Bayesian Curve Registration of Functional Data

by

Z. Zhong, A. Majumdar and R. L. Eubank

Arizona State University

July 15, 2008

Abstract. Functional data arise in numerous areas nowadays. When the functional responses evolve with respect to time, the subjects may experience events at different paces with the consequence that the sample curves are improperly aligned for inferential purposes. In particular, the sample mean function without alignment will fail to produce a satisfactory estimator of the true process mean function. In this article a new model for curve alignment or registration is developed from a Bayesian perspective. It incorporates nonparametric spline curve fitting methods with continuous Monte Carlo Markov chain (MCMC) techniques. The functional response curves are fit by nonparametric spline methods with their coefficients treated as random parameters. Similarly, the warping functions are modeled as random spline functions and random shift and amplitude coefficients are also included in the model formulation. An MCMC algorithm is created to estimate the parameters in the model. The performance of the proposed method is evaluated in an empirical study.

Key words: curve registration, functional data analysis, posterior inference, predictive distribution

1 Introduction

Functional data analysis is a relatively new area in statistics. It concerns the study of observations on a function space valued random variable. Thus, in contrast to classical statistical themes, each individual observation represents a function rather than simply a scalar or vector value at a particular point. Functional data arise in many different fields, including economics, biology and signal processing. Techniques for the analysis of functional data are described in Ramsay and Silverman (1997) while data analysis case studies are presented in Ramsay and Silverman (2002).

The standard summary measures that are used in functional data analysis include the mean and covariance functions. After functional data are collected, each subject can typically be represented

by a smooth function and these functions are, in turn, employed to construct summary measures. However, there are often systematic variations among those curves which have the consequence that direct averaging across curves will produce poor estimators of the true process average and covariance functions. In such cases the solution is to *register* or align the sample curves so that they only differ in amplitude and thereby produce a cross-sectional mean that gives a more satisfactory summary of the data. Without taking the phase variation into account, the cross-sectional mean always underestimates the amplitude of local maxima and overestimates the amplitude of local minima. This is problematic because local extrema are often the most important features of the process that is under study and their amplitude should be estimated accurately.

Various registration methods have been proposed in the literature for dealing with functionaln data. Most of this work involves the use of of warping functions. These are strictly monotone functions whose inverses transform observation time to a synchronized time scale where certain features on different curves occur simultaneously. A standard warping function model for functional data would assume that there an observed set of curve $x_1, \ldots, x_M$ stem from a model of the form

$$x_i(t) = \mu(h_i^{-1}(t)) + \epsilon_i(t), \qquad i = 1, \ldots, M$$

with $\mu(\cdot)$ the true mean function for the process, h_i^{-1} the inverse warping function for the ith subject and $\epsilon_i(\cdot)$ an error process. Registration for a sample of functional data can then be viewed as equivalent to estimating $h_1, \ldots, h_M$.

Sakoe and Chiba (1978) provide an early example of the use of warping functions to align two curves. An extension of their approach that could be employed with a sample of functional data was devloped by Gasser and Kneip (1995). Bayesian approaches to curve registration has been employed by McKeague (2005) and Alshabani, et al. (2007) for dealing with problems of signature recognition and analysis of human movement curves, respectively. More in line with our proposed methodology is the work of Telesca and Inoue (2008) that will be discussed more fully at the end of Section 2.

The continuous monotone registration method of Ramsay and Li (1996) is a popular registration technique that is available from the FDA package in R. This method derives from a model where

$$z(t) = x_i\left[h_i(t)\right] + \epsilon_i(t),$$

with $x_i(\cdot)$ an observed sample curve and $z(\cdot)$ a fixed function that provides a template for the

individual curves. The warping function is then estimated by the function h that minimizes

$$F_\lambda(z, x_i|h_i) = \int_0^{T_0} \{z(t) - x_i\,[h_i(t)]\}^2 dt + \lambda \int_0^{T_0} w_i^2(t)dt$$

with w_i representing the relative curvature: i.e., $w_i = D^2 h_i / D h_i$, with $D^2(\cdot)$ and $D(\cdot)$ being the second and first derivative, respectively. To implement the actual minimization the w_i are represented by linear combinations of B-spline basis functions. The choice of the function $z(\cdot)$ is problematic. In practice this is often taken to be the cross-sectional mean and the processes is applied in an iterative format to each of the sample curves while updating the cross-sectional mean at each step.

Our proposed approach has ties to the shape invariant models (SIM) proposed by Lawton, Sylvestre and Maggio (1972). In particular, we pursue a direction that is similar to that of Brumback and Lindstrom (2004) in a SIM context. Their model can be written as

$$x_{ij} = \theta_{i1}\mu(h^{-1}(t_{ij}, \phi_i), \beta) + \theta_{i4} + \epsilon_{ij}, \quad i = 1, \ldots, M, \quad j = 1, \ldots, n_i,$$

where μ is a common shape function dependent on parameters and h is a common parametric form for the individual warping functions. Here θ_{i1}, θ_{i4} and the ϕ_i are random, curve-specific quantities and the ϵ_{ij} are random errors. The h_i^{-1} were then modelled as monotone splines with random coefficients and μ was treated as a cubic spline. The resulting framework can be viewed as producing a nonlinear mixed effects model and estimation methodology was developed from that perspective.

Our goal in the remainder of this paper is development of an automatic, data driven method for registration of functional data. In the next section we construct a registration model from a Bayesian perspective. Using this framework we then derive the desired automatic estimation algorithm. Section 3 describes the results of a small empirical study that examines the performance of our approach relative to continuous monotone registration. We conclude in Section 4 with a summary of our principal results and a discussion of future areas of investigation.

2 Bayesian Approach

In this section we lay out our Bayesian framework for data registration. The basic premise is that we observe responses x_{ij} being obtained at time ordinates t_{ij} for the ith subject/experimental unit with $j = 1, \ldots, n_i$ representing the readings for a particular subject. The x_{ij} then satisfy

$$x_{ij} = a_{sh_i} + a_{sc_i}\mu\{h_i^{-1}(t_{ij})\} + \epsilon_{ij}, \quad i = 1, \ldots, M, \quad j = 1, \ldots, n_i, \tag{1}$$

where the ϵ_{ij} are normal random errors, a_{sh_i} and a_{sc_i} are shift and scale parameters, respectively, $\mu(\cdot)$ is a common mean function and $h_i(\cdot)$ is the warping function for the ith subject. To simplify notation we will use $\mathbf{x}_i$ and $\mathbf{t}_i$ to indicate the responses and t ordinates for the ith subject and let $N = \sum_{i=1}^{M} n_i$ in what follows. Then, for any function f we use $f(\mathbf{t}_i)$ to indicate an $n_i \times 1$ vector with elements $(f(t_{i1}), \ldots, f(t_{in_i}))$.

In the next subsection we give a detailed development of the model we will employ. The basic premise behind our formulation is relatively simple. The idea is to approximate all unknown functions by linear combinations of B-splines and the resulting set of coefficients are then estimated from the data. However, there are complications that arise due to the monotonicity restrictions that must be treated with special care in order to produce a satisfactory approach. Section 2.2 then discusses the specific Monte Carlo Markov chain algorithms that we will employ while Section 2.3 gives the details behind our estimation algorithm.

2.1 Model formulation

A fundamental tool for our approach is the B-spline basis that will be used for modeling. Following developments in de Boor (1978), we can give a recursive definition for these functions. Specifically, let $0 < \xi_1 < \ldots < \xi_k < 1$ be a set of of interior knots and define $2m$ additional knots as $\xi_{-(m-1)}, \cdots, \xi_{-1} = \xi_0 = 0$, $\xi_{k+1}, \ldots, \xi_{k+m} = T$ for some specified upper limit T. The B-splines of order m with knots at $\xi_1, \ldots, \xi_k$ are then defined recursively by

$$B_{i,m}(t) = \frac{t - \xi_i}{\xi_{i+m-1} - \xi_i} B_{i,m-1}(t) + \frac{\xi_{i+m} - t}{\xi_{i+m} - \xi_{i+1}} B_{i+1,m-1}(t), \tag{2}$$

for $i = -(m-1), \ldots, k$, with

$$B_{i,1}(t) = \begin{cases} 1, & t \in [\xi_i, \xi_{i+1}), \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

being the initial step in the recursion. In using the recursion formula we need to employ the property that a B-spline of order r corresponding to an $r+1$ coincident knot is zero. With this convention, equation (2) provides both a definition and computational formula for the basis functions.

We will represent the warping functions as linear combinations of B-splines. Since warping functions will be monotone increasing, we need to enforce this property on the corresponding B-splines. Specifically, since the derivative of a B-spline is also a B-spline of one lower degree,

monotonicity is obtained through an inequality constraint on the coefficients. Accordingly, we now take the ith warping function to be

$$h_i^{-1}(\mathbf{t}_i) = \mathbf{U}(\mathbf{t}_i)\boldsymbol{\phi}_i, \tag{4}$$

for $i = 1, \ldots, M$, with $\mathbf{U}(\mathbf{t}_i)$ representing the quadratic B-spline basis with D uniformly spaced knots evaluated at time $\mathbf{t}_i$ and $\boldsymbol{\phi}_i$ the corresponding coefficient vector whose elements must be strictly increasing. So, $\mathbf{U}(\mathbf{t}_i)$ is a $n_i \times (D+3)$ matrix and $\boldsymbol{\phi}_i$ a vector of dimension $D+3$.

Similar to the developments for warping functions, we model the common mean function μ with a cubic spline: i.e.,

$$\mu(\cdot) = \mathbf{B}(\cdot)^T\beta,$$

with $\mathbf{B}(\cdot)$ the vector of B-spline basis functions for the common shape function and β a corresponding coefficient vector. Here $\mathbf{B}(\cdot)$ is a (Q+4)-vector with Q being the number of knots. The knots are taken to be equally spaced and β is a vector of dimension $Q+4$.

