varphi(\delta^2_{k_1-1}|k_1-1) d\delta^2_{k_1-1} \delta_{S_{\boldsymbol{\mu}_{1:k_1}}}(d\boldsymbol{\mu}_{k_1-1}) \quad (72)$$

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

$$\begin{aligned}
&\mathcal{K}^{(k_{\max})}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0) \\
&= \int_{\mathbb{R}^+\times\mathbb{R}^+\times\boldsymbol{\Omega}} \mathcal{K}^{(k_1)}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_l, d\delta^2_l, l, d\boldsymbol{\mu}_{1:l}) \\
&\qquad \times \mathcal{K}^{(k_{\max}-k_1)}(\Lambda_l, \delta^2_l, l, \boldsymbol{\mu}_{1:l}; d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0) \\
&\geq \int_{\mathbb{R}^+\times\mathbb{R}^+} \int_{\{0\}\times\boldsymbol{\Omega}_0} \mathcal{K}^{(k_1)}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_l, d\delta^2_l, l, d\boldsymbol{\mu}_{1:l}) \\
&\qquad \times \mathcal{K}^{(k_{\max}-k_1)}(\Lambda_l, \delta^2_l, l, \boldsymbol{\mu}_{1:l}; d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0) \\
&= \mathcal{K}^{(k_1)}(\Lambda_{k_1}, \delta^2_{k_1}, k_1, \boldsymbol{\mu}_{1:k_1}; d\Lambda_0, d\delta^2_0, 0, d\boldsymbol{\mu}_0) \mathcal{K}^{(k_{\max}-k_1)}(\Lambda_0, \delta^2_0, 0, \boldsymbol{\mu}_0; d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0) \\
&\geq \mathbb{I}_{\{\Lambda_{k_1};\Lambda_{k_1}<M_2\}}(\Lambda_{k_1}) M_3^{k_1-1} \left(\frac{\xi c^\star}{M_1 M_2}\right)^{k_1} \varsigma^{k_{\max}-k_1} \phi(d\Lambda^*_0, d\delta^{*2}_0, 0, d\boldsymbol{\mu}^*_0) \qquad (73)
\end{aligned}$$

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

Corollary 3 *The transition kernel $\mathcal{K}$ is ϕ-irreducible. As $p(d\Lambda, d\delta^2, k, d\boldsymbol{\mu}_{1:k}|\mathbf{x}, \mathbf{y})$ is an invariant distribution of $\mathcal{K}$ and the Markov chain is ϕ-irreducible, then from (Tierney 1994, Theorem 1*, pp. 1758) the Markov chain is $p(d\Lambda, d\delta^2, k, d\boldsymbol{\mu}_{1:k}|\mathbf{x}, \mathbf{y})$-irreducible. Aperiodicity is straightforward. Indeed there is a non-zero probability of choosing the update move in the empty configuration from equation (62) and to move anywhere in $\mathbb{R}^2 \times \{0\} \times \{\boldsymbol{\mu}_0\}$. Therefore the Markov chain admits $p(d\Lambda, d\delta^2, k, d\boldsymbol{\mu}_{1:k}|\mathbf{x}, \mathbf{y})$ as unique equilibrium distribution (Tierney 1994, Theorem 1*, pp. 1758).*

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:*

$$\lim_{\Lambda\to+\infty} \mathcal{K}V(\Lambda, \delta^2, k, \boldsymbol{\mu}_{1:k}) / V(\Lambda, \delta^2, k, \boldsymbol{\mu}_{1:k}) = 0 \qquad (74)$$

where by definition:

$$\mathcal{K}V(\Lambda, \delta^2, k, \boldsymbol{\mu}_{1:k}) \triangleq \int_{\mathbb{R}^+\times\mathbb{R}^+\times\boldsymbol{\Omega}} \mathcal{K}(\Lambda, \delta^2, k, \boldsymbol{\mu}_{1:k}; d\Lambda^*, d\delta^{*2}, k^*, d\boldsymbol{\mu}^*_{1:k}) V(\Lambda^*, \delta^{*2}, k^*, \boldsymbol{\mu}^*_{1:k}) \quad (75)$$

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

$$\begin{aligned}
\mathcal{K} = &\left(b_{k_1}\mathcal{K}_{birth} + d_{k_1}\mathcal{K}_{death} + m_{k_1}\mathcal{K}_{merge} + s_{k_1}\mathcal{K}_{split} + (1 - b_{k_1} - d_{k_1} - s_{k_1} - m_{k_1})\mathcal{K}_{update}\right) \\
&\times p(\delta^2_{k_2}|\delta^2_{k_1}, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) p(\Lambda_{k_2}|k_2) \qquad (76)
\end{aligned}$$

