Biometrika (2005), **92**, 2, *pp.* 419–434

Printed in Great Britain

Hierarchical models for assessing variability among functions

BY SAM BEHSETA

Department of Mathematics, California State University, Bakersfield, California 93311, U.S.A.

sbehseta@csub.edu

ROBERT E. KASS

Department of Statistics and Center for Neural Basis of Cognition, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, U.S.A.

kass@stat.cmu.edu

AND GARRICK L. WALLSTROM

Center for Biomedical Informatics, University of Pittsburgh, Pittsburgh, Pennsylvania 15213, U.S.A.

garrick@cbmi.pitt.edu

SUMMARY

In many applications of functional data analysis, summarising functional variation based on fits, without taking account of the estimation process, runs the risk of attributing the estimation variation to the functional variation, thereby overstating the latter. For example, the first eigenvalue of a sample covariance matrix computed from estimated functions may be biased upwards. We display a set of estimated neuronal Poisson-process intensity functions where this bias is substantial, and we discuss two methods for accounting for estimation variation. One method uses a random-coefficient model, which requires all functions to be fitted with the same basis functions. An alternative method removes the same-basis restriction by means of a hierarchical Gaussian process model. In a small simulation study the hierarchical Gaussian process model outperformed the random-coefficient model and greatly reduced the bias in the estimated first eigenvalue that would result from ignoring estimation variability. For the neuronal data the hierarchical Gaussian process estimate of the first eigenvalue was much smaller than the naive estimate that ignored variability due to function estimation. The neuronal setting also illustrates the benefit of incorporating alignment parameters into the hierarchical scheme.

Some key words: Bayesian adaptive regression spline; Bayesian functional data analysis; Curve fitting; Free-knot spline; Functional data analysis; Hierarchical Gaussian process; Neuron spike train; Nonparametric regression; Reversible-jump Markov chain Monte Carlo; Smoothing.

1. INTRODUCTION

Consider the problem of describing the variability among m real-valued functions of a single variable, $f^1(t), \ldots, f^m(t)$, that have been estimated from enough data to capture sharp functional variations but not enough so that uncertainty due to estimation may be

safely ignored. The argument t is assumed to lie in a finite interval $[a, b]$. Figure 1 illustrates the situation with a sample of neuronal point-process histograms from which intensity functions have been estimated. A standard approach in both neurophysiology (Optican & Richmond, 1987) and statistics (Ramsay & Silverman, 1997) is to begin with the estimated functions $\hat{f}^i$ evaluated on a grid $t_1, \ldots, t_p$, define p-dimensional random vectors $Y^i = (\hat{f}^i(t_1), \ldots, \hat{f}^i(t_p))$, and apply techniques from multivariate analysis, such as principal components. This strategy attempts to describe the variability of the unobserved random vectors $Z^i = (f^i(t_1), \ldots, f^i(t_p))$ by instead describing the variability of the estimates Y^i. We will label this approach naive functional data analysis. This approach will be successful when there is little error in estimation relative to the variability among the functions. However, in many datasets, like that depicted in Fig. 1, the variability in the estimates $\hat{f}^i$ is substantial and the naive method will mistakenly attribute that variability to the variability among the functions. For example, the first eigenvalue of the sample covariance matrix of Y^i can be strongly biased upwards as an estimator of the first eigenvalue of the covariance matrix $V = \mathrm{cov}(Z^i)$. In this paper we present methods for estimating V that account for the variability in estimating the curves f^i.

Our approach is similar in spirit to that of Ke & Wang (2001), and we also adopt roughly the same high-level strategy as James (2002), James et al. (2000), Rice & Wu

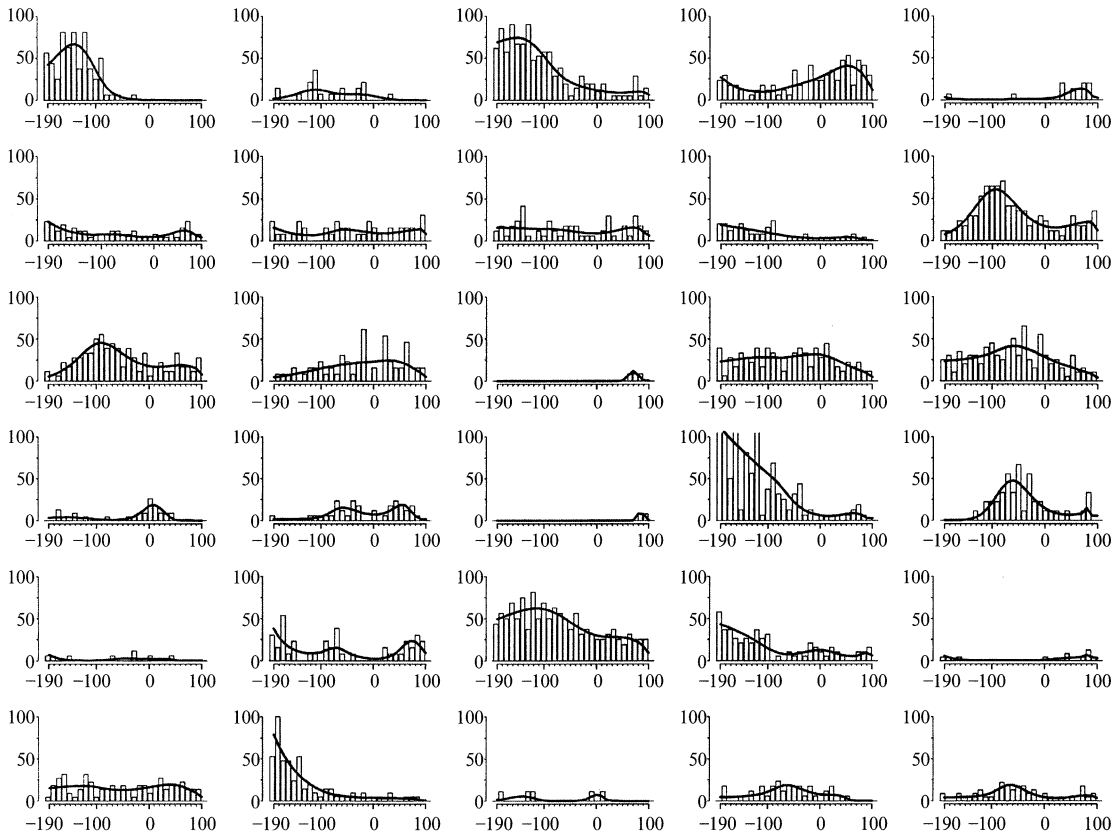

Fig. 1. Normalised histograms displaying observed firing rates for 30 neurons from an experiment described in § 5, together with fitted firing-rate curves obtained from Bayesian adaptive regression splines. Horizontal axes run from 200 milliseconds before the target hit to 100 milliseconds after; vertical axes from 0 to 100 events, i.e. neuronal spikes, per second.