Upon inserting the models for the warping and mean function into (1), we obtain

$$\mathbf{x} = \text{diag}\{\mathbf{1_{n_1}}, \ldots, \mathbf{1_{n_M}}\}\mathbf{a_{sh}} + \text{diag}\{\mathbf{B}[\mathbf{U}(\mathbf{t}_1)\boldsymbol{\phi}_1]\beta, \ldots, \mathbf{B}[\mathbf{U}(\mathbf{t}_M)\boldsymbol{\phi}_M]\beta\}\mathbf{a_{sc}} + \epsilon, \tag{5}$$

where

$$\mathbf{x} = (x_{1,1}, \ldots, x_{1,n_1}, \ldots, x_{M,1}, \ldots, x_{M,n_M})^T,$$

$\mathbf{a_{sh}} = (a_{sh_1}, \ldots, a_{sh_M})^T$,$\mathbf{a_{sc}} = (a_{sc_1}, \ldots, a_{sc_M})^T$, $\text{diag}\{A_1, \ldots, A_M\}$ denotes a diagonal matrix with diagonal elements being matrices $A_1, \ldots, A_M$ and

$$\epsilon = (\epsilon_{1,1}, \ldots, \epsilon_{1,n_1}, \ldots, \epsilon_{M,1}, \ldots, \epsilon_{M,n_M})^T \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$$

with $\mathbf{I}$ the $N \times N$ identity matrix. For each observation x_{ij}, we can write (5) as

$$x_{ij} = a_{sh_i} + a_{sc_i}\{\mathbf{B}[\mathbf{U}(\mathbf{t}_i)\boldsymbol{\phi}_i]\}_j\beta + \epsilon_{ij}, \quad i = 1, \ldots, M, j = 1, \ldots, n_i,$$

with $\{\mathbf{B}[\mathbf{U}(\mathbf{t}_i)\boldsymbol{\phi}_i]\}_j$ being the jth row of $\mathbf{B}[\mathbf{U}(\mathbf{t}_i)\boldsymbol{\phi}_i]$. For easy interpretation, we will denote $\{\mathbf{B}[\mathbf{U}(\mathbf{t}_i)\boldsymbol{\phi}_i]\}_j$ as $\mathbf{g}_{ij}^T$ in the future. The qth column of $\mathbf{g}_{ij}^T$ is the qth B-spline basis function evaluated at $\mathbf{U}(\mathbf{t}_i)\boldsymbol{\phi}_i$: i.e.,

$$(\mathbf{g}_{ij}^T)_q = B_q[\mathbf{U}(\mathbf{t}_i)\boldsymbol{\phi}_i], \tag{6}$$

With this notational convention, our model can be expressed as

$$x_{ij} = a_{sh_i} + a_{sc_i}\mathbf{g}_{ij}^T\beta + \epsilon_{ij}, \quad i = 1, \ldots, M, \ j = 1, \ldots, n_i$$

or, in matrix form, as

$$\mathbf{x} = \text{diag}(\mathbf{1_{n_1}}, \ldots, \mathbf{1_{n_M}})\mathbf{a_{sh}} + \text{diag}(\mathbf{g}^T\beta)\mathbf{a_{sc}} + \epsilon$$

with $\text{diag}(\mathbf{g}^T\beta) = \text{diag}\{\mathbf{B}[\mathbf{U}(\mathbf{t}_1)\boldsymbol{\phi}_1]\beta, \ldots, \mathbf{B}[\mathbf{U}(\mathbf{t}_M)\boldsymbol{\phi}_M]\beta\}$.

Before we begin our analysis, we put some restrictions on the model. Without loss we restrict all the curves to have a common start and end time: i.e., $t_{i,1} = 0$ and $t_{i,n_i} = T$ for all $i = 1, \ldots, M$. Since we want the real start/end time and the warped start/end time to coincide, we then restrict the time transformations to match at both the start points and the endpoints. That is, $h_i(t_1) = 0$ and $h_i(t_{n_i}) = T$ for all $i = 1, \ldots, M$. Hence, in our setting we fix $\phi_{i,1} = 0$ and $\phi_{i,D+3} = T$. This allows us to focus on the time differences among curves and makes the estimation less complicated.

As noted above, to ensure that the warping function is monotone increasing, we constrain its coefficients $\boldsymbol{\phi}_i = (\phi_{i,1}, \ldots, \phi_{i,D+3})^T$ to be strictly increasing in that $\phi_{i,1} < \phi_{i,2} < \cdots < \phi_{i,D+3}$. Thus, let $\Phi = (\phi_1, \ldots, \phi_M)^T$. As in Brumback and Lindstrom (2004) the monotone increasing constraint on Φ will then be enforced by modeling Φ with a Jupp inverse transformation that will be discussed subsequently.

For identifiability reasons, we force the average time transformation to be fixed at the identity transformation. That is, the mean of the $\boldsymbol{\phi}_i$ is the vector of identity spline coefficient. Specifically, the parameters $\boldsymbol{\phi}_i$ are constrained so that

$$\frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{D+3} U_j(t)\phi_{i,j} = t$$

This is equivalent to

$$\sum_{j=1}^{D+3} U_j(t)\{\frac{1}{M}\sum_{i=1}^{M}\phi_{i,j}\} = t \tag{7}$$

Let

$$s_j = \frac{1}{M}\sum_{i=1}^{M}\phi_{i,j}$$

and define

$$\mathbf{s} = (s_1, \ldots, s_{D+3})^T.$$

This vector can then be obtained through the recursion

$$\begin{aligned} s_1 &= 0, \\ s_{q+1} &= \frac{\gamma_{q+r} - \gamma_{q+1}}{r-1} + s_q, \quad q = 1, \ldots, D+2. \end{aligned} \tag{8}$$

In (8), γ_i, $i = 1, \ldots, D+6$ are the knots with domain [0, T]. There are D interior knots and 2 boundary knots. The boundary knots are repeated 3 times for the purpose of applying the B-spline functions. That is, $(\gamma_1, \ldots, \gamma_{D+6})^T = (0, 0, 0, \gamma_4, \ldots, \gamma_{D+3}, T, T, T)^T$ and $0 < \gamma_4 < \gamma_{D+2} < \cdots < \gamma_{D+3} < T$. In what follows we will use $\mathbf{S}$ to denote the $M \times (D+3)$ matrix with all rows equal to $\mathbf{s}^T$.

Application of the Jupp transformation can be accomplished as follows. Let $\mathbf{v}$ be an ordered vector of length Q: that is, $\mathbf{v} = (v_1, \ldots, v_Q)^T$ with $v_1 < v_2 < \cdots < v_Q$. Define the Jupp transformation as $\text{Jupp}(\mathbf{v}) = \mathbf{w} = (w_1, \ldots, w_Q)^T$, where

$$w_q = \begin{cases} v_q & q = 1, Q, \\ \log(\frac{v_{q+1} - v_q}{v_q - v_{q-1}}) & q = 2, \ldots, Q-1. \end{cases}$$

It follows from this that the inverse of the Jupp function $\mathbf{w} = \text{Jupp}^{-1}(\mathbf{v})$ will be a vector of increasing elements with $w_1 = v_1$ and $w_Q = v_Q$. It can be calculated via the following steps:

(i) Define $a_1 = 1$, $a_q = \exp(\sum_{k=2}^{q} v_k)$, for $q = 2, \ldots, Q-1$;

(ii) Let $c_k = (v_Q - v_1)a_k / (\sum_{q=1}^{Q-1} a_q)$, for $k = 1, \ldots, Q-1$;

(iii) $w_1 = v_1$, $w_Q = v_Q$, and $w_j = v_1 + \sum_{k=1}^{j-1} c_k$, for $j = 2, \ldots, Q-1$.

It is easy to verify that for a matrix A we will have $\text{Jupp}(\text{Jupp}^{-1}(A)) = A$.

We now define the Φ matrix by

$$\Phi = \text{Jupp}^{-1}([\mathbf{0} \;\; \mathbf{Z}^T\mathbf{P} \;\; \mathbf{0}] + \text{Jupp}(\mathbf{S})). \tag{9}$$

with $\mathbf{0}$ a vector whose elements are all equal to 0, $\mathbf{P}$ a matrix of unconstrained parameters and $\mathbf{Z}$ an $(M-1) \times M$, full row-rank matrix determined by $\mathbf{Z}^T\mathbf{Z} = \mathbf{W}$ with

$$\mathbf{W} \equiv -\frac{1}{M-1}\mathbf{J}_M + \frac{M}{M-1}\mathbf{I}_M.$$

Here $\mathbf{J}_M$ is an $M \times M$ matrix with all entries 1, and $\mathbf{I}_M$ is an $M \times M$ identity matrix.

Since we know that the first column of Φ is equal to $(0, \ldots, 0)^T$ and the last column of Φ is equal to $(T, \ldots, T)^T$, there are only $D+1$ columns of "free parameters" in Φ. Since $\sum_{i=1}^{M} \phi_i \doteq M\mathbf{s}$, the last row of Φ, Φ_M can be expressed in terms of the first $M-1$ rows. So that

$$\Phi_M \doteq M\mathbf{s} - \sum_{i=1}^{M-1} \phi_i$$

The row dimension of Φ is therefore equal to $M-1$ and we can define a $(M-1)\times(D+1)$ matrix, $\tilde{\Phi}$, of free parameters by taking the Φ matrix and deleting its first column, last column and last row to obtain

$$\tilde{\Phi} = \begin{pmatrix} \phi_{12} & \cdots & \phi_{1,D+2} \\ \vdots & \ddots & \vdots \\ \phi_{M-1,2} & \cdots & \phi_{M-1,D+2} \end{pmatrix}. \tag{10}$$

The matrix in (10) will be the focus of subsequent developments using MCMC technology.