Now, study the following expression:

$$
\begin{aligned}
& \mathcal{K}V(\Lambda_{k_1}, \delta_1^2, 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_{k_2}^2 | \delta_{k_1}^2, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) d\delta_{k_2}^2 \int_{\mathbb{R}+} p(\Lambda_{k_2}|k_2) \Lambda_{k_2}^v d\Lambda_{k_2} \\
& + d_{k_1} \sum_{k_2 \in \{k_1, k_1-1\}} \int_{\Phi_{k_2}} \mathcal{K}_{death} \int_{\mathbb{R}+} p(\delta_{k_2}^2 | \delta_{k_1}^2, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) d\delta_{k_2}^2 \int_{\mathbb{R}+} p(\Lambda_{k_2}|k_2) \Lambda_{k_2}^v d\Lambda_{k_2} \\
& + s_{k_1} \sum_{k_2 \in \{k_1, k_1+1\}} \int_{\Phi_{k_2}} \mathcal{K}_{split} \int_{\mathbb{R}+} p(\delta_{k_2}^2 | \delta_{k_1}^2, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) d\delta_{k_2}^2 \int_{\mathbb{R}+} p(\Lambda_{k_2}|k_2) \Lambda_{k_2}^v 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 | \delta_{k_1}^2, k_2, \boldsymbol{\mu}_{1:k_2}, \mathbf{x}, \mathbf{y}) d\delta_{k_2}^2 \int_{\mathbb{R}+} p(\Lambda_{k_2}|k_2) \Lambda_{k_2}^v 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} \int_{\mathbb{R}+} p(\delta_{k_1}^{*2} | \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}^*|k_1) \Lambda_{k_1}^{*v} d\Lambda_{k_1}^* \\
= \; & b_{k_1} \sum_{k_2 \in \{k_1, k_1+1\}} \int_{\mathbb{R}+} p(\Lambda_{k_2}|k_2) \Lambda_{k_2}^v d\Lambda_{k_2} + d_{k_1} \sum_{k_2 \in \{k_1, k_1-1\}} \int_{\mathbb{R}+} p(\Lambda_{k_2}|k_2) \Lambda_{k_2}^v d\Lambda_{k_2} \\
& + s_{k_1} \sum_{k_2 \in \{k_1, k_1+1\}} \int_{\mathbb{R}+} p(\Lambda_{k_2}|k_2) \Lambda_{k_2}^v d\Lambda_{k_2} + m_{k_1} \sum_{k_2 \in \{k_1, k_1-1\}} \int_{\mathbb{R}+} p(\Lambda_{k_2}|k_2) \Lambda_{k_2}^v d\Lambda_{k_2} \\
& + (1 - b_{k_1} - d_{k_1} - s_{k_1} - m_{k_1}) \int_{\mathbb{R}+} p(\Lambda_{k_1}^*|k_1) \Lambda_{k_1}^{*v} d\Lambda_{k_1}^*
\end{aligned}
$$

As $p(\Lambda|k)$ is a Gamma distribution, for any $0 \leq k \leq k_{\max}$ one has $\int_{\mathbb{R}+} p(\Lambda|k) \Lambda^v d\Lambda < +\infty$ and then the result immediately follows ■

Proof of Theorem 3

Proof. 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 $p(d\Lambda, d\delta^2, k, d\boldsymbol{\mu}_{1:k} | \mathbf{x}, \mathbf{y})$ as invariant distribution. Proposition 1 proved the ϕ-irreducibility and the minorisation condition with $k_0 = k_{\max}$ and Proposition 2 proved the drift condition, thus Theorem 3 applies ■

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*, PhD thesis, University Cergy-Pontoise, Paris XV, France. In French.

Andrieu, C. and Doucet, A. (1998). Efficient stochastic maximum a posteriori estimation for harmonic signals, *EUSPICO*, Island of Rhodes.

Andrieu, C., Breyer, L. A. and Doucet, A. (1999). Convergence of simulated annealing using Foster-Lyapunov criteria, *Technical Report CUED/F-INFENG/TR 346*, Cambridge University, http://www-sigproc.eng.cam.ac.uk/.

Andrieu, C., de Freitas, J. F. G. and Doucet, A. (1999a). Maximum realisable MCMC classifiers, Under review.