(2001) and Shi et al. (1996). The methods developed here, however, are distinguished by two key features: they are able to fit functions that require domain-adaptive smoothing, so long as there are sufficient data to supply individual estimates of each function, and they are applicable to non-Gaussian data.

A natural way to describe variability among curves that are fitted with a set of basis functions is to assume that the basis coefficient vectors follow a distribution. Previous work, cited above, has emphasised such random-coefficient models, and we write down the normally-distributed random coefficient model in § 2·1. A restriction is that the basis functions must be the same across curves. While this is a reasonable simplification when there are limited data from which to estimate each curve, in cases such as that illustrated in Fig. 1 it may be preferable to fit each curve with its own empirically selected basis. We are especially interested in fitting curves that have irregular variation, being smooth over much of their domain but subject to sudden jumps: neuronal firing-rate functions often change slowly for most of the duration of observation, but rapidly in some short time interval; see DiMatteo et al. (2001) and Kass et al. (2003) for good examples. We therefore investigated an alternative approach. As discussed in § 2·2, this begins with the individually obtained fits Y^i defined above and, with the assumption that the Y^i and Z^i vectors are multivariate normal, applies a familiar normal hierarchical model, the random-effects model. Thus, in the second approach, the function values rather than the coefficients are taken as the random effects. Since in principle this can be applied at any time resolution we will regard the functions $f^i(t)$ as realisations of a Gaussian process and will refer to the resulting model as a hierarchical Gaussian process model.

The method is easily implemented and very general in so far as it applies to any context in which the variation among curves may be safely modelled by a Gaussian process, and any curve-estimation method, and set of within-curve sample sizes, that produces roughly normally distributed estimators. Furthermore, as we mention in § 2·3, this asymptotic normal approximation does not need to be highly accurate. We also note there that it is possible to correct both methods for moderate nonnormality of estimators.

A point stressed by Ramsay & Silverman (1997, Ch. 5), Wang & Gasser (1997) and others is that in describing variability among functions it is often important first to align the functions, partly to avoid attributing alignment variability to between-curve variability, and partly to make the overall mean function similar to the individual functions; an extreme case occurs when curves are identical in shape but shifted by varying amounts, in which case one would first shift the curves to align them, and then discover that their shapes exhibited no variability. Thus, any analysis scheme that purports to treat functional data should accommodate alignment in some way. In § 2·4 we note the relative ease with which alignment may be incorporated into the hierarchical modelling framework, thereby providing a complete accounting for the dominant sources of variability among a set curves. This becomes an additional use of hierarchical modelling in functional data analysis.

Many methods could be used to obtain estimates $\hat{f}^i$ of the curves f^i. We apply here an especially powerful approach called 'Bayesian adaptive regression splines' (DiMatteo et al., 2001; Kass & Wallstrom, 2002), which can be applied to a wide class of nonlinear models, including Poisson and other generalised nonparametric regression models. Although we use Bayesian methods throughout this paper, the general hierarchical model in § 2·2 could be used with alternative estimation methods, such as restricted maximum likelihood in conjunction with bootstrap estimates of the variability of the fitted curves $\hat{f}^i$ based on kernel smoothers.

2. HIERARCHICAL MODELS

2·1. *Random-coefficient models*

For individual units, or neurons, $i = 1, \ldots, m$ observed at times u_{ij}, for $j = 1, \ldots, n_i$, suppose we have data W_{ij} that follow distributions

$$W_{ij} \sim p\{w_{ij} | f^i(u_{ij})\}, \tag{1}$$

where the functions f^i have the form $f^i(t) = \sum_h b_h^i(t)\beta_h^i$ for some basis functions $b_j^i(t)$. Below we will use a spline basis defined by a set of knots ξ and these will be determined empirically: we will obtain a posterior distribution on the knot set. We will therefore use ξ to index the coefficients, and so on. For the moment the formulation may be more general, with ξ standing for an abstract index that in some way defines the basis. Now suppose that each function is evaluated at points $u_1, \ldots, u_n$ and that we collect the function evaluations into a vector $X_\xi \beta_\xi^i$, where X_ξ is the design matrix and β_ξ^i is the coefficient vector for the ith function. In principle X_ξ could vary with the function, and thus could be subscripted with i, but for notational convenience we assume that all functions are evaluated at the same values of t. We then define the random-coefficient model

$$(f^i(u_1), \ldots, f^i(u_n))^{\mathrm{T}} = X_\xi \beta_\xi^i, \quad \beta_\xi^i | \xi, \alpha_\xi, D_\xi \sim N(\alpha_\xi, D_\xi), \tag{2}$$

for $i = 1, \ldots, m$ independently. Since all curves have the same basis functions, variation among curves may be described in terms of variation among the coefficients; conditionally on ξ, this is summarised by the matrix D_ξ. As a consequence of their dependence on ξ, the D_ξ matrices produced are not comparable, in the case of splines being produced with different knot sets ξ and perhaps having different dimensionalities, and therefore they should not be interpreted directly. Instead, we view the functions f^i as draws from a Gaussian process defined conditionally on ξ. For any collection of t values $(\tilde{t}_1, \ldots, \tilde{t}_p)$ the corresponding matrix $\tilde{X}_\xi D_\xi \tilde{X}_\xi^{\mathrm{T}}$ is the covariance matrix of $(f^i(\tilde{t}_1), \ldots, f^i(\tilde{t}_p))$ under (ξ, α_ξ, D_ξ). This, or some functional of it, would be the object of inference.

For a given ξ, model (2) is a generalised linear mixed model. However, fitting of (2) will be computationally demanding because it involves the estimation of the parameters $\beta_\xi^1, \ldots, \beta_\xi^m$, α_ξ and D_ξ nested within estimation of ξ. One simplification we have adopted here is to replace (1) with $\hat{\beta}_\xi^i | \xi \sim N(\beta_\xi^i, R^i)$, where $\hat{\beta}_\xi^i$ is the maximum likelihood estimator and R^i is the inverse of the observed information matrix; that is, for each ξ, we estimate $\beta_\xi^1, \ldots, \beta_\xi^m$ by maximum likelihood and then apply the conditional hierarchical model

$$\hat{\beta}_\xi^i | \xi \sim N(\beta_\xi^i, R^i), \quad \beta_\xi^i | \xi, \alpha_\xi, D_\xi \sim N(\alpha_\xi, D_\xi). \tag{3}$$

We elaborate this in § 3·1, in the context of spline fitting.