We will need to work with transformations between $\tilde{\Phi}$ and $\mathbf{P}$. To facilitate that, define $\tilde{Z}$ as the matrix Z by deleting its last column. So $\tilde{Z}$ is a $(M-1)\times(M-1)$ square matrix of full rank. Similarly, let $\tilde{S}$ be a $(M-1)\times(D+1)$ matrix obtained by deleting the first and last column and the last row of $\mathbf{S}$. Since the matrix $\tilde{Z}$ is invertible, we can now transform between the variable $\mathbf{P}$ and variable $\tilde{\Phi}$. Specifically, $\mathbf{P}$ can be determined in terms of $\tilde{\Phi}$ from the relation

$$\mathbf{P} = (\tilde{Z}^T)^{-1}(\mathrm{Jupp}(\tilde{\Phi}) - \mathrm{Jupp}(\tilde{\mathbf{S}}))$$

After the transformations on the parameters, we now have three parameters β, $\mathbf{P}$ and σ^2 to estimate in our model. Since our samples are chosen randomly and the effects come from some random variables, it is reasonable to model these parameters as random effects. Accordingly, we propose to estimate them from a Bayesian perspective and will use Monte Carlo Markov Chain methods to realize the values of the parameters.

2.2 MCMC Algorithm

Two of the most commonly used classes of MCMC algorithms are the Gibbs Sampler and Metropolis Hastings algorithm (e.g., sections 10 and 7 of Robert and Casella 2004). The Gibbs sampler can be used when the posterior distribution of the parameter of interest is of some specific form we know such as a normal distribution. When this is not the case the Metropolis- Hastings algorithm comes into play. The Metropolis-Hastings algorithm is used to generate a Markov chain that uses an acceptance/rejection rule to converge to the specified target distribution. It can be shown that the chain generated by the Metropolis-Hastings algorithm is a Markov chain and converges to the stationary distribution.

We will also apply the Langevin-Hastings Hybrid algorithm in our work. The Langevin-Hastings (Møller et al. 1998) algorithm revises the Metropolis-Hastings algorithm using a truncated proposal density $Q(\gamma|\gamma^*)$ for γ a parameter vector. Usually the truncated proposal density is chosen to be

a normal distribution wherein

$$Q(\gamma^*|\gamma) \sim \mathcal{N}\left(\gamma + \frac{\delta}{2} \bigtriangledown (\gamma)^{trunc}, \delta \mathbf{I}\right),$$

where δ is selected to make the acceptance ratio within the range of 0.20 and 0.60 and

$$\bigtriangledown(\gamma) = \frac{\partial}{\partial \gamma} \log(f(\gamma|\mathbf{x}))$$

with $f(\cdot)$ the likelihood function. The Langevin-Hastings Hybrid algorithm is a truncated version of the Langevin-Hastings algorithm obtained by replacing $\bigtriangledown(\gamma)$ in the proposal distribution by

$$\bigtriangledown(\gamma)^{trunc} = -\gamma + \max(H, \frac{\partial}{\partial \gamma} \log(l(\mathbf{x}|\gamma))$$

with $H > 0$ a user-specified parameter (Møller and Waagepetersen, 2004, p. 192). The Langevin-Hastings Hybrid algorithm has an advantage in terms of its convergence performance. The posterior samples drawn according to the Langevin-Hastings Hybrid algorithm converge faster than those drawn according to the Metropolis-Hastings algorithm.

2.3 MCMC Estimation of Model Parameters

We will now apply both the Gibbs sampler and the Langevin-Hastings Hybrid algorithm to generate a sequence of samples from the joint posterior distribution of the random variables β, σ^2, $\mathbf{a_{sh}}$, $\mathbf{a_{sc}}$ and $\mathbf{P}$ that appear in our model. The first step in the process is to derive a likelihood function for the parameters we are interested in and put priors on each parameter in order to find the posterior distribution. Thus, let $\Theta = (\beta^T, \sigma^2, \mathbf{a_{sh}}^T, \mathbf{a_{sc}}^T, vec(\mathbf{P})^T, \Sigma_{vec(\mathbf{P})})^T$, where $vec(\mathbf{P})$ stacks the columns of $\mathbf{P}^T$ in a single vector form. We then consider the likelihood of our model to be that of a multivariate normal having the form

$$l(\mathbf{x}|\Theta) = \frac{1}{(2\pi)^{N/2}\sigma^{N/2}} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{M}\sum_{j=1}^{n_i}(x_{ij} - a_{sh_i} - a_{sc_i}\mathbf{g}_{ij}^T\beta)^2\right\},$$

with $N = \sum_{i=1}^{M} n_i$ the total number of observations.

We choose an improper prior for the coefficient vector β with the prior density $p(\beta)$ proportional to the uniform function. The posterior distribution for β is then

$$p(\beta|\mathbf{x}) \propto \exp\left\{-\frac{1}{2}(\beta - \boldsymbol{\mu}_\beta)^T \Sigma_\beta^{-1} (\beta - \boldsymbol{\mu}_\beta)\right\}.$$

with

$$\begin{aligned}\boldsymbol{\mu}_\beta &= \frac{1}{\sigma^2}\Sigma_\beta \sum_{i=1}^{M}\sum_{j=1}^{n_i}(x_{ij}a_{sc_i}\mathbf{g}_{ij}^T - a_{sh_i}a_{sc_i}\mathbf{g}_{ij}^T)^T, \\ \Sigma_\beta &= \sigma^2\left[\sum_{i=1}^{M}\sum_{j=1}^{n_i}(a_{sc_i}^2\mathbf{g}_{ij}^T\mathbf{g}_{ij})\right]^{-1}\end{aligned}$$

We next choose the prior for σ^2 to be an improper prior proportional to $1/\sigma^2$. The posterior distribution of σ^2 is then found to be that of an inverse gamma with parameters

$$\alpha_{\sigma^2} = \frac{N}{2},$$

and

$$\beta_{\sigma^2} = \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{n_i}(x_{ij} - a_{sh_i} - a_{sc_i}\mathbf{g}_{ij}^T\beta)^2 + \frac{vec(\mathbf{P})^T\Sigma_{vec(\mathbf{P})}^{-1}vec(\mathbf{P})}{2}.$$

In the appendix we show that our prior specifications leads to a proper joint distributions for all the model parameters.

We model the shift effects as being independent with a_{sh_i} having a $\mathcal{N}(0,\sigma_{sh}^2)$ distribution for $i = 1,\ldots,M$, where σ_{sh}^2 is a hyper parameter. Then, the posterior distribution of a_{sh_i} follows a normal distribution with mean

$$\frac{\sigma_{sh}^2}{n_i\sigma_{sh}^2 + \sigma^2}\sum_{j=1}^{n_i}(x_{ij} - a_{sc_i}\mathbf{g}_{ij}^T\beta)$$

and variance

$$\sigma^2\sigma_{sh}^2/(n_i\sigma_{sh}^2 + \sigma^2).$$

The hyper parameter σ_{sh}^2 is given an inverse Gamma distribution with shape parameter $\alpha_{\sigma_{sh}^2} = 2$ and scale parameter $\beta_{\sigma_{sh}^2} = 0.5$. These choices for the shape and scale parameters make the hyper-prior distribution diffuse. The posterior distribution of σ_{sh}^2 is again an inverse Gamma with shape parameter $M/2 + 2$ and scale parameter $\sum_{i=1}^{M} a_{sh_i}^2/2 + 0.5$.

In our setting, the scale factors a_{sc_i} are all positive. To ensure this property, we will perform the transformation $a_{sc_i} = e^{\tilde{\alpha}_i}$, where $\tilde{\alpha}_i = \sigma_{sc}\alpha_i$ with σ_{sc}^2 being a hyper parameter and α_i being unconstrained random parameters. The hyper parameter σ_{sc}^2 is given an inverse Gamma distribution with shape parameter $\alpha_{\sigma_{sc}^2} = 2$ and scale parameter $\beta_{\sigma_{sc}^2} = 0.5$. The posterior distribution of σ_{sc}^2 is again an inverse Gamma with shape parameter $M/2+2$ and scale parameter $\sum_{i=1}^{M}\tilde{\alpha}_i^2/2+0.5$.

If we now take $\boldsymbol{\alpha} = (\alpha_1,\ldots,\alpha_M)^T$, the prior distribution of $\boldsymbol{\alpha}$, or $\pi(\boldsymbol{\alpha})$, is chosen to be $\mathcal{N}(\mathbf{0},\mathbf{I})$ with $\mathbf{I}$ being the $M \times M$ identity matrix. Because the posterior distribution of $\boldsymbol{\alpha}$ is not of a form

that allows us to easily generate random ordinates, we will apply the Langevin-Hastings Hybrid algorithm to update the posterior samples of $\boldsymbol{\alpha}$. Thus, let

$$\begin{aligned}\nabla(\boldsymbol{\alpha}) &= \frac{\partial}{\partial \boldsymbol{\alpha}} \log[\pi(\boldsymbol{\alpha}) l(\mathbf{x}|\boldsymbol{\alpha})] \\ &= -\boldsymbol{\alpha} + \frac{1}{\sigma^2}\Big\{\sum_{j=1}^{n_i}(x_{ij} - a_{sh_i} - e^{\sigma_{sc}\alpha_i}\mathbf{g}_{ij}^T\beta)\sigma_{sc}e^{\sigma_{sc}\alpha_i}\mathbf{g}_{ij}^T\beta\Big\}_{i=1}^{M},\end{aligned} \tag{11}$$

where $\{f_i(\cdot)\}_{i=1}^M$ denotes a vector of size M with ith element $f_i(\cdot)$. The proposal density for $\boldsymbol{\alpha}^*$ is then given by

$$p(\boldsymbol{\alpha}^*|\boldsymbol{\alpha}) = \mathcal{N}\left(\boldsymbol{\alpha} + \frac{\delta_1}{2}\nabla(\boldsymbol{\alpha})^{trunc}, \delta_1\mathbf{I}\right) \tag{12}$$

The updated Langevin-Hastings Hybrid algorithm for $\boldsymbol{\alpha}$ works as follows.