Andrieu, C., de Freitas, J. F. G. and Doucet, A. (1999b). Sequential Bayesian estimation and model selection applied to neural networks, *Technical Report CUED/F-INFENG/TR 341*, Cambridge University, http://svr-www.eng.cam.ac.uk/.

Andrieu, C., de Freitas, J. F. G. and Doucet, A. (1999c). Sequential MCMC for Bayesian model selection, *IEEE Higher Order Statistics Workshop*, Ceasarea, Israel.

Andrieu, C., Djurić, P. M. and Doucet, A. (1999). Model selection by MCMC computation, to appear in *Signal Processing*.

Bakshi, B. R. and Stephanopoulos, G. (1993). Wave-net: A multiresolution, hierarchical neural network with localized learning, *AIChE Journal* **39**(1): 57–80.

Baxt, W. G. (1990). Use of an artificial neural network for data analysis in clinical decision-making: The diagnosis of acute coronary occlusion, *Neural Computation* **2**: 480–489.

Bernardo, J. M. and Smith, A. F. M. (1994). *Bayesian Theory*, Wiley Series in Applied Probability and Statistics.

Besag, J., Green, P. J., Hidgon, D. and Mengersen, K. (1995). Bayesian computation and stochastic systems, *Statistical Science* **10**: 3–66.

Billings, S. A. (1980). Identification of nonlinear systems – a survey, *IEE Proceedings Part D* **127**(6): 272–285.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*, Clarendon Press, Oxford.

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

Breiman, L. (1993). Hinging hyperplanes for regression, classification, and function approximation, *IEEE Transactions on Information Theory* **39**(3): 999–1013.

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

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

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function, *Mathematics of Control, Signals, and Systems* **2**(4): 303–314.

de Freitas, J. F. G., Niranjan, M. and Gee, A. H. (1998). The EM algorithm and neural networks for nonlinear state space estimation, *Technical Report CUED/F-INFENG/TR 313*, Cambridge University, http://svr-www.eng.cam.ac.uk/~jfgf.

de Freitas, J. F. G., Niranjan, M., Gee, A. H. and Doucet, A. (1999). Sequential Monte Carlo methods to train neural network models, to appear in *Neural Computation*.

Denison, D. (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.

Duane, S., Kennedy, A. D., Pendleton, B. J. and Roweth, D. (1987). Hybrid Monte Carlo, *Physics Letters B* **195**(2): 216–222.

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

Fisher, R. A. (1929). Tests of significance in harmonic analysis, *Proceedings of the Royal Society A* **125**: 54–59.

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

Friedman, J. H. (1991). Multivariate adaptive regression splines, *The Annals of Statistics* **19**: 1–141.

Friedman, J. H. and Stuetzle, W. (1981). Projection pursuit regression, *Journal of the American Statistical Association* **76**(376): 817–823.

Gelfand, A. E. and 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. and Rubin, D. B. (1995). *Bayesian Data Analysis*, Chapman and Hall.

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

George, E. I. and Foster, D. P. (1997). Calibration and empirical Bayes variable selection, *Technical report*, Department of Management Science and Information Systems, University of Texas, http://uts.cc.utexas.edu/ eig/.

Gilks, W. R., Richardson, S. and Spiegelhalter, D. J. (eds) (1996). *Markov Chain Monte Carlo in Practice*, Chapman and Hall, Suffolk.

Girosi, F., Jones, M. and Poggio, T. (1995). Regularization theory and neural networks architectures, *Neural Computation* **7**: 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*, Wiley.

Hannan, E. J. (1961). Testing for a jump in the spectral function, *Journal of the Royal Statistical Society B* **23**: 394–404.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation*, Macmillan College Publishing Company.

Hecht-Nielsen, R. (1990). *Neurocomputing*, Addison-Wesley.

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

Holmes, C. (1999). A Bayesian approach to generalised nonlinear modelling with multivariate smoothing splines, Under Review.

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

Huber, P. J. (1985). Projection pursuit, *The Annals of Statistics* **13**(2): 435–475.

Jang, J. S. R. and Sun, C. T. (1993). Functional equivalence between radial basis function networks and fuzzy inference systems, *IEEE Transactions on Neural Networks* **4**(1): 156–159.

Juditsky, A., Hjalmarsson, H., Benveniste, A., Delyon, B., Ljung and Sjöberg, J. (1995). Nonlinear black-box models in system identification: Mathematical foundations, *Technical report*, Linköping University, Sweden.