2·2. *Hierarchical Gaussian processes*

For our alternative approach we begin by assuming that each estimated curve $\hat{f}^i$ may be considered as a Gaussian process with mean f^i and covariance function $\Gamma_{\hat{f}^i}$, which we write as

$$\hat{f}^i \sim \mathrm{GP}(f^i, \Gamma_{\hat{f}^i}). \tag{4}$$

Although the covariance functions $\Gamma_{\hat{f}^i}$ will be estimated during the procedure that fits $\hat{f}^i$ to f^i, in (4) we treat them as known. We then assume that the underlying functions are themselves independent realisations of a Gaussian process,

$$f^i \sim \text{GP}(\alpha, \Gamma_f). \tag{5}$$

We implement this by choosing a grid of values, $t_1, \ldots, t_p$, as in the naive approach, thereby obtaining a familiar Normal hierarchical model,

$$Y^i \sim N_p(Z^i, S^i), \quad Z^i \sim N_p(\mu, V), \tag{6}$$

where Y^i and Z^i are defined above, N_p is used to denote a p-dimensional multivariate normal distribution, the S^i covariance matrices are defined from the fitting procedure, and assumed known, and the covariance matrix V is the object about which we will make inferences. We will call (4) and (5) a hierarchical Gaussian process model, though we modify (4) in important ways below. Model (6) is a discrete form of (4) and (5).

Stated in this form, it is apparent that many different methods may be used to estimate the functions f^i, and thereby create alternative hierarchical Gaussian process models. Strictly speaking $S^i = \text{var}(Y^i|Z^i)$ is unknown. However, we simply plug in the estimate $\hat{S}^i$ obtained from fitting f^i; that is, we take $S^i = \hat{S}^i$ and treat this plug-in estimator as if it were fixed and known. In § 3·3 we take $S = \hat{S}^i$ to be the posterior covariance matrix obtained from Bayesian adaptive regression splines. Similarly, alternative methods may be used to estimate the 'random-effects' covariance matrix V. In § 3·3 we estimate V in a Bayesian way by completing the hierarchical model in (6) with a flat prior on μ and an inverse-Wishart prior on V.

2·3. *Improvement on normality*

Both (3) and (6) involve a replacement of (1) by normally distributed estimators. We have found these approximations to be entirely adequate for our applications. One explanation for this is that the use of (3) in place of the exact hierarchical model based on (1) is, for the estimation of second-stage parameters, equivalent to using Laplace's method at the first stage (Daniels & Kass, 1998), which has accuracy of order $O(n^{-1})$ rather than $O(n^{-\frac{1}{2}})$. Nonetheless, it may be desirable to do better. To obtain improved approximations to the posteriors corresponding to hierarchical models based on (1), we may use importance reweighting in conjunction with samples drawn from either (3) or (6). This is a special case of a quite general method that applies when a normal approximation is used in the first stage of a two-stage hierarchical model. This importance reweighting has been discussed and shown to be effective, for moderate departures from normality, by Daniels & Kass (1998, 1999). As we have said, we have found it unnecessary in our applications, so we omit any detailed examination here.

2·4. *Alignment*

In many applications it is important to include additional parameters to align or 'register' curves and make them comparable (Ke & Wang, 2001; Ramsay & Li, 1998; Wang & Gasser, 1997). In principle, alignment should be performed on the curves f^i themselves. In the usual approach where estimation variability of $\hat{f}^i$ is ignored, the alignment is instead performed on the estimated curves $\hat{f}^i$. Models (4)–(6) provide a natural

framework in which alignment of curves f^i can take place, so that variability associated with alignment estimation may be accounted for. Based on our experience with the flexible multiple-curve-fitting procedures described here, we do not expect alignment to improve fits greatly. Rather, its purpose is to apportion the variability appropriately, separating out that due to differing locations or scales of the variable t among the m different functions. This often provides a distinct interpretation of the data.

The simplest type of alignment, which will be illustrated in § 5, is to allow each curve to involve shift in time. In this case the functions $f^i(t)$ must be replaced by $f^i(t-\phi^i)$, where ϕ^i is the shift of location parameter for the ith curve. In a more general formulation, f^i is replaced by $f^i \circ h^i$, where $h^i(t)$ is a transformation of time, the 'time-warping' function. In the case of shifts, we have $h^i(t) = t - \phi^i$. More generally, h^i will depend on some vector of parameters ϕ^i. To treat alignment in hierarchical Gaussian process models we thus replace (4) with

$$\hat{f}^i \sim \mathrm{GP}(f^i \circ h^i, \Gamma_{\hat{f}^i}) \tag{7}$$

and we could modify (1) similarly. The parameter vectors ϕ^i would be incorporated into the hierarchical model (6) and, in a Bayesian application, into an appropriate Markov chain Monte Carlo implementation; that is, the model (6) is written conditionally on the parameter vector ϕ^i using (7) and then we also write

$$\phi^i \sim g(\phi^i | \gamma, \tau) \tag{8}$$

for some suitable probability density g that depends on parameter vectors γ and τ, which would typically involve location and scale parameters for the components of ϕ^i. This requires an additional Metropolis–Hastings step for the posterior of ϕ^i given the rest of the parameters, which is easily implemented.

The parameters γ and τ need to be estimated. For this, it should be recognised at the outset that some regularisation is needed to overcome nonidentifiability; the freedom in picking the time-realignment function h^i is traded against the freedom in picking the functional form f^i. In our Bayesian framework we introduce informative priors on the alignment hyperparameters γ and τ, and these priors will penalise alternative configurations differentially; see the discussion of identifiability in Wang & Gasser (1997) and Ramsay & Li (1998). For example, it would often be sensible to put a strong prior on the time-alignment hyperparameters that would force h to be close to the identity transformation; this is similar to using a strong penalty in the approach of Ramsay & Li (1998). In § 5 we illustrate with an analysis of neuronal data in which we introduce location parameters to allow each intensity function its own origin in time.

2·5. *Choice of grid*

An essential part of the implementation of (6) is the choice of the grid $t_1, \ldots, t_p$. Here p must be large enough that function variability is captured, but increasing p may create a covariance matrix of such dimensionality that estimation of it becomes difficult. In this context there may well be some advantage in crafting a specialised covariance estimation method; see Daniels & Kass (2001) and an as yet unpublished report by M. J. Daniels. Furthermore, the grid points do not need to be equally spaced. On the other hand, our experience to date suggests that, as in the usual applications of functional data analysis (Ramsay & Silverman, 1997), results are not very sensitive to choice of grid.