(i) Generate $\boldsymbol{\alpha}^*$ according to (12), where δ_1 is chosen to make the acceptance ratio within the range of 0.20 to 0.60.

(ii) Calculate

$$a_1 = \frac{\pi(\boldsymbol{\alpha}^*)}{\pi(\boldsymbol{\alpha})}\frac{l(\mathbf{x}|\boldsymbol{\alpha}^*)}{l(\mathbf{x}|\boldsymbol{\alpha})}\frac{p(\boldsymbol{\alpha}|\boldsymbol{\alpha}^*)}{p(\boldsymbol{\alpha}^*|\boldsymbol{\alpha})}, \tag{13}$$

where

$$\frac{\pi(\boldsymbol{\alpha}^*)}{\pi(\boldsymbol{\alpha})} = \exp\left\{-\frac{\boldsymbol{\alpha}^{*T}\boldsymbol{\alpha}^*}{2} + \frac{\boldsymbol{\alpha}^T\boldsymbol{\alpha}}{2}\right\}, \tag{14}$$

$$\begin{aligned}\frac{l(\mathbf{x}|\boldsymbol{\alpha}^*)}{l(\mathbf{x}|\boldsymbol{\alpha})} = \exp\Big\{ &-\frac{1}{2\sigma^2}[2(\mathrm{diag}(\mathbf{1}_{n_1},\ldots,\mathbf{1}_{n_M})\mathbf{a}_{sh})^T\mathrm{diag}(\mathbf{g}^T\beta)(e^{\sigma_{sc}\boldsymbol{\alpha}^*} - e^{\sigma_{sc}\boldsymbol{\alpha}}) \\ &-2\mathbf{x}^T\mathrm{diag}(\mathbf{g}^T\beta)(e^{\sigma_{sc}\boldsymbol{\alpha}^*} - e^{\sigma_{sc}\boldsymbol{\alpha}}) + e^{\sigma_{sc}\boldsymbol{\alpha}^{*T}}(\mathrm{diag}(\mathbf{g}^T\beta))^T diag(\mathbf{g}^T\beta)e^{\sigma_{sc}\boldsymbol{\alpha}^*} \\ &-e^{\sigma_{sc}\boldsymbol{\alpha}^T}(\mathrm{diag}(\mathbf{g}^T\beta))^T\mathrm{diag}(\mathbf{g}^T\beta)e^{\sigma_{sc}\boldsymbol{\alpha}}]\Big\},\end{aligned} \tag{15}$$

and

$$\begin{aligned}\frac{p(\boldsymbol{\alpha}|\boldsymbol{\alpha}^*)}{p(\boldsymbol{\alpha}^*|\boldsymbol{\alpha})} = \exp\Big\{ &-\frac{1}{2\delta_1}[(\boldsymbol{\alpha} - \boldsymbol{\alpha}^* - \frac{1}{2}\delta_1 \bigtriangledown (\boldsymbol{\alpha}^*))^T(\boldsymbol{\alpha} - \boldsymbol{\alpha}^* - \frac{1}{2}\delta_1 \bigtriangledown (\boldsymbol{\alpha}^*)) \\ &-(\boldsymbol{\alpha}^* - \boldsymbol{\alpha} - \frac{1}{2}\delta_1 \bigtriangledown (\boldsymbol{\alpha}))^T(\boldsymbol{\alpha}^* - \boldsymbol{\alpha} - \frac{1}{2}\delta_1 \bigtriangledown (\boldsymbol{\alpha}))]\Big\}.\end{aligned} \tag{16}$$

(iii) For $t = 1, 2, \ldots, n_{iter}$, update $\boldsymbol{\alpha}^{t+1} = \boldsymbol{\alpha}^*$ if a_1 in step 2 is greater than 1. Otherwise, generate a sample u from the uniform distribution on $[0, 1]$. Compare the value of u with that of a_1. If $u < a_1$, then let $\boldsymbol{\alpha}^{t+1} = \boldsymbol{\alpha}^*$. Otherwise, keep the value of $\boldsymbol{\alpha}^t$: that is, $\boldsymbol{\alpha}^{t+1} = \boldsymbol{\alpha}^t$.

We also choose the prior for the random effect $vec(\mathbf{P})$ to be that of a normal density with mean $\mathbf{0}$ and variance covariance matrix $\sigma^2\Sigma_{vec(\mathbf{P})}$. The hyper-parameter $\Sigma_{vec(\mathbf{P})}$ has an inverse Wishart distribution with degrees of freedom ν and scale matrix Ψ, which is obtained by deleting the first and last columns and the first and last rows of $\tilde{\Psi}$, with

$$\tilde{\Psi} = \lambda_0 \left\{ \int_0^1 \mathbf{U}''(t)\mathbf{U}''(t)^T dt \right\}, \tag{17}$$

where $\mathbf{U}''$ is the second derivative of the vector $\mathbf{U}$ in equation (4) and λ_0 is some user specified small value. The posterior distribution of $\Sigma_{vec(\mathbf{P})}$ is given by an inverse Wishart distribution with parameters $\nu + (M-1)(D+1) + 1$ and scale matrix $\tilde{\tilde{\Psi}} = (1/\sigma^2)vec(\mathbf{P})vec(\mathbf{P})^T + \Psi$.

The posterior distribution of $vec(\mathbf{P})$ does not have a standard form. As a result we will apply the Langevin-Hasting's algorithm to update $vec(\mathbf{P})$ in the MCMC iteration.

To accoplish this, first write $vec(\mathbf{P}) = \sigma\Sigma^{1/2}_{vec(\mathbf{P})}\boldsymbol{\gamma}$, so that $\boldsymbol{\gamma}$ is a $(M-1)(D+1) \times 1$ vector, which has a prior distribution that is multivariate normal with mean $\mathbf{0}$ and identity variance matrix. Then, take the proposal distribution of $\boldsymbol{\gamma}^*$ to be $p(\boldsymbol{\gamma}^*|\boldsymbol{\gamma}) = \mathcal{N}\left(\boldsymbol{\gamma} + \frac{\delta_2}{2}\triangledown(\boldsymbol{\gamma})^{\text{trunc}}, \delta_2\mathbf{I}\right)$, with

$$\begin{aligned} \triangledown(\boldsymbol{\gamma}) &= \frac{\partial}{\partial\boldsymbol{\gamma}}\log(f(\mathbf{x}|\boldsymbol{\gamma})) \\ &= \frac{d}{d\boldsymbol{\gamma}}\log\pi(\boldsymbol{\gamma})\frac{d}{d\boldsymbol{\gamma}}\log l(\mathbf{x}|\boldsymbol{\gamma}) \\ &= -\boldsymbol{\gamma} + \tilde{\triangledown}(\boldsymbol{\gamma}), \end{aligned} \tag{18}$$

where $\pi(\boldsymbol{\gamma})$ and $l(\mathbf{x}|\boldsymbol{\gamma})$ are, respectively, the prior for $\boldsymbol{\gamma}$ and the likelihood of the model with respect to $\boldsymbol{\gamma}$. We write $\frac{d}{d\boldsymbol{\gamma}}\log l(\mathbf{x}|\boldsymbol{\gamma})$ as $\tilde{\triangledown}(\boldsymbol{\gamma})$ in equation (18) and observe that it can be calculated through the relation

$$\begin{aligned} \tilde{\triangledown}(\boldsymbol{\gamma}) &= \sigma\Sigma^{1/2}_{vec(\mathbf{P})}\frac{d}{d(vec(\mathbf{P}))}\log l(\mathbf{x}|vec(\mathbf{P})) \\ &= \sigma\Sigma^{1/2}_{vec(\mathbf{P})}|J|^{-1}\tilde{\triangledown}(vec(\tilde{\Phi})). \end{aligned} \tag{19}$$

where $vec(\tilde{\Phi})$ stacks the columns of $\tilde{\Phi}$ into a single vector and J is the Jacobian matrix.

Let us take a further look at the Jacobian matrix. By the nature of the Jupp transformation, J is a $(M-1) \times (M-1)$ block tri-diagonal matrix where a particular block J_i has element $(J_i)_{lk}$,

$l, k = 1, \ldots, D+1$ in its lth row and kth column with

$$(J_i)_{lk} = \begin{cases} \frac{\phi_{il} - \phi_{i(l+2)}}{(\phi_{i(l+2)} - \phi_{i(l+1)})(\phi_{i(l+1)} - \phi_{il})}, & \text{if } l = k, \\ \frac{1}{\phi_{i(l+2)} - \phi_{i(l+1)}}, & \text{if } k = l+1, \\ 0, & \text{if } k > l+1, \\ (J_i)_{kl}, & \text{if } l > k. \end{cases} \tag{20}$$

The quantity $\tilde{\nabla}(vec(\tilde{\Phi}))$ is calculated via the relation

$$\begin{aligned} \tilde{\nabla}(vec(\tilde{\Phi})) &= \frac{\partial}{\partial(vec(\tilde{\Phi}))}(\log(\mathbf{x}|vec(\tilde{\Phi}))) \\ &= \frac{\partial}{\partial(vec(\tilde{\Phi}))}\left[\frac{-\sum_{i=1}^{M}\sum_{j=1}^{n_i}(x_{ij} - a_{sh_i} - a_{sc_i}\mathbf{g}_{ij}^T\beta)^2}{2\sigma^2}\right] \\ &= \frac{\sum_{j=1}^{n_i}(x_{ij} - a_{sh_i} - a_{sc_i}\mathbf{g}_{ij}^T\beta)a_{sc_i}(\mathbf{g}_{ij}^T)'\beta}{\sigma^2} \end{aligned} \tag{21}$$

with $\mathbf{g}_{ij}^T$ a function of $vec(\tilde{\Phi})$ through equation (6) and $(\mathbf{g}_{ij}^T)'$ being the partial derivative of $\mathbf{g}_{ij}^T$ with respect to $vec(\tilde{\Phi})$.