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

Le Cun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E. and Hubbard, W. (1989). Backpropagation applied to handwritten zip code recognition, *Neural Computation* **1**: 541–551.

Le Cun, Y., Denker, J. S. and Solla, S. A. (1990). Optimal brain damage, *in* D. S. Touretzky (ed.), *Advances in Neural Information Processing Systems*, Vol. 2, San Mateo, CA, pp. 598–605.

Liu, J., Wong, W. H. and Kong, A. (1994). Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes, *Biometrika* **81**(1): 27–40.

Lowe, D. (1989). Adaptive radial basis function nonlinearities and the problem of generalisation, *Proceedings of the IEE Conference on Artificial Neural Networks*.

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

Marquardt, D. W. and Snee, R. D. (1975). Ridge regression in practice, *American Statistician* **29**(1): 3–20.

Marrs, A. D. (1998). An application of reversible-jump MCMC to multivariate spherical Gaussian mixtures, *in* M. I. Jordan, M. J. Kearns and S. A. Solla (eds), *Advances in Neural Information Processing Systems*, Vol. 10, pp. 577–583.

Mathews, V. J. (1991). Adaptive polynomial filters, *IEEE Signal Processing Magazine* pp. 10–26.

Meyn, S. P. and Tweedie, R. L. (1993). *Markov Chains and Stochatic Stability*, Springer, New York.

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

Müller, P. and 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, *Technical Report NCRG/99/002*, Neural Computing Research Group, Aston University, Birmingham, B4 7ET, UK.

Neal, R. M. (1996). *Bayesian Learning for Neural Networks*, Lecture Notes in Statistics No. 118, Springer-Verlag, New York.

Pao, Y. H. (1989). *Adaptive Pattern Recognition and Neural Networks*, Addisson-Wesley.

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

Poggio, T. and Girosi, F. (1990). Networks for approximation and learning, *Proceedings of the IEEE* **78**(9): 1481–1497.

Refenes, A. (ed.) (1995). *Neural Networks in the Capital Markets*, John Wiley and Sons.

Richardson, S. and 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. and Müller, P. (1998). Feedforward neural networks for nonparametric regression, *Technical Report 98–02*, Institute of Statistics and Decision Sciences, Duke University. Available at http://www.stat.duke.edu.

Ripley, B. D. (1996). *Pattern Recognition and Neural Networks*, Cambridge University Press.

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

Roberts, S. J. and Penny, W. D. (1998). Bayesian neural networks for classification: How useful is the evidence framework?, to appear in *Neural Networks*.

Roberts, S. J., Penny, W. D. and Pillot, D. (1996). Novelty, confidence and errors in connectionist systems, *IEE Colloquium on Intelligent Sensors and Fault Detection*, pp. 1–10.

Robinson, T. (1994). The application of recurrent nets to phone probability estimation, *IEEE Transactions on Neural Networks* **5**(2): 298–305.

Rosenblatt, A. (1959). *Principles of Neurodynamics*, Spartan, New York.

Rumelhart, D. E., Hinton, G. E. and Williams, R. J. (1986). Learning internal representations by error propagation, *in* D. E. Rumelhart and J. L. McClelland (eds), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Cambridge, MA, pp. 318–362.

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

Scott, M. J., Niranjan, M. and Prager, R. W. (1998). Parcel: Feature subset selection in variable cost domains, *Technical Report CUED/F-INFENG/TR 323*, Cambridge University, http://svr-www.eng.cam.ac.uk.

Smith, M. and Kohn, R. (1996). Nonparametric regression using Bayesian variable selection, *Journal of Econometrics* **75**: 317–344.

Spyers-Ashby, J. M., Bain, P. and 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**: 35–43.

Swets, J. A. (1963). Information retrieval systems, *Science* **141**: 245–250.

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

Van Laarhoven, P. J. and Arts, E. H. L. (1987). *Simulated Annealing: Theory and Applications*, Reidel Publishers, Amsterdam.

Wahba, G. and Wold, S. (1969). A completely automatic French curve: Fitting spline functions by cross-validation, *Communications on Statistics, Series A* **4**(1): 1–17.

Wetherill, G. B. (1986). *Regression Analysis with Applications – Monographs on Statistics and Applied Probability*, Chapman and Hall.

Yingwei, L., Sundararajan, N. and 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 and A. Zellner (eds), *Bayesian Inference and Decision Techniques*, Elsevier, pp. 233–243.