3. Multiple curve-fitting with Bayesian adaptive regression splines

3·1. *Overview of Bayesian adaptive regression splines*

We begin with the single-curve spline-based generalised nonparametric regression model for data W_j depending on a variable t:

$$W_j \sim p(w_j|\theta_j, \zeta), \quad \theta_j = f(t_j) \tag{9}$$

with f being a spline having knots at unknown locations $\xi_1, \ldots, \xi_k$. Model (9) includes a vector of nuisance parameters ζ to indicate generality, though in the Poisson case there is no nuisance parameter. If we write $f(t)$ in terms of basis functions $b_{\xi,h}(t)$ as $f(t) = \sum_h b_{\xi,h}(t)\beta_{\xi,h}$, the function evaluations $f(t_1), \ldots, f(t_n)$ may be collected into a vector $(f(t_1), \ldots, f(t_n))^{\mathrm{T}} = X_\xi \beta_\xi$, where X_ξ is the design matrix and β_ξ is the coefficient vector. For a given knot set $\xi = (\xi_1, \ldots, \xi_k)$ model (9) poses a relatively easy estimation problem; for exponential-family responses, such as Poisson, it becomes a generalised linear model. The hard part of the problem is determining the knot set ξ, and using the data to do so provides the ability to fit a wide range of functions, as reviewed by Hansen & Kooperberg (2002). Bayesian adaptive regression splines is a Markov chain Monte Carlo based algorithm that samples from a suitable approximate posterior distribution on the knot set ξ. This, in turn, produces samples from the posterior on the space of splines. In practice, cubic splines and the natural spline basis have been used in most applications. Bayesian adaptive regression splines can be viewed as a powerful engine for searching for an 'optimal' knot set, but, because it generates a posterior on the space of splines, it produces an improved spline estimator based on model averaging (Kass & Raftery, 1995) and it also provides uncertainty assessments.

Key features of the Markov chain Monte Carlo implementation of Bayesian adaptive regression splines include the following:

(i) a reversible-jump chain (Green, 1995) on ξ after integration of the marginal density

$$p(w|\xi) = \int p(w|\beta_\xi, \xi, \zeta)\pi(\beta_\xi, \zeta|\xi)d\beta_\xi d\zeta, \tag{10}$$

the integration being performed exactly for normal data and approximately, by Laplace's method, otherwise;

(ii) continuous proposals for ξ;

(iii) a locality heuristic for the proposals that attempts to place potential new knots near existing knots.

For notational convenience here and throughout the paper we are, following Hansen & Kooperberg (2002), suppressing the dependence of the knot set ξ on the number of knots k, but Bayesian adaptive regression splines explores the space of generalised regression models defined by ξ and k and the prior on k can, in some cases, control the algorithm in important ways (DiMatteo et al., 2001; Hansen & Kooperberg, 2002; Kass & Wallstrom, 2002).

The first implementation feature, item (i) above, introduces an analytical step within the Markov chain Monte Carlo partly to simplify the problem of satisfying detailed balance and partly for the sake of Markov chain Monte Carlo efficiency, which is generally increased when parameters are integrated; see Liu et al. (1994). In addition, the method takes advantage of the high accuracy of Laplace's method in this context. In so doing the

'unit-information' prior discussed by Kass & Wasserman (1995) and Pauler (1998) has been used, and this gives the interpretation that the algorithm is essentially using BIC to define a Markov chain on the knot sets. The importance of performing the integral (10), at least approximately, has been stressed by Kass & Wallstrom (2002). Continuous proposals and the locality heuristic, items (ii) and (iii), together allow knots to be placed close to one another, which is advantageous when there is a sudden jump in the function.

For each draw $\xi^{(a)}$ from the posterior distribution of ξ, a draw $\beta_\xi^{(a)}$ is obtained from the conditional posterior of β_ξ, conditionally on $\xi^{(a)}$. The conditional posterior of β_ξ may often may be assumed normal, but it also may be obtained more accurately with additional sampling, as described by DiMatteo et al. (2001) via importance reweighting, along the lines of the reweighting discussed in § 2·3 above. We have generally found that for moderate sample sizes it is unnecessary to correct the normal approximation. From $\beta_\xi^{(a)}$ we may obtain fitted values $f^{(a)}(\tilde{t}) = \sum b_{\xi,h}(\tilde{t})\beta_{\xi,h}^{(a)}$ for selected $\tilde{t}$ and these, in turn, may be used to produce a draw $\phi^{(a)}$ from the posterior distribution of any characteristic $\phi = \phi(f)$, such as the value at which the maximum of $f(t)$ occurs. Software for Bayesian adaptive regression splines is available at www.stat.cmu.edu/~kass.

We next discuss the assessment of variability among the curves $f^1, \ldots, f^m$ via fitting with alternative versions of Bayesian adaptive regression splines.

3·2. *Random-coefficient hierarchical models*

It is natural to consider using the random-coefficient model (2) to describe the variation across multiple curves that are fitted with splines. One way of proceeding would be to devise a method for selecting a single knot set ξ^* that is reasonably effective for all the curves. Fitting model (2), or its approximate version (3), is then straightforward. A simple modification of Bayesian adaptive regression splines allows it to be used to select such a ξ^*: the m curves may be fitted simultaneously, constraining them to use the same knot set. This is carried out by replacing the single-curve marginal density (10) with the joint density

$$p(w^1, \ldots, w^m|\xi) = \prod_i \int p(w^i|\beta_\xi^i, \xi)\pi(\beta_\xi^i|\xi)d\beta_\xi^i. \tag{11}$$

A sample may then be generated from the resulting posterior on ξ and the sampled knot set having the highest posterior density, the posterior modal knot set, may be selected as ξ^*.