The updated Langevin-Hastings Hybrid algorithm for $\boldsymbol{\gamma}$ now works as follows:

(i) Generate $\boldsymbol{\gamma}^*$ according to $\mathcal{N}(\boldsymbol{\gamma} + \frac{\delta_2}{2}\nabla(\boldsymbol{\gamma})^{\text{trunc}}, \delta_2\mathbf{I})$, where δ_2 is chosen to make the acceptance ratio within the range of 0.20 and 0.60.

(ii) Calculate

$$a_2 = \frac{\pi(\boldsymbol{\gamma}^*)}{\pi(\boldsymbol{\gamma})}\frac{l(\mathbf{x}|\boldsymbol{\gamma}^*)}{l(\mathbf{x}|\boldsymbol{\gamma})}\frac{p(\boldsymbol{\gamma}|\boldsymbol{\gamma}^*)}{p(\boldsymbol{\gamma}^*|\boldsymbol{\gamma})}, \tag{22}$$

where

$$\frac{\pi(\boldsymbol{\gamma}^*)}{\pi(\boldsymbol{\gamma})} = \exp\left\{-\frac{\boldsymbol{\gamma}^{*T}\boldsymbol{\gamma}^*}{2} + \frac{\boldsymbol{\gamma}^T\boldsymbol{\gamma}}{2}\right\}, \tag{23}$$

$$\begin{aligned} \frac{l(\mathbf{x}|\boldsymbol{\gamma}^*)}{l(\mathbf{x}|\boldsymbol{\gamma})} = \exp\Big\{ &-\frac{1}{2\sigma^2}\{\sum_{i=1}^{M}\sum_{j=1}^{n_i}[-2x_{ij}a_{sc_i}(\mathbf{g}_{ij}^{*T} - \mathbf{g}_{ij}^T)\beta \\ &+2a_{sh_i}a_{sc_i}(\mathbf{g}_{ij}^{*T} - \mathbf{g}_{ij}^T)\beta + a_{sc_i}^2\beta^T(\mathbf{g}_{ij}^*\mathbf{g}_{ij}^{*T} - \mathbf{g}_{ij}\mathbf{g}_{ij}^T)\beta]\}\Big\}, \end{aligned} \tag{24}$$

and

$$\begin{aligned} \frac{p(\boldsymbol{\gamma}|\boldsymbol{\gamma}^*)}{p(\boldsymbol{\gamma}^*|\boldsymbol{\gamma})} = \exp\Big\{ &-\frac{1}{2\delta_2}[(\boldsymbol{\gamma} - \boldsymbol{\gamma}^* - \frac{1}{2}\delta_2\nabla(\boldsymbol{\gamma}^*))^T(\boldsymbol{\gamma} - \boldsymbol{\gamma}^* - \frac{1}{2}\delta_2\nabla(\boldsymbol{\gamma}^*)) \\ &-(\boldsymbol{\gamma}^* - \boldsymbol{\gamma} - \frac{1}{2}\delta_2\nabla(\boldsymbol{\gamma}))^T(\boldsymbol{\gamma}^* - \boldsymbol{\gamma} - \frac{1}{2}\delta_2\nabla(\boldsymbol{\gamma}))]\Big\}. \end{aligned} \tag{25}$$

(iii) For $t = 1, 2, \ldots, n_{iter}$, update $\boldsymbol{\gamma}^{t+1} = \boldsymbol{\gamma}^*$ if a_2 in step 2 is greater than 1. Otherwise, generate a sample u from the uniform distribution on $[0, 1]$. Compare the value of u with that of a_2. If $u < a_2$, then let $\boldsymbol{\gamma}^{t+1} = \boldsymbol{\gamma}^*$. Otherwise, keep the value of $\boldsymbol{\gamma}^t$: that is, $\boldsymbol{\gamma}^{t+1} = \boldsymbol{\gamma}^t$.

Combining all of our estimation steps, we can now draw a sequence of samples for

$$\Theta = (\beta^T, \sigma^2, \mathbf{a_{sh}}^T, \sigma^2_{sh}, \mathbf{a_{sc}}^T, \sigma^2_{sc}, vec(\mathbf{P})^T, \Sigma_{vec(\mathbf{P})})^T.$$

The components of β, σ^2, $\mathbf{a_{sh}}$, σ^2_{sh}, σ^2_{sc}, $\Sigma_{vec(\mathbf{P})}$ are updated by a Gibbs sampler algorithm, while the component of $\mathbf{a_{sc}}$ and $vec(\mathbf{P})$ are updated by the Langevin-Hastings Hybrid algorithm. The procedure is as follows.

(i) Assign starting values for

$$\Theta^{(0)} = (\beta^{(0)T}, \sigma^{2(0)}, \mathbf{a_{sh}}^{(0)T}, \sigma^{2(0)}_{sh}, \mathbf{a_{sc}}^{(0)T}, \sigma^{2(0)}_{sc}, vec(\mathbf{P})^{(0)T}, \Sigma^{(0)}_{vec(\mathbf{P})})^T. \tag{26}$$

(ii) Find the corresponding $\Phi^{(0)}$ and $\mathbf{g}^{(0)T}_{ij}$.

(iii) For $t = 1, 2, \ldots, n_{iter}$

Step 1 : Generate $\beta^{(t)}$ according to $\mathcal{N}(\boldsymbol{\mu}_\beta, \Sigma_\beta)$.

Update β to $\beta^{(t)}$, then generate $\sigma^{2(t)}$ from a $InvGamma(\alpha_{\sigma^{2(t-1)}}, \beta_{\sigma^{2(t-1)}})$ distribution.

Step 2 : Update $\mathbf{a_{sh}}$ to $\mathbf{a}^{(t)}_{sh}$ and update σ^2_{sh} to $\sigma^{2(t)}_{sh}$. Then, update $\boldsymbol{\alpha}$ according to the Langevin-Hastings Hybrid algorithm and update σ^2_{sc} to $\sigma^{2(t)}_{sc}$.

Step 3 : Find $\mathbf{a}_{sc}$ according to $\mathbf{a}_{sc} = e^{\sigma_{sc}\boldsymbol{\alpha}}$.

Step 4 : Update $\boldsymbol{\gamma}$ according to the Langevin-Hastings Hybrid algorithm. Find $vec(\mathbf{P})$ according to $vec(\mathbf{P}) = \sigma\Sigma^{1/2}_{vec(\mathbf{P})}\boldsymbol{\gamma}$ and update $\mathbf{P}$ accordingly. Calculate Φ by $\Phi = \text{Jupp}^{-1}([\mathbf{0} \;\; \mathbf{Z}^T\mathbf{P} \;\; \mathbf{0}] + \text{Jupp}(\mathbf{S}))$ and then update $\mathbf{g}$ using the new values in Φ.

Step 5 : Update the hyper prior $\Sigma_{vec(\mathbf{P})}$ to its posterior distribution using the new values in $vec(\mathbf{P})$ and return to Step 1.

(iv) Iterate until all posterior distributions converge.

To conclude let us return to the Bayesian curve registration paper by Telesca and Inoue (2008) that was mentioned in the introduction. Similar to our approach, they employ a common shape function and model this as well as their warping functions as linear combinations of B-spline basis

functions. However, they use a random walk procedure to draw posterior samples of the ϕ_i which is known to converge slowly if the proposal density is not chosen to be close to the true distribution. As a result, our use of the Langevin- Hastings Hybrid algorithm can be expected to produce faster convergence performance. Other differences between our methodology and that of Telesca/Inoue include how the hyperparameters are modeled. In particular, we allow the variance covariance matrix of $\mathbf{P}$ to be chosen as a hyper-parameter with an inverse Wishart prior distribution to allow for more flexibility in the estimation process. Perhaps the most significant difference between the Telesca/Inoue formulation and ours concerns the way the ϕ_i are modelled. Telesca and Inoue (2008) employ a multivariate normal prior for the ϕ_i which has the consequence of not ensuring monotonicity of the warping functions. By using the Jupp transformation as we have done here it is possible to avoid such difficulties.

3 Empirical Study

In this section, we examine the performance of our registration methodology in a small empirical experiment. In this simulation study we will obtain Bayesian estimators for the parameters in our model and thereby obtain registered curves that can be employed in a cross-sectional sample mean for estimation of the process mean. After that, we will compare the performance of our methodology with results obtained using the Ramsay and Li (1996) approach and with the unregistered cross-sectional mean. At the end of this section, we will discuss the sensitivity of our results to the choice of the prior distributions.

All the computations in this section were carried out in R. Original source code was created for our Bayesian algorithm while the Ramsay and Li (1996) methodology was implemented using the register.fd function from the R fda package.

The design of the simulation is motivated by the empirical work in Gervini and Gasser (2004). In particular, we use a sine function as the prototype process mean in our simulation which has the consequence of creating problems for estimation of peaks and valleys of a curve.

Let the true mean function be

$$\mu(t) = sin(2\pi t), \tag{27}$$

for $t \in [0, 1]$. This function has a peak at $\tau_{01} = 0.25$ and a valley at $\tau_{02} = 0.75$, with maximum value 1 and minimum value -1. For each sample curve, we then choose n uniformly spaced input values $t_j = (j-1)/(n-1)$, $j = 1, \ldots, n$, and generate data as follows.

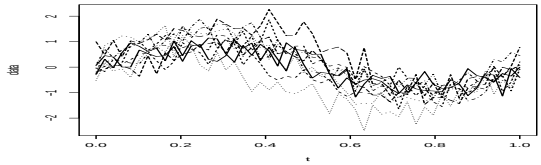

Figure 1: 10 simulated sample curves.