Selecting a single $\xi = \xi^*$ and then fitting (2), however, would miss possible benefits from model averaging, here averaging across alternative knot sets. We have, instead, used (11) and generated a sample of draws $\xi^{(a)}$ from the posterior. The nontrivial problem of fitting (3) for each of many simulated vectors $\xi^{(a)}$ is greatly reduced because, in practice, it turns out that it often suffices to use a very small posterior sample of values $\xi^{(a)}$. To explain this, suppose we wish to make inferences about a functional $\phi = \phi(f)$ representing an interesting characteristic of a curve. In the total variance formula

$$\text{var}(\phi|y) = E_{\xi|y}\{\text{var}(\phi|\xi, y)\} + \text{var}_{\xi|y}\{E(\phi|\xi, y)\}, \tag{12}$$

the second term, representing the uncertainty across alternative knot sets, will often be comparatively small, not so small that it can be ignored but small enough that small posterior samples suffice for estimating ϕ or its variance. This is related to the observation that, while free-knot spline fitting poses a difficult optimisation problem, fits that represent

suboptimal local posterior maxima, which may be noticeably inferior from visual inspection, can be close, in mean-squared difference, to the correct posterior model fit. Here we have subsampled a small set of values $\xi^{(a)}$ with $a = 1, \ldots, n_a$ and, for each $\xi^{(a)}$, used Gibbs sampling, via BUGS (Spiegelhalter et al., 1996), applied to (3) to compute the posterior mean of the desired vector of function values $\tilde{X}_{\xi^{(a)}} D_{\xi^{(a)}} \tilde{X}^{\mathrm{T}}_{\xi^{(a)}}$. This produced n_a conditional posterior means. These n_a values were averaged to obtain the marginal posterior mean; see the 2003 Carnegie Mellon Ph.D. thesis of S. Behseta for implementation details.

In principle it would be possible to build a chain on $(\xi, \beta^1_\xi, \ldots, \beta^m_\xi, \alpha_\xi, D_\xi)$ but this would be delicate and rather different from the approximate integration strategy of Bayesian adaptive regression splines. We have not attempted it. It would also be possible, in principle, to use (2) to define the joint density for the data conditionally on the knots according to

$$p(w^1, \ldots, w^m|\xi) = \prod_i \int p(w^i|\beta^i_\xi, \xi)\pi_R(\beta^i_\xi|\xi, \alpha_\xi, D_\xi)\pi_{\alpha,D}(\alpha_\xi, D_\xi)d\alpha_\xi\, dD_\xi\, d\beta^i_\xi,$$

where $\pi_R(\beta^i_\xi|\xi, \alpha_\xi, D_\xi)$ is the normal random-effects distribution. However, evaluation of this density still poses a difficult integration problem nested within the knot selection problem. Instead, our use of the unit-information prior in (11) produces very mild shrinkage of the coefficients towards the origin, while the random-effects prior would produce data-controlled shrinkage of the coefficients towards their mean. We would not expect the resulting relative weights attached to alternative knot sets to be dramatically different. In any case, as we have said, (11) provides an easily-implemented method of finding knot sets that do a reasonably good job of fitting all curves simultaneously.

3·3. *Hierarchical Gaussian process models*

The virtue of the model given in (4)–(7) is that each function may be fitted separately with different knot sets. Our approach is to apply Bayesian adaptive regression splines to each of the m curves. We pick a grid $t_1, \ldots, t_p$, and compute the posterior covariance matrix $S^i = \mathrm{var}(f^i(t_1), \ldots, f^i(t_p)|w^i)$, where w^i is the vector of all data available for the ith function, not necessarily data observed at $t_1, \ldots, t_p$. This matrix S^i is obtained as the sample covariance matrix of the Markov chain Monte Carlo draws $(f^{i(a)}(t_1), \ldots, f^{i(a)}(t_p))$ where, as in the introduction to § 3 above, these fitted-value draws are, in turn, obtained from the draws $\beta^{i(a)}_\xi$ using $f^{i(a)}(\tilde{t}) = \sum b^i_{\xi,h}(\tilde{t})\beta^{i(a)}_{\xi,h}$. We then apply (6) and estimate V using Gibbs sampling, based on a flat prior on μ and an inverse-Wishart prior on V. While we are mainly interested in the covariance function Γ_f, or its finite-dimensional representation V, it should be observed that the posterior distributions on the functions f^i will produce some shrinkage away from $\hat{f}^i$ and toward their mean, the amount depending of course on the relative magnitudes of the between-curve covariance matrix V and the within-curve estimation covariance matrices S^i.

4. Simulation study

We conducted a small simulation study using as ground truth the fitted curves displayed in Fig. 1. We compared the hierarchical Gaussian process and random-coefficient methods with naive functional data analysis in estimating the first eigenvalue and first proportion

of variance. To be specific, to obtain a single data replication we simulated Poisson data as in (2), with u_{ij} values as in Fig. 1, after simulating 30 vectors $(f^i(u_{i1}), \ldots, f^i(u_{ip}))$ from a multivariate normal distribution with mean μ and variance matrix V; we set μ and V equal to the sample mean and sample covariance matrix of the curves in Fig. 1. We replicated this procedure 30 times, thereby obtaining 30 datasets and their resulting estimates from those approaches. The mean squared error results are given in Table 1. They indicate that the hierarchical Gaussian process method yields a large reduction in mean squared error compared to the random-coefficient model and a huge reduction compared to naive functional data analysis. An explanation for the improvement of the hierarchical Gaussian process model over the random-coefficient model is that the latter uses extra knots to fit a diversity of curves and therefore undersmooths in some cases. An illustration appears in Fig. 2, which is discussed below.

Table 1: *Simulation study. Mean squared error in estimating first eigenvalue and first proportion of variance across the* 30 *data replications, with simulation standard errors in parentheses.*

	Naive-FDA	Random-coefficient BARS	HGP
Eigenvalue	2·98 (0·201)	1·335 (0·132)	0·284 (0·045)
Proportion	0·060 (0·001)	0·018 (0·003)	0·0075 (0·001)

FDA, functional data analysis; BARS, Bayesian adaptive regression splines; HGP, hierarchical Gaussian process.

5. NEURONAL DATA ANALYSIS

5·1. *The data*

The data in Fig. 1 came from a study of primary motor cortex neurons in monkeys during two conditions of a sequential pointing task (Matsuzaka et al., 2001). Relevant experimental details are summarised in the Ph.D. thesis of S. Behseta; some analysis has been reported in Behseta et al. (2002). In brief, firing times of a single neuron were recorded while a monkey completed the task, with increased firing rate of the neuron indicating increased functional activity. The task required the monkey to touch a particular sequence of three illuminated buttons, among five buttons in all. In the first experimental condition the button touches were in a predetermined and highly practised order, while in the second condition they were in random order. The experiment was repeated thousands of times, over the course of more than a year, while recordings were made on single neurons. On a given day several new neurons were typically examined; it was not possible to re-examine neurons across days. A standard practice is to aggregate firing times for a given neuron across experimental replications into 10-millisecond time bins, thereby reducing the data to a count for each bin, the information lost being negligible, and this is the form in which we have analysed the data here. Figure 1 displays the resulting histograms for 30 of the neurons, out of a total of 347, over a 300-millisecond period in the random-order condition. The histograms have been normalised by dividing by the number of experimental replications for each neuron, thereby making the units events per second per replication,

which are the units associated with the Poisson process intensity functions. For a general discussion of statistical methods in a related neurophysiological context, see Ventura et al. (2002). Among other things, that work verified that it is safe to treat such aggregated and binned data as generating Poisson-distributed counts.

5·2. *Initial variability assessment*

As we indicated in § 1 the variability among curves is often described using principal components. The degree to which first principal component summarises variability is quantified by the first proportion of variance, $\lambda_1/(\lambda_1 + \ldots + \lambda_p)$, where λ_j is the jth eigenvalue. We examined the first eigenvalue and first proportion of variance for the neuronal data using both naive functional data analysis and the hierarchical Gaussian process model.

To obtain the fitted curves $\hat{f}^i$ plotted in Fig. 1 we applied Bayesian adaptive regression splines with a Poisson model in (9). We also computed posterior variance matrices S^i, using the centres of the 30 time bins as our grid. We applied the hierarchical Gaussian process model in the form (6), using on V an inverse-Wishart prior with 31 degrees of freedom and a scale matrix equal to the harmonic mean of the matrices S^i, the inverse of the mean of the $(S^i)^{-1}$ matrices. This prior, using minimal integer degrees of freedom $31 = p + 1$, seems a sensible default to us because, in the absence of other knowledge, we would want the prior to be very diffuse and we have no other starting point for the between-curve variability than the within-curve variability. Using Gibbs sampling we obtained the posterior distributions of the first eigenvalue of V and the corresponding first proportion of variance. These are displayed in Figs 2(b) and (c). In addition, we computed the sample covariance matrix from the fitted function values, i.e. the vectors Y^i in (6), and obtained its first eigenvalue and corresponding proportion of variance due to the first principal component. These two values, which represent a standard assessment of variability and which we have here called naive functional data analysis, are displayed as vertical lines in Figs 2(b) and (c). The posterior distributions are shifted substantially downwards from the naive method values that are obtained when the fitting variability is ignored. Multiplying and dividing the Wishart prior scale matrix by two produced small but nontrivial variations in the posterior on the first eigenvalue; however, these alternative priors produced negligible alterations of the posterior on the first proportion of variance. Based on the simulation study in § 4 the hierarchical Gaussian process estimators portrayed in Figs 2(b) and (c) are highly likely to be much more accurate than their naive functional data analysis counterparts.

An additional result from the simulation study in § 4 was the superiority of the hierarchical Gaussian process model over the random-coefficient model. In Fig. 2(a) we have displayed some of the results of fitting the random-coefficient model to these data. These fits use the same basis functions for all neurons. While most fitted curves are nearly the same as those obtained from fitting the curves individually, with different basis functions, several same-basis fits are a little too wiggly, especially that in the third column of the first row. This illustrates a general experience we have had in examining similar data: the assumption of the same basis functions can lead to excessive numbers of knots and therefore to overfitting. This is apparently the cause of the much poorer simulation results for the random-coefficient model.

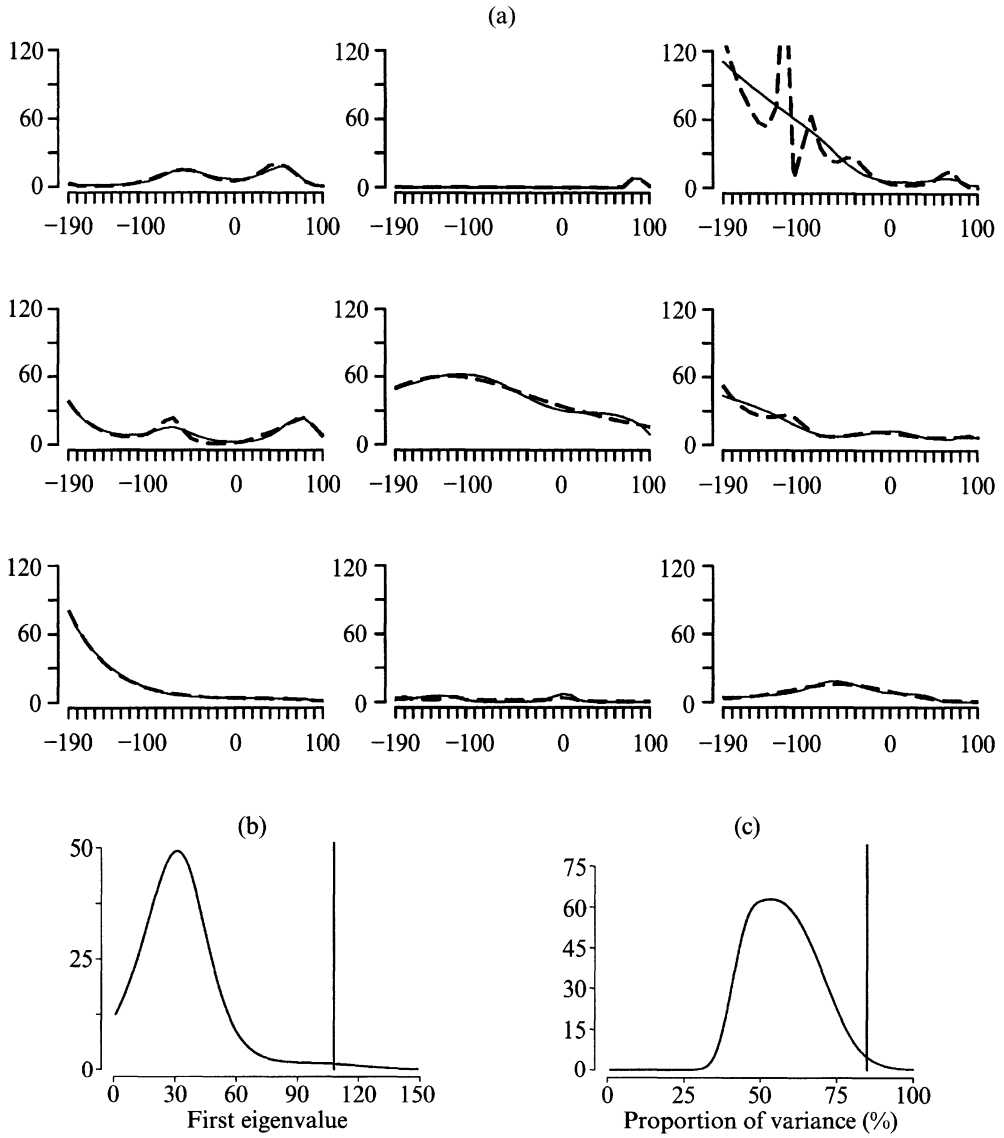

Fig. 2: Neuronal data. (a) Nine sets of fitted curves using the hierarchical Gaussian process model, shown by solid lines, and the random-coefficient model, dashed lines; the former appear in Fig. 1, rows 4–6 and columns 2–4. Horizontal axes run from 200 milliseconds before the target hit to 100 milliseconds after; vertical axes from 0 to 100 events, i.e. neuronal spikes, per second. (b) The posterior distribution of the first eigenvalue of V together with the naive functional data analysis estimate, shown by vertical bar. (c) The posterior distribution of the proportion of variance associated with the first principal component of V, together with the naive method estimate of the same quantity, vertical bar.