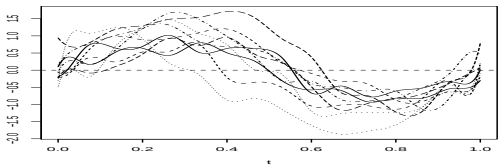

Figure 2: 10 smoothed sample curves.

(i) We first generate the location for the peak and valley for each curve randomly. Specifically, for curve i, let τ_{i1}, τ_{i2} be the locations for the peak and valley, respectively. We then take $\tau_{ik} = \tau_{01} + \xi_{ik}$, $k = 1, 2$ with $\xi_{ik} = \min\{\max\{V_{ik}/12, -0.24\}, 0.24\}$ and the V_{ik} being generated independently from a $\mathcal{N}(0, 1)$ distribution. The truncation employed to produce the ξ_{ik} is to ensure that $\tau_{i1} < \tau_{i2}$ and τ_{i1} and τ_{i2} are within $[0, 1]$.

(ii) Secondly, we choose the warping function $h_i(t)$ to be piecewise linear functions with $h_i(0) = 0$, $h_i(\tau_{01}) = \tau_{i1}$, $h_i(\tau_{02}) = \tau_{i2}$ and $h_i(1) = 1$.

(iii) We next generate the responses as $x_{ij} = a_i + b_i\mu(h_i^{-1}(t_{ij})) + \epsilon_{ij}$, where a_i and b_i are independent and identically distributed as $\mathcal{N}(0, 0.1(1 - 1/\sqrt{2}))$ and $\mathcal{N}(1, 0.1)$ random variables, respectively and the ϵ_{ij} are random errors following a $\mathcal{N}(0, 0.1)$ distribution. These choices of the variances of a_i, b_i and ϵ_{ij} make the error-to-model ratio 0.5.

(iv) We generate $M = 10$ sample curves with each curve being sampled at either $n = 20$ or 50 points to produce the actual data. Thirty replications for each combination were generated.

A specific data set with 10 curves each sampled at 50 points is shown in Figure 1. The curves in the figure include noise. Figure 2 displays smoothed versions of the curves in Figure 1. We choose a cubic B-spline basis with 18 knots to smooth the curves.

In order to estimate the mean function for the data set in Figure 1, we ran 6500 iterations of the MCMC process with the first 2500 posterior samples representing burn-in and with thinning

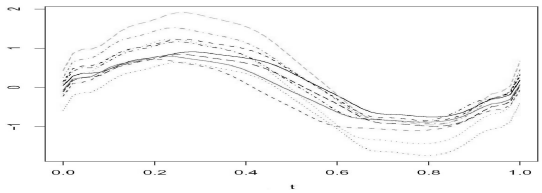

Figure 3: Registered sample curves produced by our method.

through retention of only every 20th posterior sample. The number of knots used for the warping functions was chosen to be 3 and the number of knots used for the global shape function μ was set at 18. Figure 3 shows the resulting registered sample curves and Figure 4 shows the corresponding registered mean curve together with associated 95% credible intervals. It clearly shows that our registered cross-sectional mean curve has a peak at approximately 0.25, with maximum around 1. In addition, the value of the true mean function falls within the 95% credible intervals. The curve in dotted lines in Figure 4 is the cross-sectional sample mean of the smoothed sample curves. Our registered mean can be seen to do a better job of estimating the timing of the valley.Figure 5 shows the registered posterior median with 95% credible intervals. Comparison with Figure 4 suggests that the results are similar to those obtained using the registered posterior mean. We also

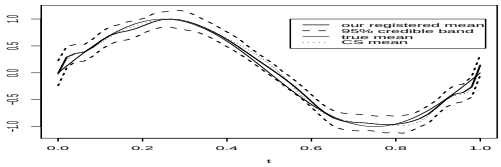

Figure 4: Our registered mean curve (in bold solid lines) with 95% credible intervals (in bold dashed lines), true mean function (in solid lines) and the cross-sectional (CS) sample mean (in dotted lines).

compared the performance of our method for the data in Figure 1 to results obtained from the Ramsay and Li (1996) continuous monotone registration method. Their method was applied twice in this example. First we chose the cross-sectional mean function as the target function and used their technique with smoothing parameter 0.005 to get an initial set of registered curves. Then, we computed a new cross-sectional mean using these registered curves and treated the new cross-sectional mean as the target function for a second application of their technique. Figure shows the registered sample curves produced by the Ramsay/Li method while Figures 7 and 8 show the

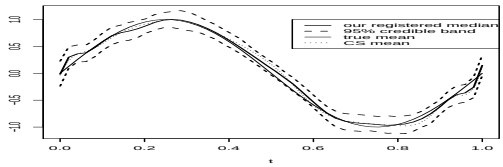

Figure 5: Our registered median curve (in bold solid lines) with 95% credible intervals (in bold dashed lines), true mean function (in solid lines) and the cross-sectional (CS) sample mean (in dotted lines).

cross-sectional mean obtained through the Ramsay/Li method in dotted lines. Their approach appears to be over-estimating the peak and under-estimating the valley of the true regression curve. There is also a bias present in estimating the timing of the valley that is visually evident in the plot. One could choose to apply the above process for a third time or, more generally, attempt to iterate to some type of convergence. But, in practice our experience has been that there is not much improvement to be gained by doing this.

For this particular data set, we obtained the posterior mean and median of the error variance σ^2 to be 0.2 with corresponding 95% credible interval $(0.17, 0.25)$. This over-estimates the true value of 0.1. We also investigated the hyper parameters σ^2_{sh} and σ^2_{sc} and found that their true

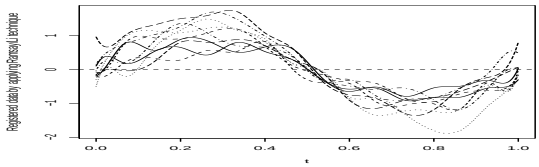

Figure 6: Registered sample curves produced by the Ramsay/Li method.

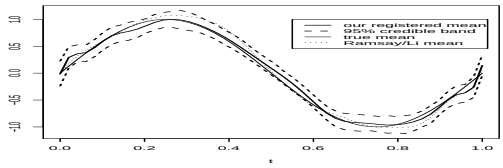

Figure 7: Our registered mean curve (in bold solid lines) with 95% credible intervals (in bold dashed lines), true mean function (in solid lines) and the Ramsay/Li registered cross-sectional mean (in dotted lines).

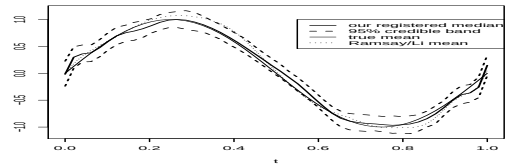

Figure 8: Our registered median curve (in bold solid lines) with 95% credible intervals (in bold dashed lines), true mean function (in solid lines) and the Ramsay/Li registered cross-sectional mean (in dotted lines).

values fall in their associated 95% credible intervals.

We used root average squared error ($RASE$) as a measure of performance for our mean function estimator where

$$RASE(\hat{\mu}) = \left[\sum_{j=1}^{n} \{\hat{\mu}(t_j) - \mu(t_j)\}^2 / n \right]^{1/2}. \tag{28}$$

Figure 9 summarizes the $RASE$ produced by four estimated mean curves: our posterior mean, our posterior median, the sample cross-sectional mean and the registered cross-sectional mean obtained through the Ramsay/Li method for sampling at $n = 20$ points per curve. The median of $RASE$ for our method is smaller than the other two.

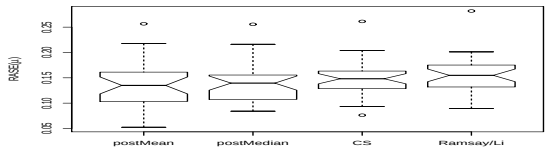

Figure 9: Simulated root average squared errors of $\hat{\mu}$ for $n = 20$.

Figure 10 provides the same information for $n = 50$ sample points per curve.

We also compare the combined $RASE$ results for the case of $n = 20$ and $n = 50$ in Figure

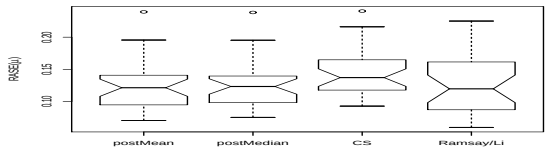

Figure 10: Simulated root average squared errors of $\hat{\mu}$ for $n = 50$.

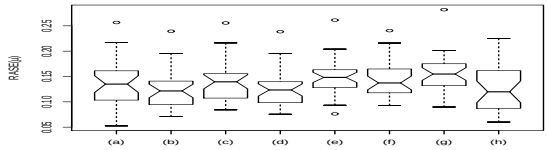

Figure 11: Simulated root average squared errors of $\hat{\mu}$: (a) Our posterior mean for $n = 20$, (b) our posterior mean for $n = 50$, (c) our posterior median for $n = 20$, (d) our posterior median for $n = 50$, (e) cross-sectional mean for $n = 20$, (f) cross-sectional mean for $n = 50$, (g) Ramsay/Li cross-sectional mean for $n = 20$, (h) Ramsay/Li cross-sectional mean for $n = 50$.

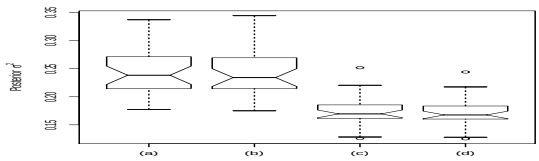

Figure 12: Box plots of estimated σ^2. (a) Posterior means of σ^2 for $n = 20$, (b) posterior medians of σ^2 for $n = 20$, (c) posterior means of σ^2 for $n = 50$, and (d) posterior medians of σ^2 for $n = 50$. The true value of σ^2 is 0.1.