5·3. *Alignment*

We now illustrate the use of alignment within the hierarchical Gaussian process model using a subset of 16 neurons. These 16 neurons were classified as having similar firing-rate curves by a functional cluster analysis performed on the 347 neurons, described in more detail in the Ph.D thesis of S. Behseta. While the similarity of the firing-rate curves for these 16 neurons indicates probable similarity of physiological function, their networked connections to muscles controlling arm movement are complex and it would be

reasonable to expect the 16 neurons to have variable lags in firing activity with respect to the experimental clock time, with lags varying by perhaps tens of milliseconds. We therefore incorporated an alignment function of the form $h^i(t) = t - \theta^i$ in (7) and took θ^i, which is then a scalar, to be normally distributed with mean and standard deviation γ and τ as in (8). We set $\gamma = 0$ and took the distribution on τ to be the square-root of an inverse-chi-squared centred at 20 milliseconds with standard deviation of 20 milliseconds, reflecting knowledge of the likely magnitude of shifts. Modest changes in the prior on τ did not appreciably alter the results. Figure 3(a) shows the 16 original fitted intensity functions, while Fig. 3(c) displays the resulting aligned versions.

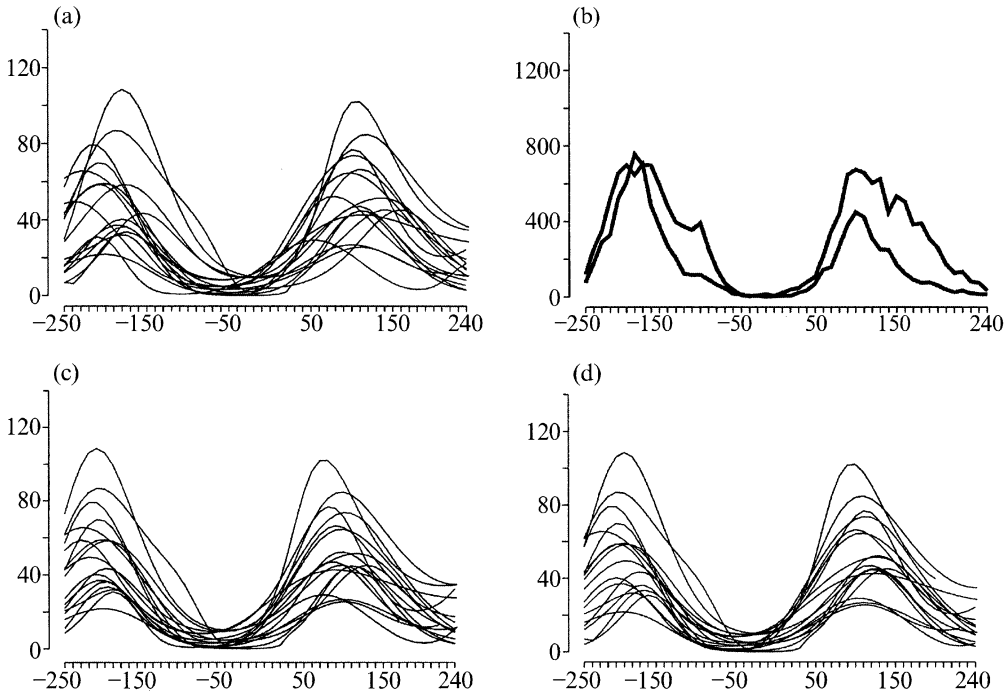

Fig. 3: Alignment. (a) Fits to firing-rate histograms from 16 neurons before alignment. (b) Electromyogram signals for the finger and wrist muscles. (c) Fits to the neuronal histograms after alignment using (8) in (7), as described in the text. (d) Fits to the neuronal histograms after alignment based on correlation with the electromyogram signals.

Some special circumstances of the experiment offered a nice opportunity to examine effectiveness of alignment according to an external criterion. As part of the experiment electromyograms for various muscle groups were also recorded for the same task. After identifying the 16 neurons by clustering we found two muscles whose electromyogram recordings have a comparable time-dependent activity pattern, shown in Fig. 3(b). These two muscles are the abductor pollicis longus, a finger muscle, and the extensor carpi radialis, a wrist muscle. We then refitted each shift parameter θ^i by maximising the correlation, over time, between the shifted Bayesian adaptive regression splines fit and the electromyogram; that is, we maximised the function

$$\rho(\theta^i) = \frac{\langle \hat{f}^i(t-\theta^i), g(t) \rangle}{\|\hat{f}^i(t-\theta^i)\| \, \|g(t)\|},$$

where g is the electromyogram signal and the numerator and denominator use the ordinary Euclidean inner product and norm, calculated from the discrete representations of the functions along a grid. The results of this analysis are shown in Fig. 3(d) with the electromyogram signals in Fig. 3(b). We observe that the functions aligned internally, in Fig. 3(c), are quite similar to those aligned externally, in Fig. 3(d). This lends support to the use of alignment in this context. Further details may be found in the Ph.D. thesis of S. Behseta.

As mentioned above, alignment is important because it is often desirable to separate the variability due to alignment from that due to variation in curve shape. For these 16 neurons we found that the proportion of variability due to the first principal component increased from 0·62 without alignment to 0·71 after alignment.