11. For all the four methods, $RASE$ decreases as n increases, with the reduction being most pronounced for our method and the Ramsay/Li method.

The posterior means and medians of the error variance for the case of $n = 20$ and $n = 50$ are displayed in the box plot in Figure 12. When $n = 20$, the median of the posterior mean/median of σ^2 is about 0.16. When the sampling points are increased to $n = 50$, the median of the posterior mean/median of σ^2 decreases to about 0.14. Thus, there appears to be a positive bias in estimation of σ^2. The posterior means and medians of the shift parameter $\mathbf{a}_{sh}$ for the $n = 20$ and $n = 50$ cases are shown in Figure 13. The median of the posterior means of $\mathbf{a}_{sh}$ is lower than 0 and the variability is about the same as that of the true values of the shift parameter, as indicated by the notches in Figure 13. We looked at the 95% credible intervals for $\mathbf{a}_{sh}$ and found that about 98% of them contain the true values. The posterior mean and median for the hyper parameter σ^2_{sh} are shown in Figure 14. About 90% of the 95% credible intervals of σ^2_{sh} contain the true value 0.03. We plotted the true shift parameter values versus the estimated values in Figure 15, which shows linear trends with slopes around 0.6 for $n = 20$ and around 0.8 for $n = 50$. Both the slopes and

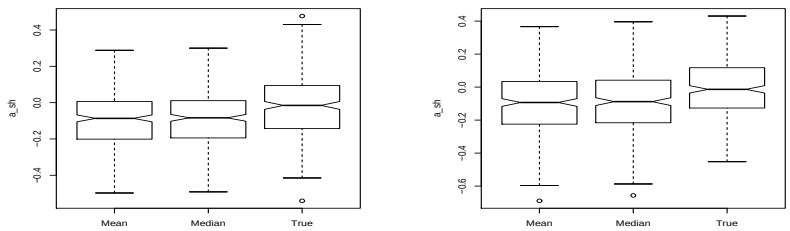

Figure 13: Box plots of a_{sh} for $n = 20$ (left panel) and $n = 50$ (right panel).

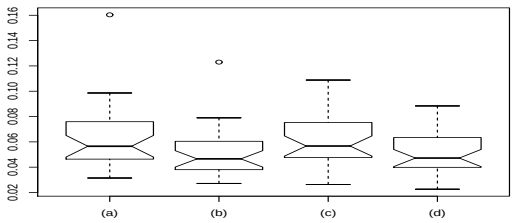

Figure 14: Box plots of estimated σ^2_{sh}. (a) Posterior means of σ^2_{sh} for $n = 20$, (b) posterior medians of σ^2_{sh} for $n = 20$, (c) posterior means of σ^2_{sh} for $n = 50$, and (d) posterior medians of σ^2_{sh} for $n = 50$. The true value of σ^2_{sh} is 0.03.

correlation coefficients increase as the number of sampling points increases. Estimation results for the $\mathbf{a}_{sc}$ are summarized by the box plots of the posterior mean/median estimators in Figure 16. The median of the posterior mean/median of $\mathbf{a}_{sc}$ is about 1, which is the true mean for the slopes that were used to generate the data. The box plot of the hyper parameter ${\sigma_{sc}}^2$ is shown in Figure 17. The posterior means/medians of the hyper parameter ${\sigma_{sc}}^2$ become closer to the true value 0.1 as the number of sampling points increase. In addition, all the 95% credible intervals include the true value 0.1.

Figure 18 shows the scatter plots of the scale parameter values for the data and the corresponding estimated values. Again, both the slopes and the correlation coefficients increase as n increases. Empirical work in Telesca and Inoue (2008) compares their approach with landmark registration and the Gervini and Gasser (2004) self-modeling warping function method under a simulation framework that is virtually identical to the one employed in this section. In combination with our comparisons with the Ramsay and Li (1996) approach, one could view the combined implication as suggesting superior performance may be obtained using a Bayesian approach (either ours or that

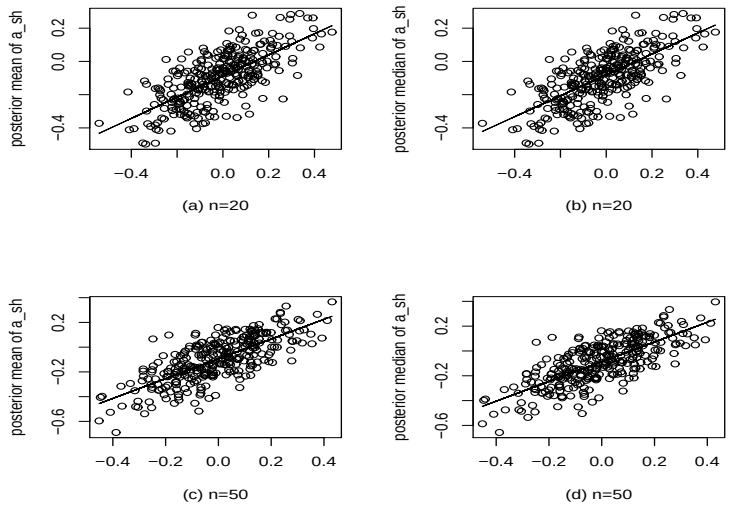

Figure 15: Sample curve shifts versus estimated values. (a) Slope 0.636679 and correlation coefficient 0.6994137, (b) slope 0.628929 and correlation coefficient 0.7012147, (c) slope 0.799683 and correlation coefficient 0.738109 and (d) slope 0.795091 and correlation coefficient 0.7416735.

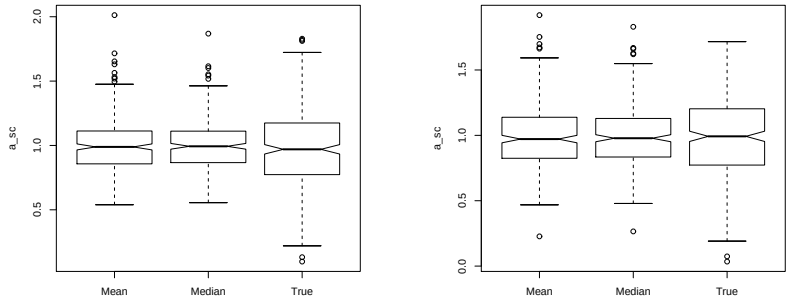

Figure 16: Box plots of a_{sc} for $n = 20$ (left panel) and $n = 50$ (right panel).

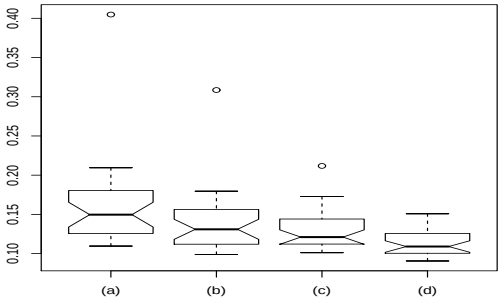

Figure 17: Box plots of estimated σ_{sc}^2. (a) Posterior means of σ_{sc}^2 for $n = 20$, (b) posterior medians of σ_{sc}^2 for $n = 20$, (c) posterior means of σ_{sc}^2 for $n = 50$, and (d) posterior medians of σ_{sc}^2 for $n = 50$. The true value of σ_{sc}^2 is 0.1.

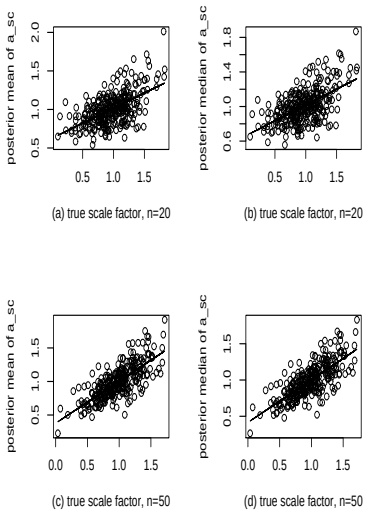

Figure 18: Sample curve scale factors versus estimated values. (a) Slope 0.39190 and correlation coefficient 0.5694445, (b) slope 0.36656 and correlation coefficient 0.5669515, (c) slope 0.62638 and correlation coefficient 0.7475686 and (d) slope 0.58813 and correlation coefficient 0.7449118.

of Telesca and Inoue, 2008) than from two of the more popular methods for curve registration. However, we are puzzled by the Telesca and Inoue (2008) result that reports superior empirical performance for landmark registration over the Gervini and Gasser (2004) method in terms of $RASE$. This disagrees with the Gervini and Gasser (2004) conclusion for this same simulation scenario where they find their method outperforms landmark registration. Until this disparity is resolved, it is difficult to interpret the implications that the Telesca and Inoue (2008) empirical findings have for our approach.

We conclude this section with a discussion of the sensitivity that we observed from the choices of the prior distributions in our empirical work. First, we note that we have tried both proper and improper prior for σ^2 and β in our particular simulated data sets and found little difference in the estimation of the posterior mean and median of the parameters σ^2, β, $\mathbf{a}_{sh}$, $\mathbf{a}_{sc}$, σ^2_{sh} and σ^2_{sc}.

We tried proper prior distributions for σ^2 that were inverse Gamma with shape parameter 2 and scale parameters 1, 0.5 or 0.1. Proper prior distributions for β that were multivariate normal with identity variance matrix were used with mean vectors of $\mathbf{0}$ or $\mathbf{1}$. These choices did not affect the performance of the registered curves and the registered mean curves. For the specific data set in Figure 5, all the proper priors and the improper prior for σ^2 and β give results with $RASE$ values all around 0.08.

For the hyper parameters σ^2_{sh} and σ^2_{sc}, we gave them hyper prior inverse gamma distribution with scale parameters of 1.5, 1 or 0.5 so that the resulting prior mean is 1.5, 1 or 0.5. These choices of the prior distribution do not influence the performance of the registered curves and the mean curve. The posterior mean and median of σ^2_{sh} and σ^2_{sc}, however, increase as their prior mean increases. But the true values still fall in the 95% credible intervals.