6. DISCUSSION

The distinction between the random-coefficient and hierarchical Gaussian process models is important. The normal random-coefficient model implies that the functions $f^i(t)$ are Gaussian processes conditionally on the knot set, while the hierarchical Gaussian process model takes them to be Gaussian processes marginally. The marginal Gaussian process assumption may seem more natural, in that conceptually the variability is between the curves rather than the somewhat artificially-invoked coefficients, but either approach may be reasonable and effective, and part of the purpose of this paper was to compare them. The random-coefficient model is likely to be most useful when the data are sparse, as for example in James et al. (2000). The hierarchical Gaussian process framework will be helpful when there are sufficient data per function that irregular variation may be estimated, but not so much that the estimation variability becomes negligible, relative to the variability across functions. To substantiate this point we performed additional simulations as in § 4, either decreasing or increasing μ so as to decrease or increase the signal-to-noise ratio. When μ was decreased by a factor of 5 we found the hierarchical Gaussian process model no longer to be superior to the random-coefficient model: the mean squared error values for the proportion of variance were 0·055 ($\pm$0·019) for the random-coefficient model and 0·066 ($\pm$0·016) for the hierarchical Gaussian process model, with 0·098 ($\pm$0·010) for naive functional data analysis. When μ was increased by a factor of 20 we found naive functional data analysis to be adequate: the mean squared error values for the proportion of variance were 0·0733 ($\pm$0·0260) for the random-coefficient model, 0·0694 ($\pm$0·0277) for the hierarchical Gaussian process model and 0·0815 ($\pm$0·0299) for naive functional data analysis.

In general, effects of curve estimation on the magnitude and direction of bias in the proportion of variance associated with the first principal component can be subtle and will depend on the relationship of the within-curve variance matrices S^i to the between-curve variance matrix V: it is possible to construct theoretical examples where the first eigenvalue is biased upwards but the bias in the proportion of variance is negligible or downwards. One explanation for the very large improvement seen here in hierarchical Gaussian process estimators compared to those of the naive method is that, in the neuronal setting we used, large estimation variability tends to occur in the peaks of the curves, which dominate the first principal component. Therefore, the within-curve estimation variability contributes quite substantially to the naive functional data analysis estimate of the first eigenvalue, exaggerating the extent to which the first principal component summarises the variability among the functions. We would expect this to be a fairly common situation, not limited to neurophysiology.

The procedure we have applied is easily implemented in two steps: first, the m separate datasets are smoothed, and then available software, such as BUGS, may be applied to fit the normal hierarchical model (6). The general framework of models (4)–(7) allows several elaborations we have not implemented. First, as we noted, importance reweighting will in some cases improve the normal approximation of (4) to the more accurate hierarchical model that would result from (1). Secondly, more complicated alignment schemes could be used. Thirdly, while we have been satisfied with the use of straightforward Bayesian estimation of V, including our choice of prior, in (6), it should be possible to devise improved procedures for choosing the grid $t_1, \ldots, t_p$ and then take advantage of the special covariance structure induced by functions.

Acknowledgement

This work is based in part on a portion of S. Behseta's Ph.D. dissertation under the supervision of R. E. Kass, and was partially supported by grants from the National Institutes of Health and the National Science Foundation. The authors are grateful for helpful comments from the referees.

References

Behseta, S., Matsuzaka, Y., Picard, N., Kass, R. E. & Strick, P. L. (2002). Muscle-like activity of M1 neurons during multi-joint movements. *Soc. Neurosci. Abstr.*, no. 61.13.

Daniels, M. J. & Kass, R. E. (1998). A note on first-stage approximation in two-stage hierarchical models. *Sankhyā* B **60**, 19–30.

Daniels, M. J. & Kass, R. E. (1999). Nonconjugate Bayesian estimation of covariance matrices and its use in hierarchical models. *J. Am. Statist. Assoc.* **94**, 1254–63.

Daniels, M. J. & Kass, R. E. (2001). Shrinkage estimators for covariance matrices. *Biometrics* **57**, 1173–84.

DiMatteo, I., Genovese, C. R. & Kass, R. E. (2001). Bayesian curve-fitting with free-knot splines. *Biometrika* **88**, 1055–73.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* **82**, 711–32.

Hansen, M. H. & Kooperberg, C. (2002). Spline adaptation in extended linear models (with Discussion). *Statist. Sci.* **17**, 2–51.

James, G. (2002). Generalized linear models with functional predictors. *J. R. Statist. Soc.* B **64**, 411–32.

James, G. M., Hastie, T. J. & Sugar, C. A. (2000). Principal component models for sparse functional data. *Biometrika* **87**, 587–602.

Kass, R. E. & Raftery, A. E. (1995). Bayes factors. *J. Am. Statist. Assoc.* **90**, 773–95.

Kass, R. & Wallstrom, G. (2002). Comment on 'Spline adaptation in extended linear models' by Mark H. Hansen and Charles Kooperberg. *Statist. Sci.* **17**, 24–9.

Kass, R. E. & Wasserman, L. A. (1995). A reference Bayesian test for nested hypotheses and its relationship to the Schwarz criterion. *J. Am. Statist. Assoc.* **90**, 928–34.

Kass, R. E., Ventura, V. & Cai, C. (2003). Statistical smoothing of neuronal data. *NETWORK: Computation in Neural Systems* **14**, 5–15.

Ke, C. & Wang, Y. (2001). Semiparametric nonlinear mixed-effects models and their applications (with Discussion). *J. Am. Statist. Assoc.* **96**, 1272–98.

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

Matsuzaka, Y., Picard, N. & Strick, P. L. (2001). Sequence learning in monkeys. *Soc. Neurosci. Abstr.*, no. 24.174.

Optican, L. & Richmond, B. (1987). Temporal encoding of two-dimensional patterns by single units in primate inferior temporal cortex. An information theoretic analysis. *J. Neurophysiol.* **57**, 162–78.

Pauler, D. K. (1998). The Schwarz criterion and related methods for normal linear models. *Biometrika* **85**, 13–27.

Ramsay, J. O. & Li, X. (1998). Curve registration. *J. R. Statist. Soc.* B **60**, 351–63.

Ramsay, J. O. & Silverman, B. W. (1997). *Functional Data Analysis.* New York: Springer.

RICE, J. & WU, C. (2001). Nonparametric mixed effects models for unequally sampled noisy curves. *Biometrics* **57**, 253–9.

SHI, M. S., WEISS, R. E. & TAYLOR, J. M. G. (1996). An analysis of paediatric CD4 counts for acquired immune deficiency syndrome using flexible random curves. *Appl. Statist.* **45**, 151–63.

SPIEGELHALTER, D. J., THOMAS, A., BEST, N. G. & GILKS, W. R. (1996). *BUGS: Bayesian inference using Gibbs sampling*, Version 0.5. Cambridge: MRC Biostatistics Unit.

VENTURA, V., CARTA, R., KASS, R. E., GETTNER, S. N. & OLSON, C. R. (2002). Statistical analysis of temporal evolution in single-neuron firing rates. *Biostatistics* **3**, 1–23.

WANG, K. & GASSER, T. (1997). Alignment of curves by dynamic time warping. *Ann. Statist.* **25**, 1251–76.

[*Received September* 2003. *Revised October* 2004]