4 Summary and Conclusions

Functional data analysis has been an active research area in recent years. As a result the curve registration problem in functional data has received much attention in the literature. In the case of time related data, a registered process mean function provides a very useful tool for understanding the timings and values of certain features that are of interest in a functional data analysis context. Most existing registration methods rely on the estimation of warping functions. One challenge here is that warping functions need to be modeled as monotone increasing function. To solve this problem, we employed monotone spline functions with coefficients produced from the Jupp

transformation whose inverse has the monotone increasing property.

In order to estimate the mean function, we framed the registration problem in a shape invariant model setting. This approach has the consequence that registered sample curves preserve the same shape but are stretched, shrunk or shifted in some way. We then developed a method to estimate the shape and amplitude from a Bayesian perspective. Our formulation treated the cross-sectional mean curves and warping functions as random realizations of certain distributions. We also treat the shifts and the amplitudes as random covariates in the model. Those parameters are estimated by taking samples through an MCMC algorithm. In the MCMC process, we incorporated the Gibbs sampler and the Langevin-Hastings Hybrid algorithms to obtain a fast converging path for the posterior samples. Limited empirical comparisons with the continuous monotone registration method of Ramsay and Li (1996) suggests that our metod is competitive with others that are commonly used in this area while having the advantage of providing the ability to obtain credible intervals for the model parameters and process mean function.

One extension of our registration method concerns the way of choosing the number and locations of knots of the B-spline basis functions that are used to model both the process mean and warping functions. The current approach is ad hoc and does not allow for data fitting adaptation in terms of knot locations. One possible way to allow for knot location flexibility is to choose a large number of potential knots from which a subset of knots can be selected. The problem then turns into a variable selection problem that could be addressed using Bayesian variable selection techniques as in Smith and Kohn (1996). We intend to investigate this approach in future work.

Another aspect of our model we want to further examine is the choice of λ_0 in the hyper-prior $\Sigma_{vec(\mathbf{P})}$. In particular, we want to explore the use of cross-validation methods or (Bayesian) generalized maximum likelihood to adaptively estimate this parameter. It is also of interest to further examine choice of the prior distribution for β. Specifically, we would like to assess the effect of choosing a prior distribution such as the one we chose for the hyper-prior $\Sigma_{vec(\mathbf{P})}$. An adaptive choice of the "smoothing parameter" could then be implemented here as well.

Acknowledgement: Majumdar's and Eubank's research was supported by grants from NSF. The authors would like to express their appreciation to Lyndia Brumback and Mary Lindstrom for making their code available to us.

Appendix

Since we have introduced the improper prior $\pi(\beta, \sigma^2) = 1/\sigma^2$ into the model defined in Section 2, we would like to show that the joint distribution is proper. That is, we would like to prove that

$$\int p(\beta, \sigma^2, \theta; \mathbf{x}) d\beta d\sigma^2 d\theta d\mathbf{x} = 1, \tag{29}$$

where θ is the vector of all the other parameters used in the model, other than (β, σ^2). To undertake this proof, we state two important results from the text "Bayesian Data Analysis" (Gelman *et al*, 2004, p. 356) below.

Result 1: Suppose that $\mathbf{y}$ is a $N \times 1$ vector and β is a $(Q+4) \times 1$ vector such that

$$\mathbf{y}|\beta, \sigma^2, \mathbf{X} \sim \mathcal{N}(\mathbf{X}\beta, \sigma^2 \mathbf{I}_N)$$

with $\mathbf{I}_N$ being the identity matrix of dimension N and the prior distribution is given by

$$p(\beta, \sigma^2|\mathbf{X}) = \frac{1}{\sigma^2}. \tag{30}$$

Then, it follows that

$$\beta|\sigma^2, \mathbf{y}, \mathbf{X} \sim \mathcal{N}(\beta, \sigma^2 V_\beta)$$

and

$$\sigma^2|\mathbf{y}, \mathbf{X} \sim \text{Inv} - \chi^2 \left(N - (Q+4), \frac{1}{N-(Q+4)} (\mathbf{y} - \mathbf{X}\hat{\beta})^T (\mathbf{y} - \mathbf{X}\hat{\beta}) \right), \tag{31}$$

where $\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$, $V_\beta = (\mathbf{X}^T\mathbf{X})^{-1}$ and $\text{Inv} - \chi^2$ is the inverse χ^2 distribution.

From result 1, it is clear that the joint posterior distribution of (β, σ^2) given $(\mathbf{y}, \mathbf{X})$ is valid. Hence,

$$p(\beta, \sigma^2, \mathbf{y}, \mathbf{X}) = \frac{l(\mathbf{y}|\beta, \sigma^2, \mathbf{X})}{\sigma^2} \tag{32}$$

defines a valid joint density for $(\beta, \sigma^2, \mathbf{y})$ with $l(\mathbf{y}|\beta, \sigma^2, \mathbf{X})$ being the likelihood. This leads to our second result:

Result 2: $\int\int\int p(\beta, \sigma^2, \mathbf{y}|\mathbf{X}) d\beta d\sigma^2 d\mathbf{y} = 1$: i.e.,

$$\int\int\int \frac{1}{(2\pi\sigma^2)^{N/2}\sigma^2} \exp\left\{ -\frac{(\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta)}{2\sigma^2} \right\} d\beta d\sigma^2 d\mathbf{y} = 1. \tag{33}$$

Next, we turn to our model (1). Let us define

$$\mathbf{y}(\mathbf{x}, \mathbf{a}_{sh}) = \mathbf{x} - \text{diag}(\mathbf{1}_{n_1}, \ldots, \mathbf{1}_{n_M})\mathbf{a}_{sh} \tag{34}$$

and

$$\mathbf{X}(\Phi, \mathbf{a}_{sc}) = \text{diag}\{\mathbf{B}[\mathbf{U}(\mathbf{t}_1)\boldsymbol{\phi}_1]\beta, \ldots, \mathbf{B}[\mathbf{U}(\mathbf{t}_M)\boldsymbol{\phi}_M]\}\mathbf{a}_{sc}. \tag{35}$$

For simplicity, we can denote $\mathbf{y} = \mathbf{y}(\mathbf{x}, \mathbf{a}_{sh})$ and $\mathbf{X} = \mathbf{X}(\Phi, \mathbf{a}_{sc})$. Then, we observe that the likelihood of the model is given by

$$l(\beta, \sigma^2, \theta; \mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left\{ -\frac{(\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta)}{2\sigma^2} \right\}. \tag{36}$$

Hence it follows from Result 2 that

$$\int\int\int l(\beta, \sigma^2, \theta; \mathbf{x})/\sigma^2 d\beta d\sigma^2 d\mathbf{y} = 1. \tag{37}$$

Note that the prior distribution for θ is proper in that

$$\int \pi(\theta) d\theta = 1. \tag{38}$$

So we have

$$\int\int\int\int \pi(\theta) l(\beta, \sigma^2, \theta; \mathbf{x})/\sigma^2 d\beta d\sigma^2 d\mathbf{y} d\theta = 1, \tag{39}$$

which proves that

$$p(\beta, \sigma^2, \theta, \mathbf{x}) = \pi(\theta) l(\beta, \sigma^2, \theta; \mathbf{x})/\sigma^2$$

is a valid density.

References

[1] Alshabani, A. K. S., Dryden, I. L., Litton, C. D., and Richardson, J. (2007). Bayesian analysis of human movement curves. *Applied Statistics* **56**, 415-428. 235-254.

[2] de Boor, C. (1978). *A Practical Guide to Splines*. New York: Springer-Verlag.

[3] Brumback, L. and Lindstrom, M. (2004). Self modeling with flexible, random time transformations. *Biometrics* **60**, 461-470.

[4] Gasser, T. and Kneip, A. (1995). Searching for structure in curve samples. *Journal of the American Statistical Association* **90**, 1179-1188.

[5] Gelman, A., Carlin, J. B., Stern, HS. and Rubin, DB. (2004). *Bayesian Data Analysis*. Boca Raton: Chapman & Hall/CRC.

[6] Gervini, D. and Gasser, T. (2004). Self-modeling warping functions. *Journal of the Royal Statistics Society* **66**, 959-971.

[7] Jupp, D. (1978). Approximation to data by splines with free knots. *SIAM Journal on Numerical Analysis* **15**, 328-343.

[8] Lawton, W., Sylvestre, E. and Maggio, M. (1972). Self modeling nonlinear regression. *Journal of the American Statistical Association* **14**, 513-532.

[9] McKeague, I. (2005). A statistical model for signature verification. *Journal of the American Statistical Association* **100**, 231-241.

[10] Møller, J., Syversveen, A. and Waagepetersen, R. P. (1998). Log Gaussian Cox process. *Scandinavian Journal of Statistics* **25**, 451-482.

[11] Møller, J. and Waagepetersen, R. P. (2004). *Statistical Inference and Simulation for Spatial Point Processes.* Boca Raton: Chapman & Hall/CRC.

[12] Ramsay, J. and Li, X. (1996). Curve registration. *Journal of the Royal Statistical Society* **60**, 351-363.

[13] Ramsay, J. and Silverman, B. (1997). *Functional Data Analysis.* New York: Springer.

[14] Ramsay, J. and Silverman, B. (2002). *Applied Functional Data Analysis: Methods and Case Studies.* New York: Springer.

[15] Sakoe, H. and Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Signal Processing* **26**, 43-49.

[16] Smith, M. and Kohn, R. (1996). Nonparametric regression using Bayesian variable selection. *Journal of Econometrics* **75**, 317-343.

[17] Telesca D. and Inoue L.Y.T. (2008). Bayesian Hierarchical curve registration. *Journal of the American Statistical Association* **481**, 328-339.